A Holistic Approach to Non-Terrestrial 5G Networking with LEO Satellites: Algorithms, Experiments, and Insights

N. CAMERON MATSON, Georgia Institute of Technology, U.S.A.
YU-TAI LIN, Georgia Institute of Technology, U.S.A.
KARTHIKEYAN SUNDARESAN, Georgia Institute of Technology, U.S.A.

Non-terrestrial networks (NTNs) have been proposed as a key component of next generation of mobile networks. Satellite networks can potentially enable connectivity anywhere on earth, and modern satellites are capable of providing high throughput connectivity. However, there remain several open questions about how NTN will work in practice. We show that blindly applying terrestrial RAN architectures and connectivity management algorithms to the NTN context fails to deliver on multiple network and user level metrics. In this work we present an NTN system that takes a holistic view of the problem, considering both the radio access network architecture as well as the algorithms that drive user session orchestration in the face of satellite mobility. Using a realistic satellite emulation platform as well as large-scale simulations we show that our proposed system outperforms baseline solutions in simultaneously balancing multiple key performance indicators including throughput, coverage, and stability by reducing the impact of satellite mobility.

CCS Concepts: • **Networks** → **Wireless access networks**; **Mobile networks**; *Network design and planning algorithms*; **Network simulations**; *Network design principles.*

Additional Key Words and Phrases: Non-Terrestrial Networking; LEO Satellites; 5G; RAN Architecture; Network Orchestration; Optimization; Emulation

ACM Reference Format:
N. Cameron Matson, Yu-Tai Lin, and Karthikeyan Sundaresan. 2025. A Holistic Approach to Non-Terrestrial 5G Networking with LEO Satellites: Algorithms, Experiments, and Insights. *Proc. ACM Netw.* 3, CoNEXT4, Article 54 (December 2025), 24 pages. https://doi.org/10.1145/3769001

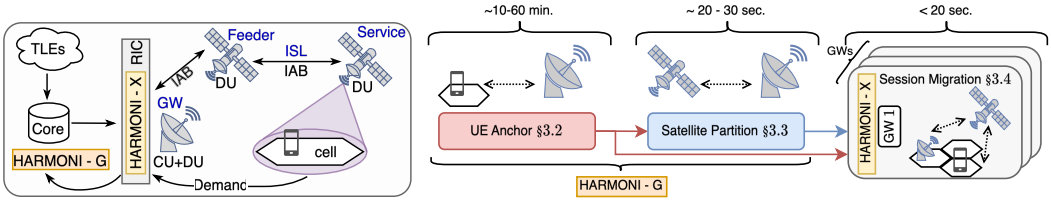
1 Introduction

Broadband networking with satellites is a reality today [16] with numerous providers, e.g. Starlink [35], Kuiper [50], OneWeb [3], etc., providing high speed internet via satellite around the world. Deploying 5G and future generation mobile networks on satellites, known as non-terrestrial networking (NTN), is the next logical step in this evolution, with the potential to provide seamless coverage anywhere on earth, particularly in hard to reach places (e.g. rural areas) [31]. Low-earth orbit satellites (LEO) are especially interesting because of their proximity to Earth; however, it is significantly more challenging to deploy 5G NTNs via LEOs than best-effort IP networks (e.g. [18, 32, 47, 52, 56]), due to their core-centric control and dense, cellular ground coverage.

Despite these challenges, NTNs are coming—officially included in standardization efforts since the 3GPP 5G Rel. 17 in 2020 [6]. Meanwhile on the commercial side as of 2024, Starlink has partnered with several mobile operators around the world to provide direct-to-cell coverage via satellite on the same spectrum as their partner operators [46]. This represents a significant advance and

Authors' Contact Information: N. Cameron Matson, Georgia Institute of Technology, Atlanta, Georgia, U.S.A., ncmatson@gatech.edu; Yu-Tai Lin, Georgia Institute of Technology, Atlanta, Georgia, U.S.A., ytlin1993@gatech.edu; Karthikeyan Sundaresan, Georgia Institute of Technology, Atlanta, Georgia, U.S.A., karthik@ece.gatech.edu.

ACM 2834-5509/2025/12-ART54
https://doi.org/10.1145/3769001



(a) HARMONI RAN Architecture (b) HARMONI Session Orchestration

Fig. 1. System Overview of HARMONI

demonstrates the physical layer feasibility of connecting handsets to LEO satellites; however, little information is available on how the various layers of the Radio Access Network (RAN) map onto the Starlink satellite network, beyond stating that the satellite "acts like a traditional cellular base station" [46]. While the initial use case of this service is to provide low-data rate applications (like emergency alerts and text messages) in remote areas, it remains to be seen if and how NTNs can support applications with more demanding requirements. Similarly, while NTNs will never replace traditional terrestrial networks in dense urban areas, it is still an open question if they can be scaled to provide dense cellular coverage on par with rural or suburban terrestrial networks.

A simple translation of terrestrial cellular systems to NTN would be to deploy gNBs (i.e. base stations) either at the satellite gateway (GW) or on-board the satellites themselves. From there, users (UEs) would decide which satellite to connect to based on signal strength as they do with terrestrial base stations. Using case studies leveraging realistic LEO constellations, we highlight that this approach fails to satisfy multiple key performance indicators (KPIs) essential to mobile network operations. Specifically, we show naively reusing terrestrial paradigms has two major downsides: First, the architecture results in handovers that are more expensive in terms of latency—up to 4× due to messaging with the core network—compared to alternatives. The accumulation of these handover events has a significant impact on the individual UE quality of experience (QoE). Second, while UE-driven connection decisions may be simple, they result in poor overall network efficiency, using $< 50\%$ of total network capacity.

To understand the root cause of why terrestrial paradigms fail in the NTN context, we need to understand the fundamental difference between the two: *infrastructure mobility*. In a terrestrial network, mobility is confined to UEs at the edge of the RAN. By contrast, in NTNs, satellites—critical, connection bearing infrastructure *within* the RAN—are highly mobile. The two key challenges in designing an NTN system are coping with this mobility inversion: 1) How should we map the different RAN functions onto the satellite network? 2) How do we satisfy multiple KPIs—throughput, latency, stability (i.e. reducing the impact and frequency of handovers)—in the face of infrastructure mobility? Stepping back, we observe that fixed-earth cells [23] and satellite GWs are analogous, albeit much larger, to the cells and BSs of a terrestrial cellular network. This provides an opportunity to abstract the satellite mobility *within* the RAN by leveraging the *stability* at the edges.

To this end, we propose a novel NTN framework, HARMONI[1], shown in Fig. 1, which specifies both a deployment *architecture* and a suite of scalable, efficient session orchestration *algorithms*. Designed with satellite mobility in mind, HARMONI enables NTN sessions that provide both the performance (e.g. high throughput, low latency) and stability necessary for broadband applications.

Specifically, HARMONI's architecture (Fig. 1a) leverages the recent Open-RAN (O-RAN) framework to disaggregate a RAN node into centralized-, distributed-, and radio-unit nodes (CU, DU, RU). We propose deploying a CU-DU at each satellite GW and a DU-RU on board satellites. Connections

[1]Holistic Architecture for RAN Management and Orchestrated NTN Integration

between satellite and GW and ISLs within the satellite constellation are enabled via an integrated access and backhaul (IAB) interface. This allows HARMONI to decouple the satellite mobility from the user-CU relationship, replacing expensive inter-CU handovers with inter-DU handovers. By using IAB and proactively establishing tunnels between GW-CUs and satellite-DUs, HARMONI performs the bulk of HO messaging outside of the critical path enabling session migration (from one satellite to another) in such a way that minimizes the effect on user QoE. Further, routing within the IAB network can be preformed using layer 2 routing methods [7], allowing HARMONI to leverage recent works on efficient inter-satellite network routing [28, 29, 44].

While HARMONI's architecture allows us to mitigate the impact of satellite mobility on the user QoE, we still need scalable, efficient algorithms to manage the connections between UEs and their anchor to the core network (i.e. the CU-GW) at the time scale of satellite dynamics. HARMONI couples its split RAN architecture with a spatially and temporally hierarchical suite of session orchestration algorithms (Fig. 1b). First, at coarse timescales it maximizes network utilization through load balancing of the cells amongst GWs. Then, at finer timescales, each GW operates independently to orchestrate the migration of sessions through the dynamic satellite network to maximize session demand satisfaction, reduce latency, and maintain stability. Several noteworthy aspects of HARMONI's algorithms include: (i) efficiency: They provide a flexible framework to jointly optimize and balance the trio of KPIs (capacity, latency, stability) simultaneously based on operator requirements; furthermore, notwithstanding the NP-hardness of the underlying assignment problems, we establish performance guarantees. (ii) scalability: The fine timescale routing decisions are entrusted to individual GWs, focusing only on a subset of satellites and cells, which allows the NTN to scale without sacrificing global performance.

We implement and evaluate HARMONI's components on a satellite network emulator [29], which combines realistic satellite trajectories and connectivity with real IP network tooling, allowing us to emulate 1000s of traces measuring throughput, delay, and routing. We consider realistic NTN scenarios consisting of dense terrestrial cells and 1500+ satellites from Starlink's phase 1 constellation. This realistic emulation is supplemented by large scale simulations to characterize HARMONI at larger network scales. Our evaluations highlight HARMONI's ability to abstract the impact of satellite mobility on 5G NTN sessions resulting in: (1) high network utilization, with total throughput up to 90% of network capacity, (2) low latency, with >95 % of users having under 30 ms of end-to-end latency, and (3) high stability, i.e. minimal session disruption, while maintaining complete cell coverage. HARMONI does this while being scalable, allowing its algorithms to run in real-time at the fine timescales of satellite dynamics.

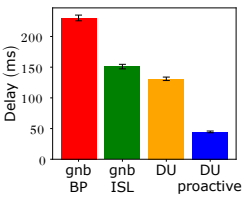


Fig. 2. Avg. HO Delay

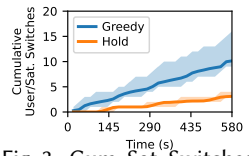


Fig. 3. Cum. Sat. Switches

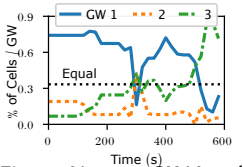


Fig. 4. Uneven GW Load

Finally, we note that many of the existing satellite network work is centered around a single aspect of deploying NTNs, e.g. the handover procedure at the UE- [15, 24, 54] or GW-level [57] or generalized IP routing within the satellite network [14, 18, 32, 42, 47, 52, 53]. Additionally many of the evaluations are limited to simulations involving individual, point-to-point connections through the satellite network, rather than covering dense cellular deployments. To the best of our knowledge, HARMONI is the first work that adopts a holistic view in understanding the complex trade-offs that exist between the architecture and session algorithms specific to an NTN while addressing multiple KPIs simultaneously, and proposes a 5G compliant framework for its efficient and scalable deployment.

2 Need for a Holistic NTN Approach

2.1 RAN Architecture

An important question in deploying NTNs is the choice of RAN architecture. There are several options, enabled by the development of O-RAN [27] and 5G RAN functional splits [39], each with potential benefits and pitfalls. Here we consider three options: transparent, monolithic gNB, and a split CU/DU architecture with CU/DU at the GW and DU/RU on board the satellite.

Transparent satellites are the simplest; the entire gNB (CU+DU+RU) is located at the satellite GW. The satellite has no on-board baseband processing and thus cannot make use of ISLs. This architecture has been included since 3GPP Release 17 [6]. By contrast, the monolithic gNB architecture places the entire gNB on board the satellite.This is an officially supported architecture in the standard as of 3GPP Release 19 [23]. The gNB-on-board solves the ISL issue of the transparent architecture, but another issue arises: whenever users switch satellites, they must also switch gNB-CUs. We claim this is a problem for two reasons: First, in a terrestrial network, inter-CU handovers only occur when a *user moves* between cells covered by unique base stations. This movement is recognized in the core network *access and mobility management function* (AMF). If it is the infrastructure (i.e. the satellite) that has moved rather than the UE, logically, we should not treat this the same. Second, and importantly, this inter-CU handover is expensive, as we will show.

To understand the impact of RAN architecture on user QoE, we perform experiments (detailed in § 4.1) in a realistic satellite network emulator [29] and recreate the handover messaging that occurs when a UE switches satellites using three RAN architectures: (1) `gNB-BP`: A full gNB-on-board without ISL. (2) `gNB-ISL`: The Rel. 19 arch. of gNB-on-board, *with* ISL allowing direct communication over the Xn interface [48]. (3) `DU`: CU-DU split RAN where CU+DU is located at the GW, with DU+RU on board the satellite.

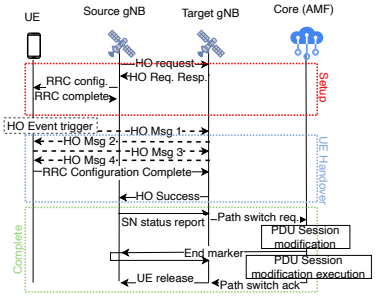


Fig. 5. Inter-CU HO Msg.

Fig. 2 shows the handover delay associated with these architectures. Both `gNB` architectures experience higher delay compared to `DU`. This is because when using a `gNB` architecture, the UE-satellite switch triggers an inter-CU handover (shown in Fig. 5) requiring a message exchange with the core network's AMF. This adds latency due to the RTT and computation within the core. Since `gNB-BP` lacks ISLs, there can be no direct gNB-gNB communication; all signaling must go through the core on the ground. We have not showed the Rel. 17 transparent architecture. Because *all* signaling terminates at the ground (including RACH), the delay is even greater than `gNB-BP`. On the contrary, in `DU` these switch events trigger inter-DU handovers, which avoid any core-based overhead because the handover happens within a single logical gNB (detailed in Fig. 6). Even though inter-DU handovers require slightly more messages, they take less time, especially in NTNs as they do not require an AMF update. The handover-induced delay reported in Fig. 2 has a direct impact on the UE QoE by reducing the instantaneous throughput, as we will see. The fourth option in Fig. 2, `DU-proactive`, used by `HARMONI` (detail in § 3.1), reduces this impact even further. The RAN architecture is only one component of the NTN system; we next explore its counterpart—session orchestration—identifying how terrestrial network algorithms fail to simultaneously satisfy multiple KPIs when used in NTNs.

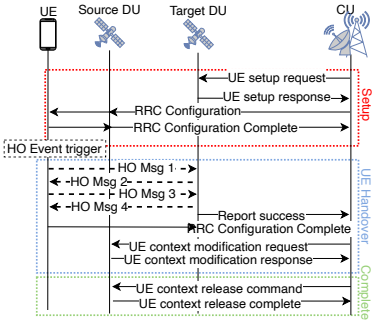


Fig. 6. Inter-DU HO Msg.

2.2 Orchestration: Balancing Multiple KPIs

For any sufficiently large satellite constellation, there are at any moment, multiple satellites in view from any one terrestrial location. This increases the total capacity of the network, but it raises several questions: Which areas on Earth will a satellite serve? If multiple satellites serve the same terrestrial area, how do UEs within that cell decide which satellite to connect to? Finally, since the end-point for all NTN user traffic is the operator's (most likely terrestrial) core network, once the traffic enters the satellite network, how do the satellites "choose" which GW is used to offload its traffic? These questions amount to a problem of session orchestration: *How is the connection from the UE, through the satellite network, to the core managed in the face of satellite mobility?* The answer depends on what we want to optimize. NTNs, like terrestrial mobile networks, need to simultaneously satisfy multiple KPIs. Important ones that we will consider here are: 1) Coverage: Ensuring total coverage requires a constellation of a sufficient size. We assume a modern LEO mega constellation with enough satellites to cover all cells, but we want to ensure that our orchestration provides service to all users at all times; 2) Latency: Due to the large distances between UEs and satellites, the orchestration should minimize delay when possible; 3) Maximizing stability: Infrastructure mobility poses a unique challenge in NTNs. While some churn is inevitable, minimizing the frequency of handover events will lead to higher QoE for UEs; and 4) Fairly satisfying user demand.

We argue that in NTNs, these KPIs present tradeoffs which are less pronounced in terrestrial networks. Mobility is restricted to the UEs which makes stability easier to manage. Base stations are much closer to the UEs, so the choice of gNB has little impact on latency. Base stations are also strategically placed to spatially distribute the network load, and when there is highly asymmetrical traffic, networks can perform load balancing. To illustrate the tradeoffs between these KPIs in NTNs, let's consider a small scenario involving a few hundred cells and three GWs (details in § 4.2) and attempt to optimize, first individually, each of our KPIs.

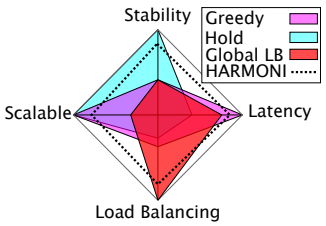


Fig. 7. KPI Tradeoff

Local Schemes for Latency and Stability: To minimize latency, each session should take the shortest path. This is achieved when both UEs and satellites take a `greedy` approach to connectivity. In this scheme, UEs periodically measure the signals of visible satellites, and connect to the strongest; for LEO satellites, the strongest signal is likely to originate from the closest. Likewise, satellites, aware of their positions (or controlled by the operator on the ground) connect to the closest GW. This mirrors terrestrial networks where the connections are largely UE-driven. Though technically, the gNB and core network are responsible for admitting UEs and facilitating handovers, decisions are predicated on local, UE-based measurement reports. A downside of `greedy` is that cells switch satellites and GWs very frequently. Recall that each switch triggers some type of handover impacting UE QoE. Fig. 3 shows the number of cell-satellite and cell-GW that `greedy` incurs over time. If instead of minimizing the latency, we want to maximize the stability, we could "hold" onto connections as long as they are valid and also choose the satellite that will be "alive" the longest (as proposed in several works including [15]). Fig. 3 shows that the `hold` approach does reduce switches.

There is an important downside to both schemes: *poor load balancing results in low network efficiency.* Sessions based on *local* decisions, end up in progressive bottlenecks throughout the network: most UEs connect to a few satellites, which connect to a handful of GWs. This congestion reduces the capacity available per cell *and* inefficiently uses network resources. To illustrate, Fig. 4 shows the percentage of cells served by each GW over time when using the `hold` method. All else being equal, in a "balanced" network, each GW would support approx. 33% of the cells, giving each

cell access to an equal share of the network capacity. On the other hand, when the majority of cells are assigned to a single GW as in Fig. 4, the capacity available per cell suffers.

Global Load Balancing Optimization: What if orchestration decisions were made globally by a centralized entity instead of locally by individual nodes? Consider an orchestration scheme which computes optimal connections between all UEs and GWs, through the satellite constellation. Its objective is to increase the network utilization and decrease congestion to increase the available capacity per cell. The details of this global load balancing formulation can be found in Appendix B. This optimization needs to be computed for the entire network continuously at the timescales of the satellite dynamics. While this may be possible for small scenarios consisting of a few UEs, GWs, and satellites, the (NP-Hard) optimization is unable to scale larger scenarios involving mega-constellations and dense terrestrial cells (e.g. in § 5 we show it takes over 20 s for a small problem involving 1200 cells and 15 GWs). In contrast, while the local session management schemes have downsides, they have the benefit of being naturally scalable.

Fig. 7 summarizes the trade-offs in these terrestrial-inspired single objective schemes: While optimal for their stated objective, each scheme exhibits unacceptable performance along other desired KPI dimensions. In the following section we present HARMONI, a holistic NTN framework designed to scalably and efficiently balance multiple KPIs simultaneously.

3 HARMONI Design

In this section we provide details on the design of HARMONI. The core components are a split CU/DU RAN architecture empowered by ISL over IAB (Fig. 1a) paired with a suite of scalable session orchestration (SO) algorithms (Fig. 1b) to jointly optimize multiple network-wide KPIs such as maximizing network utilization and minimizing session disruption due to satellite mobility.

3.1 NTN RAN Architecture

Fundamentally, HARMONI's architecture (Fig. 1a) is a mapping of RAN entities—UEs, cells, gNBs, and their interfaces—onto a satellite network— GWs, GW-facing feeder-satellites, UE-facing service-satellites, and ISLs. The distinction between service- and feeder-satellites is purely logical as satellites typically have separate RF chains for uplink and downlink in each direction. The same physical satellite may function as both service and feeder. We assume fixed-earth cells [21] to mimic static terrestrial cells. Based on the discussion in § 2.1, HARMONI splits the gNB between the GW and the satellites, locating a complete CU/DU at each GW while placing the DU/RU on board the satellites. The connection between the two, as well as all ISLs between satellite-DUs, are executed using IAB. The split RAN is critical for HARMONI's stability objective. Under this architecture, UE-satellite switches trigger lower-latency inter-DU handovers. HARMONI's SO algorithm anchors each UE to a single GW-CU, nearly eliminating the more costly inter-CU handovers altogether. Two additional important aspects of HARMONI's architecture are:

Trajectory Aware HO: To further increase the stability of NTN sessions, HARMONI takes advantage of predictable satellite trajectories, tracked via global databases of TLEs (two-line element). Using this knowledge, HARMONI's SO algorithms quickly compute when handovers are needed and *proactively* initiate the inter-DU handover procedures between satellites before the switch occurs. HARMONI precreates the necessary bearers/sessions from the GW to the target satellite-DU by performing all but the UE-involved messages in Fig. 6 *before* the actual HO event, eliminating their impact on user QoE. This enables an even lower handover latency than that of DU as shown by DU-proactive in Fig. 2. Handovers procedures are *triggered* periodically by HARMONI's session orchestration algorithms (§ 3.2). It uses knowledge of the satellite trajectories to ensure the cell-service-feeder-GW assignment is valid. **Efficient Routing via IAB:** DU-to-DU connectivity, on the feeder link as well as all ISLs, is facilitated via the integrated-access-and-backhaul (IAB) feature

available in 5G NR [33]. Routing within an IAB network is managed by the Backhaul Adaptation Protocol (BAP) which replaces IP to manage hop-by-hop addressing within the network [53]. By using IAB, HARMONI abstracts away ISL routing and focuses only on choosing the optimal service-, feeder-satellite pair for each session leaving intra-constellation routing to be handled by existing algorithms (e.g. [32, 47, 52, 53]. We note that in addition to static IAB in 5G, dynamic, mobile IAB is an important functionality [9] that future releases are expected to incorporate, especially for NTNs.

3.2 HARMONI Session Orchestration

Because a global, end-to-end optimization (even for a single KPI) cannot scale to realistic NTN scenario sizes, HARMONI session orchestration is spatially, and temporally decomposed into three subproblems (Fig. 1b). Rather than solve each layer of the multi-dimensional problem sequentially, the decomposition leverages the split RAN architecture to work "outside-in": the first submodule optimally assigns cells to GW-CUs (establishing an upper bound on performance) before the remaining two optimize satellite assignments. While not equivalent to the global optimization, this decomposition ensures scalability while each submodule remains optimal. Additionally, the submodules' optimization problems are parameterized to balance multiple KPIs. The orchestration is carried out by two entities: a centralized, core-network function HARMONI-G and a distributed, O-RAN xAPP HARMONI-X running on RAN intelligent controllers (RICs) at each GW.

At slow coarse timescales (e.g. 10s of minutes), HARMONI-G requests aggregate cell demand data from the HARMONI-X instances, and performs **UE-CU Anchoring (§ 3.2.1)**. This global optimization maximizes network utilization through load balancing while enabling the down stream modules to address additional KPIs by forming coherent "GW-regions". HARMONI-G also keeps track of satellite trajectories by pulling from a TLE database. At medium timescales (e.g. 20-30s), it uses these trajectories to compute *regional visibility* data for each GW-region and performs **Satellite Partitioning (§ 3.2.2)**—assigning disjoint subsets of satellites to each GW-region. HARMONI-G uses a novel greedy algorithm that enables fast, efficient partitioning.

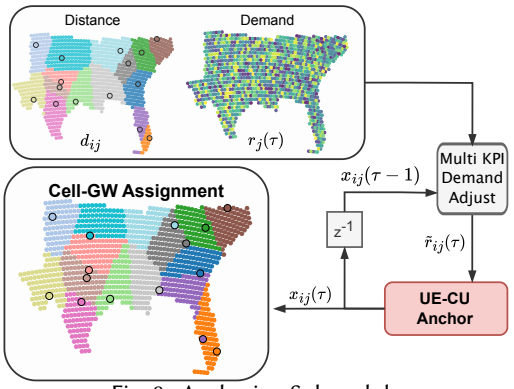


Fig. 8. Anchoring Submodule

The two assignments (cell-GW and satellite-GW), along with the satellite trajectory information, are shared with each GW's HARMONI-X instance. Then, at fine timescales (<20s), each computes *cell visibility*, and uses it to perform in parallel **Session Migration (§ 3.2.3)**, in which HARMONI-X assigns a specific service- and feeder-satellite pair for the NTN sessions from each cell. The local optimization addresses multiple KPIs simultaneously: throughput, latency, and stability in the presence of satellite dynamics.

In the following sections we detail each submodules' optimization formulation, giving particular emphasis on how the formulation relates to the specifics of the NTN session orchestration problem. Additionally, if applicable, we provide performance guarantees for the algorithms in each phase.

3.2.1 Cell-GW Session Anchoring. The first submodule of HARMONI is cell-GW session Anchoring, depicted in Fig. 8, carried out by the core network function HARMONI-G. Rather than handle individual UEs, HARMONI aggregates them within pre-defined, fixed-earth cells—analogous to, but larger than cells in terrestrial systems. The goal of Anchor is to pair each cell with a GW to balance demand across the network. At coarse timescales, HARMONI-G monitors the aggregated cell demand by

querying each GW-CU's HARMONI-X instance. Anchor hanldes multiple objectives: maximizing throughput through load balancing, minimizing the end-to-end latency, and maintaining stability.

Problem: The formulation is as follows: Consider M GWs, each with fixed capacities C_i, and N fixed-earth cells. At a coarse time step, τ, each cell, j, has an aggregate demand, $r_j(\tau)$, (top right of Fig. 8). Let $x_{ij} = 1$ if cell j is assigned to GW i; otherwise $x_{ij} = 0$. If the total demand assigned to a particular GW is exceeded, UEs will experience congestion and be unable to achieve their desired rates. NTNs (like terrestrial networks) utilize scheduled access and can "trim" the demand of each UE through the allocation of resources. In the (likely) event that total demand exceeds the total GW capacity, we apply a uniform "service ratio", λ to all cells for fairness. Since Anchor is only run at coarse intervals, e.g. 10s of minutes, so "fairness" here is different from that in traditional wireless scheduling, e.g. proportional fairness [8, 30], which operates on a faster, frame-by-frame basis. Thus, we have two objectives: maximize λ and maximize the *network utilization*, i.e. the percentage of total network capacity which is utilized. Both objectives can be solved simultaneously through a load balancing objective: by distributing the demand as evenly as possible among the GWs (through a min max formulation, also known as the *minimum makespan problem* [45]) we reduce, as much as possible, the congestion at each GW-CU. $\min_x \max_i \lambda \sum_j x_{ij} r_j(\tau)$, and then maximize λ.

Anchor(τ): (1)

$$\begin{aligned} \min_{x} \quad & \max_i \lambda \sum_j x_{ij}(\tau)\tilde{r}_{ij}(\tau) \\ \text{s.t.} \quad & \sum_i x_{ij}(\tau) = 1, \\ & \lambda \sum_j x_{ij}(\tau)\tilde{r}_{ij}(\tau) \le C_i \forall i \end{aligned}$$

How does Anchor optimize multiple KPIs? We capture all three of HARMONI's global objectives (throughput, latency, stability) by defining a new GW-dependent "virtual demand" variable, $\tilde{r}_{ij}$. To encourage stability, $\tilde{r}_{ij}(\tau) = r_j(\tau)$ if $x_{ij}(\tau - 1) = 1$, i.e. we do not modify the demand for the current GW. For all other GWs, $\{i : x_{ij}(\tau - 1) = 0\}$, we "virtually" increase the demand of cell j—since the objective is a minimization, increasing the effective demand discourages reassignment. We know that the minimal latency is achieved over a bent pipe connection, which can only occur if the cell is assigned to a GW near enough to have visible satellites in common. Thus, to minimize latency, the demand is scaled by a distance-dependent factor: $\left(\beta + \bar{d}_{ij}^{\alpha}\right)$, where $\bar{d_{ij}}$ is the normalized distance between the center of cell j and GW i (because of fixed-earth cells, this is known *a priori*). The parameters α and β control how much to increase the virtual demand based on the distance and stability objectives, respectively. The final formulation is given in Eq. 1.

Algorithm: The minimum makespan problem is NP-Hard. Leveraging the solution in [45], our algorithm is the following:
(1) Relax the integrality constraint, i.e. $x_{ij} \ge 0$ rather than $x_{ij} \in \{0, 1\}$. Perform a binary search on $\lambda \in (0, 1]$ to find the maximum value for which the LP relaxed problem has a feasible solution. Define this service ratio as λ_0 and the LP solution as $\hat{x}_{ij}$. (2) Perform the following rounding algorithm (based on [45]) to obtain integral solution x^*_{ij}: form a bipartite graph of cells and GWs. Create multiple nodes per GW for each fractional assignment in LP-relaxed solution and create weighted edges between them and the cell node according to the $\hat{x}_{ij}$. Then find a minimum cost matching between the cell and virtual GW nodes. (3) Let the demand assigned to each GW be $R_i = \sum_j x^*_{ij} r_j$ (where r_j is the *original* demand.) If $\lambda_0 R_i > C_i$, adjust λ according to the max violation: $\lambda = \min_i \frac{C_i}{R_i} \implies \lambda R_i \le C_i \forall i$.

We emphasize the utility of minimum makespan objective: It is used to control the maximum capacity constraint violation, which directly corresponds to the final value of λ and in turn the satisfaction of each UE. The output of the Anchor $x^*_{ij}(\tau)$, is a cell, GW-CU assignment, that only changes with large-scale changes in demand, *not with satellite mobility.* Because of the distance objective, the cells assigned to particular a GW-CU are close, both to each other and the GW-CU itself, forming *GW-regions* as seen in the lower left of Fig. 8, this is important for the feasibility of the

next submodule. Because Anchor is centralized, it leads to better network performance compared to locally-driven alternatives, as we will see in § 5. Finally, we present the following theorem on the performance[2] of Anchor, with a sketch of proof deferred to Appendix A.1.

Theorem 1. *The algorithm for* Anchor *satisfies no less than* $\frac{1}{2}$ *the max achievable demand.*

3.2.2 Satellite Partition. The next phase of session orchestration (performed by HARMONI-G) is to Partition the visible satellites among the GWs. This is both critical for scalability (it shrinks the global problem to a local one which can by run in parallel at each GW) and necessary from an architectural standpoint (DUs may only connect to one upstream CU in O-RAN). Fig. 9 shows an overview of Partition which takes as inputs cell-satellite visibility (computed from TLEs), Anchor assignments, and the current partition, and outputs a partition for the next time step. The problem may seem similar to Anchor, but satellite mobility induces key differences:

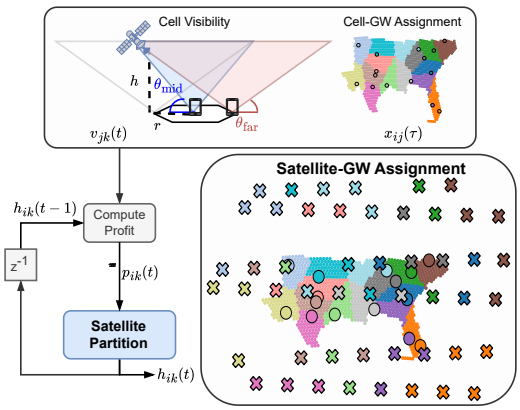


Fig. 9. Sat. Partition

Visibility: A satellite k is "visible" from a particular point on earth (e.g. cell j) if its elevation angle above the horizon, θ_{jk}, is greater than some $\theta_{\min}$. Let $v_{jk}(t) = 1$ if $\theta_{jk}(t) \geq \theta_{\min}$. We require full cell coverage, implying that within a GW-region, at least one of the assigned satellites must be visible for all cells. To ensure that a satellite is visible from all points within the cell, we increase the minimum visibility elevation angle proportionately to the size of the cell (shown in the inset of Fig. 9). **Timescale:** The session anchoring takes place over a coarse timescale (e.g. 10s of minutes), but Partition must occur faster to account for changes in satellite visibility. In choosing a timescale for the satellite partition problem there is a trade-off in the longevity of a particular assignment (too large a time step will result in invalid visibility assumptions) and the computational cost of finding the partition. We find that 20-30 seconds is a good value for sufficiently large constellations, ensuring that satellites are visible during the time step while providing ample time to run HARMONI's fast partitioning algorithm.

Problem: Our objective is to optimize h_{ik}, the assignment of satellites (k) to GWs (i), so as to maximize the number of visible satellites per cell in each GW region. This increases the total capacity per cell, decreasing congestion, and offers greater flexibility at fine timescales to achieve multiple KPIs. To do this, we define the "profit" in assigning satellite k to the GW i as the number cells in i which can be covered by k: $p_{ik}(t) = \sum_j x^*_{ij}(\tau) v_{jk}(t)$. Where $x^*_{ij}(\tau)$ is the Anchor assignment.

To increase stability, the profit in assigning satellite k to a new GW i is scaled by a parameter $\kappa \leq 1$, i.e $\tilde{p}_{ik}(t) = \kappa p_{ik}$ if $h_{ik}(t-1) = 0$. We formulate our objective as a sum-log (commonly used in rate-allocation problems to promote fairness [8, 30]) over the number of satellites per GW. The final Partition formulation is given in Eq. 2. Constraint Eq. 2b ensures a valid partition, while constraint Eq. 2c ensures complete cell coverage. GWs are included as additional "cells" in the Partition, ensuring at least one satellite is visible to the GW to serve as feeder. The existence of a feasible solution is a function of the network topology. Obviously, if there are more GWs than visible satellites, a solution does not exist; however, this is mitigated by the size of modern constellations.

[2]This is a worse case performance bound (for any arbitrary scenario). We find the average performance on realistic scenarios like the ones considered in § 5 is quite close to the upper bound.

`Partition`(t):

$$\max_{h} \quad \sum_i \log \sum_k h_{ik}(t)\tilde{p}_{ik}(t) \tag{2a}$$

$$\text{s.t.} \quad \sum_i h_{ik}(t) \leq 1 \forall k, \tag{2b}$$

$$\sum_k v_{jk}(t) \sum_i x^*_{ij}(\tau) h_{ik}(t) \geq 1 \forall j \tag{2c}$$

Algorithm: The `Partition` problem is significantly harder than the assignment problem in `Anchor`. While the partition constraint is similar to other assignment problems, the objective and cell-coverage constraint is more like a maximum coverage problem [34].

To solve `Partition` we have designed a graph-based, greedy algorithm. The details of the algorithm are described in Appendix B. An example partition is shown in Fig. 9. In § 5.3 we discuss how adjusting κ allows us control the trade-off between different NTN KPIs.

3.2.3 Satellite Session Migration. The last phase of `HARMONI`'s session orchestration (shown in Fig. 10) is to allocate satellite-DUs to connect each cell and its assigned GW and then manage the migration of these sessions in response to satellite mobility. This process runs on independent `HARMONI-X` xApp instances located at each GW-CU. The task is to assign two satellites for each session: A *service* satellite-DU to provide access to the UEs within a cell and a *feeder* satellite-DU which connects to the GW-CU. These do not need to be distinct satellites, i.e. the same physical satellite function as both the service and feeder for multiple cells. Because `HARMONI` utilizes IAB, each downstream DU may only connect to one upstream DU which allows us to simplify the joint allocation into a sequential one. The cell-GW and satellite-GW assignments from `HARMONI-G` define the sets of cells and satellites each `HARMONI-X` considers in solving `Service` and `Feeder`. Each `HARMONI-X` process takes the cell-satellite visibility and satellite trajectories (computed from TLEs), the cell demand, along with the current solution to produce a new set of assignments.

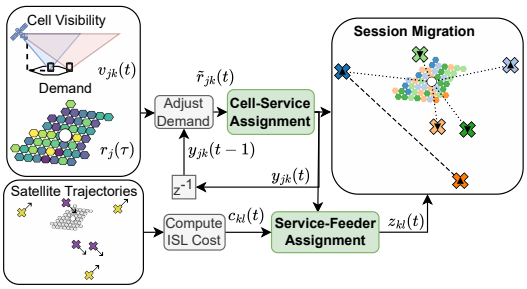


Fig. 10. `HARMONI-X` Fine-Time Process

Problem: Each sub-stage formulation is similar to the previous with the following adjustments:

θ-based Channel Quality: `HARMONI` is fundamentally a *network* level orchestration, thus the capacity constraints we present represent the *total* capacity of that RAN node as determined by its processing capabilities and its allocated bandwidth. However, we should also account for the fact that each space-to-ground link has an associated capacity dependent on the SNR. Since these channels are LOS, the SNR is a function of the distance and any atmospheric attenuation. We capture both factors through the elevation angle: low elevation angles correspond to both greater distances and more atmospheric losses; the shortest distance and the minimum attenuation both occur when the satellite is directly overhead. While `Partition` used a binary notion of visibility, this submodule uses a more continuous notion of visibility to model the quality of the space-ground channel. We define a service (feeder) visibility factor $\tilde{v}_{jk}$ based on the angle: $\tilde{v}_{jk}(t) = \sin\theta_{jk}(t)$ if $\theta_{jk} \geq \theta_{\min}$ and 0 otherwise. The elevation-adjusted demand parameter is $\rho_{jk}(t) = \frac{r_j(t)}{\tilde{v}_{jk}(t)}$. Note that if a satellite k is invisible from cell j, (i.e. $\theta_{jk} < \theta_{\min}$) then $\rho_{jk} = \infty$. For stability, we incorporate a handover cost by adjusting the demand with a parameter $\gamma \geq 1$: $\tilde{\rho}_{jk}(t) = \rho_{jk}$ if $y_{jk}(t-1) = 1$, otherwise $\tilde{\rho}_{jk} = \gamma\rho_{jk}$.

ISL Cost: `HARMONI` only specifies the service-feeder pair. It does not dictate full ISL routing and rely on network layer ISL routing protocols to efficiently connect the service and feeder satellites via IAB. However, the chosen service-feeder pair can have a large impact on the eventual ISL route chosen. Previous work [13, 47] identified that in constellations with +grid connectivity, ISL routes

`Service`$_i(t)$ (3):

$$\min_{y} \quad \max_{k} \mu_S \sum_j y_{jk}(t)\tilde{\rho}_{jk}(t)$$
$$\text{s.t.} \quad \sum_k y_{jk}(t) = 1 \quad \forall j,$$
$$\mu_S \sum_j y_{jk}(t)\tilde{\rho}_{jk}(t) \le C_S \ \forall k$$

`Feeder`$_i(t)$ (4):

$$\min_{z} \quad \max_{l} \mu_F \sum_k z_{kl}(t) q_{kl}(t)$$
$$\text{s.t.} \quad \sum_l z_{kl}(t) = 1 \quad \forall k,\ \mu_F \sum_{l,k} z_{kl}(t) q_{kl}(t) \le C_i,$$
$$\mu_F \sum_k z_{kl}(t) q_{kl}(t) \le C_F \ \forall l, \qquad \sum_{l,k} z_{kl}(t) c_{kl}(t) \le I_{\text{ISL}}$$

are shortest when the source and destination satellites travel in the same direction, e.g. north vs. south. To account for this, we introduce an ISL cost $c_{kl} \in \{1, 2, 3\}$ for assigning service satellite s_k to feeder satellite s_l, corresponding to $\{s_k = s_l$, same, and different directions$\}$, respectively. Minimizing this cost incentivizes bent-pipe routes (i.e. same satellite for both service and feeder) if possible, and otherwise empowers the routing protocol to minimize the hop count and end-to-end latency. Other costs that are functions of the particular routing protocol may also be used. This cost minimization objective can be incorporated into the minimum makespan formulation as an additional constraint without changing the algorithm or the performance guarantees [45].

Formally, let $\{s_k\}$ and $\{s_l\}$ be the sets of service and feeder satellites (they may overlap.) Each service (feeder) satellite has capacity C_S (C_F), and recall GW i has capacity C_i. Let y_{jk} (z_{kl}) be the assignment variable of service satellite k to cell j (service k to feeder l). As before we have service ratios for `Service` (μ_S) and `Feeder` ($\mu_F \le \mu_S$) to scale the satisfied demand. The formulation for the `Service` problem is given in Eq. 3. Given the solution (y^*, μ_S^*), each service satellite s_k is assigned demand $q_k = \mu_S^* \sum_j y_{jk}^* \rho_{jk}$. For a chosen ISL cost value I_{ISL} the formulation for the `Feeder` assignment is given in Eq. 4.

Algorithm: `Service` and `Partition` are solved sequentially using the algorithm from `Anchor` (§ 3.2.1), with the exception of the ISL cost in `Feeder` which may be set and adjusted independently; however, enforcing a lower ISL cost may yield a lower service ratio. We present the following theorem regarding the performance with a sketch of proof deferred to Appendix A.3:

Theorem 2. *The final μ^* following* `Feeder` *is no less than $\frac{1}{4}$ that of the optimal, joint assignment.*

A snapshot of the final session orchestration is shown in Fig. 10. Potential feeder satellites are connected to the GW via a dotted line, while the service-feeder relationship is indicated by a dashed line. Note that the service satellite connects to a potential feeder traveling in the same direction (as indicated by the arrows). `HARMONI-X` then shares the session configuration with the satellite-DUs, proactively sets up the necessary tunnel to minimize disruption, and triggers the necessary inter-DU handover procedures.

Algorithm Summary: Each of `HARMONI`'s three submodules—cell-GW anchoring, satellite partitioning, and session migration—are formulated to be efficient and flexible in catering to multiple, often competing, KPIs. By adjusting the parameters of each phase, we can adjust the balance between KPIs. Note that the optimal latency and stability schemes (i.e. `greedy` and `hold` from § 2.2) can be achieved within this framework by setting the parameters appropriately. Meanwhile, the algorithm design is hierarchical, both temporally (multi-timescale) and spatially (centralized `HARMONI-G` and distributed `HARMONI-X`) which allows it to scale to large NTN deployments without sacrificing global performance as we demonstrate in § 5.

4 Implementation and Experiments

4.1 HARMONI Implementation

To evaluate `HARMONI`, we use StarryNet [29] an open source platform for realistic satellite networking experimentation. StarryNet (Fig. 11 left) uses containers to enable collecting real network

measurements (e.g. throughput and latency) on emulated satellite deployments based on real satellite trajectories and topologies. Cell demands and satellite trajectories are generated using StarryNet and communicated to the appropriate HARMONI components (Fig. 11 right) implemented in Python. HARMONI performs session orchestration and feeds the slow- and fine-timescale assignments back to StarryNet. Given the service and feeder satellite assignment, StarryNet computes an efficient ISL route within the constellation. While useful for its satellite mobility emulation capabilities, StarryNet was designed for IP-level experimentation, and does not have by default the ability to emulate 5G systems. As such we make two major additions to the IP-level capabilities provided by StarryNet:

NTN specific network entities. In StarryNet, all nodes are either Satellites or GroundNodes. In order to emulate NTNs, we add the functionality needed to represent the different capabilities of UE and GW nodes. Additionally, we modify both the GroundNode and Satellite entities such that we can map particular NTN RAN architectures onto them, e.g. gNB-on-board, CU/DU split.

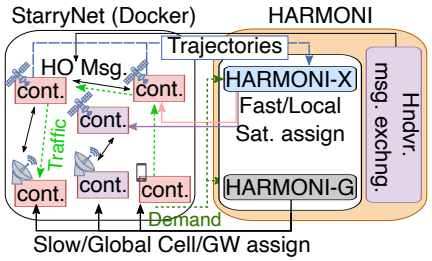


Fig. 11. HARMONI impl. w/StarryNet

Capturing HO Overhead To accurately capture the handover overhead for different deployment strategies, we implement shell programs that exchange the necessary control message of both inter-CU and inter-DU handovers based on the procedures shown in Fig. 5 and 6. As a reminder, proactive inter-DU handovers require only messages that involve the UE at the time of handover. We then deploy the programs to different entities based on their roles (i.e. UE/DU/RU). StarryNet emulates realistic transmission delays based on the geo-location of satellites and ground nodes, allowing us to accurately profile the handover overhead reported in Fig. 2 and overall network performance (e.g. sum throughput, average delay, etc.) for different NTN architectures and session orchestration algorithms. We use Linux tools `iperf` to capture network throughput and `ping` and `traceroute` to understand the end-to-end network latency.

4.2 Experiments

To represent a realistic satellite deployment, we use a subset of Starlink's GW deployment [2] and Phase 1 of their constellation [1]—a Walker-Delta [5] pattern with 1584 satellites across 72 orbits with an inclination angle of 53° each with 22 satellites at a height of roughly 550 km. We assume each GW is equipped with eight antennas/radios each capable of supporting a 20 Gbps feeder link [40], i.e. $C_i = 20$ Gbps $\forall i$. We assume each satellite has the same capacity for both the service and feeder links, i.e. $C_S = C_F = 20$ Gbps.

Name	GWs	Cells	Area (km^2)	Sats.
small	3	147	0.236 E6	24
se	15	1147	2.087 E6	54
east	21	1511	2.498 E6	76
usa	54	4569	8.102 E6	127

Table 1. Scenario Summary

We define fixed-earth cells using Ubers's H3 protocol [4] with resolution 4, producing an average cell radius of 26 km (much larger than any terrestrial system cell). As mentioned in § 3.2.1, we are concerned with aggregate demand within a cell; thus, at the coarse timescale (10 minutes) each cell's demand is generated (uniformly) randomly between 5-20 Gbps. We assume satellites are equipped with phased array antennas [40] capable of generating steerable, spot beams. For satellite visibility, based on § 3.2.2, the minimum elevation angle to 25°. We emulate satellite mobility using realistic trajectories, captured at 20 second intervals for both the medium- and fine-timescales. To emulate how an operator would deploy HARMONI, we vary the ground node deployment in several scenarios described in Table 1 and shown in Fig 26 in the Appendix. All scenarios use the full 1584 satellite Starlink constellation. The "Sats." column is the average

number of cell-visible satellites. Because StarryNet can only support a limited number of links we can only use it to evaluate `small`; thus, we evaluate the network-level performance of HARMONI's design for the remaining scenarios in simulation by using the SkyField Python library [41].

Baselines: We evaluate HARMONI at a system-level by comparing it to alternative architecture-algorithm pairs. The potential architectures are: i) `transparent` with gNB at GW (i.e. 3GPP Rel. 17), ii) gnb: full gNB-on-board (i.e Rel. 19 and the Starlink D2C deployment), and iii) `split`: a split CU/DU architecture utilizing IAB (used by HARMONI). To create a full system these architectures are paired with one of the following baseline algorithms: a) `greedy`: cells/satellites choose the closest satellite/GW at each time step. b) `hold`: like `greedy`, but cell-satellite and satellite-gw relationships are maintained as long as possible. c) `qglobal`: a quasi-globally optimal algorithm (detailed in Appendix B), which performs a sequential assignment of cells to satellites to GWs. The true global formulation is only used to demonstrate its inability to scale.

5 Evaluation

5.1 Emulated System-Level Results

We now present HARMONI's performance on several system-level benchmarks using StarryNet emulator on the `small` scenario. To measure throughput, each cell's demand is used as the requested throughput to `iperf`. Measured throughput depends on the congestion at each satellite and GW node as determined by the session orchestration algorithm. We use `ping` and `traceroute` to measure the end-to-end delay and route from cell to GW. Unless otherwise specified, HARMONI's session orchestration algorithm is configured with the parameters $(\alpha, \beta, \kappa, \gamma) = (2, 1, 0.01, 2)$ to balance KPIs (parameter choice detailed in § 5.3). To model unpredictable satellite availability, we set a "damage ratio" in StarryNet causing 5% of satellites to fail within each timestep, giving a conservative estimate for network reachability.

Network Utilization: We define "network utilization" as the ratio between total network traffic ($\sum_i \mu_i^* \sum_{j' \in \mathcal{U}_i} r_{j'}$) and instantaneous system capacity (the minimum of the total GW and visible satellite capacities). Fig. 12a shows the average network utilization over time for HARMONI as well as two baseline systems: `gnb+greedy` and `split+hold`. HARMONI achieves the highest network utilization, approx. 80% on average, nearly twice the baselines, with less variation over time.

HO Impact on QoE: While steady-state network utilization is a function of session orchestration, the different system architectures have a direct impact on how instantaneous user QoE responds to satellite mobility. To measure this impact, we look at the average throughput of cells before and after satellite switch events, shown in Fig. 12b. The network utilization prior to the event reflects the steady-state performance. All approaches experience a drop in throughput immediately after the switch event occurring at 0ms while the UE and gNB exchange handover messages; however, since HARMONI utilizes a split CU/DU architecture with pro-active path creation, its throughput recovers most rapidly (< 50ms). The other systems take longer to return to the pre-event levels, up to 200 ms for the `gnb+greedy` system. The difference in steady-state throughput is a result of averaging over multiple time-steps. The accumulated impact of these HO is explored in Appendix C.

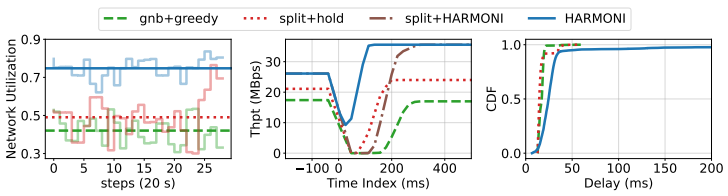


Fig. 12. (a) Network Util. (b) HO impact (c) End-to-End Delay

End-to-End Delay: Fig. 12c shows the CDF of the end-to-end (i.e. cell-to-GW) delay measured via `ping` in StarryNet for HARMONI, `greedy`, and `hold` algorithms. Note that this delay, as opposed

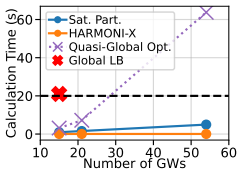


Fig. 13. Scalability

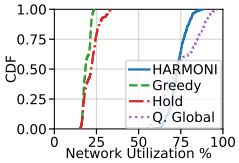


Fig. 14. Utilization

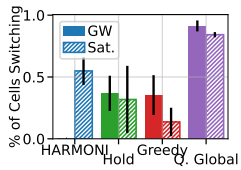


Fig. 15. Switches per Step

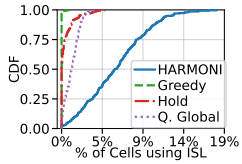


Fig. 16. % cells w/ISLs

to the HO-induced latency discussed above, is not a function of the specific NTN architecture. Compared to the two locally-driven algorithms, which result in mostly bent-pipe connections between user and GW, HARMONI experiences a comparable, but slightly longer delay: 30 ms vs 23 ms at the 85th percentile. The session orchestration algorithm in HARMONI does not explicitly incorporate the end-to-end delay (from user to GW) in its formulation. Rather the delay is minimized indirectly in two ways: 1) by anchoring sessions at nearby GW-CUs (§ 3.2.1) and 2) by minimizing the potential ISL cost of service-to-feeder satellite pairs (§ 3.2.3). In addition, it leverages external routing algorithms to establish low-latency ISL connections through the constellation. While the delay is slightly longer than the compared baselines, and has a long tail, we argue it is well justified given the superior performance of HARMONI in both the network utilization and session stability. Ultimately, this represents one of the fundamental trade-offs in satellite networks as discussed in § 2.2. Fortunately, HARMONI is parameterized to balance these tradeoffs (§ 5.3).

5.2 Algorithm Performance Simulations

We now focus specifically on the effectiveness of HARMONI's session orchestration algorithm. Due to the limitations of StarryNet, we simulate the larger scenarios in Python using the SkyField library [41] for the satellite trajectories. Most of the following results, unless specified, come from the se scenario. Additional results for remaining scenarios are available in Appendix C.

Scalability: One of the primary motivators for HARMONI's hierarchical design is to decouple any *global*, network-wide optimization (involving thousands of variables) from satellite dynamics, without sacrificing global performance. We achieve this by partitioning the cells and satellites among the GWs, and running *local* optimizations at each GW on a much smaller set of variables. To illustrate HARMONI's scalability, we compare the run time of the two time-critical submodules—**Satellite Partition** and HARMONI-X's **Session Migration**—with that of the qglobal optimization for the three larger scenarios in Table 1. The run times are measured on a workstation with 32 GB of RAM and a 24-core, Intel i9 3.5 GHz CPU. The 85th percentile run times for each are shown in Fig. 13. We first note the runtime of the true global formulation (which only optimizes load balancing) on the smallest scenario is already over 20 s. We see that qglobal does not scale with the size of the scenario, primarily due to the significant increase in the number of cells. While the compute time of HARMONI's greedy partitioning algorithm does increase with the number of GWs (and satellites), the number of visible satellites does not increase as fast as the number of cells does, thus the compute time remains manageable. The session migration compute time does not increase with scenario size. Because each GW performs it independently, for its own region, it only depends on the number of cells per GW region, which remains fairly constant (50-100) as the size of the scenario increases. Importantly, both the partitioning and session migration are achievable in well under the 20 second threshold set for HARMONI updates to handle satellite mobility.

Feasible Network Utilization: We define network utilization the same as in § 5.1 with the exception that it is computed from the service ratio delivered by the SO algorithm rather than measured directly. Fig. 14 shows the CDF of network utilization by HARMONI and the local and global baselines over 10 minutes of simulation. As expected, the locally-based algorithms severely

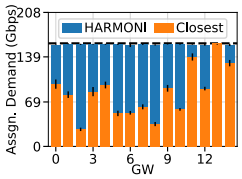


Fig. 17. GW LB

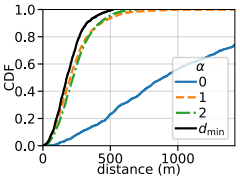


Fig. 18. Cell-GW distance

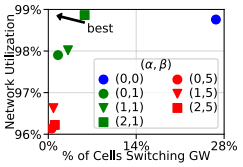


Fig. 19. Util. vs. Switches

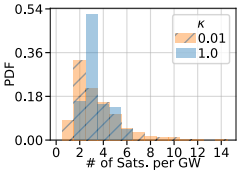


Fig. 20. Dist. of Sats.

under utilize network resources and `qglobal` is most efficient. In fact HARMONI slightly outperforms `qglobal` (which is not optimal itself) around 40% of the time.

Switch Events: Fig. 15 shows the average percentage of cells experiencing GW- (solid/left) and satellite-switches (hatched/right) every 20 seconds within a single coarse time interval of 10 minutes. Because of HARMONI's UE-GW anchoring (§ 3.2.1), users experience *zero* GW switches within the coarse time interval. While HARMONI causes more frequent satellite-switches than `greedy` and `hold`, it is comparable to the number of GW-switches caused by those schemes. In HARMONI, all satellite-switches are minimally impactful proactive, inter-DU handovers. Finally, note that the high network utilization of `qglobal` comes at the cost of extremely high churn—nearly all of the cells experience a switch event every 20 seconds, illustrating the competing nature of these KPIs. Unlike baselines, HARMONI strikes a balance between maintaining stability and high utilization.

ISL Usage: A CDF of the percentage of cells at each time step using ISL connections to reach its assigned GW (i.e. those that are not reachable via bent-pipe) is shown in Fig. 16. HARMONI requires ISLs for a larger (but small overall) proportion of users compared to baseline algorithms. This higher ISL usage directly leads to the marginal increase in end-to-end delay observed in Fig. 12c; however, as mentioned, this is a natural trade-off for the utilization and stability gains of HARMONI.

5.3 Parametric KPI Trade-offs

A key benefit of HARMONI's design is supporting multiple, often competing KPIs simultaneously. In this section we highlight the HARMONI's ability to adjust the balance of these objectives.

Anchoring: The formulation for `Anchor` (Eq. 1) has two parameters: α to minimize the GW-cell distance and β to encourage stability. One objective of `Anchor` submodule of HARMONI's session orchestration load balancing user demand. Fig. 17 shows the average assigned demand (after applying the coarse-time service ratio) to each of the 15 GWs in the `se` scenario over 10 coarse time steps when $(\alpha, \beta) = (2, 1)$. We compare this to a demand assignment which anchors cells to their nearest GW. HARMONI utilizes 99% of the total GW capacity while the "closest" strategy achieves only 52%, with an uneven distribution of demand across GWs. The final throughput is limited by the remaining submodules, so it is critical that this stage is as efficient as possible; HARMONI achieves nearly optimal load balancing. Amazingly, we achieve this efficiency while maintaining high GW-cell proximity. Fig. 18 shows a CDF of the distance between cells and their assigned GW. For non-zero α we are very close to the minimum distance assignment. Increasing GW-cell proximity increases the chance a cell and its assigned GW have similar satellite visibility which reduces likelihood of needing ISLs to connect the two, ultimately reducing end-to-end latency.

The final objective in the coarse timescale, after load balancing and minimizing GW-cell distance, is to reduce the the number of cell-GW switches that trigger expensive inter-CU handovers. We incentivize maintaining the current cell-GW assignment through the parameter β in Eq. 1; however, it comes at a cost of decreased network utilization. Fig. 19 plots the average network utilization (as a fraction of the GW capacity) over 10 coarse time steps vs. the percentage of cells that switch GWs for different (α, β) pairs. We see that increasing the value of β decreases both the number of handovers and network utilization. This is because we may need to decrease the service ratio if

demand changes and we do not adapt the assignment. *An operator may wish to vary* (α, β) *over time to find the correct balance between efficiency and stability depending on the deployment scenario.*

Partitioning: `Partition`'s objectives are to 1) ensure complete cell coverage, 2) evenly distribute satellites among GW regions, and 3) reduce HO caused by satellite-GW switches over time. `HARMONI` balances of the partition objective with switch minimization through the parameter κ. Fig. 20 shows the PDF of the number of satellites per GW for $\kappa \in \{0, 0.01\}$. From a balanced partition perspective, the "tighter" distribution resulting from $\kappa = 1$ is preferable as no one GW has significantly more or fewer satellites than any other GW. The $\kappa = 0.01$ partition has much higher variation—some GWs receive only one satellite, while others receive 10+. The trade off is that increasing κ decreases stability. Fig. 21 shows a CDF of the number of satellite-GW switches per 20 second time step over 10 minutes. We see that $\kappa = 0.01$ maintains complete cellular coverage with significantly fewer satellite-GW switches than $\kappa = 1$. *Once again,* `HARMONI` *allows operator to balance efficiency (higher utilization) and stability (fewer handovers) depending on need.*

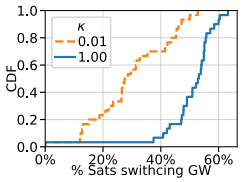


Fig. 21. Sat-GW Switch

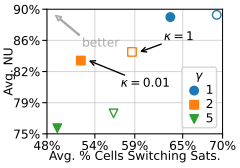


Fig. 22. Util. vs. Switch

Session Migration: As with the other two phases, the fine-time session migration tries to balance efficiency (e.g. high network utilization) with stability (minimizing cell-satellite switches)–this time through the parameter γ . Fig. 22 shows the average total network utilization vs. the percent of cells who switch satellites per time step at different values of γ and κ. Increasing γ lowers the frequency of switches (and thus HOs) at the cost of utilization, illustrating the trade-off that exists between the objectives and how γ can tune the performance of the system. Note that κ sets the operating point of this utilization-stability curve. A lower κ decreases satellite-GW switches, illustrating the trickle down effect of `HARMONI`'s hierarchical design. With low γ, we achieve nearly 100% network utilization, but require handing over nearly 50% of user cells every 20 seconds. *While* `HARMONI` *performs minimally impactful, pro-active inter-DU HOs, high volumes may still be undesirable to operators.* `HARMONI` *provides the flexibility to adapt to these competing KPIs.*

6 Related Work

Intra-Constellation Routing There has been increasing interest in research around routing within mega-constellation satellite networks [19, 20, 32, 37, 42, 53, 55, 56]. The main challenge identified is the constant (but predictable) churn of satellite neighbors. A common approach to deal with this mobility is to perform geographic routing [32, 42]. Other works treat the routing through the network as a flow problem and present graph-based solutions to devise routes that maximize flow [55, 56]. IAB routing within the satellite network has also been considered [37, 53].

Handover How to execute handovers in the face of satellite mobility has also received a lot of attention [12, 15, 24, 25, 51, 52, 54]. The survey in [12] provides a nice overview of the different scenarios and types of switch events. Many works use a local measurement (e.g. proximity, signal strength) for determining handoff, similar to terrestrial networks; however, some works have focused on NTN-specific methods like angle [54], antenna gain [24], longest potential connection [15]. An application of 5G's (terrestrial) conditional handover procedure was investigated in [25]. [51] considers the predictable motion of satellites and pre-emptively performs message passing between source and target satellites prior to the switch similar to our proactive handovers; however it is restricted to inter-CU HO between monolithic gNB satellites rather than within gNB inter-DU handovers. Finally, while most works look at the problem from the point of view of an individual user, [15, 52] investigate handover for larger groups of users simultaneously.

Architecture While the aforementioned work considers only individual aspects of implementing NTN, several recent works have studied different network architectures in the context of satellites. For example [10, 11] investigated how software-defined networking (SDN) and network-function virtualization (NFV) might be applied to satellite networks; however, they do not specifically consider the implications of this architecture on a 5G system. On the other hand, [17, 43] do look specifically at the potential 5G RAN architectural choices for NTN, but the focus of these works is primarily on the feasibility of each architecture from a PHY layer perspective. As the leading commercial satellite network, much work has gone into understanding the design of SpaceX's Starlink, e.g. [19, 22, 36, 49]. Important to our discussion is the consensus among researchers (and confirmed by SpaceX) that Starlink runs a global network reconfiguration every 15s similar to HARMONI's fine-time operation. However, because Starlink is still a black box, we still do not understand how GW assignment is performed which is critical for NTN session orchestration. Finally, while 3GPP has included the transparent and regenerative gNB in the standard as part of Rel. 17 [6] and 19 [23]; however we have shown the stability benefits of a split RAN.

7 Conclusion

In this work, we have demonstrated how reuse of terrestrial RAN architecture and session management algorithms in NTNs fails to achieve many network- and user-level KPIs because of high satellite (i.e. infrastructure) mobility. In response, we have presented HARMONI, a holistic system for NTN. HARMONI consists is both a split RAN architecture, enabling HARMONI to mitigate the impact of satellite mobility by anchoring users to CU-GWs and proactively establishing tunnels for inter-DU HO. This architecture is coupled with a hierarchical set of session orchestration algorithms that create/migrate sessions between cells, GWs, and the satellite network to jointly optimize multiple KPIs: throughput, utilization, delay, and stability. Using a realistic satellite emulator, we demonstrated HARMONI's effectiveness to enable scalable, efficient, and flexible NTN operation.

Acknowledgments

This work was supported in part by NSF (CNS 2208761).

References

[1] 2020. Attachment Narrative App SAT-MOD-20200417-00037. https://fcc.report/IBFS/SAT-MOD-20200417-00037/2274315

[2] 2023. Starlink Ground Station Locations (2024). https://starlinkinsider.com/starlink-gateway-locations/

[3] 2024. Eutelsat OneWeb and Intelsat make history delivering high-speed connectivity above the Arctic Circle. http://oneweb.net/resources/eutelsat-oneweb-and-intelsat-make-history-delivering-high-speed-connectivity-above-arctic

[4] 2024. Home | H3. https://h3geo.org/

[5] 2024. Satellite constellation. https://en.wikipedia.org/w/index.php?title=Satellite_constellation&oldid=1237721529#Walker_Constellation Page Version ID: 1237721529.

[6] 3GPP. 2022. 3GPP TS 38.300 version 17.1.0 Release 17. https://www.etsi.org/deliver/etsi_ts/138300_138399/138300/17.01.00_60/ts_138300v170100p.pdf

[7] 3GPP. 2024. 5G; NR; Backhaul Adaptation Protocol (BAP) specification. https://www.etsi.org/deliver/etsi_ts/138300_138399/138340/18.00.00_60/ts_138340v180000p.pdf

[8] Ahmed Abdel-Hadi and Charles Clancy. 2014. A utility proportional fairness approach for resource allocation in 4G-LTE. In *2014 International Conference on Computing, Networking and Communications (ICNC)*. 1034–1040. doi:10.1109/ICCNC.2014.6785480

[9] Zaid Abdullah, Steven Kisseleff, Eva Lagunas, Vu Nguyen Ha, Frank Zeppenfeldt, and Symeon Chatzinotas. 2023. Integrated Access and Backhaul via Satellites. http://arxiv.org/abs/2304.01304 arXiv:2304.01304.

[10] Jinzhen Bao, Baokang Zhao, Wanrong Yu, Zhenqian Feng, Chunqing Wu, and Zhenghu Gong. 2014. OpenSAN: A Software-defined Satellite Network Architecture. *ACM SIGCOMM COMPUTER COMMUNICATION REVIEW* 44, 4 (Oct. 2014), 347–348. doi:10.1145/2740070.2631454 Publisher: ACM SIGCOMM; Assoc Comp Machinery.

[11] Lionel Bertaux, Samir Medjiah, Pascal Berthou, Slim Abdellatif, Akram Hakiri, Patrick Gelard, Fabrice Planchou, and Marc Bruyere. 2015. Software Defined Networking and Virtualization for Broadband Satellite Networks. *IEEE COMMUNICATIONS MAGAZINE* 53, 3 (March 2015), 54–60. doi:10.1109/MCOM.2015.7060482

[12] Abhipshito Bhattacharya and Marina Petrova. 2023. Study on Handover Techniques for Satellite-to-Ground Links in High and Low Interference Regimes. In *2023 JOINT EUROPEAN CONFERENCE ON NETWORKS AND COMMUNICATIONS & 6G SUMMIT, EUCNC/6G SUMMIT*. IEEE, New York, 359–364. doi:10.1109/EUCNC/6GSUMMIT58263.2023.10188361 ISSN: 2475-6490, 2575-4912 Num Pages: 6 Series Title: European Conference on Networks and Communications Web of Science ID: WOS:001039230700070.

[13] Quan Chen, Giovanni Giambene, Lei Yang, Chengguang Fan, and Xiaoqian Chen. 2021. Analysis of Inter-Satellite Link Paths for LEO Mega-Constellation Networks. *IEEE Transactions on Vehicular Technology* 70, 3 (March 2021), 2743–2755. doi:10.1109/TVT.2021.3058126

[14] Quan Chen, Jianming Guo, Lei Yang, Xianfeng Liu, and Xiaoqian Chen. 2020. Topology Virtualization and Dynamics Shielding Method for LEO Satellite Networks. *IEEE Communications Letters* 24, 2 (Feb. 2020), 433–437. doi:10.1109/LCOMM.2019.2958132 Conference Name: IEEE Communications Letters.

[15] Veronica M Grant. 2023. Proliferated Low Earth Orbit (pLEO) Satellite Constellation Handover Cost Analysis. (June 2023).

[16] Yue Guan, Fan Geng, and Joseph Homer Saleh. 2019. Review of High Throughput Satellites: Market Disruptions, Affordability-Throughput Map, and the Cost Per Bit/Second Decision Tree. *IEEE Aerospace and Electronic Systems Magazine* 34, 5 (May 2019), 64–80. doi:10.1109/MAES.2019.2916506 Conference Name: IEEE Aerospace and Electronic Systems Magazine.

[17] Alessandro Guidotti, Alessandro Vanelli-Coralli, Matteo Conti, Stefano Andrenacci, Symeon Chatzinotas, Nicola Maturo, Barry Evans, Adegbenga Awoseyila, Alessandro Ugolini, Tommaso Foggi, Lorenzo Gaudio, Nader Alagha, and Stefano Cioni. 2019. Architectures and Key Technical Challenges for 5G Systems Incorporating Satellites. *IEEE Transactions on Vehicular Technology* 68, 3 (March 2019), 2624–2639. doi:10.1109/TVT.2019.2895263 Conference Name: IEEE Transactions on Vehicular Technology.

[18] Wei Han, Baosheng Wang, Zhenqian Feng, Baokang Zhao, and Wanrong Yu. 2016. Distributed mobility management in IP/LEO satellite networks. In *2016 3rd International Conference on Systems and Informatics (ICSAI)*. 691–695. doi:10.1109/ICSAI.2016.7811041

[19] Mark Handley. 2018. Delay is Not an Option: Low Latency Routing in Space. In *Proceedings of the 17th ACM Workshop on Hot Topics in Networks (HotNets '18)*. Association for Computing Machinery, New York, NY, USA, 85–91. doi:10.1145/3286062.3286075

[20] Yannick Hauri, Debopam Bhattacherjee, Manuel Grossmann, and Ankit Singla. 2020. "Internet from Space" without Inter-satellite Links. In *Proceedings of the 19th ACM Workshop on Hot Topics in Networks (HotNets '20)*. Association for Computing Machinery, New York, NY, USA, 205–211. doi:10.1145/3422604.3425938

[21] Mohsen Hosseinian, Jihwan P. Choi, Seok-Ho Chang, and Jungwon Lee. 2021. Review of 5G NTN Standards Development and Technical Challenges for Satellite Integration With the 5G Network. *IEEE Aerospace and Electronic Systems Magazine* 36, 8 (2021), 22–31. doi:10.1109/MAES.2021.3072690

[22] Liz Izhikevich, Manda Tran, Katherine Izhikevich, Gautam Akiwate, and Zakir Durumeric. 2024. Democratizing LEO Satellite Network Measurement. *Proc. ACM Meas. Anal. Comput. Syst.* 8, 1 (Feb. 2024), 13:1–13:26. doi:10.1145/3639039

[23] Joern Krause. 2024. Non-Terrestrial Networks (NTN). https://www.3gpp.org/technologies/ntn-overview

[24] Enric Juan, Mads Lauridsen, Jeroen Wigard, and Preben Mogensen. 2022. Handover Solutions for 5G Low-Earth Orbit Satellite Networks. *IEEE Access* 10 (2022), 93309–93325. doi:10.1109/ACCESS.2022.3203189 Conference Name: IEEE Access.

[25] Enric Juan, Mads Lauridsen, Jeroen Wigard, and Preben Mogensen. 2022. Performance Evaluation of the 5G NR Conditional Handover in LEO-based Non-Terrestrial Networks. In *2022 IEEE Wireless Communications and Networking Conference (WCNC)*. 2488–2493. doi:10.1109/WCNC51071.2022.9771987 ISSN: 1558-2612.

[26] Richard M. Karp. 1972. Reducibility among Combinatorial Problems. In *Complexity of Computer Computations*, Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger (Eds.). Springer US, Boston, MA, 85–103. doi:10.1007/978-1-4684-2001-2_9

[27] Sadayuki Abeta Toshiro Kawahara and Anil Umesh Ryusuke Matsukawa. 2019. O-RAN Alliance Standardization Trends. (2019).

[28] Tomohiro Korikawa, Chikako Takasaki, Kyota Hattori, and Hidenari Ohwada. 2024. An Adaptive Rule-based Path Selection Method using Link Information in Non-Terrestrial Networks. In *2024 IEEE 10th International Conference on Network Softwarization (NetSoft)*. 286–294. doi:10.1109/NetSoft60951.2024.10588890

[29] Zeqi Lai, Hewu Li, Yangtao Deng, Qian Wu, Jun Liu, Yuanjie Li, Jihao Li, Lixin Liu, Weisen Liu, and Jianping Wu. 2023. {StarryNet}: Empowering Researchers to Evaluate Futuristic Integrated Space and Terrestrial Networks. 1309–1324. https://www.usenix.org/conference/nsdi23/presentation/lai-zeqi

[30] L. Li, M. Pal, and Y. R. Yang. 2008. Proportional Fairness in Multi-Rate Wireless LANs. In *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*. 1004–1012. doi:10.1109/INFOCOM.2008.154

[31] Konstantinos Liolis, Alexander Geurtz, Ray Sperber, Detlef Schulz, Simon Watts, Georgia Poziopoulou, Barry Evans, Ning Wang, Oriol Vidal, Boris Tiomela Jou, Michael Fitch, Salva Diaz Sendra, Pouria Sayyad Khodashenas, and Nicolas Chuberre. 2019. Use cases and scenarios of 5G integrated satellite-terrestrial networks for enhanced mobile broadband: The SaT5G approach. *INTERNATIONAL JOURNAL OF SATELLITE COMMUNICATIONS AND NETWORKING* 37, 2, SI (April 2019), 91–112. doi:10.1002/sat.1245

[32] Lixin Liu, Hewu Li, Yuanjie Li, Zeqi Lai, Yangtao Deng, Yimei Chen, Wei Liu, and Qian Wu. 2022. Geographic Low-Earth-Orbit Networking without QoS Bottlenecks from Infrastructure Mobility. In *2022 IEEE/ACM 30TH INTERNATIONAL SYMPOSIUM ON QUALITY OF SERVICE (IWQOS)*. IEEE, New York. doi:10.1109/IWQoS54832.2022.9812903 ISSN: 1548-615X Num Pages: 10 Series Title: International Workshop on Quality of Service Web of Science ID: WOS:000853833300040.

[33] M. Luglio, C. Roseti, M. Quadrini, and F. Zampognaro. 2024. Definition of Satellite Systems Role in Integrated Access Backhaul (IAB) Architectures. In *2024 International Symposium on Networks, Computers and Communications (ISNCC)*. 1–6. doi:10.1109/ISNCC62547.2024.10758989

[34] Nimrod Megiddo, Eitan Zemel, and S. Louis Hakimi. 1983. The Maximum Coverage Location Problem. *SIAM Journal on Algebraic Discrete Methods* 4, 2 (1983), 253–261. arXiv:https://doi.org/10.1137/0604028 doi:10.1137/0604028

[35] Michael Kan. 2024. Starlink's Laser System Is Beaming 42 Million GB of Data Per Day. https://www.pcmag.com/news/starlinks-laser-system-is-beaming-42-million-gb-of-data-per-day

[36] Nitinder Mohan, Andrew E. Ferguson, Hendrik Cech, Rohan Bose, Prakita Rayyan Renatin, Mahesh K. Marina, and Jörg Ott. 2024. A Multifaceted Look at Starlink Performance. In *Proceedings of the ACM Web Conference 2024 (WWW '24)*. Association for Computing Machinery, New York, NY, USA, 2723–2734. doi:10.1145/3589334.3645328

[37] António J. Morgado, Firooz B. Saghezchi, Pablo Fondo-Ferreiro, Felipe Gil-Castiñeira, and Jonathan Rodriguez. 2024. Intelligent Backhaul Link Selection for Traffic Offloading in B5G Networks. *IEEE Access* 12 (2024), 106757–106769. doi:10.1109/ACCESS.2024.3436890 Conference Name: IEEE Access.

[38] William P. Pierskalla. 1968. Letter to the Editor—The Multidimensional Assignment Problem. *Operations Research* 16, 2 (April 1968), 422–431. doi:10.1287/opre.16.2.422

[39] Michele Polese, Leonardo Bonati, Salvatore D'Oro, Stefano Basagni, and Tommaso Melodia. 2023. Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges. *IEEE Communications Surveys & Tutorials* 25, 2 (2023), 1376–1411. doi:10.1109/COMST.2023.3239220

[40] Mike Puchol. 2022. Modeling Starlink capacity. https://mikepuchol.com/modeling-starlink-capacity-843b2387f501

[41] Brandon Rhodes. 2019. Skyfield: High precision research-grade positions for planets and Earth satellites generator. *Astrophysics Source Code Library* (July 2019), ascl:1907.024. https://ui.adsabs.harvard.edu/abs/2019ascl.soft07024R ADS Bibcode: 2019ascl.soft07024R.

[42] Manuel Roth, Hartmut Brandt, and Hermann Bischl. 2021. Implementation of a geographical routing scheme for low Earth orbiting satellite constellations using intersatellite links. *Int. J. Satell. Commun. Netw.* 39, 1 (Jan. 2021), 92–107. doi:10.1002/sat.1361 Num Pages: 16 Place: Hoboken Publisher: Wiley Web of Science ID: WOS:000543115000001.

[43] Siva Satya Sri Ganesh Seeram, Luca Feltrin, Mustafa Ozger, Shuai Zhang, and Cicek Cavdar. 2024. Feasibility Study of Function Splits in RAN Architectures with LEO Satellites. doi:10.48550/arXiv.2404.09186 arXiv:2404.09186 [cs].

[44] Keyi Shi, Jingchao Wang, Hongyan Li, and Kan Wang. 2024. Enhancing Resource Utilization of Non-Terrestrial Networks Using Temporal Graph-Based Deterministic Routing. *IEEE Transactions on Vehicular Technology* 73, 6 (2024), 9211–9216. doi:10.1109/TVT.2024.3361969

[45] David B. Shmoys and Éva Tardos. 1993. An approximation algorithm for the generalized assignment problem. *Mathematical Programming* 62, 1-3 (Feb. 1993), 461–474. doi:10.1007/BF01585178

[46] Starlink. 2024. SPACEX SENDS FIRST TEXT MESSAGES VIA ITS NEWLY LAUNCHED DIRECT TO CELL SATELLITES. https://api.starlink.com/public-files/DIRECT_TO_CELL_FIRST_TEXT_UPDATE.pdf

[47] Gregory Stock, Juan A. Fraire, and Holger Hermanns. 2022. Distributed On-Demand Routing for LEO Mega-Constellations: A Starlink Case Study. In *2022 11th Advanced Satellite Multimedia Systems Conference and the 17th Signal Processing for Space Communications Workshop (ASMS/SPSC)*. 1–8. doi:10.1109/ASMS/SPSC55670.2022.9914716 arXiv:2208.02128 [cs].

[48] Alain Sultan. [n. d.]. 5G System Overview. https://www.3gpp.org/technologies/5g-system-overview

[49] Hammas Bin Tanveer, Mike Puchol, Rachee Singh, Antonio Bianchi, and Rishab Nithyanand. 2023. Making Sense of Constellations: Methodologies for Understanding Starlink's Scheduling Algorithms. In *Companion of the 19th International Conference on emerging Networking EXperiments and Technologies (CoNEXT 2023)*. Association for Computing Machinery, New York, NY, USA, 37–43. doi:10.1145/3624354.3630586

[50] Thomas Kohnstamm. 2024. Everything you need to know about Project Kuiper, Amazon's satellite broadband network. https://www.aboutamazon.com/news/innovation-at-amazon/what-is-amazon-project-kuiper

[51] Jiasheng Wu, Shaojie Su, Xiong Wang, Jingjing Zhang, and Yue Gao. 2024. Accelerating Handover in Mobile Satellite Network. arXiv:2403.11502 [cs] doi:10.48550/arXiv.2403.11502

[52] Huiting Yang, Wei Liu, Jiandong Li, and Tony Q. S. Quek. 2023. Space Information Network With Joint Virtual Network Function Deployment and Flow Routing Strategy With QoS Constraints. *IEEE Journal on Selected Areas in Communications* 41, 6 (June 2023), 1737–1756. doi:10.1109/JSAC.2023.3273704 Conference Name: IEEE Journal on Selected Areas in Communications.

[53] Hao Yin, Sumit Roy, and Liu Cao. 2022. Routing and Resource Allocation for IAB Multi-Hop Network in 5G Advanced. *IEEE Transactions on Communications* 70, 10 (Oct. 2022), 6704–6717. doi:10.1109/TCOMM.2022.3200673 Conference Name: IEEE Transactions on Communications.

[54] Jina Yu, Tae-Yoon Kim, and Jae-Hyun Kim. 2023. WIP: Performance Evaluation of Angle-based Handover in LEO-based NTN. In *2023 IEEE 24TH INTERNATIONAL SYMPOSIUM ON A WORLD OF WIRELESS, MOBILE AND MULTIMEDIA NETWORKS, WOWMOM*. IEEE Computer Soc, Los Alamitos, 300–303. doi:10.1109/WoWMoM57956.2023.00045 Num Pages: 4 Series Title: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks Web of Science ID: WOS:001046685900032.

[55] Senbai Zhang, Aijun Liu, Chen Han, Xiang Ding, and Xiaohu Liang. 2021. A Network-Flows-Based Satellite Handover Strategy for LEO Satellite Networks. *IEEE Wirel. Commun. Lett.* 10, 12 (Dec. 2021), 2669–2673. doi:10.1109/LWC.2021.3111680 Num Pages: 5 Place: Piscataway Publisher: Ieee-Inst Electrical Electronics Engineers Inc Web of Science ID: WOS:000728140800015.

[56] Tao Zhang, Jiandong Li, Hongyan Li, Shun Zhang, Peng Wang, and Haiying Shen. 2020. Application of Time-Varying Graph Theory over the Space Information Networks. *IEEE Network* 34, 2 (March 2020), 179–185. doi:10.1109/MNET.001.1900245 Conference Name: IEEE Network.

[57] Yuke Zhou, Jiang Liu, Ran Zhang, Man Ouyang, and Tao Huang. 2023. A Novel Feeder Link Handover Strategy for Backhaul in LEO Satellite Networks. *Sensors* 23, 12 (June 2023), 5448. doi:10.3390/s23125448 Num Pages: 18 Place: Basel Publisher: MDPI Web of Science ID: WOS:001015584100001.

APPENDIX

A Performance Guarantees

A.1 Coarse-Time

Note that in the final step of the coarse-time algorithm we return to the actual demand r_j rather than the adjust demand $\tilde{r}_{ij}$. The capacity constraint was met with $\tilde{r}_{ij}$, and since $\tilde{r}_{ij} \geq r_j$ for all i, and the the solution is still guaranteed to be feasible in terms of r_j. The result from [45] guarantees, that the integral solution after rounding will have a makespan of at most 2× the capacity constraint, i.e. $\max_i \frac{\lambda_0 R_i}{C_i} \leq 2$. This implies that the $\min_i \frac{C_i}{R_i} \geq \frac{1}{2}$, and consequently $\lambda \geq \frac{1}{2}\lambda_0$ ■

A.2 Medium-Time

We present the following greedy algorithm to solve `Partition`: Begin by forming a bipartite graph for the time step t $\mathcal{G}(t) = (S, G, E(t))$ with satellite nodes, S, and gateway nodes G. Each edge, e_{ik}, between GW i and satellite k is weighted according to both the profit, $\tilde{p}_{ik}(t)$, and a cost which is used to ensure the coverage constraint is met. The cost, $c_{ik}(t)$, is the number of currently *uncovered* cells in *other* GWs, which could be covered by satellite k: $c_{ik}(t) = \sum_{i' \neq i} \sum_{j \in \mathcal{U}_{i'}} x_{i'j}(\tau) v_{jk}(t)$, where $\mathcal{U}_{i'}$ is the set of uncovered cells in GW-region i'. The weight of each edge is the $\min\{\tilde{p}_{ik}(t), \frac{\tilde{p}_{ik}(t)}{c_{ik}(t)}\}$, where the min ensures numerical stability. An example graph $\mathcal{G}(t)$ is shown in Fig. 23. GWs are represented by squares at the bottom, and satellites are depicted by small "X"s at the top. For visibility, the edges are quantized into three levels based on the number cells in a GW region covered by a satellite. This is the "profit" used in greedy algorithm. The "cost", i.e. the number of potential number of cells not covered by that satellite, is not explicitly shown.

The algorithm proceeds by choosing the edge, e_{ik} with maximal weight and assigning the corresponding satellite k to GW i. After each selection, remove the chosen satellite k from the graph and recompute the cost component of the remaining edge weights based on the remaining *uncovered* cells. The algorithm continues until all satellites have been assigned.

The above algorithm has a computational complexity of $O(MK)$ where M is the number of GWs and K the number of visible satellites. In § 5 we will show that this allows us to scale to large scenarios efficiently without sacrificing performance. `Partition` being a combination of both an assignment and coverage problem is significantly harder than the two in isolation, hence it is difficult to provide worst-case performance guarantees. However, our greedy algorithm incorporates both the profit and the cost to balance both constraints effectively and produces satellite partitions that contribute to a close-to-optimal performance as we will see in evaluations.

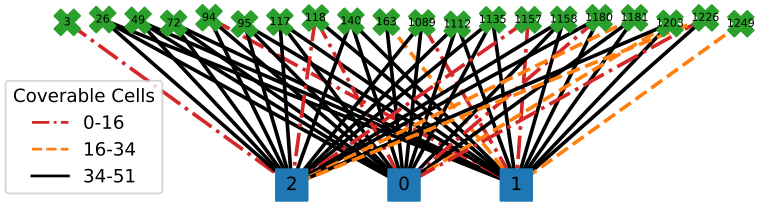


Fig. 23. Example Graph for Partitioning

A.3 Fine-Time

Let maximum service ratio achieved by the optimal two-stage assignment be μ^*. Let the service ratio obtained by the LP-relaxed `Service` problem be $\hat{\mu}_S$; this is obviously greater than the optimal

solution: $\mu^* \leq \hat{\mu}_S$. Using the same logic as in Appendix A.1, the service ratio following rounding, μ_S is no less than half the LP solution: $\frac{1}{2}\hat{\mu}_S \leq \mu_S \leq \hat{\mu}_S$.

This service ratio μ_S must serve as an upper bound on the search for a feasible solution to `Feeder`, yielding $\hat{\mu}_F \leq \mu_S$. Once again we apply the half-approximation on the rounding of the LP-relaxed solution to `Feeder`: $\frac{1}{2}\hat{\mu}_F \leq \mu_F \leq \hat{\mu}_F$.

If μ_S is a feasible service ratio for the LP-relaxed `Feeder` problem, then we get $\mu_F \geq \frac{1}{2}\hat{\mu}_F = \frac{1}{2}\mu_S \geq \frac{1}{4}\mu^*$. If there is no feasible solution, then $\mu^* \leq \mu_F$, and we get $\mu_F \geq \frac{1}{2}\hat{\mu}_F \geq \frac{1}{2}\mu^* \geq \frac{1}{4}\mu^*$ ■

Discussion: Whether or not we achieve the tighter guarantee of $\frac{1}{2}\mu^*$ depends on where the bottleneck capacity lies within the two stage assignment. Because of the unique assignment constraint, there is a many-to-one relationship between service and feeder satellites, making it likely that the bottleneck is on the feeder side (the case in all our chosen scenarios). In that case the optimal service ratio depends on the feeder capacity, and our result guarantees the half-approximation. There is a possibility, perhaps due to asymmetrical hardware for uplink/downlink on-board the satellite that the total feeder capacity is much greater than service capacity. In this case the $\frac{1}{4}$ guarantee applies.

B Global Optimization Formulation

Let us consider a global, end-to-end, optimization of the network utilization. We will use a similar minimum makespan formulation as in § 3.2, but because we have to assign flows through each layer of the network we also have a form of a multi-dimensional assignment problem [38] which is also NP-Hard. The formulation for this problem is as follows: At some time t we have a graph like the one in Fig. 24. Each cell u_j generates some demand $r_j(t)$. The task is to find an assignment, i.e. a vector $\mathbf{x}(t)$, which selects edges in the graph creating flows from each cell, through the satellite network, to a GW. Our load balancing objective is to minimize the maximum load on any node (service satellite, feeder satellite, or GW). We can do this by formulating a problem that constrains the load on each node to be less than some variable L which we then minimize

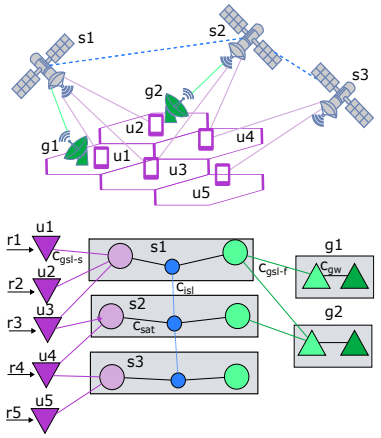


Fig. 24. Scenario and associated flow graph.

$$
\begin{aligned}
\min_{x} \quad & L \\
\text{s.t.} \quad & \sum_f \sum_s \sum_j x_{ifsj}(t) c_{ifsj}(t) \leq \eta_i L \, \forall i, \\
& \sum_i \sum_s \sum_j x_{ifsj}(t) c_{ifsj}(t) \leq \xi_f L \forall f, \\
& \sum_i \sum_f \sum_j x_{ifsj}(t) c_{ifsj}(t) \leq \epsilon_s L \, \forall s, \\
& \sum_i \sum_f \sum_k x_{ifsj}(t) = 1 \quad \forall j
\end{aligned}
$$

Where we have introduced the terms η, ξ, and ϵ as scaling factors to account for the different capacities. The cost parameter is inversely proportional to the visibility-adjust demand, similar to § 3.2, $c_{ifsj}(t) \propto \frac{r_j(t)}{v_{sj}(t) w_{fi}(t)}$, where $v_{sj}(t) \propto \sin\theta_{sj}(t)$ if $\theta_{sj} \geq \theta_{\min}$ and 0 otherwise, and similarly for w. Being multidimensional, this problem is harder than the traditional makespan problem. Indeed, the technique used throughout this paper (based on [45]) for the typical two-dimensional matching problem relies on finding a matching in the bipartite graph formed from the solution of the LP-relaxed problem. If we wanted to use a similar strategy for the multidimensional version, we would then need to find multidimensional matching, which is itself a

NP-Complete problem [26]. In practice, we have used the commercial solver Gurobi to optimize the above problem for various problem sizes. To optimize end-to-end the se scenario with only 15 GWs, Gurobi takes nearly 20s. For a slightly larger problem with only 600 cells and 20 GWs, Gurobi takes 609 s to bound the solution to within 2% of the optimal value. The number of nodes scales exponentially with the number of cells and GWs as each additional cell or GW increases the problem space by the number of edges it adds to the graph.

This formulation, despite producing an optimal load balanced solution, does not account for multiple KPIs, nor does it meet some of the NTN- and architecture-specific constraints. For example, the only level of the problem required to be a one-to-one matching is the users to service satellites. A single service satellite may be assigned to multiple feeder satellites, likewise a single feeder to multiple GWs. This may be feasible for certain satellite network or NTN architectures, but not when utilizing HARMONI's split architecture. Because each GW is a CU-DU, each satellite is a DU, and they are connected via IAB, each service satellite, may only connect to one upstream feeder satellite, and GW.

Instead we consider the following constrained global optimization. Let y_{kj}, z_{lk} and x_{il} be the assignment variables between cell j, service satellite k, feeder satellite l and GW i. The formulation would be:

$$\begin{aligned} \max_{\lambda, x, y, z} \quad & \lambda \sum_i \sum_l x_{il} \sum_k z_{lk} \sum_j y_{kj} r_j \\ \text{s.t.} \quad & \sum_k y_{kj} = 1 \ \forall j, \quad \sum_l z_{lk} = 1 \ \forall k, \quad \sum_i x_{il} = 1 \ \forall l, \\ & q_k = \lambda \sum_j y_{kj} r_j \le C_S \ \forall k, \quad \rho_l = \sum_k z_{lk} q_k \le C_F \ \forall l, \quad \sum_l x_{il} \rho_l \le C_i \ \forall i, \end{aligned} \tag{5}$$

This formulation is still complicated (considering four variables) and still fails to address multiple KPIs. As such we consider the following "quasi" global optimization. We eliminate the service-feeder optimization by assigning the service satellite to the feeder that minimizes the ISL cost as defined in § 3.2.3; let this be $\overline{z}_{lk}$. Then we optimize the cell-service assignment, subject to the feeder capacity via Problem 6. The service ratio λ is found via binary search as in HARMONI. Finally, we assign feeder satellites (and their assigned demand ρ_l) to GWs with the service ratio μ via solving Problem 7.

$$\begin{aligned} \max_{y} \quad & \sum_k \sum_j y_{kj} r_j \\ \text{s.t.} \quad & \sum_k y_{kj} = 1 \quad \forall j, \\ & q_k = \lambda \sum_j y_{kj} r_j \le C_S \forall k, \\ & \sum_k \overline{z}_{lk} q_k \le C_F \forall l \end{aligned} \tag{6}$$

$$\begin{aligned} \max_{x} \quad & \mu \sum_i \sum_l x_{il} \rho_l \\ \text{s.t.} \quad & \sum_i x_{il} = 1 \quad \forall l, \\ & \mu \sum_l x_{il} \rho_l \le C_i \forall i \end{aligned} \tag{7}$$

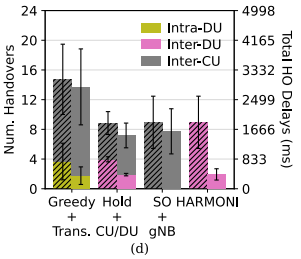


Fig. 25. Total HO

While not equivalent to the true end-to-end optimization of Eq. 5, this two-stage approach performs the global cell to satellite optimization which is missing from HARMONI due to its partitioning step, giving it more flexibility in maximizing network utilization. However, the complexity of such a large optimization does not scale well to the large numbers of satellites and cells, in contrast with HARMONI.

C Additional Results

Accummulated QoE Impact While the architecture determines the *quality* of the HO, the session orchestration algorithm determines their *frequency*. Combined we can quantify the *accumulated* impact of HOs over time. Consider Fig. 25 which shows both the total number of each type of handover event (intra-du, inter-du, inter-cu) an average cell experiences using various systems (left) as well as the total delay caused by those HOs (right) over 10 minutes. We consider three other NTN systems: `trans+greedy`, `split+hold`, and `gnb+SO`, where SO indicates the session orchestration used in HARMONI. By using HARMONI, users experience fewer HOs of all types which result in far less disruption—less than half the total delay compared to other systems—leading to more stable, higher QoE sessions.

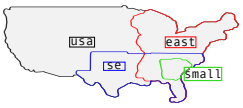


Fig. 26. Scenario Maps

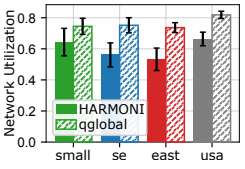


Fig. 27. Netw. Util.

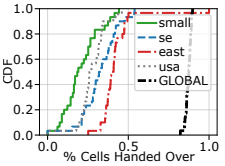


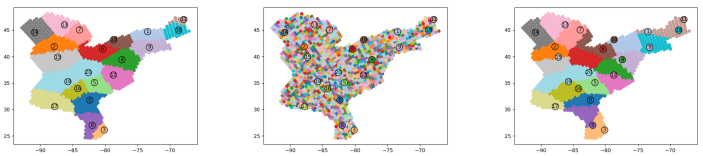
Fig. 28. Cell-Sat Switch

Algorithm Results vs Scenario: Scenario maps are shown in Fig. 26. In this section we compare HARMONI's multi KPI performance across each scenario.

Fig. 27 compares the average network utilization over time of HARMONI, with parameters $(\alpha, \beta, \gamma, \kappa) = (2, 1, 2, 0.01)$, with that of the `global` optimization in all four scenarios. We are within 10% network utilization to the global optimization across all scenario sizes, demonstrating HARMONI's suitability for varied deployments.

The CDF of the percentage of cells which switch satellites each time step is shown in Fig. 28. As with network utilization, HARMONI's performance is consistent across scenarios. At 85% of the time, fewer than half of the cells are switched to a new satellite. While this is more churn than local algorithms like `greedy` or `hold`, HARMONI vastly outperforms those in terms of network utilization, and it is still much less than the switches caused by `global` (an example CDF from the `se` scenario is also shown in Fig. 28) which switches nearly every cell every 20 seconds for marginal network utilization gains. This further illustrates HARMONI's ability to balance multiple KPIs and to do it across different NTN deployments.

Fig. 29 shows three different cell-GW assignment maps for the `east` scenario. The first (Fig. 29a) assigns each cell to the closest GW. Notice how the formed GW-regions contain very different numbers of cells, which indicates poor load balancing. The second (Fig. 29b) is an assignment based only on cell demand, equivalent to setting $\alpha = 0$. In this case, no GW regions are formed at all, which prevents us from performing satellite partitions as there is no similarity in the satellite visibility between cells assigned to the same GW. Finally, the last figure (Fig. 29c) gives the assignment produced by HARMONI which forms GW regions which are both coherent (i.e. low cell-GW distance) and even (i.e. proper load balancing).



(a) Distance Only (b) Demand Only (c) HARMONI

Fig. 29. GW Anchor Assignment Maps for `east`

Received June 2025; accepted September 2025