PhD-FSTM-2025-139

The Faculty of Science, Technology and Medicine

DISSERTATION

Defence held on 28 November 2025 in Luxembourg

to obtain the degree of

DOCTEUR DE L'UNIVERSITE DU LUXEMBOURG
EN INFORMATIQUE

by

Sovit BHANDARI
Born on 22 March 1993 in Kailali (Nepal)

INTELLIGENT RESOURCE MANAGEMENT AND SERVICE OPTIMIZATION IN FULLY REGENERATIVE NON-TERRESTRIAL NETWORKS

Dissertation defense committee

Dr. Eva Lagunas, Chair
Assistant Professor, University of Luxembourg

Dr. Thang Xuan Vu, Dissertation Supervisor
Research Scientist, University of Luxembourg

Dr. Symeon Chatzinotas, Member
Professor, University of Luxembourg

Dr. Shashi Raj Pandey, Member
Assistant Professor, Aalborg University

Dr. Zheng Chang, Member
Assistant Professor, University of Jyväskylä

Intelligent Resource Management and Service Optimization in Fully Regenerative Non-Terrestrial Networks

Sovit BHANDARI

"You have a right to perform your prescribed duties,

but you are not entitled to the fruits of your actions!"

– *Translated from BHAGAVAD GITA (2.47)*

(or ancient Reinforcement Learning??)

Abstract

Recent advancements in satellite launch technologies and phased array antennas have enabled large constellations of spaceborne non-terrestrial networks (NTNs), such as medium Earth orbit (MEO) and low Earth orbit (LEO) systems, to provide global broadband and direct-to-cell connectivity, complementing terrestrial network infrastructure. However, current NTN payloads are mostly (digitally) transparent, functioning as bent-pipe relays with limited channelization, which restricts their ability to dynamically optimize scarce communication and computational resources (e.g., power, bandwidth, and computing capabilities). This, in turn, has led to inefficient satellite spectrum utilization, increased reliance on ground stations, and challenges in maintaining ubiquitous connectivity during frequent satellite handovers (HOs).

To achieve seamless connectivity and meet the key performance indicators (KPIs) envisioned by the International Mobile Telecommunications (IMT)-2030 framework for sixth-generation (6G) networks, this thesis investigates next-generation non-terrestrial network satellite architectures. These architectures integrate full gNodeB (gNB) functionality along with on-board caching to support user-centric intelligent services that adapt dynamically to changing user demands by fully exploiting the available communication and computational resources. They are designed to be fully reconfigurable and incorporate advanced on-board interference mitigation techniques enabled by flexible bandwidth resource management strategies, as well as inter-satellite link (ISL) connectivity to support frequent HOs without reliance on ground stations. This advanced regenerative architectural (fully digital) approach addresses many of the inherent challenges of NTNs, particularly those associated with their non-geostationary nature, and aligns with ongoing standardization efforts led by key telecommunications regulatory bodies such as the 3rd Generation Partnership Project (3GPP).

Building on this foundation, the thesis makes three main contributions to fully leverage the capabilities of such systems: (i) the design of the flexible resource management algorithm for LEO satellites (FLARE-LEO), which efficiently optimizes satellite radio resources for both HO and non-HO scenarios, enabling the delivery of the user-centric media contents in the soonest manner; (ii) the development of an on-demand, flexible Satellite-as-a-Service (SaaS) framework, capable of performing and delivering critical Earth observation (EO) tasks in near-real-time through on-board computation within space-air-ground integrated network (SAGIN) nodes under diverse beam management schemes; and (iii) the integration of IMT-envisioned advanced capabilities, including integrated sensing and communication (ISAC) and artificial intelligence (AI), to enable intelligent and seamless HOs, ensuring uninterrupted connectivity.

Firstly, FLARE-LEO is proposed to jointly optimize bandwidth, power, and spot beam coverage based on realistic geographic user distributions. It employs unsupervised K-means clustering followed by a successive convex approximation (SCA)-based iterative algorithm. The design also supports multi-spot beam multicasting, spatial multiplexing, caching, and HO-aware transmission. Two joint transmission architectures are proposed for HO periods, leveraging deep learning (DL) for downlink channel state information (CSI) prediction. The simulation results demonstrate a significant reduction in delivery time and robustness under imperfect CSI.

Secondly, a three-tier SaaS framework comprising a serving LEO satellite (first-tier), adjacent intra- and inter-orbit LEO satellites (second-tier), and a cloud-aided gateway (third-tier) to collaboratively perform on-demand EO tasks is formulated to minimize worst-case service completion time through optimized computation and communication allocation. The resulting non-linear, non-convex, non-deterministic polynomial-time (NP)-hard problem is addressed via an SCA-based iterative algorithm. Additionally, HO-aware delivery strategies are proposed for both Earth-fixed and Earth-moving beam configurations. Simulations indicate a noticeable improvement in task completion time over reference strategies, with robust performance under imperfect CSI.

Lastly, a novel make-before-break HO mechanism for utilizing ISAC at ground terminals (GTs) is designed. This novel design generates realistic three-dimensional (3D) beampatterns, enabling GTs to sense approaching LEO satellites while maintaining communication with the currently serving LEO satellite. The design problem is highly challenging due to its non-convexity and mixed-integer nature, which is solved using Riemannian manifold optimization and closed-form solutions. For multi-GT scenarios, a multi-agent deep reinforcement learning (MADRL) framework is introduced to mitigate sensing collisions and ensure quality-of-service under shared spectrum constraints. Numerical results validate improved communication–sensing trade-offs, smooth HO performance, and robustness under imperfect CSI.

Preface

This Ph.D. thesis was conducted between June 2022 and November 2025 at the Interdisciplinary Centre for Security, Reliability, and Trust (SnT) at the University of Luxembourg. The research was supervised by Dr. Thang Xuan Vu of the University of Luxembourg, Luxembourg. Periodic evaluations were carried out by the CET members: Prof. Symeon Chatzinotas and Dr. Thang Xuan Vu (both from the University of Luxembourg, Luxembourg), and Prof. Zheng Chang (University of Jyväskylä, Finland). Additionally, the thesis was completed in partnership with Société Européenne des Satellites (SES), Luxembourg, represented by Mr. Tom Christophory.

Support of the Thesis

This Ph.D. thesis was supported by the Luxembourg National Research Fund (FNR) through the project INSTRUCT (ref. IPBG19/14016225/INSTRUCT) and the project RUTINE (ref. C22/IS/17220888/RUTINE).

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my supervisor, Dr. Thang Xuan Vu, for his invaluable guidance, insightful feedback, and for providing countless opportunities for my professional growth. Without his persistent supervision and unwavering support, the completion of this thesis would not have been possible. I am also equally grateful to Prof. Symeon Chatzinotas for his excellent technical advice and timely constructive feedback throughout my PhD journey. My sincere thanks also go to Prof. Zheng Chang from the University of Jyväskylä for his insightful technical suggestions during each CET meeting, which greatly helped me to refine my research findings. Furthermore, I would like to thank Prof. Björn Ottersten for reviewing one of my research papers during the initial stages of my doctoral studies and for providing valuable feedback that helped me improve it significantly.

I also extend my gratitude to the FNR and SES S.A. for funding my research and offering internship opportunities as part of the PPP model. These experiences provided me with valuable insight into real-world industrial practices and helped me navigate my research direction by bridging industry and academia. I am equally thankful to my internship supervisors for their guidance during this period.

Last but not least, I owe my loving thanks to my family and friends for always inspiring and motivating me through thick and thin. The completion of this work would not have been possible without the help of all of them.

Sovit BHANDARI

Luxembourg, Dec. 2025

Contents

List of Figures

List of Tables

Abbreviations

2D	Two Dimensional
3D	Three Dimensional
3GPP	3rd Generation Partnership Project
4G	Fourth-Generation
5G	Fifth-Generation
6G	Sixth-Generation
AI	Artificial Intelligence
AO	Alternating Optimization
API	Application Programming Interface
AR	Augmented Reality
AUG	Associate User Group
AWGN	Additive White Gaussian Noise
B5G	Beyond Fifth-Generation
BH	Beam-Hopping
BN	Batch Normalization
BnB	Branch-and-Bound
bps	Bits Per Second
CapEx	Capital Expenditure
c.u.	Channel Uses
CC	Centralized Collaboration
CCI	Congestion Control Information
CDN	Content Delivery Network

CMAF	Common Media Application Format
CNN	Convolutional Neural Network
CNN-CSI	CNN-based Downlink CSI
CRM	Computational Resource Management
CSI	Channel State Information
CU	Central Unit
D2C	Direct-to-Cell
DASH	Dynamic Adaptive Streaming over HTTP
DC	Distributed Collaboration
DL	Deep Learning
DNN	Deep Neural Network
DNS	Domain Name System
DPort	Destination Port
DQN	Deep Q Network
DRL	Deep Reinforcement Learning
DSAC	Dedicated Sensing and Communication
DST	Destination IP
DTH	Direct-To-Home
DTP	Digital Transparent Payload
DU	Distributed Unit
DVB-I	DVB-Internet
DVB-NIP	Digital Video Broadcasting - Native IP
DVB-S	Digital Video Broadcasting Satellite
DVB-S2	Second Generation of DVB-S
DVB-S2X	Extended Version of DVB-S2
EBC	Everest Base Camp
EO	Earth Observation
ESI	Encoding Symbol ID
ETSI	European Telecommunications Standards Institute
FCNN	Fully Connected Neural Network
FDD	Frequency-Division Duplexing
FFR	Full-Frequency Reuse

FLARE-LEO	Flexible Resource Management Algorithm for LEO Satellites
FLUTE	File Delivery over Unidirectional Transport
FoV	Field of View
FR1-NTN	Frequency Range 1 NTN
GEO	Geostationary Earth Orbit
gNB	gNodeB
GSO	Geostationary Orbit
GT	Ground Terminal
GWs	Gateways
HAPs	High Altitude Platforms
HLS	HTTP Live Streaming
HO	Handover
HPBW	Half-Power Beamwidth
HTS	High Throughput Satellite
IEEE	Institute of Electrical and Electronics Engineers
IMT	International Mobile Telecommunications
IoT	Internet of Things
IP	Internet Protocol
ISAC	Integrated Sensing and Communication
ISAC-HO	ISAC-assisted HO
ISLs	Inter-Satellite Links
ITU	International Telecommunication Union
KPI	Key Performance Indicators
LCT	Layered Coding Transport
LEO	Low Earth Orbit
LHS	Left Hand Side
LoS	Line of Sight
LP	Linear Programming
LTE	Long Term Evolution
mABR	Multicast Adaptive Bitrate
MADRL	Multi-Agent Deep Reinforcement Learning
MEC	Mobile Edge Computing

MEO	Medium Earth Orbit
MILP	Mixed-Integer Linear Programming
MIMO	Multiple-Input Multiple-Output
MINLP	Mixed-Integer Non-Linear Programming
MPC	Most Popular Caching
MSE	Mean Squared Error
MU	Multi-User
NGSO	Non-Geostationary Orbit
NLoS	Non-Line of Sight
NP	Non-Deterministic Polynomial Time
NTN	Non-Terrestrial Network
OFDM	Orthogonal Frequency-Division Multiplexing
OpEx	Operating Expenditure
OTT	Over-The-Top
PCAP	Packet Capture
Plen	Payload Length
QoS	Quality of Service
R&D	Research and Development
RC	Random Caching
ReLU	Rectified Linear Unit
RF	Radio Frequency
RHS	Right Hand Side
RL	Reinforcement Learning
RRM	Radio Resource Management
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
SaaS	Satellite-as-a-Service
SAGIN	Space-Air-Ground Integrated Networks
SAR	Synthetic Aperture Radar
SBN	Source Block Number
SCA	Successive Convex Approximation
SE	Spectral Efficiency

SES	Société Européenne des Satellites
SINR	Signal-to-Interference-plus-Noise Ratio
SNR	Signal-to-Noise Ratio
SOTA	State-of-the-Art
Sport	Source Port
SRC	Source IP
SU	Single-User
SVD	Singular Value Decomposition
Tanh	Hyperbolic Tangent
TDD	Time-Division Duplexing
TD-SAC	Time-Division Sensing and Communication
TN	Terrestrial Network
TOI	Transport Object Identifier
TS	Timestamp
TSI	Transport Session Identifier
TV	Television
UAVs	Unmanned Aerial Vehicles
UC	Uniform Caching
UDP	User Datagram Protocol
UE	User Equipment
UL	Uplink
UPAs	Uniform Planar Arrays
VoD	Video on Demand
VR	Virtual Reality
VSAT	Very Small Aperture Terminal
ZF	Zero-Forcing

Notations

$(\cdot)^H$	The Hermitian (conjugate) transpose
$\mathrm{Diag}(\cdot)$	It forms a diagonal matrix from a vector
$\mathrm{diag}(\cdot)$	It extracts the diagonal elements of a matrix
$\det(\cdot)$	The determinant of a matrix
$\mathbb{E}[\cdot]$	The expectation operator
$\mathbb{I}(\cdot)$	The indicator function
$\mathrm{tr}(\cdot)$	The trace of a matrix
$\lceil\cdot\rceil$	The ceiling operator
$\log_x(\cdot)$	The logarithm with base x
$\lvert\cdot\rvert$	The absolute value (for scalars) or set cardinality (for sets)
$\lvert\lvert\cdot\rvert\rvert$	The ℓ_2-norm (Euclidean norm) of a vector or matrix
max	The maximum operator
min	The minimum operator
$\odot$	The Hadamard (element-wise) product
$\boldsymbol{Z}$	The bold uppercase letters denote matrices
$\boldsymbol{z}$	The bold lowercase letters denote vectors
z or Z	The non-bold letters denote scalars

Chapter 1

Introduction

1.1 Background and Motivation

With the unveiling of the iPhone 14's SOS emergency text message service via satellite on September 16, 2022 [1], the space industry has undergone significant disruption, driven by efforts to connect the world to the Internet via low Earth orbit (LEO) satellites. These efforts aim to address the unprecedented demand for mobile data traffic and to support the vision of a 'Global Village' with 100% connectivity [2]. Companies such as SpaceX, Eutelsat OneWeb, Amazon's Project Kuiper, Telesat, and Shanghai Spacecom Satellite Technology are competing to deploy large constellations of LEO satellites [3], with the aim of complementing terrestrial networks (TNs) by providing global broadband and direct-to-cell (D2C) services through their large footprints.

The latest industrial advancements in this domain include the ongoing deployment of SpaceX's second-generation Starlink satellites (V2 Mini), which began with their first launch in Feb. 2023. These satellites offer up to four times the capacity of their predecessors, enable D2C connectivity, and come equipped with advanced technologies like optical inter-satellite links (ISLs), phased-array beamforming, digital signal processing, and on-board eN-odeB modems [4]. These features facilitate the provision of Long-Term Evolution (LTE) and fourth-generation (4G) services, including messaging, data, and video calls on standard mobile devices. Demonstrations showcasing these service features have been carried out in collaboration with terrestrial telecom operators such as T-Mobile and Entel. The design of Starlink V2 Mini makes use of steerable high-gain beams and optical ISLs to improve satellite coordination, enhance spectrum efficiency, and ensure compliance with International

Telecommunication Union (ITU) regulations, while also remaining compatible with both terrestrial and non-terrestrial systems [5].

Similarly, OneWeb, in collaboration with the European Space Agency and the UK Space Agency Sunrise Programme, deployed JoeySat in May 2023 to demonstrate beam-hopping (BH) and on-board processing technologies intended for its upcoming satellite constellation [6]. Although JoeySat does not currently facilitate ISLs due to the company's decision in Jul. 2018, driven by existing regulatory constraints, it relies on more than 40 ground-based gateways (GWs) worldwide for global service, each capable of handling satellite connections up to 4,000 kilometers (slant distance). This ground-based reliance causes latency variation and limits seamless handovers (HOs). Therefore, alternative approaches must be considered to achieve seamless HOs with the assistance of ground terminals (GTs).

Meanwhile, the European Union is set to launch its second-generation Galileo satellites (Galileo G2), with initial launches scheduled from 2026 onward. These advanced versions of Galileo satellites will fully feature digital navigation payloads, electric propulsion, improved antennas, ISLs, and advanced atomic clocks to deliver more robust global positioning, navigation, and timing services [7].

Thanks to SpaceX's partially reusable Falcon launch system, the cost of orbital launches has dropped significantly [8]. This reduction has enabled the deployment of payloads capable of performing orbital edge computing, substantially enhancing the computational power of non-terrestrial network (NTN) satellites, accelerating progress toward beyond fifth-generation (B5G) and sixth-generation (6G) objectives such as seamless global connectivity and support for high-demand applications across geographical barriers.

To support these advancements, fully regenerative payload architectures are preferred over transparent payload architectures for orbital edge computing. The key reason is that a fully regenerative payload supports on-board baseband processing, allowing the NTNs to implement complete gNodeB (gNB) functionalities such as modulation/demodulation, encoding/decoding, switching, routing, etc., directly on the platform. In contrast, a transparent payload architecture simply functions as a bent-pipe relay, capable only of frequency translation and power amplification for uplink and downlink signals, without any signal regeneration or intelligent processing capabilities [9]. Moreover, this architectural shift highlighted in Release (Rel.) 16 of 3rd Generation Partnership Project (3GPP) [9], is well aligned with the International Mobile Telecommunications (IMT)-2030 vision [10, 11]. As illustrated in Figure 1.1, this vision outlines the foundation for 6G, which aims to support three new transformative capabilities: ubiquitous connectivity, integrated artificial intelligence (AI) and communications, and integrated sensing and communication (ISAC). To realize these capabilities, 6G introduces enhanced key performance indicators (KPIs). For e.g., the user-experienced data rate is expected to reach 1 Gbps, and spectral efficiency (SE) is targeted to improve sixfold.

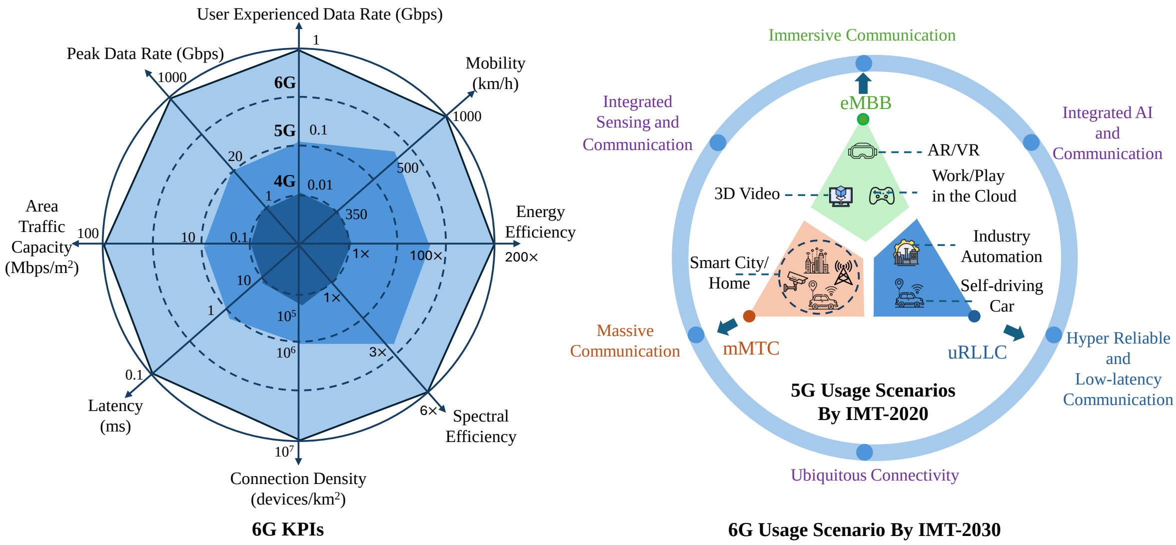

Figure 1.1: The evolution of KPIs and usage scenarios of 6G [10, 11].

In this thesis, we focus on achieving ubiquitous connectivity while focusing on several key 6G KPIs, including SE, user-experienced data rate, and latency, as outlined by IMT-2030. We prioritize fully regenerative payload-enabled spaceborne NTNs, which include geostationary Earth orbit (GEO), medium Earth orbit (MEO), and LEO satellites, over airborne NTNs such as unmanned aerial vehicles (UAVs) and high-altitude platforms (HAPs), due to their orbital stability and global coverage capabilities. Moreover, refueling airborne NTN satellites poses significant operational challenges and inefficiencies while maintaining continuous user connectivity [12].

These considerations are crucial for proposing next-generation satellite system architectures that can support intelligent, autonomous operations. In the following section, we explore the technical evolution ranging from conventional bent-pipe systems to fully reconfigurable next-generation satellite system architectures, and examine how this shift is being shaped by ongoing standardization efforts such as those led by Digital Video Broadcasting (DVB) and 3GPP.

1.2 Evolution of SatCom in Line with the Standardization Perspective

This section presents the essential technical background for the work in this thesis on on-board communication and computation, emphasizing the architectural shift from conventional satellite systems to the proposed next-generation satellite architecture (see Figure 1.2), which is aligned with ongoing standardization efforts.

Conventional single-beam satellite systems typically operate as bent-pipe relays, where the on-board transparent payload is limited to basic radio frequency (RF) operations such as uplink signal filtering, frequency conversion, and power amplification prior to downlink transmission [9]. The Digital Video Broadcasting – Satellite (DVB-S) standard, created and published by the European Telecommunications Standards Institute (ETSI) in 1995 [13], governs the widespread usage of these systems for direct-to-home (DTH) television services.

To meet the growing demand for satellite services, modern high-throughput satellite (HTS) systems have advanced beyond conventional single-beam architectures by significantly enhancing capacity and SE. In contrast to traditional satellites, HTS uses multiple tightly focused beams, frequency-color reuse, and BH techniques to optimize spectrum utilization and reduce interference [14]. These advancements are made possible through on-board digital transparent payloads (DTP) and phased array antennas, which enable restricted flexible channelization (not fully flexible), switching, and routing capabilities tailored for broadcast and multicast services [15].

The DVB Project has played a crucial role in the standardization of HTS technologies, particularly through the development of DVB-S2 (the second generation of the DVB-S standard) and DVB-S2X (an enhanced extension released in 2014), which introduce several key improvements such as higher modulation orders, finer modulation and coding schemes, superframes for backhaul, and enhanced operational modes tailored to diverse applications and environments [16].

Despite these advancements, the capabilities of HTS still remain constrained due to limited channelization and only rudimentary power control/sharing among carriers. Additionally, frequency reuse strategies based on color reuse scheme become increasingly complex with the growth of spot beams and satellites in NTNs, requiring careful planning. More importantly, these systems lack the ability to dynamically reconfigure beam patterns or autonomously adapt to changing traffic demands, highlighting the need for software-defined, intelligent satellite architectures.

This ongoing evolution, from single-beam to HTS to intelligent, reconfigurable next-generation satellites, is illustrated in Figure 1.2, which highlights the progressive enhancement in on-board functionality, flexibility, and service capability.

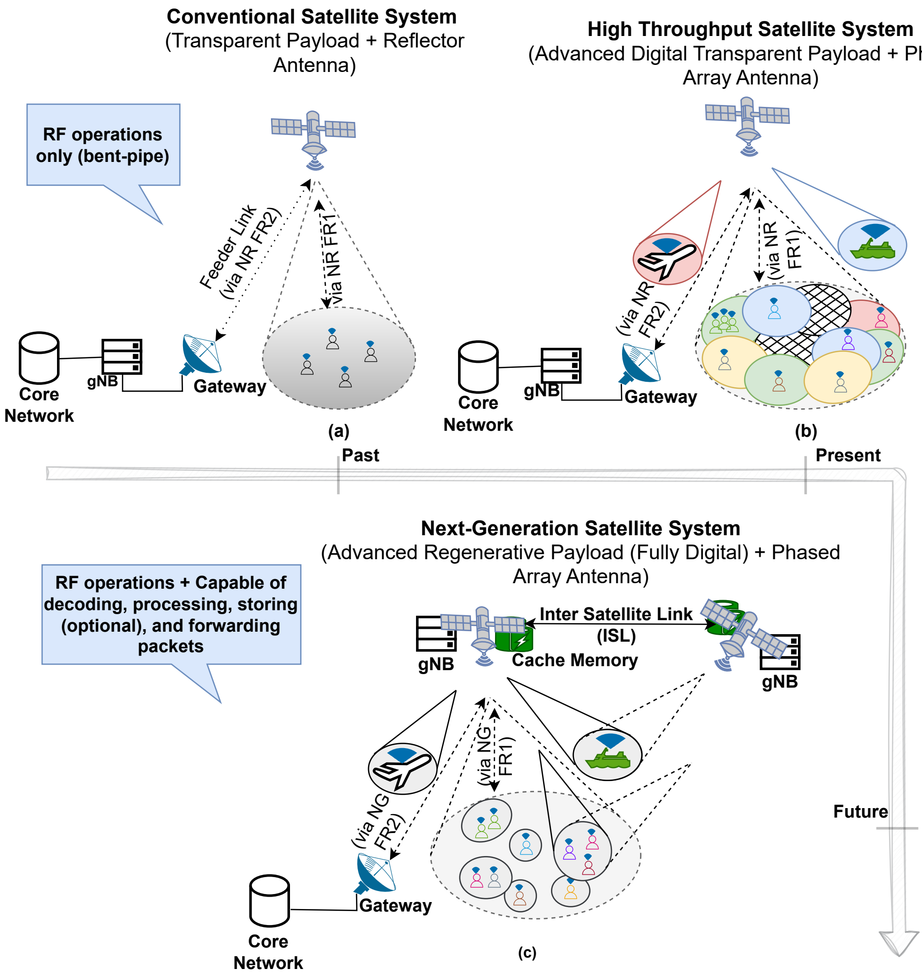

Figure 1.2: Evolution of SatCom from single-beam generating conventional satellites to next-generation satellites capable of producing multiple real-time adaptive beams.

In recent years, to support emerging services such as on-demand broadband and direct-to-cell connectivity, satellite operators have been exploring techniques to deliver performance to complement TNs. One key approach is the adoption of full-frequency reuse (FFR) to maximize SE. However, FFR introduces significant interference, which makes signal decoding at GTs more difficult. Mitigating this interference effectively requires on-board analysis of incoming data packets. This, in turn, demands gNB-like functionality that includes decoding, processing, and forwarding, directly on the satellite. From a standardization perspective, this marks a shift from transparent to regenerative payloads and enables intelligent on-board processing.

Researchers are also exploring ways to split the full gNB functionality on-board NTNs into Central Unit (CU) and Distributed Unit (DU) components, to reduce the high capital expenditures (CapEx) and operating expenditures (OpEx) associated with payload launch costs and high on-board power consumption. This architectural shift has been under discussion within the NTN community since the 3GPP Rel. 16 meetings [17]. The CU typically manages higher-layer protocols such as Radio Resource Control, Packet Data Convergence Protocol, and Service Data Adaptation Protocol, and supports functions including network slicing, connection management, and mobility management. In contrast, the DU may optionally host both the lower and higher physical layer functionalities, or only the higher physical layer, along with Medium Access Control and Radio Link Control protocols responsible for data segmentation and integration, scheduling, and multiplexing [18].

Initially, leading companies working in this domain focused primarily on minimizing payload complexity, often overlooking other aspects of the communication system. When 3GPP initiated Rel. 19 work package in Jan. 2023 to guide the transition from fifth-generation (5G) to 6G, the CU-DU functional split architecture gained strong support. Inspired by the next-generation radio access network split gNB model, which was introduced in 3GPP Rel. 15, this approach appeared to reduce on-board complexity. As a result, placing the gNB-DU on the satellite and the gNB-CU on the ground became the preferred configuration. This choice was primarily driven by payload cost and power constraints, rather than in-depth analysis of key communication performance metrics such as SE, which are significantly affected by signaling overheads. Ongoing discussions surrounding this architecture are expected to continue into 3GPP Rel. 20, which will also incorporate 6G studies as part of 3GPP's standardization roadmap (see Figure 1.3) [17].

However, relying solely on the gNB-DU on-board does not inherently support feeder link switching or store-and-forward operations, which limits system flexibility. This configuration

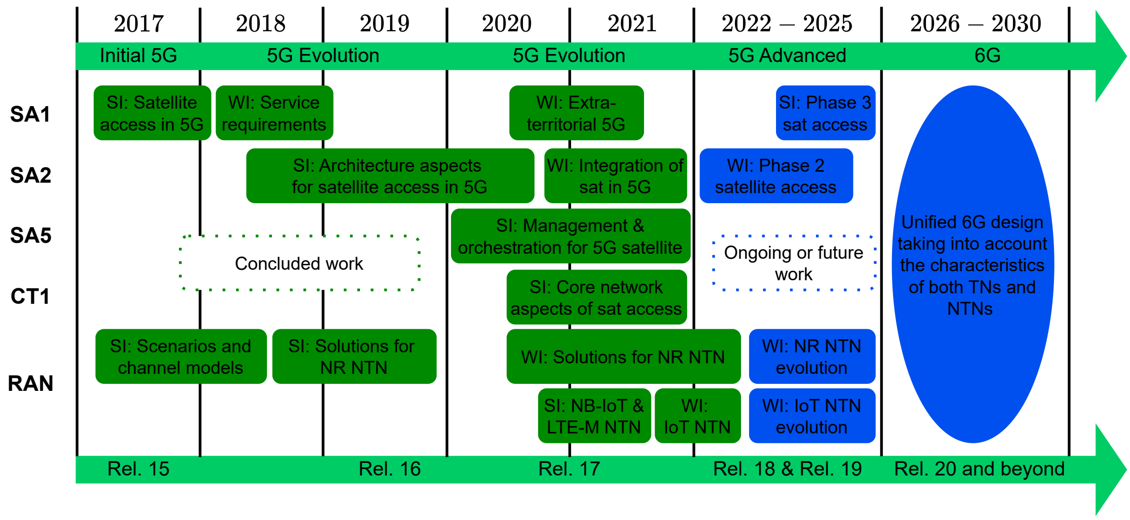

Figure 1.3: Evolution roadmap of 3GPP standards toward 6G [17].

also lacks support for standardized direct inter-satellite interfaces, hindering efficient inter-satellite mobility. Furthermore, signaling over the F1 interface (the interface that connects a gNB CU to the gNB DU) on the feeder link would still be required, and to a much greater extent than with a full gNB on-board. Thus, to enable the next-generation satellite system to meet the KPIs envisioned by IMT-2030 for 6G, this thesis considers the implementation of a full gNB on-board.

The next-generation satellite system illustrated in Figure 1.2(c) is designed to integrate storage capabilities with full gNB functionalities, aiming to mitigate the substantial backhaul latency typically associated with NTNs through the caching techniques. This architecture is particularly relevant for satellite operators such as Société Européenne des Satellites (SES), which is developing the Digital Video Broadcasting - Native IP (DVB-NIP) infrastructure to deliver video-on-demand (VoD) and over-the-top (OTT) content via DVB-S2/DVB-S2X links in response to Internet Protocol (IP)-based user requests [19]. Furthermore, the system is capable of supporting dynamic beamforming with variable spot beam sizes, enabling intelligent adaptation to user demand based on geographical distribution, and facilitating both broadcast and multicast services. Using this next-generation satellite system, an FFR scheme can be employed to fully exploit the available system bandwidth, and the interference introduced by FFR can be effectively mitigated through advanced precoding strategies, which will be further discussed in Chapter 3. The architecture also provides flexibility in power allocation per spot beam, supported by a software-defined regenerative processor, which can be leveraged to improve the quality-of-service (QoS) for marginal or weak users. In addition, ISLs can be fully leveraged to enable cooperation amongst NTN satellites within the same or adjacent orbits, enhancing overall system performance and ensuring seamless service continuity. Chapters 3 to 5 of this thesis will present the algorithmic designs that enable these advanced NTN capabilities, addressing the key inherent challenges due to their non-geostationary nature. These challenges are briefly discussed in the following section.

1.3 Research Challenges

Despite having better orbital stability and global coverage capability, spaceborne NTNs face several technical challenges. Some of them are listed below [12]:

- *Propagation Delay and Path Loss:* NTNs exhibit significant challenges such as large propagation delays and path loss compared to TNs due to their high orbital positions in space. For e.g., GEO, MEO, and LEO satellites experience two-way round-trip propagation latencies of approximately 600–800 milliseconds, 125–250 milliseconds, and 30–50 milliseconds, respectively [20]. These values are significantly higher than those of TNs, where propagation latency is typically on the order of microseconds.
- *Doppler Effect:* It occurs when there is relative motion between transceivers, causing a shift in the signal frequency from the carrier frequency [21]. Due to the high orbital velocities of non-geostationary orbit (NGSO), particularly LEO and MEO satellites, the Doppler effect is more significant in their communication links with ground users compared to GEO satellites or TNs. In the case of GEO satellites or TNs, the satellite or TN components remain stationary, while ground users may experience motion, such as users in high-speed trains or cars.
- *Channel Estimation:* Channel estimation based on pilot signals is essential for establishing communication links between an access node and a user terminal, whether in TNs or NTNs. In NGSO, the channel coherence time is significantly shorter than in TNs due to the satellites' high orbital velocities. For e.g., LEO satellites at an orbital altitude of 550 km above the Earth's surface travel at approximately 7.6 km/s [22], resulting in rapid channel variations. Additionally, the large propagation delays in NTNs increase the likelihood that the estimated channel state information (CSI) becomes outdated

before it can be effectively used to establish a communication link.

- *Mobility Management:* Due to their high orbital velocities, NGSO satellites have short orbital periods, typically ranging from 1.5 to 12 hours [23], resulting in limited service durations during each satellite pass. Specifically, for LEO satellites employing Earth-fixed beam management, the serving duration at a particular location is generally around 10 to 15 minutes. Therefore, to maintain uninterrupted continuity of service between ground users and the serving LEO satellite, an efficient HO mechanism is essential.

- *Radio Resource Management (RRM):* According to the 3GPP Rel. 18 standard [24], the frequency-division duplexing (FDD) band for frequency range 1 NTN (FR1-NTN) is allocated between 1.6 and 2.5 GHz, which is the same range that Starlink plans to use for direct-to-cell communication and is already assigned to terrestrial LTE services [25]. This spectral overlap requires efficient bandwidth management to minimize interference between TN and NTN while maximizing SE. Furthermore, the high free-space path loss in NTN systems demands substantial transmit power to maintain reliable satellite-to-ground links. However, due to the power limit at the satellite, an optimal power allocation strategy is also required to meet the minimum QoS requirements.

- *Computational Resource Management (CRM):* Due to the high launch cost per payload size [8], space-borne NTNs have more limited computational capabilities than TNs. To efficiently utilize these constrained resources and support gNB-like functionality in space, collaboration among integrated space, air, and ground nodes is essential. This cooperation must also adapt to the time-varying topology of NGSO constellations.

1.4 Research Questions

This thesis aims to address several important research questions, building on the motivation and challenges outlined in this work, which focuses on achieving ubiquitous connectivity using fully regenerative payload-enabled spaceborne NTNs. These questions arise primarily from the inherent non-geostationary characteristics of spaceborne NTNs, which necessitate the development of new models, strategies, algorithms, and architectures. Aligned with the IMT-2030 vision, the objective is to effectively tackle some of these challenges and propose innovative solutions that contribute to the enhancement of selected 6G-envisioned KPIs, such as SE, user-experienced data rate, and latency.

> **Research Question 1 (RQ1)**
>
> How can KPIs be monitored for spaceborne NTNs at a GT in near-real time automatically?

It is more difficult to evaluate actual KPIs in a deployed system than to assess theoretical KPIs. To keep pace with technological advancements and meet the performance targets envisioned for 6G systems, it is essential to implement automated mechanisms for near real-time evaluation of actual KPIs for NTNs. Such automation also contributes to optimizing the research and development (R&D) process.

Traditionally, NTNs, particularly GEO satellites, have been used for satellite television (TV) broadcasting, supported by the DVB-S standard [26]. Currently, the largest satellite operator in terms of consumer reach for DVB-based DTH transmission is SES, a Luxembourg-based satellite company. Most recently, SES has been planning to deliver digital content via IP requests through the development of the DVB-NIP system. DVB-NIP enables a two-way, IP-based connection capable of transmitting both video and data, effectively bridging the gap between traditional satellite broadcasting and IP-based content delivery. This system will be deployed via GEO satellites, supporting on-demand digital video streaming services, including OTT content, through an IP-based delivery infrastructure [19].

However, to implement such systems by deploying them in live networks, it is essential to ensure that actual system KPIs closely align with theoretical expectations. This is especially critical for companies with large consumer bases, where performance inconsistencies could result in the loss of loyal customers. Therefore, a mechanism for real-time performance assessment is essential. Large operators like SES cannot afford to rely on manual evaluation methods, which are time-consuming and significantly slow down the R&D process, making large-scale deployments both costly and inefficient. To tackle this challenge, part of this thesis proposes an automated evaluation framework that leverages recent advancements in software development technologies. This framework enables near real-time KPI assessment from GTs, significantly reducing reliance on traditional manual packet analyzers (e.g., Wireshark).

Research Question 2 (RQ2)

What strategies can be employed to effectively minimize propagation delays and path loss in spaceborne NTNs?

The large propagation delays and path loss in NTNs are primarily due to the significant physical distance between satellites and GTs. Among the various spaceborne NTN orbits, i.e., LEO, MEO, and GEO, **satellites in the LEO (a subset of NGSO satellites) offer significantly lower round-trip latency and path loss, making them the primary focus of this thesis**. Nevertheless, LEO satellites still experience higher propagation delays and path loss compared to TNs. Traditionally, satellites have been used for video broadcasting services. To reduce round-trip latency significantly, this thesis considers incorporating on-board caching functionality in LEO satellites, made possible by fully regenerative payload-enabled designs. To further mitigate path loss, a joint approach is proposed that combines adaptive beamforming with an optimal power allocation strategy.

Research Question 3 (RQ3)

How can RRM be effectively performed in NGSO satellites to mitigate interference and enhance user-centric KPIs?

The main radio resources of satellites are power, bandwidth, and spatial dimension. Thus, to effectively utilize these radio resources, part of this thesis proposes a novel optimization problem that jointly optimizes power, bandwidth, and satellite beam coverage, which constitutes a key contribution. In addition, this thesis proposes an optimal precoding design strategy to mitigate interference resulting from the adoption of a full-frequency reuse strategy. Furthermore, it leverages functionalities such as spatial multiplexing and multicasting to utilize these radio resources effectively.

Research Question 4 (RQ4)

What strategies can be developed to manage HO scenarios in NGSO satellites with minimal communication overhead, to improve user-experienced data rate as envisioned in 6G networks?

To support ubiquitous connectivity, this thesis focuses on the use of LEO satellites, as they offer lower propagation delays and path loss compared to other NGSO satellites. Due to their non-geostationary nature and limited field-of-view (FoV), each LEO satellite can serve a given geographic location for only a short duration of time, i.e., a few minutes, per orbital pass in an Earth-fixed beam configuration. Therefore, large LEO constellations are needed to ensure global coverage, and for global connectivity, a seamless HO mechanism is crucial. To address this, part of this thesis proposes novel centralized and distributed HO schemes that leverage ISLs, eliminating the need for ground-based GWs for HO. This consequently reduces signaling overhead and enhances user-centric data rates during HO events.

Research Question 5 (RQ5)

What techniques can be employed to minimize HO duration in the NGSO satellite systems without causing service disruptions, by leveraging advanced capabilities envisioned for 6G networks?

Part of this thesis focuses on minimizing HO duration without any breaks by incorporating ISAC capabilities in GTs, as HO duration in LEO satellite networks can be significantly longer than in TNs. This integrated functionality enables simultaneous communication and sensing using shared power, bandwidth, and antenna resources. Although still in the early stages of research, this feature is envisioned by IMT-2030 as one of the advanced capabilities for 6G systems, as illustrated in Figure 1.1. The proposed novel HO mechanism differs from both the centralized and distributed HO schemes introduced in the earlier research question of this thesis and from the break-before-make HO approach defined in 3GPP Rel. 17 [27].

Research Question 6 (RQ6)

In what ways can the limited computational capabilities of NGSO satellites be overcome to enable advanced service features anticipated in 6G networks, thereby minimizing end-to-end latency?

While large-scale orbital launches and reusable satellite launch systems could enable space-based gNBs, next-generation fully regenerative LEO satellites will have comparatively lower computational capabilities than their MEO, GEO, or TN counterparts. This is primarily due to the high orbital velocities and atmospheric drag experienced in LEO, which reduce their operational lifespan, which is approximately five years [28]. Moreover, with the rapid advancement of augmented reality (AR)/virtual reality (VR) technologies and other data-intensive applications such as real-time video analytics and two-dimensional (2D)/three-dimensional (3D) scene observation [29, 30], LEO satellites are expected to play a crucial role in capturing high-resolution Earth observation (EO) imagery [31] in near real time, owing to their proximity to Earth. However, processing this imagery in near real time requires substantial computational resources to support time-sensitive functions such as disaster prediction and response, evacuation planning, and damage assessment [32, 33]. To address these challenges, part of this thesis proposes a parallel and partial task offloading strategy across the serving LEO satellite, neighboring LEO satellites, and cloud-assisted GWs. Task allocation is optimized by considering factors such as the available computational capacity across the serving LEO, neighboring LEOs, and cloud-aided GWs; the computational effort required for specific tasks; task deadlines; ISL and feeder link performance; and transmission delays. Furthermore, the downlink bandwidth for EO service delivery is jointly optimized with task offloading to minimize overall end-to-end latency, taking into account both HO and non-HO scenarios.

Research Question 7 (RQ7)

How can the challenges of outdated CSI and channel non-reciprocity be addressed in NGSO satellites, and what is the impact of imperfect CSI estimation on the system-level performance?

Due to the large propagation delays in LEO/MEO satellites, the issue of outdated CSI is significantly more severe in NGSO satellites than in TNs. Moreover, unlike TNs where uplink/downlink channel reciprocity often holds, this assumption does not apply to satellite channels because of the use of widely separated frequency bands (typically by hundreds of megahertz) [27], the presence of large-scale fading, and rapid channel variations caused by high orbital velocities and Doppler effects [21]. To address the challenge of outdated CSI, part of this thesis leverages recent advances in AI by proposing a supervised deep learning (DL)-based approach [34] to predict downlink CSI for future time slots using historical downlink channel data. The model is trained on high-performance cloud servers located near the GW, and the trained weights are deployed on fully regenerative LEO/MEO satellites for inference. This architecture enables transfer learning and supports real-time prediction capabilities. Furthermore, the impact of CSI prediction/estimation errors on KPIs, such as user-experienced data rate, latency, and SE, is evaluated to demonstrate the robustness of the proposed frameworks.

> **Research Question 8 (RQ8)**
>
> How can radio resources be efficiently allocated for concurrent uplink transmissions among uncoordinated GTs in NGSO satellite systems, in the absence of a central controller, while minimizing interference?

In the NGSO satellite system, downlink transmission (e.g., from a central access node such as a LEO satellite to multiple GTs) naturally operates as a centralized system. This configuration facilitates efficient resource allocation and decision-making at the satellite level. In contrast, uplink transmission (e.g., from multiple GTs to the LEO satellite) functions as a decentralized system, and in the absence of central coordination, spectrum allocation becomes significantly more challenging. Within the same LEO satellite's FoV, concurrent transmissions from uncoordinated GTs increase the risk of spectrum collisions, thereby degrading overall system performance. To address this challenge, part of this thesis draws inspiration from deep reinforcement learning (DRL)-based dynamic spectrum access techniques in cognitive radio systems [35], and proposes a multi-agent deep reinforcement learning (MADRL) [36] framework for decentralized uplink spectrum allocation.

1.5 Research Methodologies

This section briefly presents the research process followed to provide a complete picture of this thesis, which is clearly illustrated in Figure 1.4. As shown in the figure, the research process starts with a comprehensive review of telecommunications standards, including those from 3GPP, ITU, and ETSI, along with relevant state-of-the-art (SOTA) literature primarily sourced from Institute of Electrical and Electronics Engineers (IEEE) Communications Society venues, both of which assist in identifying research gaps of practical relevance over the next 5 to 10 years. Once suitable problems are identified, aligning with both industrial applicability and academic novelty, system models are developed with a focus on fully regenerative spaceborne NTNs. Based on the identified gaps, novel problems are formulated, often involving the optimization problems of a non-convex nature that reflect the inherent

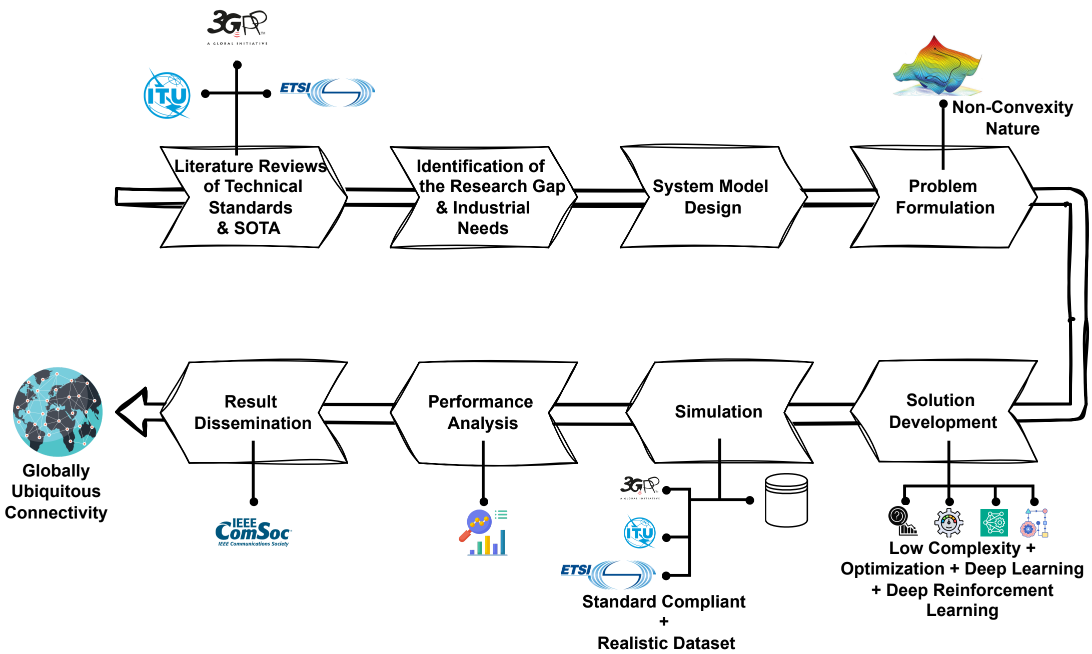

Figure 1.4: Research Methodology.

characteristics of NGSO satellites. These are then addressed through detailed analysis, with a focus on low-complexity algorithm design due to the limited computational capabilities of spaceborne NTNs. To solve the problems effectively, various optimization approaches, such as branch-and-bound (BnB) [37], successive convex approximation (SCA) [38], and Riemannian manifold optimization [39], as well as AI methods such as DL and MADRL, are implemented using software tools such as MATLAB with CVX solvers (e.g., MOSEK [40]) and Python. The proposed solutions are rigorously evaluated and compared against benchmark schemes through realistic simulation scenarios that incorporate practical datasets and parameter settings aligned with telecommunications standards. Finally, the research findings are disseminated via top-tier IEEE Communication Society venues, and, if found economically viable and impactful, they may contribute to future standardization efforts and industrial implementations supporting global ubiquitous connectivity.

1.6 Thesis Outline and Contributions

The primary contributions of this dissertation are organized into six chapters, with each chapter detailing and highlighting its key contribution. Additionally, this work has resulted in several publications and submissions to top-tier, peer-reviewed journals and conference proceedings. These are listed below, where **C** denotes conference papers and **J** denotes journal articles.

Chapter 1

This chapter provides a brief overview of the background and motivation for considering **fully regenerative spaceborne NTN satellites** as the central focus of this thesis. It then explores how satellite architectures have evolved in line with standardization efforts. This is followed by a discussion of the key research challenges arising from the inherent characteristics of such systems. Building on the research motivation and identified challenges, a series of research questions is raised, and finally, the research methodologies adopted to address these questions are presented.

Chapter 2

This chapter addresses **RQ1** by proposing an automated framework for near-real-time monitoring of KPIs for IP traffic over satellite links, using SES's in-house DVB-NIP infrastructure for the delivery of video content, including both OTT and on-demand services. The system analyzes packets received via the transparent payload GEO satellite SES Astra 2E at the GT. This framework enables near real-time evaluation of received data rates for both unicast and multicast IP traffic, significantly reducing manual effort and accelerating the R&D process. **Although this study is based on a satellite system employing a transparent payload architecture, the methodology and automation techniques presented here are equally applicable to fully regenerative payload-enabled next-generation NTNs.**

The achievement of this chapter has been submitted to the following venue:

Related submission:

[C1] **S. Bhandari**, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Automated Near Real-Time Visualization and Monitoring of DVB-NIP Data Rates," *GLOBECOM 2025 - 2025 IEEE Global Communications Conference*, Taiwan, 2025, pp. 1–6 (Submitted).

Chapter 3

This chapter tackles a series of questions, i.e., **RQ2**, **RQ3**, **RQ4**, and **RQ7**, by introducing a novel framework called the flexible resource management algorithm for LEO satellites

(FLARE-LEO). The proposed approach jointly optimizes bandwidth allocation, power distribution, and spot beam coverage to efficiently serve the users distributed across a large geographical area. It incorporates multi-spot beam multicasting, spatial multiplexing, caching, and HO scenarios. Specifically, to optimize the spot beam coverage, an unsupervised K-means algorithm is applied to realistic user demands. This is followed by a proposed SCA-based iterative algorithm for optimizing radio resources. Furthermore, to address the HO scenario, two novel joint transmission architectures, namely centralized collaboration and distributed collaboration, are proposed. These architectures jointly estimate the downlink CSI using DL techniques and optimize the transmit power of the LEO satellites involved in the HO process, such that the overall system throughput is enhanced. Simulation results show that the proposed approach outperforms the reference strategies and demonstrates a significant improvement in minimizing the user-centric maximum content delivery time. Moreover, the robustness of the proposed design is validated by its performance under imperfect CSI conditions, achieving results close to the perfect CSI case, which addresses the second part of **RQ7**.

The outputs of this chapter have been published in the following venues:

Related publications:

[C2] **S. Bhandari**, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Efficient Content Caching for Delivery Time Minimization in the LEO Satellite Networks," *2023 IEEE International Conference on Communications Workshops (ICC Workshops)*, Rome, Italy, 2023, pp. 1246–1252, doi: 10.1109/ICCWorkshops57953.2023.10283798.

[J1] **S. Bhandari**, T. X. Vu, and S. Chatzinotas, "User-Centric Flexible Resource Management Framework for LEO Satellites With Fully Regenerative Payload," in *IEEE Journal on Selected Areas in Communications*, vol. 42, no. 5, pp. 1246–1261, May 2024, doi: 10.1109/JSAC.2024.3365883.

Chapter 4

This chapter focuses on **RQ6**, and partially on **RQ7**. To provide a solution for **RQ6**, this chapter proposes a hierarchical computation framework in LEO satellites for remote Earth observation-related services such as disaster prediction, 2D/3D scene observation, route finding, and rescue operations based on satellite images/videos, while considering different beam management techniques. The proposed framework supports parallel and partial task offloading strategies, optimizing the communication and computation resources across the serving LEO satellite, adjacent LEO satellites, and a cloud-aided GW. The primary objective is to minimize the worst-case service completion time by jointly optimizing the computation and communication allocation among LEO satellites and the GW. The formulated multi-objective optimization problem is characterized as non-linear, non-convex, and non-deterministic polynomial time (NP)-hard. To tackle this challenge, an iterative algorithm based on the SCA method is introduced. Additionally, a delivery strategy is proposed for returning completed task results during HO periods under both Earth-fixed beam and Earth-moving beam configurations. Simulation results based on realistic system parameters and datasets show that the proposed design reduces the task completion time by at least 16% compared to the reference strategies. The impact of computational cycles, ISL rates, feeder link rates, and HO decisions under different beam configurations is analyzed in terms of mean task completion time. Moreover, to address the second part of **RQ7**, this chapter demonstrates the robustness of the proposed design under imperfect CSI conditions by achieving performance close to the perfect CSI case.

The contributions of this chapter have been disseminated in the following venues:

Related publication/submission:

[C3] **S. Bhandari**, T. X. Vu, and S. Chatzinotas, "LEO Satellite-assisted Task Offloading for a Near Real-time Earth Observation Service," *GLOBECOM 2024 - 2024 IEEE Global Communications Conference*, Cape Town, South Africa, 2024, pp. 3033–3038, doi: 10.1109/GLOBECOM52923.2024.10901122.

[J2] **S. Bhandari**, T. X. Vu, and S. Chatzinotas, "LEO-based Edge Computing Service Platform for Challenging Geographical Terrain," *IEEE Open Journal of the Communications Society*, doi: 10.1109/OJCOMS.2025.3638149.

Chapter 5

This chapter explores **RQ5**, part of **RQ7** and, **RQ8**. In response to **RQ5**, it considers ISAC-assisted HO (ISAC-HO) protocol that enables make-before-break HOs via ISAC-enabled GTs. A novel ISAC design capable of generating a 3D beampattern under realistic conditions allows GTs to sense approaching LEO satellites without significantly compromising communication with the currently serving LEO satellite. The design problem is highly challenging due to its non-convexity and mixed-integer nature. To tackle these challenges, this chapter proposes an approach that leverages Riemannian manifold optimization and closed-form solutions. Numerical results confirm that the proposed ISAC design significantly improves the communication–sensing trade-off and enables smooth HO. To address **RQ8**, the chapter presents a MADRL framework for resource allocation among multiple GTs, aiming to mitigate sensing collisions and ensure QoS under shared spectrum constraints. Furthermore, to examine the second part of **RQ7**, the robustness of the proposed design under imperfect CSI is evaluated.

The outputs of this chapter have been submitted to the following venues:

Related submissions:

[C4] **S. Bhandari**, T. X. Vu, N. T. Nguyen, and S. Chatzinotas, "ISAC Design for Intelligent Handover in LEO Satellites," *2026 IEEE International Conference on Communications*, Glasglow, pp. 1–6, 2025 (Submitted).

[J3] **S. Bhandari**, T. X. Vu, N. T. Nguyen, and S. Chatzinotas, "ISAC-Enabled Handover Design in LEO Satellite Networks," *IEEE Transactions on Communications*, pp. 1–16, 2025 (Major Revision, Revised & Submitted).

Chapter 6

This chapter summarizes the key findings of the thesis and outlines potential directions for future research.

Chapter 2

Automated Near Real-Time Visualization and Monitoring of DVB-NIP Data Rates

With the onset of Industry 4.0, automation has become crucial for industries to reduce repetitive manual and intellectual tasks, meet deadlines, and maintain competitiveness. The satellite industry, in particular, is harnessing automation to accelerate technology development and market launches, optimizing the research and development (R&D) process in a rapidly evolving, multi-billion dollar market. As part of this trend, key satellite players are focusing on streaming video content to meet the growing demand for video services, regardless of geographical constraints. To achieve this, they have developed the Digital Video Broadcasting - Native IP (DVB-NIP) system, with ongoing R&D efforts to ensure its effectiveness. In this chapter, we propose an automated mechanism to visualize the achieved average data rate of the DVB-NIP system across various scenarios. This system reduces the need for repetitive intellectual work, offering a more efficient alternative to conventional packet analysis methods. The proposed solution enhances the R&D process and can be extended with additional functionalities as required to monitor key performance indicators.

2.1 Introduction

According to Ericsson, it is predicted that by the end of 2028, video traffic will account for 80% of global mobile data traffic [41, 42]. The rapid proliferation of the Internet of Things (IoT) devices, data-intensive applications such as social media-based streaming services, and advancements in augmented reality (AR) technologies are expected to drive this growth

[42, 43]. Recently, there has been exponential growth in revenues for on-demand digital video streaming services, including over-the-top (OTT) services and it is expected that the revenue will reach 235 billion U.S. dollars by 2028 [43].

The most popular and conventional method of transmitting OTT content is via a content delivery network (CDN) based on Internet Protocol (IP) requests. Content is delivered to end users through terrestrial internet infrastructure, with the CDN optimizing distribution across different regions. This approach is most effective when end users are located relatively close to the CDN's points of presence or *local caches where the requested content is stored*[2.1]. However, users in geographically challenging areas, such as remote islands, during flights, or on cruise ships, may struggle to access popular content due to limited technological infrastructure, highlighting the digital divide as a multifaceted issue shaped by geographic, economic, and technological factors [42, 43].

Recent advances in satellite technology have enabled satellite-based delivery of OTT content based on IP requests, referred to as DVB-NIP, to address the aforementioned challenges. Still in its early stages, this approach, detailed in the DVB-NIP standard published by the European Telecommunications Standards Institute (ETSI) in September 2024, supports the IP-based distribution of Common Media Application Format (CMAF), Dynamic Adaptive Streaming over HTTP (DASH), and HTTP Live Streaming (HLS) content. It facilitates live video and file distribution over traditional broadcast channels to devices such as smartphones, tablets, and smart televisions [26].

Building on these advancements, Société Européenne des Satellites (SES), a leader in satellite technology for content delivery, is driving the development and implementation of the DVB-NIP specification, alongside key DVB-Internet (DVB-I) satellite features. SES collaborates with satellite operators, broadcasters, and consumer electronics manufacturers to expand these technologies. The company plays a pivotal role in providing broadband connectivity and end-to-end video services to rural and remote areas. At DVB World 2024, SES showcased its DVB-NIP Revision 2-compliant software, demonstrating its contributions to both DVB-NIP and DVB-I. The broadcast featured live and offline content from its Betzdorf headquarters, delivered via an internally developed multicast Adaptive Bitrate (mABR) server and commercial partner servers [19].

Currently, SES is using geostationary Earth orbit (GEO) satellites for the unicast and

[2.1] The contents to be placed can be predicted based on historical popularity.

multicast delivery of OTT content [19]. However, in the near future, if the inherent challenges of non-geostationary orbit (NGSO) satellites, such as high-speed Doppler effects and frequent handover requirements resulting from fixed service periods, are effectively addressed, SES may also utilize NGSO satellites for OTT content delivery, due to their comparatively lower propagation delays and better large-scale fading characteristics [42]. This would align with companies like SpaceX, Eutelsat OneWeb, Amazon's Project Kuiper, Telesat, and Shanghai Spacecom Satellite Technology, which are competing to launch large constellations of LEO satellites to provide global broadband and direct-to-cell services [44].

The transmission algorithms from the origin CDN to the NIP gateway via GEO satellites require trial-and-error processes. A key performance indicator (KPI) for DVB-NIP's robustness is comparing the received average data rate with the achievable rate for each NIP reception. The media packet metadata received at the NIP gateway are captured in packet capture (PCAP) format, and the most commonly used tool for evaluating the received data rate is using Wireshark, a popular packet-analyzer software [45]. However, Wireshark's manual packet analysis is unsuitable for industries transmitting large volumes of media files to global destinations, as near real-time evaluation is essential for optimizing the DVB-NIP algorithm, aligning the received and achievable data rates, and ensuring a smooth, risk-free transition from existing broadcasting technologies to DVB-NIP.

Our contributions can be summarized as follows:

- Algorithm 2.1 is designed to automate the near real-time extraction of key metadata from DVB-NIP PCAP files and present it in a tabular format.
- Algorithm 2.2 is designed to enable the near real-time visualization and analysis of DVB-NIP streams, facilitating debugging with optional human intervention.
- The algorithms are tested using DVB-NIP streams generated from SES's Astra 19.2 TP 1.045, demonstrating significant improvements in metadata extraction and data rate visualization (see figures 4 and 6) compared to manual packet analysis with tools like Wireshark. Additionally, recommendations are provided for evaluating the sampling time to assess packet arrival rates.

The rest of this chapter is structured as follows. Section 2.2 describes the DVB-NIP end-to-end system architecture and presents the automated framework for rate evaluation from

PCAP files. Section 2.3 presents the algorithms for automatic rate evaluation in both unicast and multicast scenarios. Section 2.4 illustrates the simulation results. Section 2.5 outlines the recommendations and limitations of this work, followed by a summary in Section 2.6.

2.2 DVB-NIP End-to-End System Architecture and PCAP Rate Evaluation Framework

2.2.1 DVB-NIP End-to-End System Architecture

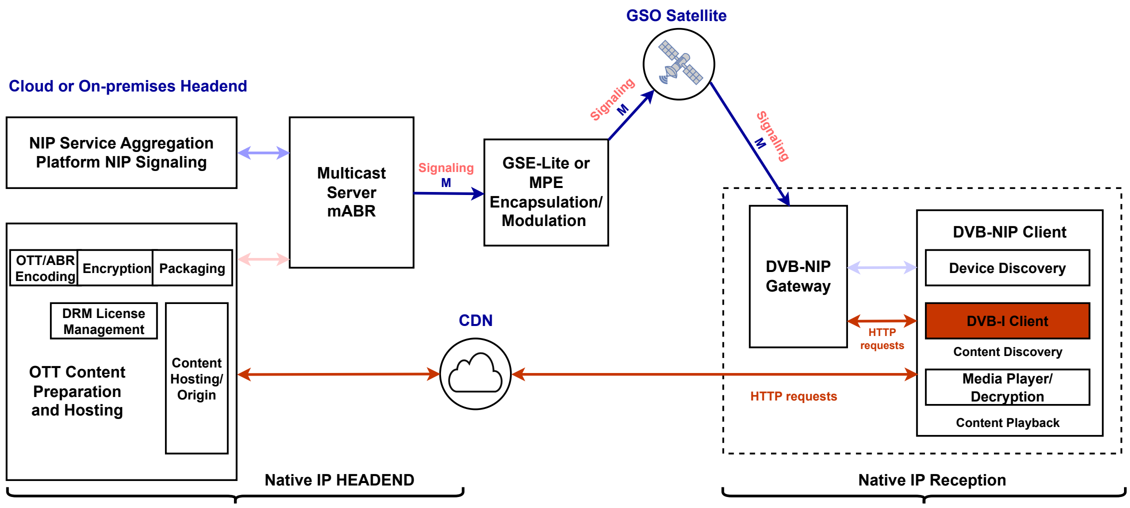

Figure 2.1: Simplified version of end-to-end system architecture of DVB-NIP [19, 26].

Fig. 2.1 illustrates a simplified end-to-end system architecture of DVB-NIP. Unlike traditional transport stream-based broadcast architectures, the DVB-NIP system sources all content, including OTT, through the *content preparation and hosting platform*. The whole NIP architecture is divided into *NIP headend* and *NIP receiver* functions [26].

The *NIP headend* function comprises several components: the *OTT content preparation and hosting platform* for media encoding, encryption, packaging, and hosting; the *NIP service aggregation platform* for compiling DVB-I service lists; the content provider platform for content acquisition, scheduling, and monetization; the *NIP signaling server* for building the multicast signaling channel; the *multicast server* for generating multicast transport sessions carrying one or more DVB-DASH services; the *service provisioning platform* for managing headend operations and service reporting; and the *encapsulation and modulation platform* for encapsulating IP multicast datagrams into NIP streams. The *NIP receiver* functions consist of *NIP gateway* and *NIP client* functions. The *NIP gateway* functions include receiving IP multicast streams, parsing the NIP announcement channel, temporarily storing broadcast assets, serving assets to clients via HTTP, retrieving content or metadata from broadband networks, and supporting remote configuration. The *NIP client* functions involve interacting with the DVB-I service list entry points registry server and service list server, displaying DVB-I service lists to end users, discovering NIP gateways, and requesting and displaying NIP services [26].

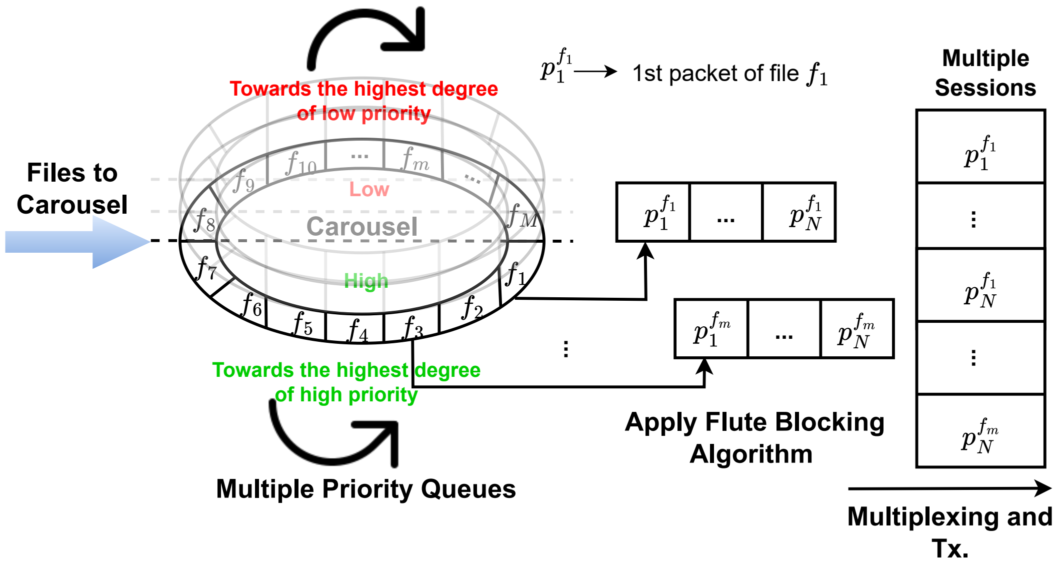

Figure 2.2: Flute carousel in the mABR server [19, 26].

Considering the pre-defined radio resource system, the mABR server is primarily responsible for the overall KPI of the DVB-NIP system. It handles the multiplexing of streams and diverse range of files (such as video on demand (VoD), signaling manifests, content guides, and DVB-I materials) by organizing files in the carousel based on their priority order as shown in Fig. 2.2. SES has developed a DVB-NIP mABR server that integrates a multiplexer optimized for efficient file delivery, accommodating specific file characteristics and transmission requirements. The server produces DVB-NIP streams that are broadcast and accessible via Astra 19.2 TP 1.045 [19].

The scheduler, or File Delivery over Unidirectional Transport (FLUTE) carousel, prioritizes tasks based on priority order and user-defined tolerance limits. It implements the FLUTE blocking algorithm, rescheduling tasks if no packet is received within a predefined interval. Upon packet reception, data is forwarded to the User Datagram Protocol (UDP)/IP multicast output.

The FLUTE protocol enables file transfer over UDP/IP networks by multiplexing files within a FLUTE carousel, allowing parallel transmission with assigned priority levels. Lower-priority carousels are transmitted only when no higher-priority ones are active. The carousel organizes files into priority queues, with each queue transferring one file per session for efficient multiplexing [19].

The token bucket algorithm is widely used for controlling the transmission bitrate of files in network traffic shaping [46], limiting both the overall bitrate of a NIP stream and the bitrate of a FLUTE carousel. This mechanism allows files to begin transmission at a specific time, with the bitrate dynamically adjusted to ensure completion by a predefined end time. Such techniques optimize transmission efficiency, meet real-time constraints, reduce zapping time, and minimize jitter in live content delivery.

To achieve these objectives and optimize multiplexing, the SES mABR server implements a scheduling algorithm that governs the execution of tasks at fixed intervals. Each task is responsible for emitting packets and re-scheduling itself for subsequent runs. However, to fully adapt this algorithm to the system's specific requirements, proper tuning is necessary. This process involves multiple trial-and-error iterations, during which the received packets must be analyzed in near real-time to enable effective debugging and facilitate the algorithm's refinement through human intervention.

2.2.2 PCAP Rate Evaluation Framework

In this section, we present the DVB-NIP data rate evaluation framework for unicast and multicast scenarios, as shown in Fig. 2.3. The process begins with extracting useful metadata from a PCAP file containing a UDP/IP stream or directly from a UDP/IP stream in near real-time. To achieve this efficiently, a fast and memory-optimized programming language like Rust is desired. Rust's high-speed performance and multiprocessing capabilities make it well-suited for processing large PCAP files with minimal resource overhead.

However, for demonstration purposes, we use Python to illustrate the procedure for extracting useful information from a PCAP file. Python is chosen for its simplicity, ease of use, and the availability of robust open-source libraries such as Scapy [47] and FLUTE [48].

Once all the useful metadata are extracted from the PCAP files using the parsing tech-

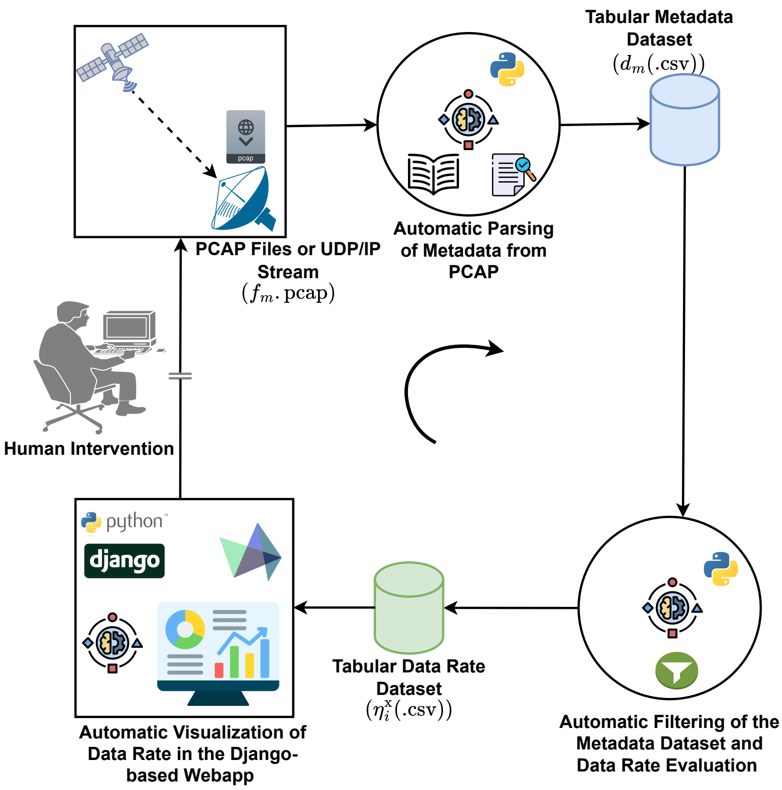

Figure 2.3: An automatic data rate evaluation framework from PCAP files.

nique automatically, the extracted information is stored in a tabular format in the system, which can be easily read by another Python script responsible for filtering the dataset for the multicast scenario to evaluate the data rate. After the data rate is evaluated for the unique transmission and reception case, it is stored in a tabular format, making it easily accessible to the backend of the web application. Django [49] can be utilized for backend creation, as it is a high-level, open-source web framework for Python. For dynamic graph visualization, Highcharts can be used, as it is a popular JavaScript charting library that allows you to create interactive, responsive, and aesthetically appealing charts for web applications, following a similar procedure as in [50] for dynamic webapp creation. To ensure the desired results are produced and for debugging purposes, human intervention may be required to adjust parameter values if the results deviate from expectations, as shown in Fig. 2.3. The detailed procedure for metadata extraction and corresponding data rate evaluation will be discussed in the next section.

2.3 Solution Development

To automate the visualization of data rate graphs for unicast and multicast scenarios, Algorithms 2.1 and 2.2 are designed and are discussed in detail in the following sub-sections.

2.3.1 Algorithm 2.1: Automatic Metadata Extraction from PCAP Files

Algorithm 2.1 Algorithm to Extract Key Metadata from DVB-NIP PCAP Files

Input: $\mathcal{F} = \{f_1, f_2, \ldots, f_M\}$
Output: $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$
Init.: TS_m, PLen_m, SRC_m, SPort_m, DST_m, DPort_m, LCT_m, TSI_m, TOI_m, SBN_m, ESI_m = []
for m in range(M) **do**
  PKTS ← rdpcap(f_i's path)
  for PKT in PKTS **do**
    if IP in PKT and UDP in PKT **then**
      // Extraction of UDP Metadata from PCAP
      TS_m.append(PKT.time)
      PLen_m.append(PKT.payload.len)
      SRC_m.append(PKT.payload.src)
      SPort_m.append(PKT.payload.sport)
      DST_m.append(PKT.payload.dst)
      DPort_m.append(PKT.payload.dport) *// Parse UDP payload to extract LCT metadata* ▽
      try
        UPKT ← PKT[UDP].payload.load
        LCT ← receiver.LCTHeader(UPKT)
        CCI_m.append(LCT.cci)
        TSI_m.append(LCT.tsi)
        TOI_m.append(LCT.toi
        SBN_m.append(LCT.sbn)
        ESI_m.append(LCT.esi)
      // If Packet DNS ▽
      except AttributeError as e
        pass
    end if
  end for
  Save the df containing the cols. $\{\mathrm{TS}_m$, PLen_m, SRC_m, SPort_m, DST_m, DPort_m, CCI_m, TSI_m, TOI_m, SBN_m, $\mathrm{ESI}_m\}$ as d_m
end for

In Algorithm 2.1, the meaningful metadata information from a list of PCAP files $m \in \mathcal{F} = \{f_1, f_2, \ldots, f_M\}$ is extracted, and this information is stored in a standardized tabular

Algorithm 2.2 Automatic Data Rate Calc. for DVB-NIP

Input: $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$, T_m^s
Output: η_m^{x}, $\mathrm{x} \coloneqq \{\mathrm{SRC[i1], SPort[i2], DST[i3], DPort[i4], TSI[i5]}\}$
Init.: $\mathrm{df, SRC, DST, SPort, DPort, TSI}$
for m in range(M) **do**
  d ← read(d_m's path); SRC ← unique(d.SRC); DST ← unique(d.DST); SPort ← unique(d.SPort); DPort ← unique(d.DPort); TSI ← unique(d.TSI) // *Fn for sampling* ▽
  def Sampling(T_m^s, p, PLA, TCA, TS)
    PL = 0; TC = 0
    while TC $\leq T_m^s$ **then**
      TD ← TS[TS.idx[p + 1]] − TS[TS.idx[p]]
      if TC + TD $\leq T_m^s$ **then**
        TC ← TC + TD
        PL ← PL + PLen[PLen.idx[p + 1]]
      else
        TCA[0] ← TC; PLA[0] ← PL
        TC ← T_m^s + 1e18
        TCA[1] ← TC; PLA[1] ← 0
      end if
    end while
  return(p, PLA, TCA) // *Filtering the df* ▽
  for i1 in range(0, len(SRC)) **do**
    df1 ← df[df['SRC'] == SRC[i1]]
    for i2 in range(0, len(SPort)) **do**
      df2 ← df1[df1['SPort'] == SPort[i2]]
      for i3 in range(0, len(DST)) **do**
        df3 ← df2[df2['DST'] == DST[i3]]
        for i4 in range(0, len(DSTPort)) **do**
          df4 ← df3[df3['DPort'] == DST[i4]]
          for i5 in range(0, len(TSI)) **do**
            df5 ← df4[df4['TSI'] == TSI[i5]]
            PLen ← df5['PLen']; TS ← df5['TS'];p = 0; PAr = []; TAr = []
            while p ≤ len(df5) − 1 **do**
              PLA = []; TCA = [];
              [p, PLA, TCA] = Sampling(T_i^s, p, PLA, TCA, TS)
              PAr.append(flatten(PLA)); TAr.append(flatten(TCA)); p ← p + 1
            end while
            η_{i5}^{x}.append(divide(PAr, TAr))
            Save and **Plot** η_{i5}^{x} (using Highcharts)
          end for
        end for
      end for
    end for
  end for
end for

format (CSV or XLSX) as d_m.

The dataset should include UDP as well as Layered Coding Transport (LCT) header fields. The UDP headers are timestamp (TS), payload length (PLen), source IP (SRC), source port (SPort), destination IP (DST), and destination port (DPort). The LCT headers are extracted by parsing the UDP payload and should contain information such as Congestion Control Information (CCI), Transport Session Identifier (TSI), Transport Object Identifier (TOI), Source Block Number (SBN) and Encoding Symbol ID (ESI), if the packet type is not DNS. The snapshot of the information extracted from the PCAP file for the unicast scenario, presented in a tabular format, is shown in Fig. 2.4.

SN	TS	Plen	SRC	SPort	Dst	Dport	CCI	TSI	TOI	SBN	ESI	Target Bit Rate
1006	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1007	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1008	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1009	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1010	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1011	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1012	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1013	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1014	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1015	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1016	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1017	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1018	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1019	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1020	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1021	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1022	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1023	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1024	1711459020	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1025	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1026	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1027	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1028	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1029	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1030	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1031	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576
1032	1711459022	1064	127.0.0.1	2222	224.0.23.14	3937	0	1	0	0	0	5048576

Problem: The packet takes significantly longer to transmit, reducing the average data rate

Possible Solution: Examine the Flute carousel and the network traffic shaping algorithm

Figure 2.4: Snapshot of the PCAP information extracted in the tabular format for the unicast scenario.

2.3.2 Algorithm 2.2: Data Rate Evaluation for Uni/Multi-cast

In Algorithm 2.2, the data rate is evaluated for each unique combination of headend and reception address, defined as x $\triangleq$ {SRC, SPort, DST, DPort, CCI, TSI}, while considering the unicast or multicast nature of DVB-NIP streams. To compute the data rate for x in the m-th PCAP file, the payloads of consecutive unique packets are summed within the user-defined sampling period T_m^s, as illustrated in the Sampling function of Algorithm 2.2. If the sampling period condition is met, i.e., TC $> T_m^s$, the loop terminates. An outage (drop in the packet) condition is declared, i.e., PL $= 0$, when no payload can be accommodated within the specified sampling period, as demonstrated by packet number 1025 in Fig. 2.4.

2.4 Simulation Results on Realistic System Scenario

We used the SES DVB-NIP system to receive packet streams via Astra 19.2 TP 1.045. For demonstration purposes, we consider a unicast scenario, and the key elements of the database for this particular DVB-NIP stream are shown in Fig. 2.4. The target bit rate for this particular NIP reception is around 5 Mbps.

Upon receiving the PCAP file of this particular stream, we analyzed it using the Wireshark tool, considering $T_m^s = 50\,\text{ms}$. The resulting graph is shown in Fig. 2.5, where a total of nine outages were observed over an 18 s duration of the stream, each lasting approximately 1.6 s.

Similarly, Fig. 2.6 presents a comparable graph generated using Python with the implementations of Algorithms 1 and 2. The graph reveals that the achieved average data rate is less than half of the desired rate. This indicates inefficient utilization of bandwidth resources, highlighting the need to identify potential issues and perform effective troubleshooting. This approach avoids relying on manual graph visualization, which becomes significantly more challenging in multicast DVB-NIP stream transmission scenarios.

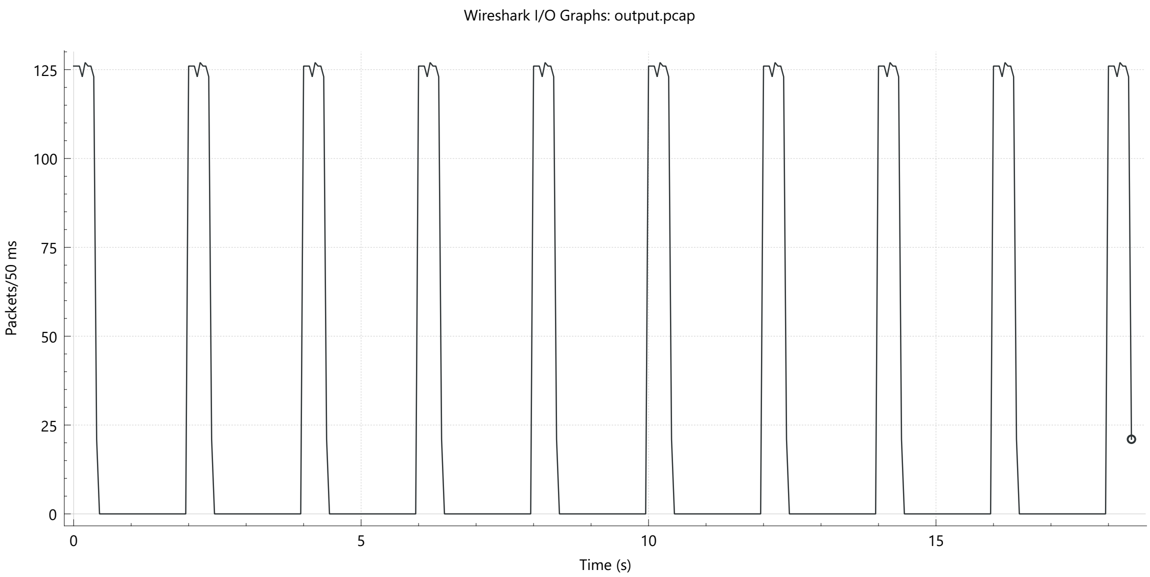

Figure 2.5: Graph generated using Wireshark to visualize DVB-NIP stream data rates.

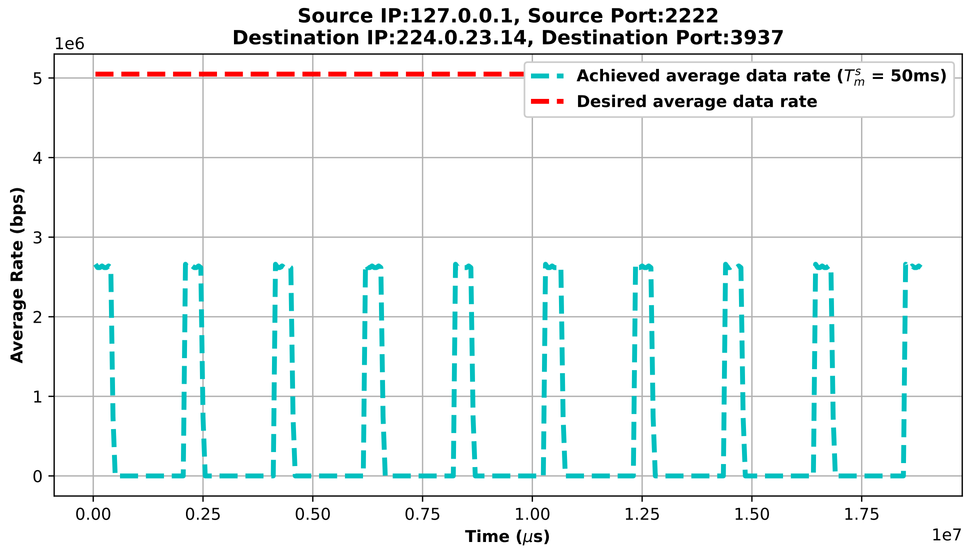

Figure 2.6: Automated visualization of DVB-NIP stream data rates using Python/Rust.

2.5 Recommendations and Limitations

2.5.1 Recommendations

The following are a few recommended steps that can help minimize the gap between the desired and achieved average data rates:

i. Initially, review and optimize the implemented FLUTE carousel algorithm as needed.

ii. Next, the parameters involved in traffic network shaping should be properly tuned.

iii. Carefully define the sampling period for the particular stream, as it may vary depending on the nature of the stream.

iv. If steps (i) – (iii) do not improve the data rate as planned, try transmitting content that takes less time than the sampling period during outage scenarios.

v. As a last resort, optimize transmission bandwidth and power.

2.5.2 Limitations

The limitations for evaluating DVB-NIP data rate are as follows:

i. The packet-to-packet timestamps in the PCAP file lack accuracy, necessitating the grouping of packets by sampling time and the averaging of data points using a moving time window approach to derive meaningful results.

ii. The transmission time for data packets from the GEO satellite (e.g., Astra 19.2) is not included in the metadata. However, this information can be embedded in the packet streams, which requires additional bandwidth and increased effort to extract the baseband frame and generic stream encapsulation frames.

iii. It is very difficult to determine the sampling period and averaging window time, and it becomes even more challenging in multicast and interleaved scenarios. This can only be determined through a trial-and-error mechanism.

2.6 Summary

In this chapter, we propose a framework for the automated evaluation of the received data rate of DVP-NIP data streams (both unicast and multicast) from PCAP files in a realistic system setup. The framework operates to monitor the KPIs of the DVB-NIP system and facilitate the debugging process in near real-time. In addition, the chapter presents key recommendations for improving the average data rate and discusses the limitations of the current work.

Chapter 3

User-Centric Flexible Resource Management Framework for LEO Satellites With Fully Regenerative Payload

The regenerative capabilities of next-generation satellite systems offer a novel approach to designing low Earth orbit (LEO) satellite communication systems, enabling full flexibility in bandwidth and spot beam management, power control, and on-board data processing. These advancements allow the implementation of intelligent spatial multiplexing techniques, addressing the ever-increasing demand for future broadband data traffic. Existing satellite resource management solutions, however, do not fully exploit these capabilities. To address this issue, a novel framework called flexible resource management algorithm for LEO satellites (FLARE-LEO) is proposed to jointly design bandwidth, power, and spot beam coverage optimized for the geographic distribution of users. It incorporates multi-spot beam multicasting, spatial multiplexing, caching, and handover (HO). In particular, the spot beam coverage is optimized by using the unsupervised K-means algorithm applied to the realistic geographical user demands, followed by a proposed successive convex approximation (SCA)-based iterative algorithm for optimizing the radio resources. Furthermore, we propose two joint transmission architectures during the HO period, which jointly estimate the downlink channel state information (CSI) using deep learning and optimize the transmit power of the LEOs involved in the HO process to improve the overall system throughput. Simulations demonstrate superior performance in terms of delivery time reduction of the proposed algorithm over the existing solutions.

3.1 Introduction

In the context of satellite constellations, the LEO constellation is considered suitable for broadband services due to its small round-trip delay compared to other satellite constellations. Thanks to advanced payload technology, the LEO satellites are now seen as key enablers for the beyond 5G (B5G) and sixth-generation (6G) communications systems, as they can intelligently deliver low-cost, higher-throughput broadband services to underserved areas [51, 52].

The success of LEO satellites to B5G/6G relies on the operating and payload architecture. Traditionally, two main configurations have been prevalent: the wide beam and the multiple spot beam configurations. The wide-beam configuration is characterized by wide coverage and is mainly used for broadcasting applications, while the multiple-spot beams are specifically designed for broadband services [53]. These configurations excel at providing dedicated services but lack the flexibility to effectively handle dynamic and complex situations such as targeted users' mobility and time-varying demand. This lack of flexibility is favored in traditional satellite architectures due to the high cost and delays associated with the payload changes [54]. However, recent advancements in payload technologies, such as digital transparent payload (DTP) and active on-board antennas, enable efficient and reconfigurable hybrid broadcast/broadband modes [55].

Current DTP, however, has limited capabilities such as restricted flexible channelization and rudimentary power control/sharing among carriers [56]. Thus, to address the shortcomings of the DTP, satellite companies are shifting their focus to incorporating advanced regenerative (fully digital) payload technology, which integrates a regenerative processor, electronically steered phased-array antennas, and optional memory units [57]. This transition allows for the optimization of various functionalities, including beamforming, spot beam coverage patterns, signal quality, bandwidth, and power as per the traffic demand [58,59]. To minimize overall latency and further enhance the quality of service (QoS), the regenerative payload's optional memory unit can be used for caching in LEO satellites. This approach is favored over the terrestrial networks because data cached in terrestrial networks must traverse multiple hops, which causes frequent HOs at the gateways (GWs) unless the requesting user equipment (UEs) are adjacent to edge nodes [60]. Moreover, the regenerative payload of the satellite constellation allows for the flexibility of on-demand multicasting services, potentially enabling the simultaneous delivery of cached content to different communities of users spread across different geographic areas [53, 61].

The successful launch of the OneWeb's LEO satellite, JoeySat, in May 2023, funded by the European Space Agency and UK Space Agency, showcases the incorporation of the flexible software-defined regenerative processor along with multi-spot beam electronically steered phase array antennas. This implementation fulfills the demand-based beam tailoring and steering capability [6]. However, a complete package of algorithm design is needed to fully leverage the functionality of fully flexible regenerative payload-enabled satellites.

3.1.1 Related Works

Several studies have been conducted to partially exploit the flexible payload capabilities [54, 59, 62–66]. In [59], a bandwidth and power optimization method is proposed for non-geostationary orbit (NGSO) based on realistic demands. In [62], a demand-driven geostationary orbit (GEO) beam steering and beam patterning method using flexible regenerative payload capabilities is proposed. The authors of [63–65] propose a caching policy in LEOs using flexible on-board regenerative payload capabilities to minimize content delivery delay and maximize the probability of successful delivery based on predefined beam coverage, transmit power, and operating bandwidth. In [54], the authors aim to leverage the regenerative payload-enabled capabilities, such as digital beamforming, caching, and bandwidth optimization, considering the realistic demands. Inspired by [54], to address the shortcomings of [63–65] to some extent, considering the flexible regenerative payload enabled LEOs capabilities, in [66], an optimization problem is formulated at two different time scales to maximize the utility function in the integrated satellite-terrestrial network by considering the joint design of cache placement, multicast-beamforming, base station and satellite clustering, and transmit power. However, the optimal use of satellite operating bandwidth and spot beam coverage was not considered therein.

Since the LEO satellites can only provide uninterrupted service to the particular area in the earth fixed beam scenario for about 10 to 15 minutes during one orbital, it is crucial to consider the HO scenario via inter-satellite link (ISL) [67]. Unfortunately, the existing literature lacks proper algorithm designs for HO duration involving multiple LEOs. To fulfil this research gap and address the shortcomings of [63–66], we propose a flexible resource management algorithm that fully leverages the flexible regenerative payload capabilities and efficiently utilizes the ISL during the HO periods.

3.1.2 Contributions

In this chapter, we propose FLARE-LEO, a collaborative algorithm that leverages the flexible payload and electronically steered phased array antennas embedded in LEOs. FLARE-LEO incorporates various capabilities of LEOs, including demand-based adaptive beam patterning and steering, multi-spot beam multicasting, caching, bandwidth and power optimization, as well as ISL-HO. Our contributions can be summarized as follows:

- We formulate a joint design of spot beam coverage, operating bandwidth, and multi-user precoding vectors to minimize the average delivery time in LEO-assisted caching networks, including HO scenarios. Although flexible bandwidth has been considered in satellite communications, to the best of our knowledge, this is the first work exploiting spatial multiplexing [67] technique within each spot beam in LEO-enabled caching systems thanks to fully flexible regenerative payload and electronically steered phased array antennas capabilities.

- We propose to solve the joint optimization problem via two sub-problems: beam coverage design and radio resource allocations. Unlike other clustering strategies [68], our approach guarantees non-overlapping and non-empty clusters, aligning with our goal of creating distinct spot beams and optimizing their coverage area. To tackle the non-convexity of the second sub-problem, we reformulate it using a difference-of-convex representation and propose two SCA-based iterative algorithms for joint optimization of frequency bandwidth and multi-user precoding vectors, applied to both optimal and zero-forcing (ZF) precoding designs. It is worth noting that the solution in [69] is not applicable in our system since it does not consider the bandwidth allocation.

- We propose novel architectures for joint resources optimization between two LEOs during the HO period, namely *centralized architecture*, in which the joint optimization is executed in the GW, and *distributed architecture*, in which each LEO optimizes its own radio resources and exchanges parts of the outputs to the other via ISL. These architectures differ in their computational capabilities, packet overhead, and communication needs between two LEOs. In addition, a deep learning (DL)-based CSI prediction is proposed during the HO period to improve the effective system throughput.

- Finally, the advantages of the proposed framework are demonstrated via numerical results based on the realistic Movielens dataset [70]. Simulation results indicate that the adaptive beam scenario outperforms the fixed beam scenario by at least 1.22 times in terms of effective mean data rate when the total power of the LEOs is varied between (25–35) dBW. Additionally, the effective mean data rate of the proposed design in HO periods is at least 1.5× higher than the conventional method without joint transmission.

The remainder of this chapter is organized as follows. Section 3.2 describes the system model and parameters. Section 3.3 presents the problem formulation and proposed solution. Section 3.4 introduces the HO scenario and DL-based CSI prediction scheme. Section 3.5 presents the different HO schemes based on computational capability and overhead. Section 3.6 demonstrates the effectiveness of the proposed scheme using numerical results. Finally, Section 3.7 concludes the chapter.

Table 3.1: Summary of Main Notations

Symbols	**Description**
M	Number of spot beams
N	Spatial multiplexing gain in each spot beam
G_m, G_u	Antenna gain of spot beam m and UE u
θ_m, ϕ_m	Elevation angle and azimuth angle
D_m, r_m	Slant distance and radius of coverage of beam m
$T, \tau_{slot}, \tau_{csi}, \tau_{pro}$	Service duration time, time slot duration, channel estimation time, and processing time
$\boldsymbol{h}_{u,k,a,m}$	Downlink channel coefficient
$\boldsymbol{w}_{k,a,m}$	Precoding vector for the multicast group k of AUG a
$\check{I}_m, \hat{I}_m$	Intra-beam and inter-beam interferences
$b_{a,m}$	Bandwidth of associated group a within spot beam m
$\mathcal{L}_{\mathscr{J}}(\Theta)$	CNN-CSI loss function

3.2 System Model

We consider a LEO constellation providing services in a given area, in which a LEO satellite is serving the users at a given time. HO occurs when the serving LEO satellite is departing and a new LEO is joining the area. Without loss of generality, the considered system not in the HO period comprises a LEO satellite serving a set $\mathcal{U} = \{1, 2, ..., u, ..., U\}$ of U single-antenna

UEs within its coverage, a GW, and a centralized cloud, as shown in Fig. 1. The operation during the HO period will be presented in Section IV. The LEO satellite is equipped with a cache-enabled flexible regenerative payload and electronically steered phased array antennas that can generate M spot beams of arbitrary shapes [71] to adaptively serve UEs within its footprint. For ease of analysis, we assume the shape of spot beam m to be circular and to leverage the users clustering and spot beam optimization, we assume a uniform antenna radiation pattern within a spot beam. It is worth noting that once the spot beams are determined, actual location-dependent path losses are employed to compute the received signal power. The spot beam gain $G_m(\theta_m, \phi_m)$ can be computed as [72]:

$$G_m(\theta_m, \phi_m) = \frac{\text{Area of isotropic sphere}}{\text{Area of spot beam rad. pattn.}} = \frac{4D_m^2}{r_m^2}, \forall m, \tag{3.1}$$

where θ_m and ϕ_m are the elevation and azimuth angles in radians relative to the boresight of the spot beam m, r_m is the spot beam's radius, and D_m is the slant distance between the LEO satellite and the m-th spot beam. To efficiently serve the UEs over a geographical area, the spot beams are designed to be non-overlapping and adequately spaced, which allows full-

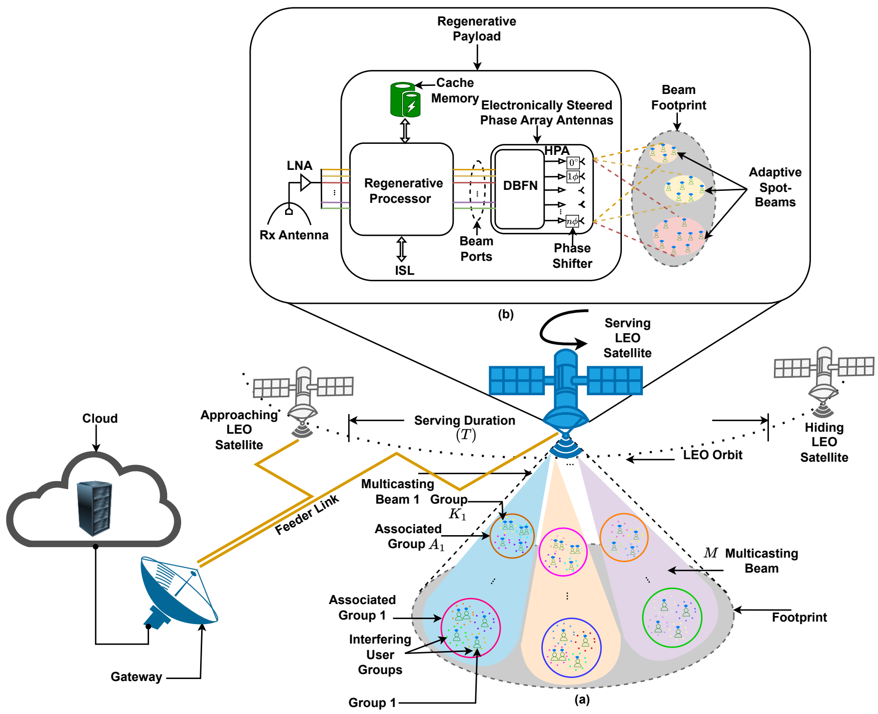

Figure 3.1: A downlink multicasting LEO satellite system with a regenerative payload architecture. (a) Multicasting LEO satellite system. (b). Regenerative payload architecture.

frequency reuse. Thanks to the advanced payload technology, each LEO satellite can deliver up to N spatial multiplexing data streams in each spot beam [73]. The main notations used in this chapter are summarized in Table 3.1.

3.2.1 Caching Model

By equipping with the advanced flexible regenerative payload, the LEO satellite is able to process data and has a limited cache memory of C bits. The U UEs are interested in the content library of $\mathcal{F} = \{1, 2, ..., f, ..., F\}$ at the centralized cloud consisting of F files. Due to the non-geostationary nature, the LEO satellite has a limited service duration of T for each satellite pass of a considered area. We consider offline caching policy [74] in which the demand vector $\boldsymbol{L}$ is obtained in advance, e.g., via historical average or prediction model [34]. Based on $\boldsymbol{L}$, the cache placement is executed at the beginning of each service duration based on generic caching models, such as most popular caching (MPC), uniform caching (UC), and random caching (RC). We focus on the transmission design in the delivery phase.

3.2.2 User Grouping

To exploit the flexible multi-beam capability, the users are served in groups depending on their geographical locations and requested contents. Denote the set of UEs in each spot beam m as $\mathcal{U}_m \subseteq \mathcal{U}$, which is further divided into K_m groups. The users within the same group request the same content file. If $K_m \leq N$, all user groups can be served simultaneously using the whole bandwidth B via spatial multiplexing techniques, i.e., multi-user precoding. Otherwise, K_m groups are divided into $A_m = \lceil \frac{K_m}{N} \rceil$ associate user groups (AUGs). Different AUGs are served via orthogonal frequency bandwidths, while the users within one AUG are served simultaneously via multi-user precoding technique. Let $\mathcal{A}_m = \{a_1^m, a_2^m, ..., a_{A_m}^m\}$ denote the set of A_m AUGs in spot beam m, and $\mathcal{K}_{a,m}$ denote the set of users belong to the AUG a of spot beam m.

To illustrate, suppose 8 users $\{u_1, u_2, ..., u_8\}$ within spot beam m requests the corresponding files $\{f_1, f_2, f_3, f_3, f_4, f_5, f_5, f_6\}$, which includes $K_m = 6$ distinct files. If $N = 4$, then there are *two AUGs, which are:* $\mathcal{K}_{1,m} = \{u_1, u_2, u_3, u_4, u_5\}$ *and* $\mathcal{K}_{2,m} = \{u_6, u_7, u_8\}$[3.1].

[3.1]This is one of possible AUG partitions. Optimal user grouping is not considered in this work.

3.2.3 Transmission Model

We focus on the signal transmission during the delivery phase in which the LEO satellite serves the users' requested contents. The service duration is divided into multiple time slots, whose duration is determined by the channel coherence time. The satellite-user channels are assumed quasi-static within one time slot and vary from one time slot to another. For a particular time slot, the signal received by the UE u in group k of AUG a ($k \in \mathcal{K}_{a,m}$) in spot beam m not in the HO time (see Section 3.4 for HO transmission) can be written as:

$$y_{u,k,a,m} = \boldsymbol{h}_{u,k,a,m}^H \boldsymbol{w}_{k,a,m} s_{k,a,m} + \check{I}_m + \hat{I}_m + n_u, \tag{3.2}$$

where $\boldsymbol{h}_{u,k,a,m} \in \mathbb{C}^{N\times 1}$ is the downlink channel coefficient to UE u of multicast group k of AUG a; $\boldsymbol{w}_{k,a,m} \in \mathbb{C}^{N\times 1}$ is the precoding vector designed for the multicast group k of AUG a; $s_{k,a,m} \in \mathbb{C}$ is the data symbol requested by UEs of group k of AUG a via multicast spot beam m with $\mathbb{E}[|s_{k,a,m}|^2] = 1$; and $n_u \sim \mathcal{CN}(0, \sigma_u^2)$ is the additive white gaussian noise (AWGN). $\check{I}_m$ is intra-spot beam interference that is caused by the concurrent transmission to different user groups within the same AUG a of spot beam m, and $\hat{I}_m$ represents inter-spot beam interference caused by power leakage from the adjacent beams. They are computed as follows:

$$\check{I}_m \triangleq \sum_{k' \in \mathcal{K}_{a,m} \setminus \{k\}} \boldsymbol{h}_{u,k,a,m}^H \boldsymbol{w}_{k',a,m} s_{k',a,m}; \ \hat{I}_m \triangleq \eta_m \frac{P_{\sum_{m'}}}{B} b_{a,m},$$

where η_m is the aggregate of the m-th spot beam inter-spot beam attenuation factor and the free-space path loss, $P_{\sum_{m'}}/B$ represents the accumulated interference density caused by the adjacent spot beams, and $b_{a,m}$ is the frequency bandwidth allocated to AUG a within spot beam m. The typical value for the inter-spot beam attenuation factor is around -30 dB. After the Doppler compensation, we have $\boldsymbol{h}_{u,k,a,m} = g_u \boldsymbol{v}_u(\varphi_u)$, where g_u is the channel gain and $\boldsymbol{v}_u(\varphi_u) \in \mathbb{C}^{N\times 1}$ is the downlink array response vector for UE u, wherein φ_u is the angle of departure (AoD) [75].

We assume a Rician fading channel $g_u = \sqrt{\beta_u}\hat{g}_u$, where $\beta_u = G_m G_u M \lambda^2/(4\pi D_u)^2$ is the large-scale fading and $\hat{g}_u = \alpha_u (h_{LoS}\sqrt{\kappa_u/(\kappa_u+1)} + h_{NLoS}\sqrt{1/(\kappa_u+1)})$ denotes the small-scale fading channel model, with κ_u represents the Rician factor, $\alpha_u = \mathbb{E}\{|\hat{g}_u|^2\}$, h_{LoS} is the deterministic line-of-sight (LoS) part, and h_{NLoS} represents the non-LoS (NLoS) component.

Other parameters are given in Table 3.1. The real and imaginary parts of $\hat{g}_u$ are independently and identically distributed as $\mathcal{N}(\sqrt{\kappa_u \alpha_u / 2(\kappa_u + 1)},\, \alpha_u / 2(\kappa_u + 1))$. Assuming perfect CSI at the satellite, the signal-to-interference-plus-noise ratio (SINR) of UE u of AUG a at spot beam m is calculated as follows:

$$\gamma_{u,k,a,m} = \frac{|\boldsymbol{h}_{u,k,a,m}^H \boldsymbol{w}_{k,a,m}|^2}{\sum_{k' \in \mathcal{K}_{a,m} \setminus \{k\}} |\boldsymbol{h}_{u,k,a,m}^H \boldsymbol{w}_{k',a,m}|^2 + \hat{I}_{agg} b_{a,m}}, \tag{3.3}$$

where $\hat{I}_{agg} \triangleq (\eta_m (P_{\sum_{m'}} / B) + N_0)$ and N_0 is the noise spectral density. The impact of imperfect CSI is studied in Section 3.6.6.

The effective transmission rate of a group k within AUG a, determined by the weakest users in the group, in spot beam m, is calculated as follows:

$$R_{k,a,m} = \Phi b_{a,m} \log_2(1 + \min_u \{\gamma_{u,k,a,m}\}). \tag{3.4}$$

where $\Phi \triangleq 1 - \frac{\tau_{csi} + \tau_{pro}}{\tau_{slot}}$ accounts for the effective transmission time, τ_{slot} is the time slot duration, τ_{csi} is the max channel estimation time, and τ_{pro} is the processing time whose value largely depends upon the beamforming techniques and the hardware capability of the regenerative payload.

3.3 Problem Formulation and Proposed Solution

3.3.1 Problem Formulation

In the pursuit of full exploitation of the flexible payload, we aim to jointly design the spot beam coverage r_m, frequency bandwidth allocation, and precoding vectors towards minimizing the worst-case average delivery latency. The joint optimization problem is formulated as follows:

$$\mathcal{P}: \min_{\{\boldsymbol{w},\boldsymbol{b},\boldsymbol{r},\boldsymbol{K}\}} \quad t(\boldsymbol{w},\boldsymbol{b},\boldsymbol{r},\boldsymbol{K}) \tag{3.5a}$$

$$\text{s.t.} \quad R_{k,a,m} \geq R_{req}, \quad \forall k,a,m, \tag{3.5b}$$

$$\sum\nolimits_{k\in\mathcal{K}_{a,m}} \sum\nolimits_{a\in\mathcal{A}_m} \|\boldsymbol{w}_{k,a,m}\|^2 \leq P_{\sum}(K_m/K), \forall m, \tag{3.5c}$$

$$\sum\nolimits_{a\in\mathcal{A}_m} b_{a,m} \leq B, \quad \forall m, \tag{3.5d}$$

$$\bigsqcup\nolimits_{m\in\mathcal{M}} \pi r_m^2 \geq A_{\sum}, \tag{3.5e}$$

where $\boldsymbol{w} \triangleq \{\boldsymbol{w}_{k,a,m}\}_{\forall k,a,m}$, $\boldsymbol{b} \triangleq \{b_{a,m}\}_{\forall a,m}$, $\boldsymbol{r} \triangleq \{r_m\}_{m=1}^M$, and $\boldsymbol{K} \triangleq \{K_m\}_{m=1}^M$ are the short-hand notations; R_{req} is the minimum QoS requirement; $\mathcal{A}_m$ is the set of AUGs in spot beam m; $P_{\sum}$ is the total transmit power of LEO satellite; $K = \sum_{m=1}^M K_m$; and $A_{\sum}$ is the total service area of the LEO satellite.

The objective function $t(\boldsymbol{w},\boldsymbol{b},\boldsymbol{r},\boldsymbol{K})$ of problem (3.5) is the end-to-end transmission latency, assuming the FastForward capability [76], is computed as follows:

$$t(\boldsymbol{w},\boldsymbol{b},\boldsymbol{r},\boldsymbol{K}) = \max_{\{k,a,m\}} \left(\max\left((q_k/R_{\mathtt{X}} + D_k/c), \Pi_k\right)\right) \tag{3.6}$$

where $\mathtt{X} \triangleq \{k,a,m\}$ is the short-hand indexes, q_k is the file size, $D_k = \max_u(D_u)$ is the slant distance between LEO satellite and k-th group, c is the speed of light. In (3.6), $\frac{q_k}{R_{\mathtt{X}}}$ and $\frac{D_k}{c}$ are the transmission and the propagation delays, respectively, incurred while sending files from the LEO satellite to UEs of group k; and $\Pi_k \triangleq \frac{(1-\mu_k)q_k}{R_{BH}} + \frac{D_0}{c}$ is the transmission and propagation delay accured in the backhaul link when sending the uncached file parts from the centralized cloud to the LEO satellite, where $\mu_k \in [0,1]$ denotes the fraction of the k-th file on LEO satellite, R_{BH} is the backhaul transmission rate, and D_0 is the slant distance between the GW and the LEO satellite.

In problem $\mathcal{P}$, constraint (3.5b) guarantees the minimum users' QoS requirement; constraint (3.5c) limits the power allocated to each LEO satellite spot beam; constraint (3.5d) sets the total bandwidth at spot beam m not exceeding B. Finally, constraint (3.5e) ensures that the non-overlapping union of the coverage areas of a total number of spot beams covers at least the total service area of LEO satellite.

Difficulty to solve problem $\mathcal{P}$: The challenge in solving problem $\mathcal{P}$ lies in both the non-convexity of the objective function and constraints (3.5b) and (3.5e), which result in a non-deterministic polynomial time hard problem. In particular, the spot beam coverage partition does not only affect the user grouping but also the antenna radiation patterns and hence the effective channel gains.

3.3.2 Proposed Solution

One might optimize the spot beam coverages jointly with the bandwidth and precoding vectors for every time slot. This method, however, imposes significant computation and operating costs. Instead, we design the spot beam coverages for the whole service duration T and *decouple the original problem* $\mathcal{P}$[3.2] into two sub-problems: one optimizes the spot beam coverage area for the long-time scale T and the other optimizes bandwidth and precoding vectors for the short-time scale, e.g., on a time slot basis.

Minimization of the Spot Beam Coverage Area

Unlike conventional payload, the full-digital payload offers full flexibility to design spot beam shapes optimized to the geographical users distribution. Since the effective channel gain is inversely proportional to the spot beam coverage, we aim to minimize the total multi-spot beams coverage while guaranteeing all the users are within the LEO satellite's coverage. The multi-spot beam coverage design is formulated as follows:

$$
\begin{aligned}
\mathcal{P}_1 : \min_{\{r_m\}} \quad & \textstyle\sum_{m \in \mathcal{M}} \pi r_m^2 && \text{(3.7a)}\\
\text{s.t.} \quad & \textstyle\bigsqcup_{m \in \mathcal{M}} \pi r_m^2 \geq A_{\sum}, \quad \forall m, && \text{(3.7b)}\\
& 0 < r_m \leq r_{\text{Max}}, \forall m, && \text{(3.7c)}
\end{aligned}
$$

where r_{Max} is the maximum spot beam radius.

Intuitively, problem $\mathcal{P}_1$ aims at finding the optimal radius of M non-overlapping spot beams, while ensuring that all the users are within the coverage of the designed spot beams, as stated in constraint (3.7b). To solve the problem $\mathcal{P}_1$, we employ the K-Means++ [77] clustering technique. The clustering is done based on the position of U UEs that demand the service, so the spot beam center is likely to point in the direction where the number of UEs is dominant. Since there are M spot beams, the U UEs are categorized into M clusters such

[3.2] It is only efficient when the geographical distribution of users and requested contents changes at a much slower rate than the time slot duration.

that m-th spot beam serves m-th cluster. The problem $\mathcal{P}_1$ can be reformulated in terms of clustering as follows:

$$\mathcal{P}_1^{'} : \min_{\{\mathcal{U}_m, \boldsymbol{c}_{\mathcal{U}_m}, r_m\}} \quad \sum\nolimits_{m\in\mathcal{M}} \pi r_m^2 \tag{3.8a}$$

$$\text{s.t.} \quad (3.7c); \; \bigsqcup\nolimits_{m\in\mathcal{M}} \mathcal{U}_m == \mathcal{U}, \tag{3.8b}$$

$$(\boldsymbol{y}_u - \boldsymbol{c}_{\mathcal{U}_m})(\boldsymbol{y}_u - \boldsymbol{c}_{\mathcal{U}_m})' \leq r_m^2, \quad \forall u, m, \tag{3.8c}$$

where $\boldsymbol{y}_u$ is the 2-D coordinate of user u, $\mathcal{U}_m$ is the set of UEs in the m-th cluster and $\boldsymbol{c}_{\mathcal{U}_m}$ is the two-dimensional (2D) centroid of m-th cluster. Constraint (3.8b) ensures that all unique UEs lie within the total service area of LEO satellite; constraint (3.8c) guarantees that UEs are clustered based on the Euclidean distance between $\boldsymbol{y}_u$ and $\boldsymbol{c}_{\mathcal{U}_m}$, which is bounded by the radius of coverage of the cluster, i.e., r_m.

The procedure to obtain $\mathcal{U}_m$, $\boldsymbol{c}_{\mathcal{U}_m}$, and r_m is shown in Algorithm 3.3. To find the boundaries of the clusters, Voronoi tessellation technique [78] is used, where the boundaries of the Voronoi polygons are computed using $\boldsymbol{c}_{\mathcal{U}_m}$. However, for mathematical tractability, the coverage area of the spot beam is considered circular. Using the outputs of Algorithm 3.3, user grouping is done as shown in Section 3.2.2 to get $\mathcal{A}_m$, $\mathcal{K}_{a,m}$, and K_m, which are used in solving the second sub-problem.

Minimization of Content Delivery

Once the spot beams are determined, we are ready to optimize the bandwidth allocation and precoding vectors to minimize content delivery latency. We assume that the time slot duration is sufficient for the satellite to serve the current users' requests, and the joint bandwidth and precoding vectors design is formulated as follows:

$$\mathcal{P}_2 : \min_{\{\boldsymbol{b}, \boldsymbol{w}\}} \quad t(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{r}, \boldsymbol{K}) \tag{3.9a}$$

$$\text{s.t.} \quad (3.5c); \; (3.5d); \; b_{a,m} \Phi \log_2(1 + \min_u \{\gamma_{u,k,a,m}\}) \geq R_{req}, \forall\, k, a, m, \tag{3.9b}$$

where $t(\boldsymbol{w}, \boldsymbol{b}, \boldsymbol{r}, \boldsymbol{K})$ is given in (3.6).

The problem $\mathcal{P}_2$ is non-convex due to the objective function and the constraint (3.9b). To tackle this difficulty, we introduce slack variables $z_{k,a,m}$, $\gamma_{k,a,m}$ and reformulate $\mathcal{P}_2$ into a more tractable form as follows:

$$
\begin{aligned}
\mathcal{P}_2^{'}: & \min_{\{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\gamma},\boldsymbol{z}\}} \max_{k,a,m} \left(\max\left(\frac{q_k}{z_{k,a,m}}+\frac{D_k}{c},\Pi_k\right)\right) && \text{(3.10a)}\\
\text{s.t. } & b_{a,m}\Phi\log_2(1+\gamma_{k,a,m})\geq z_{k,a,m}, \forall k,a,m, && \text{(3.10b)}\\
& (|\boldsymbol{h}^H_{u,k,a,m}\boldsymbol{w}_{k,a,m}|^2)/(\textstyle\sum_{k'\in\mathcal{K}_{a,m}\setminus k}|\boldsymbol{h}^H_{u,k,a,m}\boldsymbol{w}_{k',a,m}|^2+\hat{I}_{agg}b_{a,m})\geq\gamma_{k,a,m}, \quad \forall u,k,a,m, && \text{(3.10c)}\\
& b_{a,m}\Phi\log_2(1+\gamma_{k,a,m})\geq R_{req}, \quad \forall k,a,m, && \text{(3.10d)}\\
& \text{(3.5c)}, \text{(3.5d)},
\end{aligned}
$$

where $\boldsymbol{\gamma}\triangleq\{\gamma_{k,a,m}\}_{\forall k,a,m}$ and $\boldsymbol{z}\triangleq\{z_{k,a,m}\}_{\forall k,a,m}$.

The main challenge in solving problem $\mathcal{P}_2^{'}$ lies in the first three constraints, i.e., (3.10b), (3.10c), and (3.10d). We can handle constraint (3.10b) by considering the slack variable $x_{k,a,m}$, which can be reformulated as:

$$\Phi\log_2(1+\gamma_{k,a,m})\geq x_{k,a,m}, \tag{3.11}$$

$$b_{a,m}x_{k,a,m}\geq z_{k,a,m}. \tag{3.12}$$

Constraint (3.11) is convex, and to deal with constraint (3.12), we use an equivalent representation as:

$$\text{(3.12)} \Leftrightarrow (b_{a,m}+x_{k,a,m})^2\geq 2z_{k,a,m}+b^2_{a,m}+x^2_{k,a,m}, \tag{3.13}$$

which has a difference-of-convex form as both sides are convex functions. The difference-of-convex programming in constraint (3.13) can be easily tackled using the iterative-based SCA method by taking the first-order approximation of the left-hand-side (LHS) of the constraint (3.13). Let $\bar{b}_{a,m}$ and $\bar{x}_{k,a,m}$ be the feasible values of the constraint (3.13) in the current iteration. In the next iteration, the constraint (3.13) can be approximated as a convex constraint as:

$$2(b_{a,m}+x_{k,a,m})(\bar{b}_{a,m}+\bar{x}_{k,a,m})-(\bar{b}_{a,m}+\bar{x}_{k,a,m})^2\geq 2z_{k,a,m}+b^2_{a,m}+x^2_{k,a,m}. \tag{3.14}$$

To tackle the non-convexity of constraint (3.10c), we represent it in an equivalent form as:

$$(|\boldsymbol{h}^H_{u,k,a,m}\boldsymbol{w}_{k,a,m}|^2)/\gamma_{k,a,m}\geq\textstyle\sum_{k'\in\mathcal{K}_{a,m}\setminus\{k\}}|\boldsymbol{h}^H_{u,k,a,m}\boldsymbol{w}_{k',a,m}|^2+\hat{I}_{agg}b_{a,m}. \tag{3.15}$$

Algorithm 3.3 Iterative Algorithm to Solve (3.8a)

Input: U, $A_{\sum}$, $\boldsymbol{y}_u$, M
Output: $\mathcal{U}_m$, $\boldsymbol{c}_{\mathcal{U}_m}$, r_m
  $Init : \{\boldsymbol{c}_{\mathcal{U}_m}\}_{m=1}^{M}$, $i = 1$, I_{max}, $dis = []$, ϵ, $err = 1$
Based on $\boldsymbol{y}_u$, apply the K-Means++ cluster Alg.
while $err > \epsilon$ and $i < I_{max}$ **do**
  Calculate $\mathcal{U}_m$ using K-Means++ Alg.
  Compute $\boldsymbol{c}_{\mathcal{U}_m}^{(i)} = \text{mean}\{\boldsymbol{y}_u \mid u \in \mathcal{U}_m\}$
  Compute $err = |\boldsymbol{c}_{\mathcal{U}_m}^{(i)} - \boldsymbol{c}_{\mathcal{U}_m}^{(i-1)}|$
  Update $\boldsymbol{c}_{\mathcal{U}_m}^{(i-1)} \leftarrow c_{\mathcal{U}_m}^{(i)}$; $i \leftarrow i+1$
end while
Calculate radius r_m of the cluster $\mathcal{U}_m$
for $m = 1$ to M **do**
  for $j = 1$ to length$\{\mathcal{U}_m\}$ **do**
    Compute
    $dis^{(j)} = \sqrt{(\boldsymbol{y}_u - \boldsymbol{c}_{\mathcal{U}_m})(\boldsymbol{y}_u - \boldsymbol{c}_{\mathcal{U}_m})'}$
  end for
  Compute $r_m = \lceil(\max\{dis\})\rceil$
end for

Since the constraint (3.15) is also in difference-of-convex form, we use the SCA method to solve it iteratively. Taking $\bar{w}_{k,a,m}$ and $\bar{\gamma}_{k,a,m}$ as the feasible value, (3.15) can be approximated as:

$$\frac{2\boldsymbol{w}_{k,a,m}^{H}\boldsymbol{H}_{u,k,a,m}\bar{\boldsymbol{w}}_{k,a,m}}{\bar{\gamma}_{k,a,m}} - \gamma_{k,a,m}\frac{\bar{\boldsymbol{w}}_{k,a,m}\boldsymbol{H}_{u,k,a,m}\bar{\boldsymbol{w}}_{k,a,m}}{\bar{\gamma}_{k,a,m}^{2}} \geq \sum\nolimits_{k' \in \mathcal{K}_{a,m} \backslash k} \boldsymbol{w}_{k',a,m}^{H}\boldsymbol{H}_{u,k,a,m}\boldsymbol{w}_{k',a,m} + \hat{I}_{agg}b_{a,m}, \tag{3.16}$$

where $\boldsymbol{H}_{u,k,a,m} \triangleq \boldsymbol{h}_{u,k,a,m}\boldsymbol{h}_{u,k,a,m}^{H}$. Using (3.14) and (3.16), the problem $\mathcal{P}_2^{'}$ can be approximated by a convex optimization problem $\mathcal{P}_2^{''}$ as:

$$\mathcal{P}_2^{''}(\bar{\boldsymbol{w}},\bar{\boldsymbol{b}},\bar{\boldsymbol{x}},\bar{\boldsymbol{\gamma}}) : \min_{\{\boldsymbol{w},\boldsymbol{b},\boldsymbol{\gamma},\boldsymbol{x},\boldsymbol{z}\}} \max_{k,a,m} \left(\max \left(\frac{q_k}{z_{k,a,m}} + \frac{D_k}{c}, \Pi_k \right) \right) \tag{3.17a}$$

$$\text{s.t. } (3.5c), (3.5d), (3.11), (3.14), (3.16),$$

$$\Phi \log_2(1 + \gamma_{k,a,m}) \geq \frac{R_{req}}{b_{a,m}} \; \forall k, a, m, \tag{3.17b}$$

where $\boldsymbol{x} \triangleq \{x_{k,a,m}\}_{\forall k,a,m}$ and (3.17b) is directly obtained from (3.10d).

The problem $\mathcal{P}_2^{''}$ is a convex problem and it can be solved directly using the interior

---

Algorithm 3.4 Iterative Algorithm to Solve (3.9a)

Input: $\mathcal{A}_m$, $\mathcal{K}_{a,m}$, K_m, $\boldsymbol{h}_{u,k,a,m}$, μ_k, D_k, c, R_{BH}, η_m, $P_{\sum_{m'}}/B$
Output: $\boldsymbol{w}^*_{k,a,m}$, $b^*_{a,m}$, $x^*_{k,a,m}$, $\gamma^*_{k,a,m}$, $z^*_{k,a,m}$
Init: $\bar{\boldsymbol{w}}_{k,a,m}$, $\bar{b}_{a,m}$, $\bar{x}_{k,a,m}$, $\bar{\gamma}_{k,a,m}$, $\bar{z}_{k,a,m}$, $i = 1$, I_{max}, ϵ, $err = 1$
while $err > \epsilon$ and $i < I_{max}$ **do**
    Solve (3.17a) to get $\boldsymbol{w}^*_{k,a,m}$, $b^*_{a,m}$, $x^*_{k,a,m}$, $\gamma^*_{k,a,m}$, $z^*_{k,a,m}$
    Compute $t^{(i)}$
    Compute $err = |t^{(i)} - t^{(i-1)}|$
    Update $\bar{\boldsymbol{w}}_{k,a,m} \leftarrow \boldsymbol{w}^*_{k,a,m}$; $\bar{b}_{a,m} \leftarrow b^*_{a,m}$; $\bar{x}_{k,a,m} \leftarrow x^*_{k,a,m}$; $\bar{\gamma}_{k,a,m} \leftarrow \gamma^*_{k,a,m}$; $t^{(i-1)} \leftarrow t^{(i)}$; $i \leftarrow i + 1$
end while

---

point method [79]. Since the solutions of problem $\mathcal{P}_2^{''}$ should satisfy all the constraints of problem $\mathcal{P}_2$, the solution provided by problem $\mathcal{P}_2^{''}$ is sub-optimal for problem $\mathcal{P}_2$ and also depends largely on the initialization of the parameters $\bar{\boldsymbol{w}}, \bar{\boldsymbol{b}}, \bar{\boldsymbol{x}}$, and $\bar{\boldsymbol{\gamma}}$. Therefore, we propose Algorithm 3.4 to solve (3.9).

3.3.3 Complexity of the Proposed Algorithm

The computational complexity of Algorithm 3.3 is $\mathcal{O}(2MUI_{max} + MK_m)$ [80]. Assuming that the interior point method is used to solve the convex problem (3.6), in the worst case the complexity is equal to the cube of the number of real variables [79]. Since there are $\lceil K_m/N \rceil N^2 + \lceil K_m/N \rceil N$ real variables in the problem (3.6), the complexity for solving (3.6) is $\mathcal{O}(MI_{max}\left(\lceil K_m/N \rceil N^2 + \lceil K_m/N \rceil N\right)^3)$.

3.4 HO Scenario and Channel Prediction

Due to the short service duration of each LEO satellite pass, HO is important in LEO satellite constellations to guarantee a smooth service. To ensure proper HO in LEO satellite networks, the satellites involved in the HO process must have sufficient time to communicate with each other via an ISL, provided that they are capable of providing service to the same target area [66]. Additionally, at the beginning of the HO period, the first (departing) LEO satellite informs the second (approaching) LEO satellite about the allocation of spot beams.

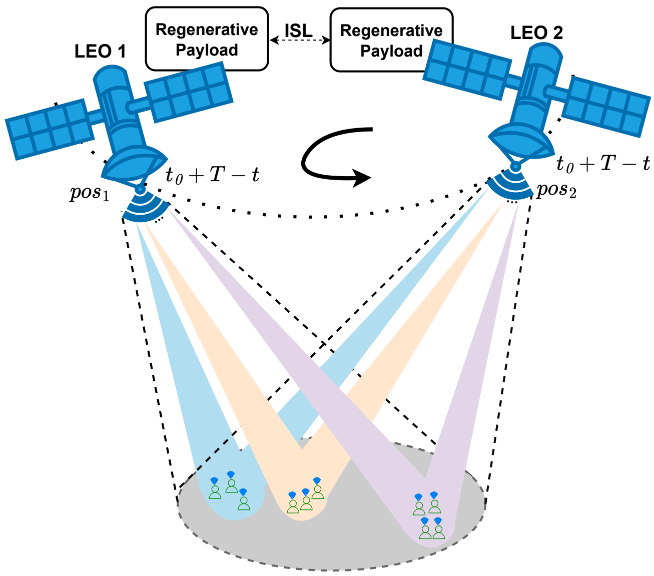

Figure 3.2: HO scenario in a downlink multicasting LEO satellite system.

3.4.1 Joint Transmission during the HO Period

To improve the service performance during the HO period, we propose a joint transmission scheme, in which two satellites are jointly sending data to the same UEs for spatial diversity, assuming that the LEOs involved in the HO process are perfectly synchronized, as depicted in Fig. 3.2. Denote $\mathtt{X} \triangleq \{k, a, m\}$, the signal received by UE u can be written as:

$$y_{u,\mathtt{X}}^{HO} = (\boldsymbol{h}_{1,u,\mathtt{X}}^H \boldsymbol{w}_{1,\mathtt{X}} + \boldsymbol{h}_{2,u,\mathtt{X}}^H \boldsymbol{w}_{2,\mathtt{X}}) s_{\mathtt{X}} + \check{I}_m^{HO} + \hat{I}_m^{HO} + n_u, \tag{3.18}$$

where $\boldsymbol{h}_{i,u,\mathtt{X}}$ and $\boldsymbol{w}_{i,\mathtt{X}}$ are the downlink channel coefficient and precoding vectors from LEO satellite $i = 1, 2$ to the target UE, and $\check{I}_m^{HO}$ and $\hat{I}_m^{HO}$ are the intra-spot beam interference and inter-spot beam interference, respectively. $\check{I}_m^{HO}$ and $\hat{I}_m^{HO}$ are defined as follows:

$$\check{I}_m^{HO} \triangleq \sum_{k' \in \mathcal{K}_{a,m} \setminus k} \left(\boldsymbol{h}_{1,u,\mathtt{X}}^H \boldsymbol{w}_{1,\mathtt{X}'} + \boldsymbol{h}_{2,u,\mathtt{X}}^H \boldsymbol{w}_{2,\mathtt{X}'} \right) s_{\mathtt{X}'}, \tag{3.19}$$

$$\hat{I}_m^{HO} \triangleq \left(\eta_{1,m} P_{1,\sum_{m'}} / B + \eta_{2,m} P_{2,\sum_{m'}} / B + N_0 \right) b_{a,m}^{HO}, \tag{3.20}$$

where $\mathtt{X}' \triangleq \{k', a, m\}$, $\eta_{1,m}$ and $\eta_{2,m}$ represent the aggregated inter-spot beam attenuation factor and the free-space path loss radiated by LEO satellite 1 and 2, respectively. Similarly, $(P_{1,\sum_{m'}}/B)$ and $(P_{2,\sum_{m'}}/B)$ represent the accumulated interference density caused by the adjacent spot beams radiated by LEO satellite's 1 and 2, respectively, and $b_{a,m}^{HO}$ is the bandwidth of AUG a within spot beam m during the HO period.

The SINR during the HO period can be written as $\gamma_{u,\mathtt{X}}^{HO} =$

$$\frac{|\boldsymbol{h}_{1,u,\mathtt{X}}^{H}\boldsymbol{w}_{2,\mathtt{X}} + \boldsymbol{h}_{2,u,\mathtt{X}}^{H}\boldsymbol{w}_{2,\mathtt{X}}|^2}{\sum_{k'\in\mathcal{K}_{a,m}\setminus\{k\}}|\boldsymbol{h}_{1,u,\mathtt{X}}^{H}\boldsymbol{w}_{1,\mathtt{X}'} + \boldsymbol{h}_{2,u,\mathtt{X}}^{H}\boldsymbol{w}_{2,\mathtt{X}'}|^2+\hat{I}_{agg}^{HO}b_{a,m}^{HO}}, \tag{3.21}$$

where $\hat{I}_{agg}^{HO} \triangleq \Big(\eta_{1,m}(P_{1,\sum_{m'}}/B) + \eta_{2,m}(P_{2,\sum_{m'}}/B) + N_0\Big)$.

The minimum effective transmission rate during the HO period can be expressed as:

$$R_{\mathtt{X}}^{HO} = b_{a,m}^{HO}\Phi\log_2\Big(1 + \min_u\big\{\gamma_{u,\mathtt{X}}^{HO}\big\}\Big). \tag{3.22}$$

It is worth noting that the achievable rate in (3.22) can be only realized if the precoding vectors are properly designed, which requires the CSI from both LEO satellites. Due to the difference in operating frequencies between the uplink and downlink in satellite communications, the UEs must provide feedback on the downlink CSI to the LEO satellite to design the precoding vectors.

During the HO process, LEO satellite 1, located at position pos_1, sends pilot signals to single-antenna UEs to estimate downlink CSI and maintain active links. The UEs provide the estimated CSI to LEO satellite 1, which then applies precoding and initiates data transmission. The estimated downlink CSI is assumed to be perfect and remains unchanged when received by LEO satellite 1. Meanwhile, LEO satellite 2 at position pos_2 uses a DL-based model to determine the downlink channel, which allows synchronized transmission of the same data symbols as LEO satellite 1. In Section 3.5, we explore various HO techniques to achieve this synchronization.

3.4.2 DL-based Downlink CSI Prediction

The conventional communication protocol is not designed to facilitate joint transmission between two LEO satellites, where the channel estimation period is designated to estimate CSI from one LEO satellite at a time. With a single antenna, the UEs cannot estimate the CSI from both LEO satellites without having the communication protocol modified, e.g., a change in the frame structures. To avoid such modification and minimize the CSI estimation time, we propose a DL-based channel prediction scheme applied to LEO 2 (the

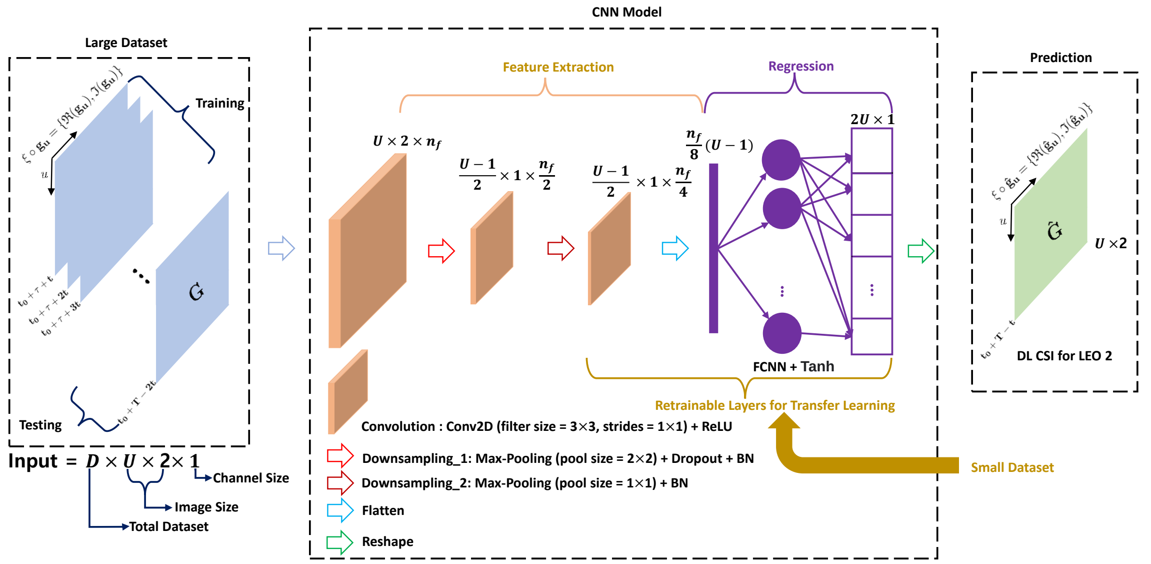

Figure 3.3: DL-based 2D-CNN model for downlink CSI prediction.

departing satellite) during the HO period, and apply channel estimation to LEO 1 (the entering satellite), as shown in Fig. 3.3. Since the departing satellite has already served the UEs in the current serving period, we can utilize the historical CSI estimates to predict the CSI during the HO time. On the other hand, the entering satellite does not possess any historic CSI measurements. Hence, its CSI can only be estimated through conventional pilot-assisted CSI estimation.

In particular, a DL-based 2D-convolutional neural network (CNN) model [34] is employed to predict the downlink CSI for LEO satellite 2 at position pos_2. The downlink CSI, i.e., $\boldsymbol{h}_u$ depends on the downlink channel gain, i.e., g_u and array response vector, i.e., $\boldsymbol{v}_u(\varphi_u)$. Taking into account that the UEs position is static, $\boldsymbol{v}_u(\varphi_u)$ can be pre-determined based on the position of the LEO satellite and the position of the UE. Thus, to predict $\hat{g}_u$ of U UEs for HO duration, i.e., $(t_\circ + T - \tau)$ at once, the g_u of U UEs during time t are vertically stacked to form $\boldsymbol{G}_t \in \mathbb{C}^{U\times 1}$ matrix. The historical data of the $\boldsymbol{G}$ is taken as image input data with two channels, then processed by the l_c convolution layers sequentially, which is three in our case, then flattened and processed by the single fully connected neural network (FCNN), and finally reshaped to get $\hat{\boldsymbol{G}}_{t_\circ+T-\tau}$. The u-th row of $\hat{\boldsymbol{G}}_{t_\circ+T-\tau}$ corresponds to $\hat{g}_{u,t_\circ+T-\tau}$. Thus, $\hat{\boldsymbol{h}}_{u,t_\circ+T-\tau} = \hat{\boldsymbol{G}}_{t_\circ+T-\tau}(u,:)\cdot\, \boldsymbol{v}_{u,t_\circ+T-\tau}$. To meet the operational requirements of the neural network, we introduce the operator ξ to map the $\boldsymbol{G}$ from the complex domain to the real domain, i.e., $\xi \circ \boldsymbol{G} = \{\Re(\boldsymbol{G}), \Im(\boldsymbol{G})\}$. The real part $\Re(\boldsymbol{G})$ and imaginary part $\Im(\boldsymbol{G})$ can be considered as the first channel and second channel, respectively. In addition, the inverse mapping of the operator ξ is ξ^{-1}.

In Fig. 3.3, the CNN-based downlink CSI (CNN-CSI) prediction model utilizes convolutional layers to extract spatial features from the channel gain. The number of filters in the convolutional layers is set to n_f, $n_f/2$, and $n_f/4$ respectively. The first convolutional layer takes a 2D input of size $U \times 2$ and uses n_f filters of size 3×3 with a stride of 1×1. The output is then passed through a downsampling layer using max-pooling with a pool size of 2×2, along with batch normalization (BN) and dropout techniques [81, 82]. The resulting downsampled data is fed into the second convolutional layer, which further reduces the number of filters. The process is repeated in the third convolutional layer, resulting in feature maps of size $((U-1) \times 1 \times n_f)/(2 \times 4)$. These features are flattened and passed to a FCNN for regression. Each convolutional layer utilizes a rectified linear unit (ReLU) activation function to introduce non-linearity. The FCNN layer uses the hyperbolic tangent (tanh) activation function to produce outputs in the range of [-1, 1] [83]. Finally, the results of the FCNN layer are reshaped into the real and imaginary part of the channel gain matrix to get the predicted output $\{\Re(\hat{\boldsymbol{G}}_{t_\circ+T-\tau}), \Im(\hat{\boldsymbol{G}}_{t_\circ+T-\tau})\}$ of size $U \times 2$. Then, the future downlink CSI can be obtained, i.e., $\hat{\boldsymbol{h}}_u(t_\circ + T - \tau) = (\xi^{-1} \circ \{\Re(\hat{\boldsymbol{G}}_{t_\circ+T-\tau}(u,:)), \Im(\hat{\boldsymbol{G}}_{t_\circ+T-\tau}(u,:))\}) \cdot \boldsymbol{v}_u(t_\circ + T - \tau)$.

Our regression problem utilizes the mean square error (MSE) [84] method for training, and the Adam optimizer [85] is employed for weight and learning rate updates. The training of the CNN-CSI model follows the mini-batch gradient descent approach, where the dataset of size $\mathcal{D}$ is divided into $\mathcal{D}$ batches of size one. Consequently, the loss function based on the data points (pixel)-based MSE [86], for each batch $(\mathscr{d} \in \mathcal{D})$ is computed as follows:

$$\mathcal{L}_{\mathscr{d}}(\Theta) = \frac{\left\|\Re(\boldsymbol{G}_{\mathscr{d}}) - \Re(\hat{\boldsymbol{G}}_{\mathscr{d}})\right\|^2 + \left\|\Im(\boldsymbol{G}_{\mathscr{d}}) - \Im(\hat{\boldsymbol{G}}_{\mathscr{d}})\right\|^2}{d \times U \times 2}, \tag{3.23}$$

where d in the denominator represents the mini-batch size, and $U \times 2$ represents the total number of data points that make up $\boldsymbol{G}$ (including both real and imaginary parts).

The CNN-CSI model weights (Θ) are updated after each batch by minimizing the loss function $\mathcal{L}(\Theta)$. To reduce training time and propagation delay in the live network, transfer learning is employed. Initially, the CNN-CSI model is trained at the GW. Then, a new CNN model is created by freezing the first two layers of the previous model (Fig. 3.3). This new model is trained again on the computationally constrained LEO satellite 2, transferring more general features learned by the initial layers just before the HO period.

Complexity of the Proposed Algorithm

The computational complexity to train the CNN-CSI model with l_c layers is given by $\mathcal{O}(E_{max}\mathcal{D} \sum_{l=1}^{l_c} n_{f,l-1} s_{f,l}^2 n_{f,l} 2U)$ [87], where $n_{f,l}$ is the number of filters in the l-th layer, $s_{f,l}$ is the spatial size of the filters in the l-th layer, $\mathcal{D}$ is the total number of batches, and E_{max} is the maximum number of training epochs required to train the model.

3.5 Joint Precoding Vectors Design during the HO Period

In this section, we present the joint design of the precoding vectors at the two LEO satellites during the HO time, given the predicted CSI from the previous section. Perfect synchronization is assumed between the two LEO satellites during the HO. We propose two collaboration schemes for computing the precoding vectors: i) *centralized collaboration (CC)*, in which the precoding vectors are computed centrally [69, 88, 89] at the GW and ii) *distributed collaboration (DC)*, in which two LEO satellites jointly compute the precoding vectors via ISL link without using the GW.

3.5.1 Centralized Collaboration

In this collaboration mode, all the computation is performed centrally at the GW which requires CSI feedback from the LEO satellites. From the system point of view, two LEO satellites are considered as parts of the *compound* antenna arrays of size $2N$. Denote $\boldsymbol{h}_{jnt,u,k,a,m} = [\boldsymbol{h}_{1,u,k,a,m}^H, \hat{\boldsymbol{h}}_{2,u,k,a,m}^H]^H \in \mathbb{C}^{2N\times 1}$ as the aggregated channel gains from two LEO satellites to the UE. We aim to design the optimal precoding vector $\boldsymbol{w}_{jnt,k,a,m} \in \mathbb{C}^{2N\times 1}$ for user group k applied to both LEO satellites.

The effective achievable rate of group k of associated group a of spot beam m during the CC-based HO (CC-HO) period, using optimal-based precoding design can be given as:

$$R_{k,a,m}^{CC,opt} = b_{a,m}^{opt} \Phi \log_2(1 + \min_u \{\gamma_{u,k,a,m}^{CC,opt}\}) \tag{3.24}$$

where $\gamma_{u,k,a,m}^{CC,opt} \triangleq \frac{|\boldsymbol{h}_{jnt,u,k,a,m}^H \boldsymbol{w}_{jnt,k,a,m}|^2}{\sum_{k'\in\mathcal{K}_{a,m}\setminus\{k\}} |\boldsymbol{h}_{jnt,u,k,a,m}^H \boldsymbol{w}_{jnt,k',a,m}|^2 + \hat{I}_{agg}^{HO} b_{a,m}^{opt}}$, wherein $b_{a,m}^{opt}$ and $\boldsymbol{w}_{jnt,k,a,m}$ are the bandwidth allocation and joint precoding vectors, respectively.

Although each LEO satellite is seen as parts of the compound antenna array of dimension $2N \times 1$, there are specific restrictions in designing the precoding vectors $\boldsymbol{w}_{jnt,k,a,m}$ in meeting the per-LEO satellite transmit power constraints. Because the first N rows of $\boldsymbol{w}_{jnt,k,a,m}$ will be applied to the LEO satellite 1 and the last N rows are applied to the LEO satellite 2, we introduce binary diagonal selection matrices $\boldsymbol{J}_1 = \text{diag}([\mathbf{1}_N, \mathbf{0}_N]) \in \{0,1\}^{2N\times 2N}$ and $\boldsymbol{J}_2 = \text{diag}([\mathbf{0}_N, \mathbf{1}_N]) \in \{0,1\}^{2N\times 2N}$. Then, the joint bandwidth and precoding vectors design can be formulated as follows:

$$\mathcal{P}^{CC,opt}: \min_{\{\boldsymbol{w},\boldsymbol{b}\}} \max_{k,a,m} \left(\max\left(\frac{q_k}{R_{\mathtt{X}}^{CC,opt}} + \frac{D_k}{c}, \Pi_k \right) \right) \tag{3.25a}$$

$$\text{s.t. } R_{\mathtt{X}}^{CC,opt} \geq R_{req}, \forall k,a,m, \tag{3.25b}$$

$$\sum\nolimits_{k\in\mathcal{K}_{a,m}} \sum\nolimits_{a\in\mathcal{A}_m} \|\boldsymbol{J}_1 \boldsymbol{w}_{jnt,\mathtt{X}}\|^2 \leq \frac{P_{\sum} K_m}{K}, \forall m, \tag{3.25c}$$

$$\sum\nolimits_{k\in\mathcal{K}_{a,m}} \sum\nolimits_{a\in\mathcal{A}_m} \|\boldsymbol{J}_2 \boldsymbol{w}_{jnt,\mathtt{X}}\|^2 \leq \frac{P_{\sum} K_m}{K}, \forall m, \tag{3.25d}$$

$$\sum\nolimits_{a\in\mathcal{A}_m} b_{a,m}^{opt} \leq B, \quad \forall m, \tag{3.25e}$$

where $\mathtt{X} \triangleq \{k,a,m\}$, $\boldsymbol{w} \triangleq \{\boldsymbol{w}_{jnt,\mathtt{X}}\}_{\forall k,a,m}$, and $\boldsymbol{b} \triangleq \{b_{a,m}^{opt}\}_{\forall a,m}$ are the short-hand notations for indexes, precoding vectors and bandwidth allocation, respectively.

We observe that problem (3.25) is similar to problem (3.9) except constraints (3.25c) and (3.25d). Fortunately, these constraints are convex, thus we can adopt the similar technique developed in Section 3.3.2. Upon obtaining the optimal precoding vectors $\boldsymbol{w}_{jnt,k,a,m}$, the GW sends the corresponding precoding coefficients to the two LEO satellites for data transmission.

3.5.2 Distributed Collaboration

Although the centralized collaboration scheme offers the optimal precoding vectors, it requires excessive signalling overhead, which motivates us to propose the distributed precoding design scheme. In this scheme, each LEO satellite compute the bandwidth allocation and precoding vectors based only on its local CSI and limited exchanged information from the other LEO satellite. Assuming a high-capacity ISL, the two LEO satellites iteratively improve its solutions via iterations. It is noted that the LEO satellites in the distributed collaboration only exchange their power scaling factors, while the bandwidth resource per user group is optimized locally.

To minimize the exchanged overhead and computation load, we consider ZF-based joint bandwidth and power allocation in this scenario. Let $\boldsymbol{W}_{i,a,m} = \boldsymbol{H}_{i,a,m}^H(\boldsymbol{H}_{i,a,m}\boldsymbol{H}_{i,a,m}^H)^{-1}$

denote the ZF-beamforming matrix for AUG a of spot beam m of LEO satellite i, $i = 1, 2$, where $\boldsymbol{H}_{i,a,m}$ is the corresponding aggregated channel matrix. Under the ZF design, the precoding vector designed at LEO satellite $i \in \{1, 2\}$ for the group $\mathtt{X} \triangleq \{k, a, m\}$ is given as $\boldsymbol{w}_{i,\mathtt{X}}^{ZF} = \sqrt{p_{i,\mathtt{X}}}\tilde{\boldsymbol{w}}_{i,\mathtt{X}}$, where $\tilde{\boldsymbol{w}}_{i,\mathtt{X}}$ is the k-th column of the ZF precoding matrix, and $p_{i,\mathtt{X}}$ is the power scaling factor. By definition, $\boldsymbol{h}_{i,u,k,a,m}^H \tilde{\boldsymbol{w}}_{i,k',a,m} = \delta_{k,k'}, \forall i, u, a, m$ assuming the accurate CSI estimation and prediction. As a result, the achievable effective rate during the DC-based HO (DC-HO) period is computed as:

$$\begin{aligned} &R_{\mathtt{X}}^{DC,ZF} \\ &= b_{a,m}^{ZF} \Phi \log_2 \Big(1 + \min_u \Big\{ \big(|\boldsymbol{h}_{1,u,\mathtt{X}}^H \boldsymbol{w}_{1,\mathtt{X}}^{ZF} + \boldsymbol{h}_{2,u,\mathtt{X}}^H \boldsymbol{w}_{2,\mathtt{X}}^{ZF}|^2\big) / \\ &\quad \Big(\sum_{k' \in K_{a,m} \setminus \{k\}} |\boldsymbol{h}_{1,u,\mathtt{X}}^H \boldsymbol{w}_{1,\mathtt{X}'}^{ZF} + \boldsymbol{h}_{2,u,\mathtt{X}}^H \boldsymbol{w}_{2,\mathtt{X}'}^{ZF}|^2 + \hat{I}_{agg}^{HO} b_{a,m}^{ZF} \Big) \Big\} \Big) \\ &= b_{a,m}^{ZF} \Phi \log_2 \Big(1 + \frac{(\sqrt{p_{1,\mathtt{X}}} + \sqrt{p_{2,\mathtt{X}}})^2}{\hat{I}_{agg}^{HO} b_{a,m}^{ZF}} \Big), \forall k, a, m. \end{aligned} \tag{3.26}$$

Denote $\alpha_{1,\mathtt{X}} \triangleq \|\tilde{\boldsymbol{w}}_{1,\mathtt{X}}\|^2$ and $\alpha_{2,\mathtt{X}} \triangleq \|\tilde{\boldsymbol{w}}_{2,\mathtt{X}}\|^2$, the short-term delivery period minimization during DC-HO period under the ZF design can be formulated as:

$$\mathcal{P}^{DC,ZF} : \min_{\{\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{b}\}} \max_{\{k,a,m\}} \left(\max \left(\frac{q_k}{R_{\mathtt{X}}^{DC,ZF}} + \frac{D_k}{c}, \Pi_k \right) \right) \tag{3.27a}$$

$$\text{s.t. } R_{\mathtt{X}}^{DC,ZF} \geq R_{req}, \quad \forall k, a, m, \tag{3.27b}$$

$$\sum\nolimits_{k \in \mathcal{K}_{a,m}} \sum\nolimits_{a \in \mathcal{A}_m} \alpha_{1,\mathtt{X}} p_{1,\mathtt{X}} \leq \frac{P_{\sum} K_m}{K}, \forall m, \tag{3.27c}$$

$$\sum\nolimits_{k \in \mathcal{K}_{a,m}} \sum\nolimits_{a \in \mathcal{A}_m} \alpha_{2,\mathtt{X}} p_{2,\mathtt{X}} \leq \frac{P_{\sum} K_m}{K}, \forall m, \tag{3.27d}$$

$$\sum\nolimits_{a \in \mathcal{A}_m} b_{a,m}^{ZF} \leq B, \quad \forall m, \tag{3.27e}$$

where $\boldsymbol{p}_1 \triangleq \{p_{1,\mathtt{X}}\}_{\forall k,a,m}$, $\boldsymbol{p}_2 \triangleq \{p_{2,\mathtt{X}}\}_{\forall k,a,m}$ and $\boldsymbol{b}$ are the short-hand notations.

The problem $\mathcal{P}^{DC,ZF}$ is non-convex due to the objective function (3.27a) and (3.27b), respectively. From the implementation perspective, the computation of the precoding vectors, as well as the bandwidth allocations, have to be executed at each LEO satellite separately. Furthermore, the bandwidth allocation $\boldsymbol{b}$ must be synchronized such that they allocate the same bandwidth to the requesting UEs. To achieve this goal, we propose an iterative algorithm in which two LEO satellites consecutively optimize their power factors and the bandwidth allocation, assuming the output of the other LEO satellite is shared. In the initialization, the problem $\mathcal{P}^{DC,ZF}$ is solved in the LEO 1 considering arbitrary feasible power value $\sqrt{\bar{p}_{2,k,a,m}}$ at LEO 2. The resulting joint optimization problem can be written as:

$$\mathcal{P}^{DC,ZF}_{LEO1}(\bar{\boldsymbol{p}}_2): \min_{\{\boldsymbol{p}_1,\boldsymbol{b}_1\}} \max_{\{k,a,m\}} \Big(\max\Big(\frac{q_k}{R^{DC,ZF}_{1,\mathtt{x}}} + \frac{D_k}{c}, \Pi_k\Big)\Big) \tag{3.28a}$$

$$\text{s.t.} \quad b_{1,a,m}\Phi\log_2\Big(1 + (\sqrt{p_{1,\mathtt{x}}} + \sqrt{\bar{p}_{2,\mathtt{x}}})^2/(\hat{I}^{HO}_{agg} b_{1,a,m})\Big)$$
$$\geq R_{req}, \ \forall k,a,m; \ (3.27c), (3.27e), \tag{3.28b}$$

where $R^{DC,ZF}_{1,\mathtt{x}} \triangleq b_{1,a,m}\Phi\log_2(1 + \frac{(\sqrt{p_{1,\mathtt{x}}}+\sqrt{\bar{p}_{2,\mathtt{x}}})^2}{(\hat{I}^{HO}_{agg} b_{1,a,m})})$.

The main challenge in solving problem $\mathcal{P}^{DC,ZF}_{LEO1}$ lies in the objective function and the first constraint (3.28b). We can handle the constraint (3.28b) by considering the slack variables $x_{1,k,a,m}$, which can be reformulated as:

$$b_{1,a,m}\Phi\log_2\Big(1 + \frac{x_{1,k,a,m}}{\hat{I}_{agg} b_{1,a,m}}\Big) \geq R_{req}, \ \forall k,a,m, \tag{3.29}$$

$$\sqrt{p_{1,k,a,m}} + \sqrt{\bar{p}_{2,k,a,m}} \geq \sqrt{x_{1,k,a,m}}. \tag{3.30}$$

Proposition 3.1 *The rate function under the ZF design in (3.29) is jointly concave in $b_{1,a,m}$ and $x_{1,k,a,m}$.*

The proof of Proposition 3.1 is shown in Appendix A. The constraint (3.30) is a difference-of-convex form as both sides are convex functions. Thus, it can be efficiently solved using the iterative-based SCA method by taking the first-order approximation of the right-hand-side (RHS) of the constraint (3.30). Let $\bar{x}_{1,k,a,m}$ be a feasible value of the constraint (3.30) in the current iteration. In the next iteration, the constraint (3.30) can be approximated as a convex constraint as:

$$\sqrt{p_{1,k,a,m}} + \sqrt{\bar{p}_{2,k,a,m}} \geq \frac{x_{1,k,a,m}}{2\sqrt{\bar{x}_{1,k,a,m}}} + \frac{\sqrt{\bar{x}_{1,k,a,m}}}{2}. \tag{3.31}$$

Now, the problem $\mathcal{P}^{DC,ZF}_{LEO1}$ can be approximated by a convex optimization problem

$\mathcal{P}^{DC,ZF'}_{LEO1}(\bar{\boldsymbol{p}}_2, \bar{\boldsymbol{x}}_1)$:

$$\min_{\{\boldsymbol{p}_1,\boldsymbol{b}_1,\boldsymbol{x}_1\}} \max_{\{k,a,m\}} \left(\max \left(\frac{q_k}{R^{DC,ZF'}_{1,\mathtt{X}}} + \frac{D_k}{c}, \Pi_k \right) \right) \tag{3.32a}$$

$$\text{s.t.} \quad (3.29), (3.31), (3.27c), (3.27e),$$

where $\boldsymbol{x}_1 \triangleq \{x_{1,k,a,m}\}_{\forall k,a,m}$ and $\bar{\boldsymbol{x}}_1 \triangleq \{\bar{x}_{1,k,a,m}\}_{\forall k,a,m}$ are the short-hand notations.

Let $p^*_{1,\mathtt{X}}$ be the solution of problem (3.32). This value will be communicated to the LEO 2 to compute its optimal transmit power, i.e., $p_{2,\mathtt{X}}$. It is worth noting that there is no need to exchange the bandwidth allocation $b_{1,a,m}$ to perform the optimization in LEO satellite 2, but only for checking the termination criteria. The optimization problem in LEO 2 can be formulated similarly as in (3.28) with the satellite subscript 1 switched with 2. Following the same method, the problem $\mathcal{P}^{DC,ZF}_{LEO2}$ can be solved by using the SCA approach of its approximated convex problem $\mathcal{P}^{DC,ZF'}_{LEO2}(\bar{\boldsymbol{p}}_1, \bar{\boldsymbol{x}}_2)$:

$$\min_{\{\boldsymbol{p}_2,\boldsymbol{b}_2,\boldsymbol{x}_2\}} \max_{\{k,a,m\}} \left(\max \left(\frac{q_k}{R^{DC,ZF'}_{2,\mathtt{X}}} + \frac{D_k}{c}, \Pi_k \right) \right) \tag{3.33a}$$

$$\text{s.t.} \quad b_{2,a,m} \Phi \log_2 \left(1 + x_{2,\mathtt{X}} / (\hat{I}^{HO}_{agg} b_{2,a,m})\right) \geq R_{req}, \forall k, a, m,$$

$$\sqrt{\bar{p}_{1,\mathtt{X}}} + \sqrt{p_{2,\mathtt{X}}} \geq \frac{x_{2,\mathtt{X}}}{\sqrt{\bar{x}_{2,\mathtt{X}}}} + \frac{\sqrt{\bar{x}_{2,\mathtt{X}}}}{2}, \forall k, a, m; \ (3.27d), (3.27e),$$

where $R^{DC,ZF'}_{2,\mathtt{X}} \triangleq \Phi_{a,m} b_{2,a,m} \log_2 \left(1 + x_{2,\mathtt{X}} / (\hat{I}^{HO}_{agg} b_{2,a,m})\right)$.

The optimal output powers $p^*_{2,\mathtt{X}}$ of $\mathcal{P}^{DC,ZF'}_{LEO2}$ will be forwarded to LEO satellite 1 to execute the next iteration of optimization. The iterations will continue until the convergence of the optimal transmit powers and bandwidth allocation. The detailed steps of the proposed algorithms are presented in Algorithms 3.5 and 3.6.

3.5.3 Major Technical Challenges & Their Solutions

For the signals emitted by the LEO satellites involved in the HO process introduced in Sections 3.5.1 and 3.5.2, such that the signals add constructively to increase the total signal strength. Some of the major technical challenges that could be encountered when implementing our proposed HO approach in a real system, along with possible directions to address them, are provided below:

Algorithm 3.5 Iter. Alg. to Solve (3.27a)

Input: $\mathcal{A}_m$, $\mathcal{K}_{a,m}$, K_m, $\boldsymbol{h}_{1,u,k,a,m}$, $\boldsymbol{w}_{1,k,a,m}$, μ_k, D_k, c, R_{BH}, $\eta_{1,m}$, $P_{1,\sum_{m'}}/B$

Output: $p^*_{1,k,a,m}$, $b^*_{1,a,m}$, $x^*_{1,k,a,m}$

Initialization: $\bar{p}_{1,k,a,m}$, $\bar{x}_{1,k,a,m}$, $\bar{b}_{1,a,m}=1$, $\bar{b}_{2,a,m}=5$, $i=1$, I_{max}, ϵ, $err=1$

while ($\bar{b}_{1,a,m}\,!\!=\bar{b}_{2,a,m}$) **do**
  Solve (3.32) at LEO Satellite 1
  while $err>\epsilon$ and $i<I_{max}$ **do**
    Solve (3.32) to get $p^*_{1,k,a,m}$, $b^*_{1,a,m}$, $x^*_{1,k,a,m}$
    Compute $t^{DC,ZF(i)}_{LEO1}$ & $err=|t^{DC,ZF(i)}_{LEO1}-t^{DC,ZF(i-1)}_{LEO1}|$
    Update $\bar{x}_{1,k,a,m}\leftarrow x^*_{1,k,a,m}$; $t^{DC,ZF(i-1)}_{LEO1}\leftarrow t^{DC,ZF(i)}_{LEO1}$; $i\leftarrow i+1$
  end while
  Update $\bar{p}_{1,k,a,m}\leftarrow p^*_{1,k,a,m}$, $\bar{b}_{1,a,m}\leftarrow b^*_{1,a,m}$
  Send/Receive via ISL
end while

$$\xrightarrow[\bar{b}_{1,a,m}]{\bar{p}_{1,k,a,m}}$$

$$\xleftarrow[\bar{b}_{2,a,m}]{\bar{p}_{2,k,a,m}}$$

Algorithm 3.6 Iter. Alg. to Solve (3.27a)

Input: $\mathcal{A}_m$, $\mathcal{K}_{a,m}$, K_m, $\hat{\boldsymbol{h}}_{2,u,k,a,m}$, $\hat{\boldsymbol{w}}_{2,k,a,m}$, μ_k, D_k, c, R_{BH}, $\eta_{2,m}$, $P_{2,\sum_{m'}}/B$

Output: $p^*_{2,k,a,m}$, $b^*_{2,a,m}$, $x^*_{2,k,a,m}$

Initialization: $\bar{p}_{2,k,a,m}$, $\bar{x}_{2,k,a,m}$, $\bar{b}_{1,a,m}=1$, $\bar{b}_{2,a,m}=5$, $i=1$, I_{max}, ϵ, $err=1$

while ($\bar{b}_{1,a,m}\,!\!=\bar{b}_{2,a,m}$) **do**
  Solve (3.33) at LEO satellite 2
  while $err>\epsilon$ and $i<I_{max}$ **do**
    Solve (3.33) to get $p^*_{2,k,a,m}$, $b^*_{2,a,m}$, $x^*_{2,k,a,m}$
    Compute $t^{DC,ZF(i)}_{LEO2}$ & $err=|t^{DC,ZF(i)}_{LEO2}-t^{DC,ZF(i-1)}_{LEO2}|$
    Update $\bar{x}_{2,k,a,m}\leftarrow x^*_{2,k,a,m}$; $t^{DC,ZF(i-1)}_{LEO2}\leftarrow t^{DC,ZF(i)}_{LEO2}$; $i\leftarrow i+1$
  end while
  Update $\bar{p}_{2,k,a,m}\leftarrow p^*_{2,k,a,m}$, $\bar{b}_{2,a,m}\leftarrow b^*_{2,a,m}$
  Send/Receive via ISL
end while

- ISL-based bandwidth synchronization: When two satellites are widely separated, pointing, tracking, and acquisition required to establish ISL connection between them requires on-board special hardware embedded in them [90].

- Difference of slant distance between satellites involved in HO and a reference location [91]: To address this timing offset before transmission is required, which becomes more challenging when both the transmitter and receiver are in motion.

- CSI prediction accuracy: In the DC-HO scheme, predicting the CSI for the leaving LEO satellite relies on historical CSI information, the spatial and temporal correlation between data points, and the specific machine learning techniques used for prediction.

3.6 Performance Evaluation on Realistic System Parameters

In this section, we evaluate the performance of the proposed framework based on realistic LEO satellite parameters and Movielens dataset.

3.6.1 LEO Satellite Footprint

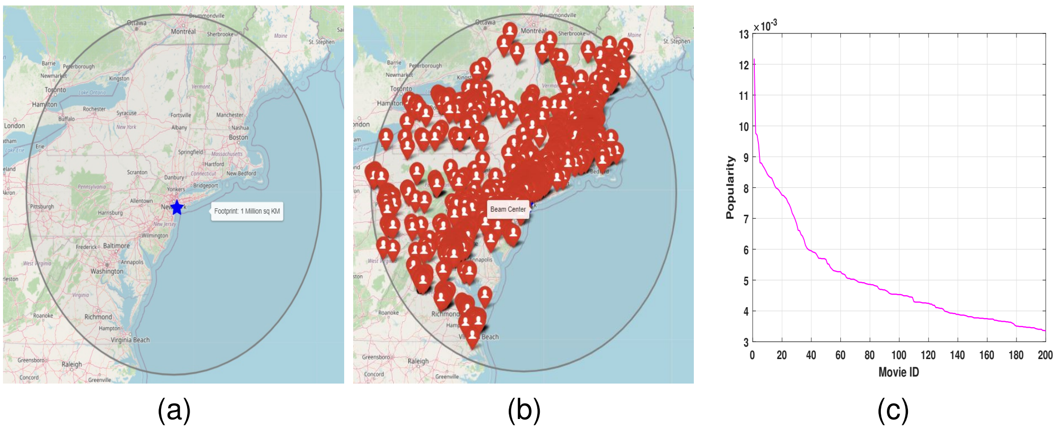

Figure 3.4: LEO satellite beam coverage and location-based UE traffic. (a) Footprint. (b) Traffic. (c) Content demand vector.

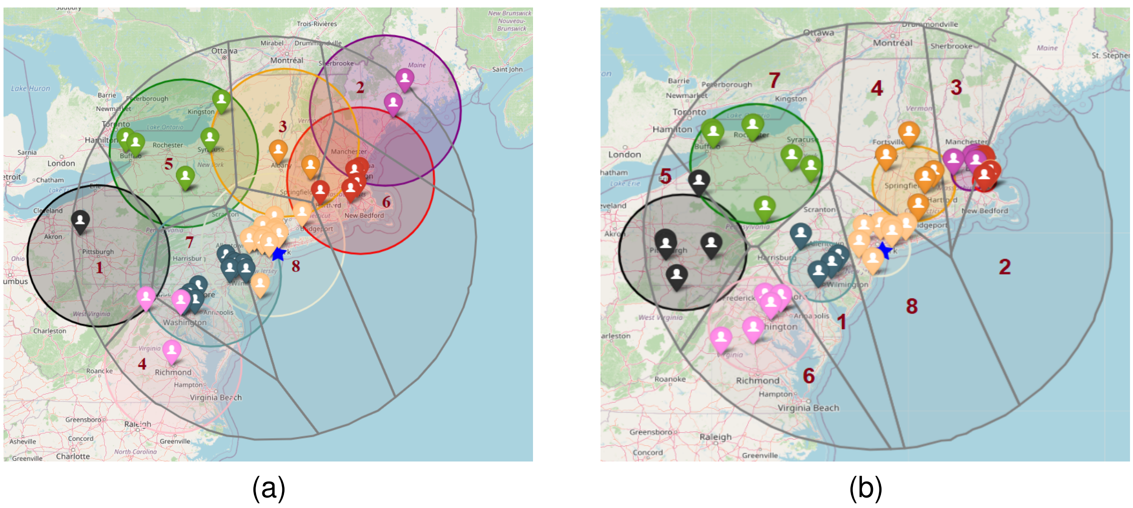

Figure 3.5: LEO satellite spot beam at time t. (a) Fixed beam. (b) Adaptive beam.

The Starlink LEO satellite 4798 is assumed to be in orbit just above New York (NY) [92]. The LEO satellite is at an altitude (H_s) of 550 km just above Earth's surface. The elevation angle ($\epsilon_\circ$) of the satellite with the Earth's center is assumed to be 40°. Based on H_s and $\epsilon_\circ$,

$A_{\sum}$ is about 1.05 million km^2 with NY as the beam center and a coverage radius (R_{LEO}) of ≈ 578 km [93]. The footprint of the LEO satellite is shown in Fig. 3.4(a).

3.6.2 Content Popularity Based on the Movielens Dataset

The content popularity is generated from the location-based Movielens dataset, in which 1M movie ratings are provided. The dataset contains UE IDs, UE locations (ZIP code), movie IDs, movie genres, and rating time, from which we can calculate the distribution of requests in any given time period. We use the ZIP code information to accurately determine the geographic distribution of requests by mapping the ZIP codes with the corresponding latitude and longitude. Since the 1M dataset covers the entire U.S., only UEs falls under the LEO's coverage of the upper part of the U.S. East Coast are considred, shown in Fig. 3.4(b). After calculating the content popularity $\boldsymbol{L}$ within the covered region, only top 200 movie ID's are taken into account. The most popular movie is indexed as 1, while the least popular is indexed as 200. The popularity of the top 200 movies within LEO satellite beam coverage region is shown in Fig. 3.4(c). For each t duration, both the location and the content requested by UEs are randomly changed based on the historical probability distribution.

3.6.3 Earth Fixed Beam Duration of LEO Satellite

In this sub-section, we calculate the elevation angle (ϵ_u) of UEs located within the footprint of the Starlink LEO satellite 4798 (see Fig. 3.5) for a total connection time of 11 to 12 minutes during one orbital period, taking into account the earth fixed beam scenario. We compute ϵ_u based on the inner product of the LEO satellite's position vector in its orbit and the position vector of the UE, using the earth-centered-earth-fixed coordinate system. Specifically, we use the formula $\epsilon_u = \sin^{-1}\left(\frac{\boldsymbol{y}_u \cdot (\boldsymbol{y}_s - \boldsymbol{y}_u)}{||\boldsymbol{y}_u||||\boldsymbol{y}_s - \boldsymbol{y}_u||}\right)$ [94], where $\boldsymbol{y}_s$ and $\boldsymbol{y}_u$ are the position vectors of the satellite and UE, respectively.

Fig. 3.6 presents the effective mean data rate for optimal and ZF-based precoding designs over time. The figure reveals a noticeable pumping effect in the data rate when the communication time between UEs and LEO satellite reaches its midpoint. This effect occurs because the latched UEs are positioned at an elevation of around 90° with respect to the LEO satellite during that period.

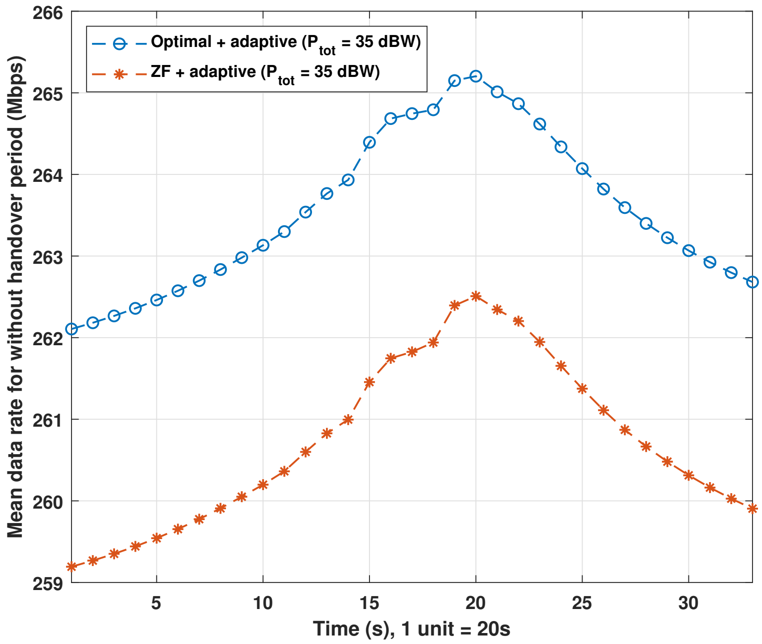

Figure 3.6: Effective mean data rate of the UEs as a function of time (elevation angle).

3.6.4 Environment Setup for the CNN-CSI Model

The training and testing dataset required for the CNN-CSI model is generated by assuming that $U = 45$ UEs are randomly located within the footprint of LEO satellite as shown in Fig. 3.5(b). The channels are time-varying and we apply the Jakes model [95] to generate the channel matrix. For data generation, we assume that the channel coefficient changes every second. Considering the earth fixed beam scenario, LEO satellite can provide service to all the requesting UEs within its footprint for around 11 minutes as shown in Fig. 3.6. Thus, the dataset of size $\mathcal{D} = 60 \times 11$ was generated to train, validate, and test the CNN-CSI model. The temporal correlation model used is $\boldsymbol{G}_t = \sqrt{\rho}\boldsymbol{G}_{t-1} + \sqrt{1-\rho}\boldsymbol{E}_t$, where $\rho \in [0,1]$ represents the correlation coefficient, and $\boldsymbol{E}_t$ is a time-independent random matrix. A correlation coefficient of 1 indicates complete channel correlation, while a coefficient of 0 indicates channel independence across different time slots.

As to the CNN-CSI model, we employ $l_c = 3$ layers where the number of filters and the filter size of corresponding layers are $\{16, 8, 4\}$ and $\{3 \times 3, 3 \times 3, 3 \times 3\}$, respectively. Other parameters are summarized in Table 3.2. Fig. 3.7 illustrates the accuracy of the CNN-CSI model in terms of MSE for different training epochs. The training and test datasets were generated with different correlation coefficient values, denoted ρ. As shown in Table 3.2, the total number of training, validation, and test dataset are 528, 130, and 1, respectively.

Since we are interested in predicting the channel coefficients of LEO satellite 2 during the HO time slot, only one test dataset is required. This means that, after the CNN-CSI model is well-trained, the data from the 659^{th} time slot is used as input to predict the next time slot's channel coefficient for LEO satellite 2. Specifically, we considered ρ values of 1, 0.9, and 0, for which the corresponding MSE values for training the model were 0.000385, 0.00185, and 0.0105, respectively. As from these results, it can be concluded that the prediction accuracy of the model CNN-CSI is higher when the temporal correlation of the channel coefficient is high and vice-versa. Additionally, the test error is slightly lower than the training error for a smaller number of training epochs, primarily due to the significantly smaller size of the test dataset and the model being inadequately trained during these early epochs. However, as the number of training epochs increases, both the training and test errors converge, and the MSE decreases, demonstrating that our CNN-CSI model is well-trained for datasets with different ρ values.

3.6.5 Performance Evaluations

Table 3.2: CNN-CSI Training Parameters

Parameters	**Value**
2D data size	45×2
Total training dataset	528
Total validation dataset	130
Test dataset	1
Maximum training epochs, E_{max}	20
Dropout	0.4
Optimizer	Adam
Activation functions	ReLU & Tanh
Loss function	MSE
Learning rate, α	0.0001

In this part, we conduct the numerical results considering a scenario where LEO satellite has a total of 8 spot beams serving 45 UEs. The UEs are randomly distributed in the coverage area as shown in Fig. 3.5. It is assumed that each spot beam is capable of transmitting $N = 4$ parallel data streams. The coverage area and the number of UEs within each spot beam is calculated using Algorithm 3.3. The spot beam with the lowest number of UEs is marked as 1, while the spot beam with the highest number of UEs is marked as 8. Fig. 3.5(a) displays

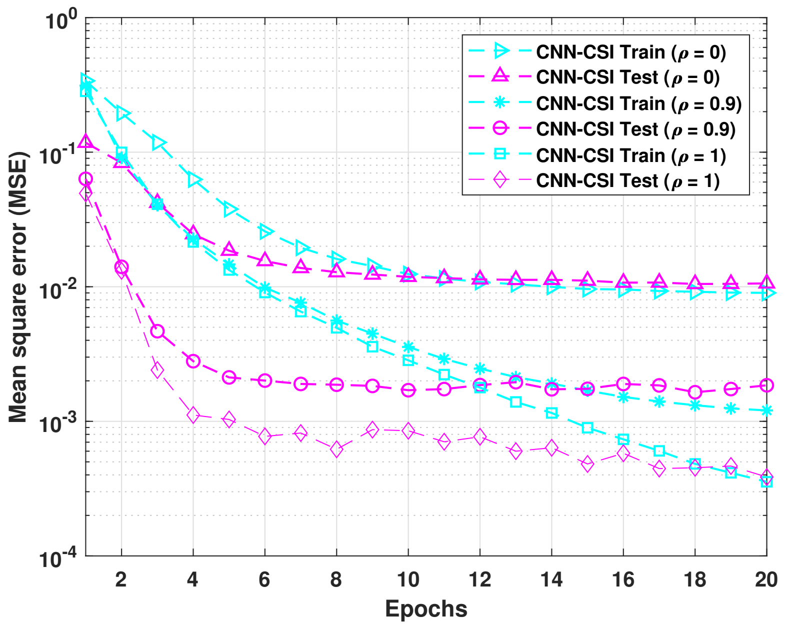

Figure 3.7: MSE vs. training epochs for CNN-CSI model.

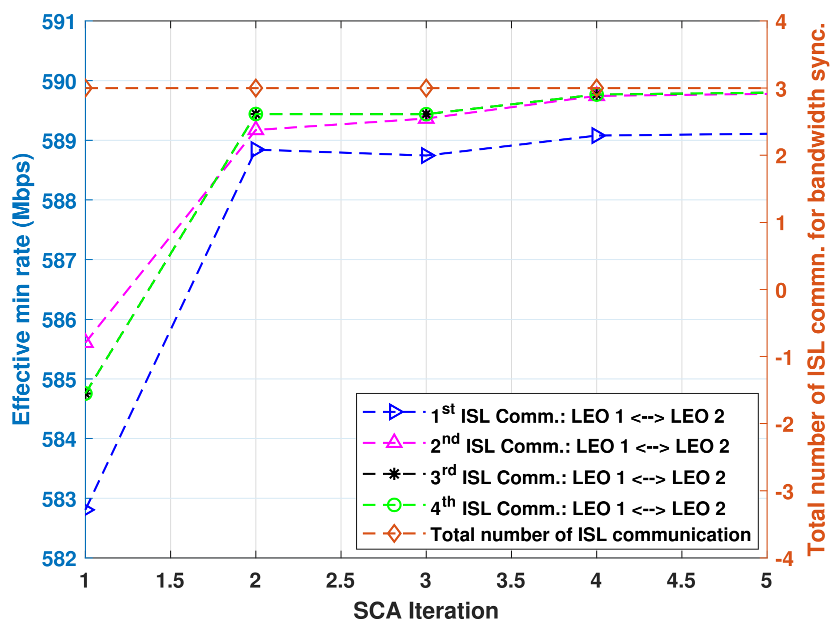

Figure 3.8: Convergence of the DC-HO algorithm.

Table 3.3: System and Channel Parameters

Parameters	**Value**
Number of UEs, U	45
Parallel data streams, N	4
Number of spot beams, M	8
Carrier frequency, f_c	2 GHz
Bandwidth, B	100 MHz
Rician factor, κ_u	10 dB
Spot beam gain, G_m	U [18 dBi, 36 dBi]
AoD, φ_u	U [-0.5, 0.5]
UE's elevation angles, ϵ_u	U [40°, 90°]
Transmit power, $P_{\sum}$	25 dBW – 35 dBW
QoS, R_{req}	$0.25 \times b_{a,m}$
Backhaul rate, R_{BH}	0.25 Gbps – 1 Gbps
Size of movies files, q	U [0.3 Gb, 1.8 Gb]
Cache storage capacity, C	215 Gb
Noise spectral density, N_0	-203.9772 dBW/Hz
Adaptive spot beam radius, r_m	U [25 km, 200 km]
Fixed spot beam radius	200 km
Earth's radius	6371 km

a fixed-spot beam, while Fig. 3.5(b) demonstrates a steerable adaptive spot beam. We adopt the LTE specifications [96], where one c.u. lasts one symbol duration, which is equal to 66.7 μs, and one block duration comprises 300 c.u. The LEO satellite is assumed to spend 1 c.u. to solve one convex optimization problem, resulting in M c.u. for solving each proposed algorithm [97]. The system and channel parameters used in the simulations are summarized in Table 3.3. The simulation results are averaged over the 100 random channel realizations. We compare the proposed framework with the following references:

- *Baseline 1:* The precoding vectors are designed based on the ZF-based approach.
- *Baseline 2:* The optimal approach is used to calculate the precoding vectors for the combined channel coefficients of the participating LEO satellites in the CC-HO method. In this approach, both satellites estimate the CSI via pilot transmission. As a result, the portion of time responsible for data transmission changes to $(1 - \frac{2\tau_{csi}+\tau_{pro}}{\tau_{slot}})$.
- *Baseline 3:* The channel estimation is similar in *Baseline 2*, except that the precoding vectors are design based on the ZF method.

The comparison with the terrestrial multicasting solution is not considered as satellite systems is design to complement the terrestrial networks.

Convergence of the Proposed Iterative Algorithm

To demonstrate the convergence of our proposed algorithms, Fig. 3.8 presents the objective function of the iterative algorithms for baseline 2 with a focus on min rate maximization (maximum content delivery delay minimization). The total transmit power of LEO satellite 1 and LEO satellite 2 is considered 25 dBW each. It is evident from the plot that the iterative algorithms exhibit rapid convergence to the sub-optimal values, requiring fewer than 5 iterations and 3 ISL communications for the DC-HO scenario.

Without HO Scenario

Fig. 3.9 portrays the mean/min data rate and outage probability between the proposed optimal design and baseline 1 for various values of LEO satellite antenna element spacing, i.e., d_{ant}. The optimal design consistently achieves higher mean/min data rates than baseline 1. At a lower d_{ant} of 0.1λ, the optimal design achieves about 3.35 times the mean/min data rate of baseline 1. In contrast, at an d_{ant} of 0.75λ, the optimal design achieves a mean/min data rate about 1.02 times that of baseline 1. Baseline 1 performs poorly at lower d_{ant} due to more sidelobes and lower spot beam directivity, leading to significant outage scenarios caused by channel correlation. In contrast, the optimal design experiences no outages. The figure reveals that increasing the d_{ant} results in higher data rates for both designs. However, increasing d_{ant} beyond 0.5λ raises the risk of grating lobes, making d_{ant} of 0.5λ desirable. At this spacing, the mean data rate for the optimal design reaches around 267 Mbps, while baseline 1 achieves approximately 257 Mbps.

Fig. 3.10 depicts the mean/min data rate and outage probability as a function of κ_u for both the optimal and baseline 1 approaches considering the adaptive beam scenario. For this result, d_{ant} is considered to be 0.5λ and $P_{\sum}$ to be 35 dBW. The κ_u is varied between 1.0233 (0.1 dB) and 100 (20 dB). From the figure, it can be seen that the data rate (mean/min) increases as the value of κ_u increases and that the data rate obtained with the optimal precoding design is higher than that of baseline 1 regardless of the κ_u value. When κ_u is 0.1 dB, the outage probability due to baseline 1 is about 17% and when κ_u is 100, the outage probability due to the baseline 1 design is 12%, while there is no outage scenario due to the optimal precoding design for the given R_{req}. Since the rates and outage probability are not significantly different between κ_u of 10 (10 dB) and 100 (20 dB), κ_u of 10 dB is considered, which is a realistic assumption.

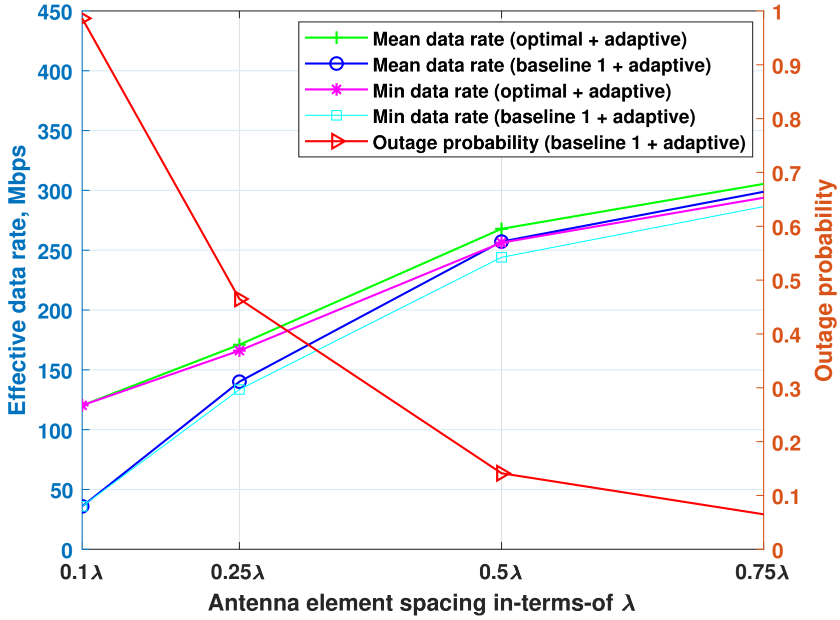

Figure 3.9: Effective mean data rate and the outage probability versus the LEO satellite array antenna element spacing (d_{ant}).

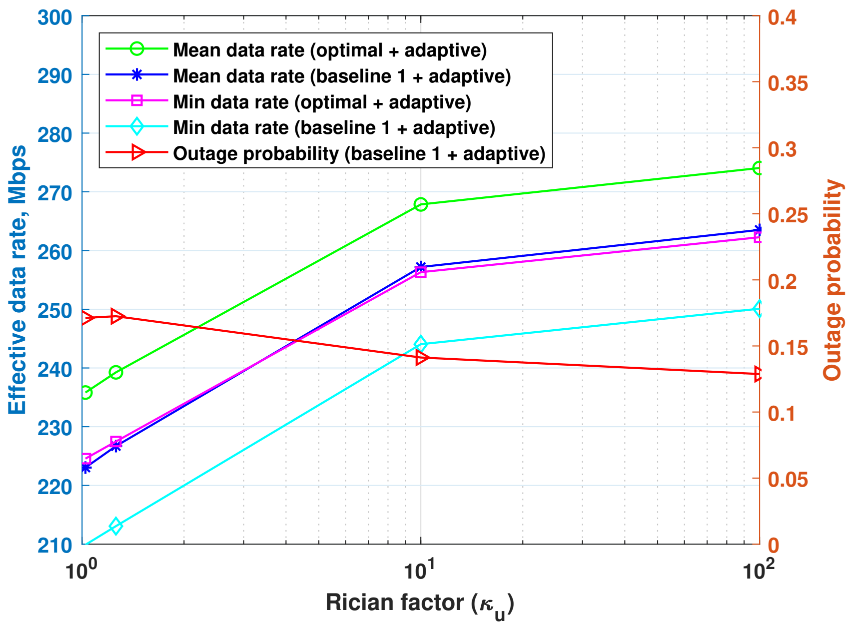

Figure 3.10: Effective mean data rate and the outage probability versus the Rician factor (κ_u).

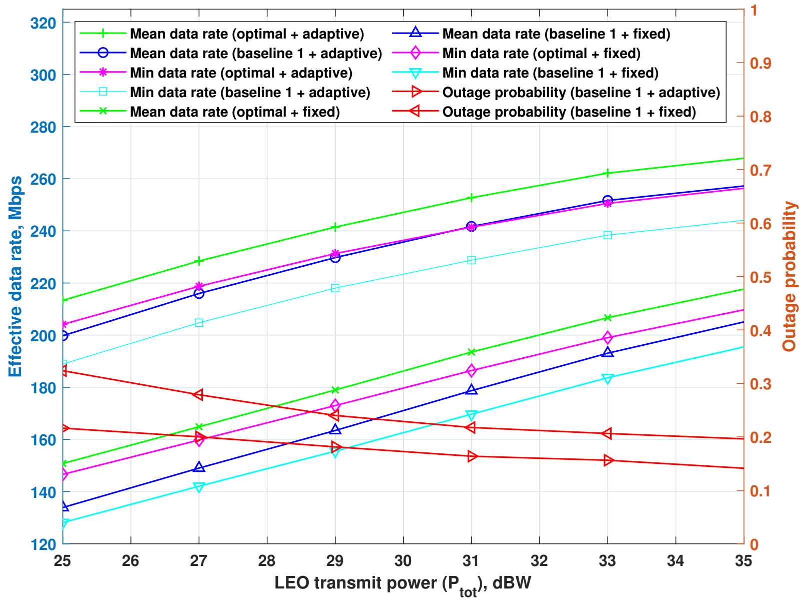

Figure 3.11: Effective data rate and the outage probability versus the total transmit power of the LEO satellite.

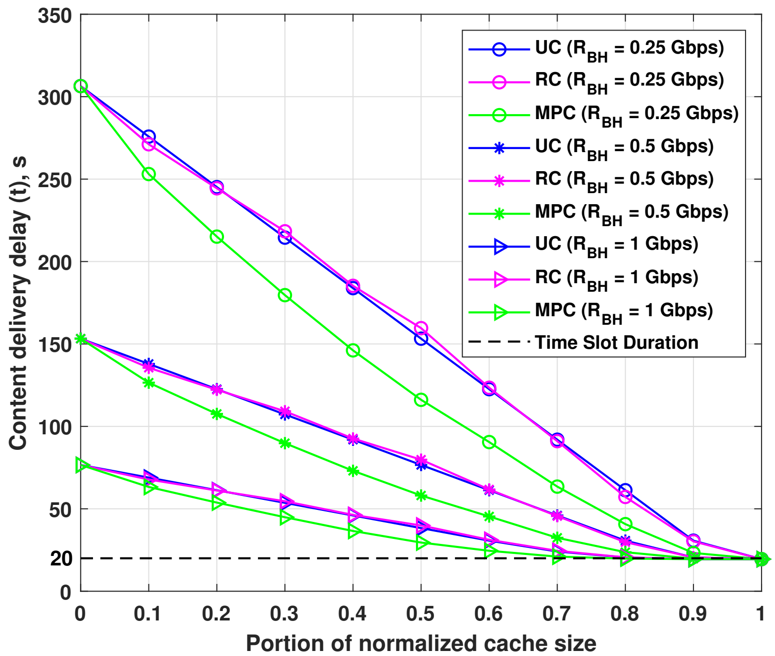

Figure 3.12: Content delivery delay versus the portion of normalized cache size for different generic caching scenarios.

Fig. 3.11 demonstrates the mean/min data rate and outage probability as a function of $P_{\sum}$, with $d_{ant} = 0.5\lambda$, $\kappa_u = 10$dB. The figure demonstrates that the mean/min data rate achieved by the adaptive beam scenario is at least 1.22 times higher than that of the fixed beam scenario, regardless of the precoding approaches. In the fixed beam scenario, all the spot beams have equal radius of 200 km, whereas the spot beam's radius in the adaptive beam scenario varies between 25 km and 200 km. The adaptive beam scenario, benefiting from improved beam directivity, outperforms the fixed-beam scenarios in terms of both data rate and outage probability. Moreover, optimal precoding achieves a significantly higher mean/min data rate than baseline 1 because it effectively eliminates intra-spot beam interference. There is no outage scenario in the optimal precoding-based design, while there are high outage probability in the baseline approach. When $P_{\sum}$ is increased, the outage probability decreases, and therefore the data rate improves comparatively more in the baseline approach than in the optimal approach. However, the increase in data rate tends to saturate when the power increase exceeds a certain limit because the inter-spot beam interference increases due to the power leakage from the neighboring beams.

Fig. 3.12 illustrates the delay as a function of normalized cache capacity for the caching models such as MPC ($\mu_k \in \{0, 1\}$), UC ($\mu_k \in [0, 1]$), and RC ($\mu_k \in \{0, 1\}$). From the figure, it can be seen that the delivery time is lower in the MPC approach compared to the UC and RC methods for small normalized cache sizes, and RC almost approaches the UC method due to the averaging of a large number of channel realizations. When the normalized cache size increases, both UC and RC approach the performance of the MPC. It can also be seen from the figure that R_{BH} also significantly affects the delivery latency for the different cache size values.

HO Scenario

Fig. 3.13 shows the relationship between the transmit power and the effective mean data rate during the HO process. The figure reveals that regardless of the precoding scheme used, the effective mean data rate during HO is consistently 1.5 times higher than the without HO period. Comparing the proposed approaches with the baselines, the ZF precoding design-based CC-HO approach outperforms baseline 3, and the optimal precoding design-based CC-HO approach outperforms baseline 2. This is because the proposed CC-HO approach estimates the CSI using a prediction model, while baseline 2 and baseline 3 estimate the CSI via pilot

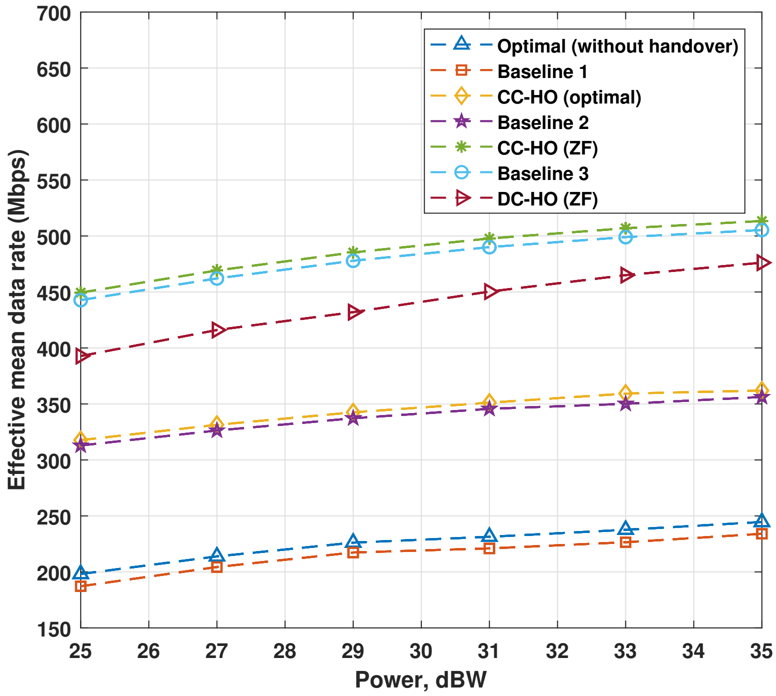

Figure 3.13: Effective mean data rate versus the total transmit power of the LEO satellites for the HO scenario.

transmission. Despite utilizing ZF-based precoding design in the CC-HO scenario, it outperforms other approaches due to the combined consideration of downlink channel coefficients and evaluation of the precoding vectors based on the combined channel coefficients, which enhances spatial diversity. The ZF-based CC-HO outperforms ZF-based DC-HO, even though both employ ZF-based precoding. This is because the former allows for full control in the design of the precoding vector and bandwidth allocation. In contrast, the latter requires synchronization of resources between the LEO satellites involved in the HO, which might not always be guaranteed.

3.6.6 Impact of Imperfect CSI

In previous sections, the proposed framework assumes perfect CSI at the satellite. In realistic conditions, the satellite operates based on the imperfect channel estimation $\hat{\boldsymbol{h}}_{u,k,a,m} = \boldsymbol{h}_{u,k,a,m} + \boldsymbol{e}$, where $\boldsymbol{h}_{u,k,a,m}$ is the true channel and $\boldsymbol{e}$ is the estimation error that is independent from the true channel and follows $\mathcal{CN}(0, \boldsymbol{I}\sigma_e^2)$. Since the precoding vectors are designed

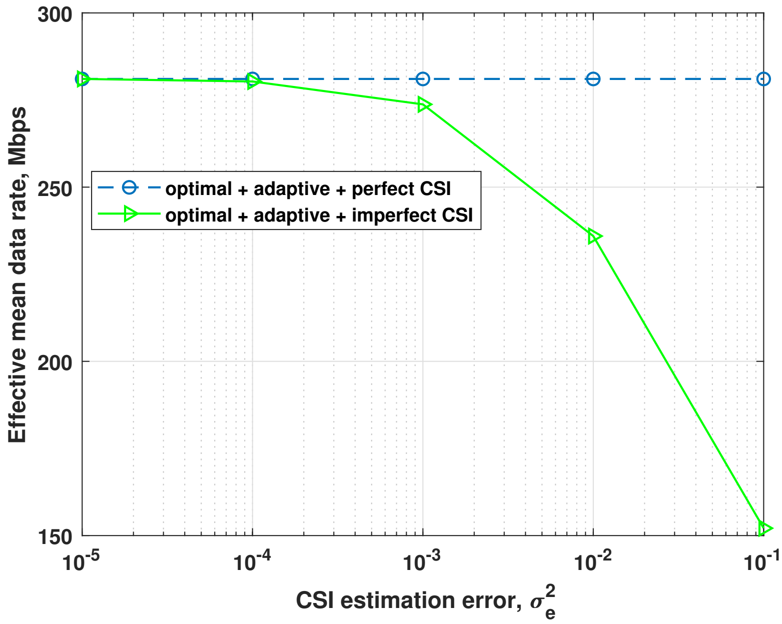

Figure 3.14: Effective mean data rate versus error variance (σ_e^2) on the imperfect CSI for non-HO scenario for a multi-spot beam multicasting LEO satellite system ($P_{\sum} = 35$ dBW).

based on the estimated channels, the SINR in this case equals to:

$$\frac{|\hat{\boldsymbol{h}}_{u,k,a,m}^H \boldsymbol{w}_{k,a,m}|^2}{\sum_{k' \neq k} |\hat{\boldsymbol{h}}_{u,k,a,m}^H \boldsymbol{w}_{k',a,m}|^2 + \sum_{\forall k'} \sigma_e^2 \|\boldsymbol{w}_{k',a,m}\|^2 + \hat{I}_{agg} b_{a,m}}, \tag{3.34}$$

where the summation of k' in the denominator is over $\mathcal{K}_{a,m}$. Optimal bandwidth allocation and precoding vectors design under imperfect CSI can be obtained similarly in Section 3.3.2, with one modification of adding $\sum_{\forall k'} \sigma_e^2 \|\boldsymbol{w}_{k',a,m}\|^2$ to the RHS of (3.16).

Fig. 3.14 presents the impact of imperfect CSI on the proposed optimal precoding design. The robustness of our design is demonstrated via a close performance to the perfect CSI case for estimation error up to 10^{-3}. When the estimation error is large, the achievable rate dramatically degrades as expected.

3.7 Summary

In this chapter, we have proposed a FLARE-LEO framework that effectively exploits the fully flexible regenerative payload capability of LEO satellites via joint design of spot beam coverage, adaptive beamforming, caching, multiuser precoding, and dynamic bandwidth allocation using realistic system parameters. In addition, we proposed innovative HO architectures that consider computational capability and overhead requirements. Using numerical results, we demonstrated that our adaptive beamforming design outperforms the fixed beam design in terms of both data rate and outage probability. We also showed that collaboration between two LEO satellites during the HO period significantly boosts the system performance. In general, the optimal precoding design outperforms the ZF-based precoding design, resulting in both an improvement in data rate and a reduction in the content delivery latency. Furthermore, we have shown that the MPC-based caching method performs better than caching strategies based on RC and UC and significantly improves the average content delivery latency for content delivery compared to a scenario without caching.

Chapter 4

LEO-based Edge Computing Service Platform for Challenging Geographical Terrain

The next-generation fully regenerative payload-enabled low Earth orbit (LEO) satellites enable Satellite-as-a-Service (SaaS) capabilities. In this work, we propose a three-tier SaaS platform for on-demand computing services such as disaster forecast, 2D/3D scene observation, route finding, and rescue operations based on satellite images/videos, taking into account different beam management techniques. Unlike existing works where LEO satellites collaborate with ground users, in the considered system, users only request a service index, while all computation is fully handled within the satellite network and/or its connected gateway (GW). The primary objective is to minimize the worst-case service completion time by optimizing the computation and communication allocation among the satellites and the GW. The formulated multi-objective optimization problem is characterized as non-linear, non-convex, and non-deterministic polynomial-time (NP)-hard. To tackle this challenge, we propose an iterative algorithm based on the alternating optimization (AO) framework. This approach solves two sub-problems in sequence during each iteration: i) optimizing the link selection and task offloading decisions via a successive convex approximation (SCA) method, and ii) optimizing computational resources and downlink communication bandwidth via a convex formulation. Furthermore, we propose a delivery strategy during handover (HO) periods for returning the task results under both Earth-fixed beam and Earth-moving beam configurations. Simulation results based on realistic system parameters and datasets show that our proposed design reduces the task completion time by at least 16% compared to the reference strategies. The impact of computational cycles, inter-satellite link rates, feeder link rates, and HO decisions under different beam configurations is illustrated in terms of mean task completion time.

4.1 Introduction

With the proliferation of virtual reality (VR) and data-hungry applications, such as real-time video analytics [29] and 2D/3D scene observation [30], the video data traffic is predicted to constitute at least 80% of global mobile data traffic [41] by 2028. As data-hungry applications are more likely to consume high computational efforts and energy, various research efforts have been undertaken to partially or fully offload tasks from resource-constrained user equipment (UE) and the internet of things (IoT) devices to terrestrial mobile edge computing (MEC) server [29, 98, 99]. Such offloading is especially critical for near real-time services [100].

Although MEC in terrestrial networks makes it possible to push the heavy computational burden to edge computing systems, the limited coverage of terrestrial edge equipment make it difficult to provide sustainable services to large number of UEs/IoT devices. Various studies have suggested alternative offloading solutions, such as utilizing unmanned aerial vehicles (UAVs) [101–103] and high altitude platform station (HAPS) systems [104–106]. However, the limitations in coverage and concurrent massive connections for both UAVs and HAPS make them unsuitable for scenarios where telecom operators aim to provide services to larger areas with limited resources. Additionally, maneuvering HAPS is challenging due to their shapes and the atmospheric drag they experience [107].

Fortunately, recent advances in fully digital payload LEO satellite networks, characterized by their low propagation delays compared to other constellations [42], provide an alternative for backhaul/fronthaul connectivity. These satellites also support near real-time broadband services, offering a larger field of view (FoV) than UAVs and HAPs in all geographical terrain [108]. Fueled by SpaceX [109] and OneWeb [110], ongoing LEO constellation projects aim to launch thousands of satellites to establish a mega constellation, collaborating with traditional operators to enhance high-capacity and seamless communication services. One of the important research problems is how to optimize the limited computational capabilities of

LEO satellites to provide efficient LEO-based computing services [111–120].

4.1.1 Related Works

In recent years, several studies have been focused on task offloading design in space-air-ground integrated networks (SAGIN) [111–120]. In [111], an LEO-assisted edge cloud computing scheme is presented, in which a computation task can be executed at either ground users, LEO satellites, or a cloud-aided gateway (GW). This computational architecture assumes binary task-offloading under limited communication and computation resources. The authors in [112, 113] overcome the assumption of fixed communication and computation resources through a joint optimization of binary task offloading and computational resource allocation, focusing on rural areas without access to the terrestrial network. Similarly, in [114], the system model is similar to [112, 113], except that it incorporates a caching scenario for task offloading. The authors of [115, 116] minimize the total energy consumption of the task offloading by jointly optimizing communication, computation, and binary offloading decisions, constrained by the tolerable deadlines of tasks. It should be noted that these works overlook HO, a key feature in LEO systems, and task output, often assuming the output size is negligible. While this assumption holds for tasks like classification or problem-solving, it falls short for 2D/3D scenery tasks, such as generating images of interest areas, where the output size is significant.

To tackle the dynamics in SAGIN, reinforcement learning (RL)-based task offloading approaches have been proposed [116–120]. Pioneering work on RL-based task offloading in an SAGIN environment is presented in [117] to minimize total delay. However, the work therein does not consider the dynamic topology of satellites, UAVs, and task arrivals. To address the drawback of [117], the authors of [118] consider the topology dynamics and propose an optimal offload route to minimize the total delay assuming static communication bandwidth and task arrivals. Additional network dynamics such as channel rates, task arrival times and time-varying topology are considered in [119, 120].

It is worth noting that the aforementioned works [111–120] primarily adopt binary or a few discrete levels for offloading decisions in satellite networks. This approach may be attributed to the lightweight and non-fragmentable nature of tasks under consideration, as well as the significant computational demands associated with RL-based offline training and potential scalability challenges. In addition, the inter-satellite propagation delay and HO periods, which are critical in LEO satellite networks, remain unaddressed. To address this research gap and the limitations of [111–120], there is a need for a scalable and efficient algorithm with practical applicability and the potential to incorporate additional layers for data protection and privacy [121].

Existing works [111–120] primarily focus on transmitting data from ground-based sensors for computation in SAGIN, whereas the chapters in [121, 122] explore a futuristic approach to joint communication and computation within SAGIN or orbital networks. This approach enables high-resolution imagery services in remote areas without transmitting raw data. Applications include water monitoring, smart oilfields, precision agriculture, wildlife tracking, disaster management, and search and rescue. It also supports data-intensive technologies like VR, autonomous driving, remote sensing, and space situational awareness.

4.1.2 Contributions

In this chapter, we propose a three-tier SaaS platform for on-demand computing, focusing on near-real-time imagery-based Earth observation (EO) services [31]. This platform leverages real-time synthetic aperture radar (SAR) and optical imagery-based EO services to enable 2D/3D scene observation, improve disaster prediction and response, and support evacuation planning and damage assessment [32, 33].

In the proposed system, when a UE located in challenging geographical terrain wants to access the imagery service, it sends a lightweight request to its serving LEO satellite. This request contains only a service index or the coordinates of the target area. Upon receiving the request, the serving LEO satellite can perform all required computations for the requested service, or offload parts of the task to neighboring satellites and/or the GW. Once all computations are completed, the serving LEO satellite delivers the final result, such as a 2D/3D image of the requested area, to the UE. It is worth noting that our computation task model is fundamentally different from those in [111–120].

Our objective is to jointly optimize computation and communication resources, considering key factors such as LEO service duration, task output size, tolerable deadline, HO impact, and imperfect channel state information (CSI).

The specific contributions of this work are summarized as follows:

- We formulate a multi-objective optimization framework for SaaS edge computing, leveraging parallel and partial task offloading to optimize computation and communication resources. The problem, classified as NP-hard, considers realistic constraints such as LEO satellites' service duration, task requirements, and energy limitations.

- We propose an alternating optimization (AO) framework to solve the proposed NP-hard problem, decomposing it into a non-convex (mixed-integer) task offloading sub-problem and a convex resource allocation sub-problem. The former is solved via a successive convex approximation (SCA) method, shown to converge to the optimal branch-and-bound (BnB) solution.

- Unlike prior works [111–120] that assume negligible output sizes, we focus on imagery-related processing, where output data size is significant. Firstly, we optimize communication strategies across feeder, ISL, and return links alongside computation resources. Additionally, we address HO challenges in LEO satellites, considering beam management techniques like *Earth-fixed beam* and *Earth-moving beam*, and analyze the impact of imperfect CSI via a robust transmission design [123].

- Finally, we conceptually illustrate the practical implementation of the proposed SaaS architecture using a container-based platform (e.g., Kubernetes), highlighting vertical and horizontal autoscaling based on dynamic task loads [124–127]. Numerical results show that our proposed method reduces the task completion time by at least 16% compared to the reference offloading strategies. The study also analyzes how computational cycle requirements, ISL, and feeder-link rates influence offloading decisions, resource utilization, and mean task completion time.

The remainder of this chapter is organized as follows. Section 4.2 describes the system model and parameters. Section 4.3 presents the problem formulation. Section 4.4 introduces the decomposition-based approach to address the proposed problem. Section 4.5 presents the impact of beam management techniques on task offloading and delivery strategies. Section 4.6 demonstrates the effectiveness of the proposed scheme using numerical results. Section 4.7 presents an open discussion on key research questions, such as *how the serving LEO satellite can identify its adjacent neighboring LEO satellites* and *how the proposed framework can be practically implemented using an open-source container orchestrator*. Finally, Section 4.8 concludes the chapter.

4.2 System Model

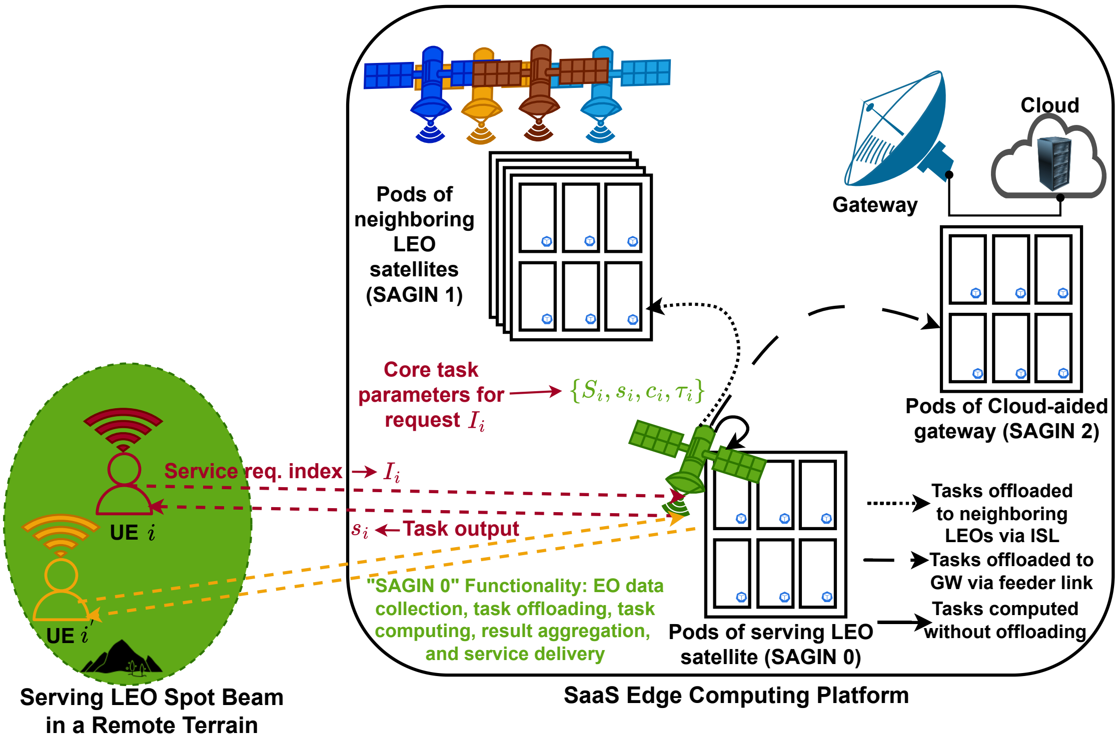

Figure 4.1: LEO satellite-assisted on-demand EO service task offloading.

We consider a *space-ground integrated*[4.1] hierarchical (three-tier) edge computing system [108], comprising a serving LEO satellite (first-tier), adjacent LEO satellites including both intra- and inter-orbit satellites (second-tier), and the cloud-aided GW (third-tier), which are collectively referred to as SAGIN nodes (SAGINs). The FoV of the LEO networks is considered to be non-overlapping; therefore, the ground users are connected to only one serving LEO satellite at a time. This system is designed to dynamically transfer a portion of tasks from the serving LEO satellite to both adjacent LEO satellites and the cloud-aided GW based on traffic load, ensuring near real-time services in remote terrains, as depicted in Fig. 4.1.

For ease of presentation, we index the three-tier SAGIN hierarchical edge computing architecture using $n \in \mathcal{N} = \{0, 1, \ldots, N-1\}$, where $N = 3$. Specifically, SAGIN 0 ($n = 0$) represents the serving LEO satellite, SAGIN 1 ($n = 1$) denotes a logical group of neighboring

[4.1] Aerial platforms such as UAVs and HAPs are not considered in our system model due to their frequent energy requirements and the limited availability of energy sources in remote terrains with harsh weather conditions.

Table 4.1: Summary of Main Notations

Notations	**Description**
M	Number of adjacent LEO satellites
T, Δ, δ^{mon}	Total number of time slots in the service duration of each LEO satellite, the time slot duration, and the uplink service request duration (monitoring window) within a time slot
I_i	On-demand EO service task request from UE i to serving LEO satellite
$S_{i,t}$, $s_{i,t}$, $c_{i,t}$, $\tau_{i,t}$	Input size, output size, computational cycle, and maximum tolerance time of the task i, respectively
$\alpha_{i,t}^{(1)}, \alpha_{i,t}^{(2)}$	Binary variables to indicate the selection of SAGINs 1 and 2, respectively, to handle the offloaded task i
$\beta_{i,t}^{(n)}$	Partial offloading decision of the requested task i to the SAGIN n
$f_{i,t}^{(n)}$	Computational capability to handle task i within SAGIN n
$b_{i,t}$	Downlink bandwidth required to transmit task i result from SAGIN 0 to the UE
$\kappa^{(n)}$	Energy factor of LEO satellites and cloud

LEO satellites treated collectively as one entity for simplicity[4.2], and SAGIN 2 ($n = 2$) corresponds to the GW. As illustrated in Fig. 4.1, the computation at each SAGIN node is containerized into pods to enable parallel and partial task execution, thereby reducing processing latency. Each pod executes a portion of the tasks associated with a specific application.

Given the focus on challenging geographical terrains, the applications involve real-time image/video processing tasks, such as disaster detection, emergency response coordination, and scene observation [32, 33]. The system supports imagery-based applications where users request inference tasks from raw data (images or videos) captured by SAGIN 0. Due to limited computational capacity, SAGIN 0 may offload some tasks to SAGIN 1 or SAGIN 2.

Having established the hierarchical architecture of the SAGIN system, we next describe its high-level operation and the workflow for service request submission and computation, with detailed analysis of communication, computation, and return link latency presented in subsequent subsections.

[4.2] The selection of a specific neighboring LEO satellite is not considered in this work, and neighboring satellites are assumed to be always available when needed. A detailed discussion on satellite selection is provided in Section 4.7.

4.2.1 System Operation Overview

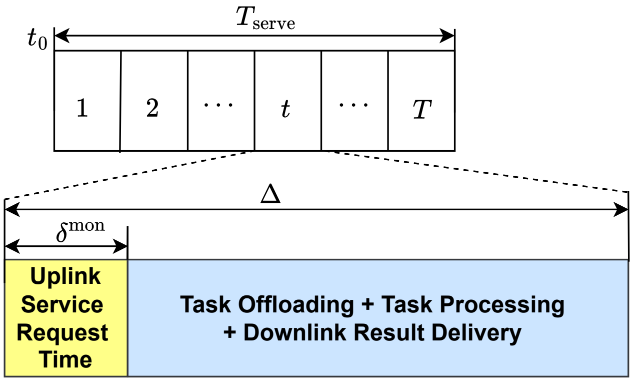

Figure 4.2: Block diagram of a single communication (uplink request and downlink transmission) block occurring within the service duration T_{serve} of the serving LEO satellite.

In the proposed framework, UEs in the remote terrains first initiate the computation requests for EO-based services, which are received by the SAGIN 0 (the serving LEO satellite). To ensure low-latency and adaptive service delivery, the total service duration T_{serve} of SAGIN 0 is divided into T equal configuration time slots of duration Δ, as illustrated in Fig. 4.2. The set of these time slots is denoted by $\mathcal{T} = \{1, \ldots, t, \ldots, T\}$. SAGIN 0 is assumed to operate in time-division duplexing (TDD) mode, which offers greater flexibility than frequency-division duplexing (FDD) by allowing the dynamic adjustment of uplink and downlink periods based on instantaneous traffic demands [128]. The main notations used in this chapter are summarized in Table 4.1.

At the beginning of each time slot t, the system monitors incoming requests during a short, fixed monitoring interval of duration δ^{mon}. Only requests arriving within this monitoring window are considered for processing in the current slot, while those arriving during the remaining interval, i.e., $\Delta - \delta^{\text{mon}}$, must wait until the next time slot. Based on the demand observed during δ^{mon}, computational resources across the hierarchical SAGINs are dynamically reallocated.

Following resource reallocation, the requested tasks are executed using a parallel and partial offloading strategy, wherein different portions of a task may be distributed across multiple SAGINs. The computed results are then aggregated and transmitted back to the requesting UEs within the same slot, ensuring timely and efficient service delivery.

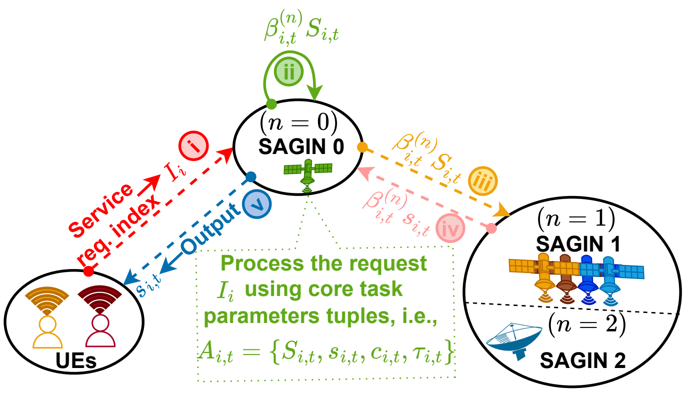

Figure 4.3: Service requests and computation process.

4.2.2 Service Requests and Computation

The LEO SaaS platform is capable of providing a fixed list of pre-defined EO services, such as object detection, high-resolution imagery for scene observation (25 to 50 cm), weather forecasting, disaster prediction, route finding during disasters, damage assessment, reconstruction monitoring, and more [32,33]. These services can be accessed through web apps on the requesting UEs' devices within the current LEO service duration, denoted as T_{serve}. Each service is listed in the web app with a unique index number. To request a service, the UE must provide the corresponding EO service index number, the priority order of the task, and the preferred latitude and longitude location within the FoV of the serving LEO satellite.

Let $\mathcal{U}_t = \{1, 2, \ldots, u, \ldots, U_t\}$ denote the set of U_t UEs requesting computational services at the time slot t, which includes both new arrival users and unserved UEs from the previous period. For simplicity, the subscript i is used to represent the user index within $\mathcal{U}_t$. As illustrated in Fig. 4.3, when UE i submits a request I_i at time t, SAGIN 0 (the serving LEO) processes the request using the core parameter tuple, i.e., $\mathcal{A}_{i,t} = \{S_{i,t}, s_{i,t}, c_{i,t}, \tau_{i,t}\}$, where $S_{i,t}$ is the input EO data size limit (in bits) to be collected by SAGIN 0 (by using optical lens or SAR), $s_{i,t}$ is the expected output size (in bits), $c_{i,t}$ is the computational cycle requirement (in cycles/sec), and $\tau_{i,t}$ is the maximum tolerance time (in sec) to return the output to UE.

The metadata of the request I_i includes contextual information such as coordinates and expected resolution. This information is provided to SAGIN 0 in JavaScript object notation (JSON) format by the UEs for on-demand service, whose size is negligible. When SAGIN 0 is unable to duly compute all requested tasks, it offloads (indicated by offloading decision variable $\beta^{(n)}$) part of the tasks to SAGIN 1 (the neighboring LEO satellites) or SAGIN 2 (the GW), depending on the link quality and computational capabilities of those SAGINs. The detailed process of end-to-end latency involved in performing an on-demand service request on a SaaS platform and delivering the final output to the requesting UEs is discussed in Sections 4.2.3–4.2.5.

4.2.3 Communication Model

This subsection presents the transmissions between SAGIN 0 (the serving LEO satellite) and the UEs, SAGIN 0 and SAGIN 1 (the neighboring LEO satellites), and between SAGIN 0 and SAGIN 2 (the GW).

Communications between SAGIN 0 and UEs

We consider a TDD system where UE communicates with the SAGIN 0 through strictly separated uplink (UE to SAGIN 0) and downlink (SAGIN 0 to UE) transmissions. Each communication block consists of an uplink request phase with duration δ^{mon} and a complementary downlink result transmission phase occupying the remaining slot duration $\Delta - \delta^{\text{mon}}$, as shown in Fig. 4.2. The system operates using the full available bandwidth $B^{(0)}_{\text{UE},\sum}$ in both directions. We assume frequency-division multiple access (FDMA) for both uplink and downlink, where the total bandwidth is equally divided among the active UEs in each time slot.

Although the size of the service request I_i (i.e., $\mathsf{s}_i^{\text{req}}$ bits) is typically negligible (as discussed in Section 4.2.2), its uplink latency, given by $l^{\text{ul}}_{i,t} = \mathsf{s}_i^{\text{req}}/r^{\text{ul}}_{i,t}$, must satisfy $l^{\text{ul}}_{i,t} \leq \delta^{\text{mon}}$ for a successful request. This latency is often small compared to other delay components. Nonetheless, for completeness, we explicitly model both transmission directions.

The uplink rate for UE i at time t is given by:

$$r^{\text{ul}}_{i,t} = b^{\text{ul}}_{i,t} \log_2\left(1 + \frac{p^{\text{ul}}_{i,t} g^{\text{ul}}_{i,t}}{b^{\text{ul}}_{i,t} N^{\text{ul}}_0}\right), \quad \forall i, t, \tag{4.1}$$

where $b^{\text{ul}}_{i,t} \triangleq \frac{B^{(0)}_{\text{UE},\sum}}{U_t}$ is the uplink bandwidth allocated to UE i, $p^{\text{ul}}_{i,t}$ is the uplink transmit power, $g^{\text{ul}}_{i,t}$ is the uplink channel gain, and N^{ul}_0 is the uplink noise power spectral density.

Given the negligible size of I_i, $l^{\text{ul}}_{i,t}$ does not influence the task allocation optimization, as computation is performed entirely within the space segment. Therefore, we shift our focus to the transmission of task results from SAGIN 0 to the UEs. Assuming that there is perfect CSI available (imperfect CSI will be discussed in Section 4.6.6), the achievable transmit rate from the serving LEO satellite to the UE i at the time slot t can be written as:

$$r_{i,t}^{(0)} = b_{i,t} \log_2 \left(1 + \frac{p_{i,t}^{(0)} g_{i,t}^{(0)}}{b_{i,t} N_0}\right), \quad \forall i, t, \tag{4.2}$$

where the superscript (0) in the symbols refers to SAGIN 0 (the serving LEO), $b_{i,t}$ is the downlink bandwidth allocated to UE i at the time slot t, $g_{i,t}^{(0)}$ is the downlink channel gain, $p_{i,t}^{(0)} \triangleq \frac{P_{\mathrm{UE},\sum}^{(0)}}{U_t}$ is the transmit power of the serving LEO for the single-antenna UE i, wherein $P_{\mathrm{UE},\sum}^{(0)}$ stands for the total transmit power. Furthermore, N_0 represents the downlink noise power spectral density. The downlink channel coefficient between the serving LEO satellite and UE i after Doppler compensation can be modeled using a Rician fading channel as in [75].

Communication between SAGIN 0 and SAGIN 1

The ISL communication between SAGIN 0 and its neighbor SAGIN 1 is usually a free-space optical link generated from vertical-cavity surface-emitting laser diodes, with the assumption that spaceborne laser terminals in the LEO satellites possess acquisition, pointing, and tracking capabilities [129, 130]. Thus, the ISL capacity R_{ISL} is usually large. In principle, the ISL rate $r_{i,t}^{(1)}$ bits-per-second (bps) allocated to UE i at time slot t can be jointly optimized with other design variables, e.g., offloading decision and computation resource. However, such joint consideration further complicates the overall system design. Therefore, to focus on cooperative computations among SAGIN nodes and communications in the return-link, we allocate the ISL capacity based on the task input and output sizes, given as: $r_{i,t}^{(1)} = (s_{i,t} + S_{i,t}) R_{\mathrm{ISL}} / (\sum_{i'=1}^{U_t} (s_{i',t} + S_{i',t}))$. Obviously, $\sum_{i \in \mathcal{U}_t} r_{i,t}^{(1)} = R_{\mathrm{ISL}}$.

Communication between SAGIN 0 and SAGIN 2

The feeder link rate between SAGIN 0 and SAGIN 2 for handling the request of UE i at the time slot t can be represented as $r_{i,t}^{(2)}$ bps for both uplink and downlink. Similarly to the ISL case, we allocate the feeder link rate to UE i as: $r_{i,t}^{(2)} = (s_{i,t} + S_{i,t}) R_{\mathrm{GW}} / (\sum_{i'=1}^{U_t} (s_{i',t} + S_{i',t}))$, where R_{GW} is the total feeder link capacity.

4.2.4 Computation Model

Denote $\beta_{i,t}^{(n)} \in [0,1], n = 0,1,2$ as the fraction task for user i at time slot t performed at SAGIN n. Similarly, $f_{i,t}^{(n)}$ denotes the computation capacity devoted for user i at time slot t at SAGIN n. In this chapter, we only consider processing time, transmission time, and propagation delay in total latency, while excluding potential queuing delays that may arise from system overload due to a high volume of on-demand service requests. Existing queuing policies in the SAGIN environment [119, 120, 131] can be modified to align with digital video broadcasting-native internet protocol (DVB-NIP) standard [26], allowing seamless integration into our system with minimal modification, thereby reducing queuing latency by prioritizing task backlogs. Additionally, incorporating blockchain technology into the SaaS platform can enhance EO data protection by recording data with immutable cryptographic signatures through hash functions and consensus mechanisms [132], without altering our proposed algorithm, while Kubernetes facilitates the easier deployment of this security enhancement.

Computation at SAGIN 0 (the serving LEO satellite)

The computation latency for performing a portion of the task in SAGIN 0 at time slot t can be written as:

$$l_{i,t}^{(0)} = \beta_{i,t}^{(0)} c_{i,t} / f_{i,t}^{(0)}, \quad \forall i, t. \tag{4.3}$$

The corresponding energy consumption in the serving LEO satellite can be written as [111, 114–116, 118, 119]:

$$\begin{aligned} e_{i,t}^{(0)} = & \underbrace{\beta_{i,t}^{(0)} \kappa^{(0)} (f_{i,t}^{(0)})^2 c_{i,t}}_{\mathrm{I:Processing}} + \underbrace{\frac{\beta_{i,t}^{(1)} p_{i,t}^{(1)} S_{i,t}}{r_{i,t}^{(1)}}}_{\mathrm{II:comms.\ to\ SAGIN\ 1}} + \\ & \underbrace{\frac{\beta_{i,t}^{(2)} p_{i,t}^{(2)} S_{i,t}}{r_{i,t}^{(2)}}}_{\mathrm{III:comms.\ to\ SAGIN\ 2}} + \underbrace{\frac{(\beta_{i,t}^{(0)} + \beta_{i,t}^{(1)} + \beta_{i,t}^{(2)}) p_{i,t}^{(0)} s_{i,t}}{r_{i,t}^{(0)}}}_{\mathrm{IV:Result\ Delivery\ to\ UE}}, \end{aligned} \tag{4.4}$$

where $\kappa^{(0)}$ is the energy factor of SAGIN 0, $r_{i,t}^{(n)}$ is defined in Section 4.2.3, and $p_{i,t}^{(0)}, p_{i,t}^{(1)}, p_{i,t}^{(2)}$ are the power used for communications between SAGIN 0 and the UE, SAGIN 1, and SAGIN 2, respectively. The overall energy consumption in (4.4) involves Part I for task execution in SAGIN 0, Parts II and III for transmitting fractions of the raw dataset to SAGIN 1 and

SAGIN 2, and Part IV for returning the task results back to the requesting UE.

Computation at SAGIN 1 (the neighboring LEO satellites)

The total computation time to perform a portion of the task in SAGIN 1 at time slot t can be written as:

$$l_{i,t}^{(1)} = \underbrace{\beta_{i,t}^{(1)} \frac{c_{i,t}}{f_{i,t}^{(1)}}}_{\text{I: Processing}} + \underbrace{\alpha_{i,t}^{(1)} \frac{2d_{0,t}^{(1)}}{c_0}}_{\text{II: Propagation}} + \underbrace{\frac{\beta_{i,t}^{(1)}(S_{i,t} + s_{i,t})}{r_{i,t}^{(1)}}}_{\text{III: Transmission}}, \quad \forall i, t, \tag{4.5}$$

where $d_{0,t}^{(1)}$ represents the ISL distance, $\alpha_{i,t}^{(1)} \in \{0, 1\}$ is a binary variable indicating whether task i has been offloaded to the SAGIN 1 ($\alpha_{i,t}^{(1)} = 1$) or not ($\alpha_{i,t}^{(1)} = 0$), and c_0 is the speed of light. The overall latency in (4.5) comprises the computation time (Part I), the round-trip propagation delay (Part II), and the cumulative transmission delays of the task's inputs and outputs (Part III) between SAGIN 0 and SAGIN 1.

The energy consumption at SAGIN 1 for performing parts of task i at the time slot t can be written as:

$$e_{i,t}^{(1)} = \beta_{i,t}^{(1)} \kappa^{(1)} (f_{i,t}^{(1)})^2 c_{i,t} + \frac{\beta_{i,t}^{(1)} p_{0,i,t}^{(1)} s_{i,t}}{r_{i,t}^{(1)}}, \forall i, t, \tag{4.6}$$

where $\kappa^{(1)}$ represents the energy factor of SAGIN 1, and $p_{0,i,t}^{(1)}$ is the transmission power used by SAGIN 1. In (4.6), the first part represents the energy consumed during the processing of a portion of the task, while the second part indicates the communication energy at SAGIN 1 for sending (parts of) results of task i back to SAGIN 0.

Computation at SAGIN 2 (the Gateway)

Let $\alpha_{i,t}^{(2)} \in \{0, 1\}$ denote the offloading variable to SAGIN 2, which indicates whether parts of the task i have been offloaded to SAGIN 2 ($\alpha_{i,t}^{(2)} = 1$) or not ($\alpha_{i,t}^{(2)} = 0$). The total computation time in performing a portion $\beta_{i,t}^{(2)} \in [0, 1]$ of task i at SAGIN 2 can be computed as:

$$l_{i,t}^{(2)} = \beta_{i,t}^{(2)} \frac{c_{i,t}}{f_{i,t}^{(2)}} + \alpha_{i,t}^{(2)} \frac{2d_{0,t}^{(2)}}{c_0} + \frac{\beta_{i,t}^{(2)}(S_{i,t} + s_{i,t})}{r_{i,t}^{(2)}}, \forall i, t, \tag{4.7}$$

where $d_{0,t}^{(2)}$ represents the slant distance between SAGIN 0 and SAGIN 2 at time slot t. In (4.7), the first part represents the computational delay, the second part is the round-trip propagation delay during the offloading of parts of task i to SAGIN 2, and the last part represents the cumulative transmission delays.

The energy consumption at the cloud-aided GW is omitted, as it is usually supplied by a constant power source.

4.2.5 Return Link Latency

Unlike existing task offloading works [111–120] that assume a negligible task output size, the outputs of EO-based services are usually 2D/3D scenery images, and thus have a considerable size. Once the requested service has been computed, the SAGIN 0 (the serving LEO satellite) must return the result to the corresponding UE via the return link. It is noted that SAGIN 0 serves its connected UEs via FDMA, as presented in Section 4.2.3. The latency for returning the task result to the requesting UE i at time slot t can be written as:

$$l_{i,t}^{\mathrm{rtn}} = \frac{(\sum_{n\in\mathcal{N}} \beta_{i,t}^{(n)}) s_{i,t}}{r_{i,t}^{(0)}} + \frac{d_{i,t}^{(0)}}{c_0}, \forall i, t, \tag{4.8}$$

where $r_{i,t}^{(0)}$ is given in (4.2), and $d_{i,t}^{(0)}$ is the slant distance between SAGIN 0 and UE i at time slot t. The overall latency in (4.8) consists of transmission delay (first part) and propagation delay (second part).

4.3 Problem Formulation

We aim to optimize the computational resources of the SAGIN nodes and the return link bandwidth to serve the maximum number of on-demand EO tasks in the shortest possible time, constrained by LEO serving duration, task deadlines, and limited system resources.

Let $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{f}$, and $\boldsymbol{b}$, respectively, denote the short-hand notations of $\{\alpha_{i,t}^{(n)}\}_{\forall n\in\{1,2\},i,t}$, $\{\beta_{i,t}^{(n)}\}_{\forall n,i,t}$, $\{f_{i,t}^{(n)}\}_{\forall n,i,t}$, and $\{b_{i,t}\}_{\forall i,t}$. Then the overall worst-case latency for delivering the requested tasks to the users can be expressed as:

$$L(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{f}, \boldsymbol{b}) = \max_i \left\{ l_{i,t}^{\mathrm{proc}} + l_{i,t}^{\mathrm{rtn}} \right\}, \tag{4.9}$$

where $l_{i,t}^{\mathrm{rtn}}$ is given in (4.8) and $l_{i,t}^{\mathrm{proc}} = \max\{l_{i,t}^{(0)}, \ldots, l_{i,t}^{(n)}, \ldots, l_{i,t}^{(N-1)}\}$ represents the worst-case parallel task processing time.

Mathematically, the multi-objective optimization problem can be formulated as follows:

$$\mathcal{P}: \max_{\{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{f},\boldsymbol{b}\}} w_1 \sum\nolimits_{i\in\mathcal{U}_t} \sum\nolimits_{n\in\mathcal{N}} \beta_{i,t}^{(n)} - (1-w_1) L(\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{f},\boldsymbol{b}) \tag{4.10a}$$

$$\text{s.t. } l_{i,t}^{\text{proc}} + l_{i,t}^{\text{rtn}} \leq \min\{\Delta, \tau_{i,t}\}, \quad \forall i,t, \tag{4.10b}$$

$$r_{i,t}^{(0)} \geq \eta_{\text{req}}^{(0)}, \forall i,t; \quad \sum\nolimits_{i\in\mathcal{U}_t} b_{i,t} \leq B_{UE,\sum}^{(0)}, \forall t, \tag{4.10c}$$

$$\sum\nolimits_{n\in\mathcal{N}} \beta_{i,t}^{(n)} \leq 1; \; 0 \leq \beta_{i,t}^{(n)} \leq 1, \quad \forall i,t,n \tag{4.10d}$$

$$\beta_{i,t}^{(n)} \leq \alpha_{i,t}^{(n)}, \quad \forall n \in \mathcal{N}\backslash\{0\}, i, t, \tag{4.10e}$$

$$\sum\nolimits_{i\in\mathcal{U}_t} f_{i,t}^{(n)} \leq F_{\sum}^{(n)}, \quad \forall n,t, \tag{4.10f}$$

$$f_{i,t}^{(n)} \leq \beta_{i,t}^{(n)} F_{\sum}^{(n)}, \quad \forall n,i,t, \tag{4.10g}$$

$$\sum\nolimits_{i\in\mathcal{U}_t} e_{i,t}^{(n)} \leq E_{\sum}^{(n)}, \quad \forall n \in \mathcal{N}\backslash\{2\}, t, \tag{4.10h}$$

$$\alpha_{i,t}^{(n)} \in \{0,1\}, \quad \forall n \in \mathcal{N}\backslash\{0\}, i, t, \tag{4.10i}$$

where w_1 is the weight factor to control the multi-objective optimization problem; $\eta_{\text{req}}^{(0)}$ is the minimum quality-of-service (QoS) requirement from the UEs; $B_{\text{UE},\sum}^{(0)}$ is the total return-link bandwidth allocated for the UEs from the serving satellite; $F_{\sum}^{(n)}$ is the total available computational capabilities of SAGIN n; and $E_{\sum}^{(n)}$ represents each SAGIN $n = 0, 1$.

In problem $\mathcal{P}$, constraint (4.10b) ensures the requested task to be completed either within the configuration duration or the task's tolerant latency. Constraint (4.10c) guarantees minimum QoS requirement and restricts the sum of sub-channel downlink bandwidth for UE. Constraint (4.10d) is to avoid duplicate execution of the tasks. Constraint (4.10e) ensures the task portion is only offloaded to selected SAGINs. Constraints (4.10f) and (4.10g) regulate the computational capabilities of all SAGIN pods to perform each requested task. Constraint (4.10h) limits the energy consumption in SAGINs 0 and 1 during task execution. Constraint (4.10i) controls the binary nature of SAGIN selection variables for offloading.

Difficulty to solve problem $\mathcal{P}$: Problem $\mathcal{P}$ is challenging to solve due to the non-concavity of its objective function and the non-convex constraints (4.10b), (4.10h), and (4.10i). Moreover, it is a non-convex mixed-integer non-linear programming (MINLP) problem and is NP-hard, primarily due to the binary nature of the offloading SAGIN selection variables $\{\alpha_{i,t}^{(n)}\}$.

Solution Development: To achieve an efficient solution for problem $\mathcal{P}$ in (4.10), we de-

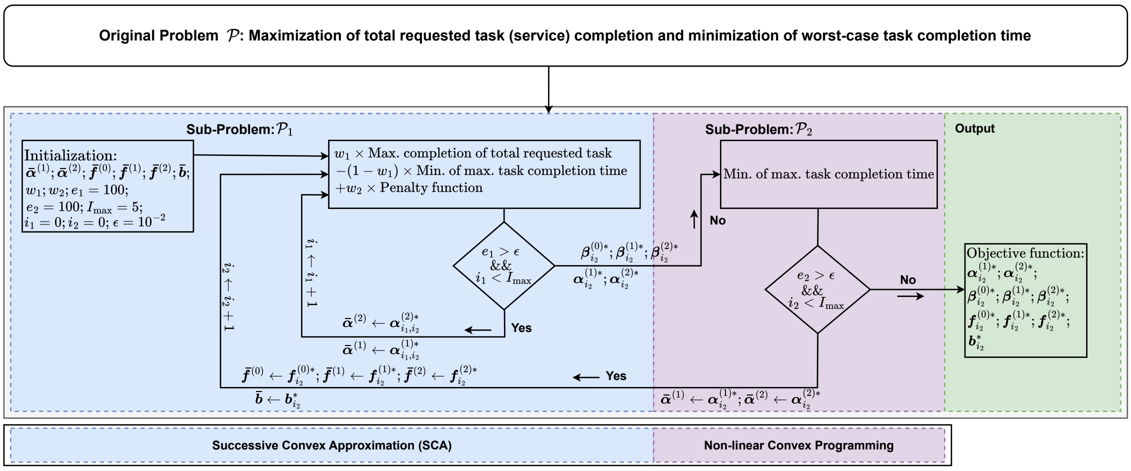

Figure 4.4: Roadmap to solve the original problem $\mathcal{P}$ using the iterative AO framework.

compose the original optimization problem into two sub-problems:

i. The first sub-problem, denoted as $\mathcal{P}_1$, focuses on selecting SAGINs where parts of tasks can be executed and optimizing the task offloading portions. Specifically, we optimize $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to maximize the completion of the total tasks within the soonest manner.

ii. The second sub-problem, denoted as $\mathcal{P}_2$, optimizes the computation allocation of the pods, i.e., $\boldsymbol{f}$, and the downlink delivery bandwidth, i.e., $\boldsymbol{b}$. The aim of this sub-problem is to serve the requested tasks in the shortest possible duration while adhering to the tolerable limits specified by the serving LEO satellite and the UEs.

A detailed procedure for solving these sub-problems using the AO framework is presented in Section 4.4, and an overview diagram illustrating the same is provided in Fig. 4.4.

4.4 Problem Decomposition

In this section, a step-by-step algorithm is designed to solve the two *sub-problems* providing a sub-optimal solution to the original problem (4.10). The optimization is performed at the beginning of each configuration time slot. Therefore, without lost of generality, we drop the time slot subscript t in the following sections for ease of presentation. Furthermore, the convergence and complexity analysis will also be provided. It is worth noting that the unserved tasks from the previous time slot will be added to the set of tasks to be executed in the current time slot.

4.4.1 Sub-Problem $\mathcal{P}_1$: Optimizing $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$

Since problem $\mathcal{P}_1$ only optimizes the node selection and task offloading decisions, the latency in (4.10) is simplified as $L(\boldsymbol{\alpha}, \boldsymbol{\beta})$. Mathematically, the problem $\mathcal{P}_1$ is formulated as follows:

$$\mathcal{P}_1 : \underset{\{\boldsymbol{\alpha},\boldsymbol{\beta}\}}{max} \quad w_1 \sum_{i\in\mathcal{U}} \sum_{n\in\mathcal{N}} \beta_i^{(n)} - (1 - w_1) \times L(\boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{4.11a}$$

$$\text{s.t. } (4.10b), (4.10d), (4.10e), (4.10g) - (4.10i),$$

which is a MINLP problem. We are interested in converting this problem to the mixed-integer linear programming (MILP) problem. The non-linearity in the constraint (4.10b) can be converted into the linear form by consideration of the slack variable $\{x_i\}$ and reformulate constraint (4.10b) as:

$$l_i^{(n)} \le x_i, \quad \forall n, i; \; x_i + l_i^{\mathrm{rtn}} \le \min\{\Delta, \tau_i\}, \quad \forall i. \tag{4.12}$$

With the inclusion of the slack variable $\{x_i\}$, the non-linear component of the second-part of the objective function (4.11a), i.e., $L(\boldsymbol{\alpha}, \boldsymbol{\beta}) \triangleq \max_i \left\{l_i^{\mathrm{proc}} + l_i^{\mathrm{rtn}}\right\}$, can be expressed as $\max_i \left\{x_i + l_i^{\mathrm{rtn}}\right\}$. However, this expression remains non-linear due to the max operation. To tackle this, a slack variable y is introduced, which is taken as the upper bound. Consequently, problem $\mathcal{P}_1$ can be equivalently reformulated as:

$$\mathcal{P}_1^{'} : \underset{\{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{x},y\}}{max} \quad w_1 \sum_{i\in\mathcal{U}} \sum_{n\in\mathcal{N}} \beta_i^{(n)} - (1 - w_1) y \tag{4.13a}$$

$$\text{s.t. } \max_i \left\{x_i + l_i^{\mathrm{rtn}}\right\} \le y, \quad \forall i, \tag{4.13b}$$

$$(4.12), (4.10d), (4.10e), (4.10g) - (4.10i),$$

where $\boldsymbol{x}$ denote the short-hand notation of $\{x_i\}_{\forall i}$.

Since the objective function, as well as all the constraints, are linear (the energy consumption is a linear function with respect to β), and involve binary variables in (4.13), it can be solved using the standard BnB method, e.g., with MOSEK in the CVX solver [40], at an expense of high computational complexity. Although the BnB method provides optimal solutions to subproblem $\mathcal{P}_1^{'}$, its exponential complexity prevents it from practical usage.

To address this computational challenge, we choose to relax the binary variables, converting them into continuous ones. This can be achieved by adding a penalty function

$\Psi^{(n)} = \left(\sum_{i\in\mathcal{U}}(\alpha_i^{(n)})^2\right) - \left(\sum_{i\in\mathcal{U}}\alpha_i^{(n)}\right), n = 1, 2$ to the objective function (4.13a) to facilitate convergence.

However, it is crucial to highlight that $\Psi^{(n)}$ takes on a quadratic form, thereby transforming the problem into quadratic programming, which cannot be directly tackled using a linear programming (LP) solver. In response, we linearize this quadratic penalty function by applying a first-order approximation. In particular, the penalty function can be approximated at any feasible point $\bar{\alpha}_i^{(n)}$ as follows:

$$\bar{\Psi}^{(n)} \triangleq \sum_{i\in\mathcal{U}} 2\alpha_i^{(n)} \sum_{i\in\mathcal{U}} \bar{\alpha}_i^{(n)} - \sum_{i\in\mathcal{U}} (\bar{\alpha}_i^{(n)})^2 - \sum_{i\in\mathcal{U}} \alpha_i^{(n)}, \forall \ n. \tag{4.14}$$

By using (4.14), problem $\mathcal{P}_1^{'}$ can be approximated with a penalty parameter w_2 as:

$$\mathcal{P}_1^{'}: \underset{\{\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{x},y\}}{max} \ w_1 \sum_{i\in\mathcal{U}} \sum_{n\in\mathcal{N}} \beta_i^{(n)} - (1-w_1)y + w_2 \sum_{n\in\{1,2\}} \bar{\Psi}^{(n)} \tag{4.15a}$$

$$\text{s.t. } (4.10\text{d}), (4.10\text{e}), (4.10\text{g}), (4.10\text{h}), (4.12), (4.13\text{b}),$$

$$0 \le \alpha_i^{(n)} \le 1 \ \ \forall n, i. \tag{4.15b}$$

It is observed that the problem (4.15), which is the relaxed version of (4.11), is an LP problem and can be efficiently solved using standard optimization techniques, such as the interior-point method [79]. Although problem (4.15) satisfies all the constraints of the original problem (4.11), it generally yields a sub-optimal solution due to its strong dependence on the initialization of the parameter $\bar{\boldsymbol{\alpha}}$. Therefore, we propose Algorithm 4.7 to solve (4.11). In Algorithm 4.7, $\text{obj}(\boldsymbol{\alpha}_{i_1}, \boldsymbol{\beta}_{i_1})$ denotes the objective function of (4.15) at iteration i_1, while $\text{obj}(\boldsymbol{\alpha}_{i_1-1}, \boldsymbol{\beta}_{i_1-1})$ represents the objective function of (4.15) at iteration $i_1 - 1$.

4.4.2 Sub-Problem $\mathcal{P}_2$: Optimizing $\boldsymbol{f}$ and $\boldsymbol{b}$

The sub-problem $\mathcal{P}_2$ optimizes the computation resources $\boldsymbol{f}$ and bandwidth $\boldsymbol{b}$ given fixed values $\boldsymbol{\alpha}, \boldsymbol{\beta}$. Therefore, the objective function of $\mathcal{P}_2$ becomes minimizing the worst-case latency, which is stated as follows:

$$\mathcal{P}_2: \underset{\{\boldsymbol{f},\boldsymbol{b}\}}{min} \max_i \left\{ l_i^{\text{proc}} + l_i^{\text{rtn}} \right\} \tag{4.16a}$$

$$\text{s.t. } (4.10\text{b}), (4.10\text{c}), (4.10\text{f}) - (4.10\text{h}).$$

It is observed that problem $\mathcal{P}_2$ is a non-linear convex problem, which can be solved efficiently using the interior point method. The interior point method yields values for $\boldsymbol{f}^*$ and $\boldsymbol{b}^*$. These values are then utilized as input for initializing the parameters $\bar{\boldsymbol{f}}$ and $\bar{\boldsymbol{b}}$ to solve the optimization problem (4.11) using Algorithm 4.7. This iterative process continues until convergence is achieved, providing sub-optimal solutions to the primary optimization problem (4.10). Consequently, we introduce Algorithm 4.8 as a solution approach for (4.10). In Algorithm 4.8, $\mathrm{obj}(\boldsymbol{f}_{i_2}, \boldsymbol{b}_{i_2})$ denotes the objective function of (4.16) at iteration i_2, while $\mathrm{obj}(\boldsymbol{f}_{i_2-1}, \boldsymbol{b}_{i_2-1})$ represents the objective function of (4.16) at iteration $i_2 - 1$.

Algorithm 4.7 Iterative SCA Algorithm to Solve (4.11)

Input: U, w_1, $\boldsymbol{f}$, $\boldsymbol{b}$, τ_i, Δ, $B^{(0)}_{\mathrm{UE},\sum}$, $F^{(n)}_{\sum}$, $E^{(n)}_{\sum}$, S_i, s_i, c_i, c_0, $d^{(0)}_i$, $d^{(1)}_0$, $d^{(2)}_0$, $p^{(n)}_i$, $p^{(1)}_{0,i}$, $p^{(2)}_{0,i}$, $\kappa^{(n)}$, $r^{(1)}_i$, $r^{(2)}_i$, N_0

Output: $\alpha^{(1)*}_i$, $\alpha^{(2)*}_i$, $\beta^{(0)*}_i$, $\beta^{(1)*}_i$, $\beta^{(2)*}_i$, x^*_i, $y*$

Initialization: $\bar{\alpha}^{(1)}_i$, $\bar{\alpha}^{(2)}_i$, $i_1 = 1$, $I_{\max} = 5$, $\epsilon = 10^{-2}$, $e_1 = 100$

while $e_1 > \epsilon$ and $i_1 < I_{\max}$ **do**
  // Solve (4.15) to get $\boldsymbol{\alpha}^*$, $\boldsymbol{\beta}^*$
  Compute $\mathrm{obj}(\boldsymbol{\alpha}_{i_1}, \boldsymbol{\beta}_{i_1})$
  Compute $e_1 = |\mathrm{obj}(\boldsymbol{\alpha}_{i_1}, \boldsymbol{\beta}_{i_1}) - \mathrm{obj}(\boldsymbol{\alpha}_{i_1-1}, \boldsymbol{\beta}_{i_1-1})|$
  Update $\mathrm{obj}(\boldsymbol{\alpha}_{i_1-1}, \boldsymbol{\beta}_{i_1-1}) \leftarrow \mathrm{obj}(\boldsymbol{\alpha}_{i_1}, \boldsymbol{\beta}_{i_1})$; $\bar{\boldsymbol{\alpha}} \leftarrow \boldsymbol{\alpha}^*$; $i_1 \leftarrow i_1 + 1$
end while

4.4.3 Initialization of the overall AO-based Algorithm 4.8

The proposed Algorithm 4.8, which provides a sub-optimal solution to the original problem $\mathcal{P}$, operates iteratively following the AO framework illustrated in Fig. 4.4. Initialization is an important step in any iterative algorithm, influencing the feasibility of optimization. Considering this, we provide the initial values of $\bar{\boldsymbol{\alpha}} = \mathbf{1}^{1\times U}$, $\bar{\boldsymbol{f}} = \frac{F^{(n)}_{\sum} \times \mathbf{1}^{1\times U}}{U}$, and $\bar{\boldsymbol{b}} = \frac{B^{(0)}_{\mathrm{UE},\sum} \times \mathbf{1}^{1\times U}}{U}$.

4.4.4 Convergence and Complexity Analysis of the overall AO-based Algorithm 4.8

Convergence Analysis of Algorithm 4.8

Here, the convergence of the proposed AO algorithm can be explained through four key properties. Let $\boldsymbol{\Pi}_{i_1,i_2} = (\boldsymbol{\alpha}^*_{i_1,i_2}, \boldsymbol{\beta}^*_{i_1,i_2}, \boldsymbol{f}^*_{i_1,i_2}, \boldsymbol{b}^*_{i_1,i_2})$ represents the optimal solution to the approximated problem, i.e., the combination of (4.15) and (4.16), at the current SCA iteration

Algorithm 4.8 Iterative AO Algorithm to Solve (4.10)

Input: U, w_1, τ_i, Δ, $B^{(0)}_{\mathrm{UE},\sum}$, $F^{(n)}_{\sum}$, $E^{(n)}_{\sum}$, S_i, s_i, c_i, c_0, $d^{(0)}_i$, $d^{(1)}_0$, $d^{(2)}_0$, $p^{(n)}_i$, $p^{(1)}_{0,i}$, $p^{(2)}_{0,i}$, $\kappa^{(n)}$, $r^{(1)}_i$, $r^{(2)}_i$, N_0

Output: $\alpha^{(1)*}_i$, $\alpha^{(2)*}_i$, $\beta^{(0)*}_i$, $\beta^{(1)*}_i$, $\beta^{(2)*}_i$, $f^{(0)*}_i$, $f^{(1)*}_i$, $f^{(2)*}_i$, b^*_i, x^*_i, $y*$

Initialization: $\bar{\alpha}^{(1)}_i$, $\bar{\alpha}^{(2)}_i$, $\bar{f}^{(0)}_i$, $\bar{f}^{(1)}_i$, $\bar{f}^{(2)}_i$, $\bar{b}_i$, $i_1 = 1$, $i_2 = 1$, $I_{\max} = 5$, $\epsilon = 10^{-2}$, $e_1 = 100$, $e_2 = 100$

1: **while** $e_2 > \epsilon$ and $i_2 < I_{\max}$ **do**
2: // Solve **Algorithm 4.7** to get $\boldsymbol{\alpha}^*$, $\boldsymbol{\beta}^*$
3: // Solve (4.16) to get $\boldsymbol{f}^*$, $\boldsymbol{b}^*$
4: Compute $\mathrm{obj}(\boldsymbol{f}_{i_2}, \boldsymbol{b}_{i_2})$
5: Compute $e_2 = |\mathrm{obj}(\boldsymbol{f}_{i_2}, \boldsymbol{b}_{i_2}) - \mathrm{obj}(\boldsymbol{f}_{i_2-1}, \boldsymbol{b}_{i_2-1})|$
6: Update $\mathrm{obj}(\boldsymbol{f}_{i_2-1}, \boldsymbol{b}_{i_2-1}) \leftarrow \mathrm{obj}(\boldsymbol{f}_{i_2}, \boldsymbol{b}_{i_2})$; $\bar{\boldsymbol{f}} \leftarrow \boldsymbol{f}^*$; $\bar{\boldsymbol{b}} \leftarrow \boldsymbol{b}^*$; $i_2 \leftarrow i_2 + 1$
7: **end while**

i_1 and AO iteration i_2, and let $\mathbb{F}(\boldsymbol{\Pi}_{i_1,i_2})$ denote the corresponding objective value of the approximated problem $\mathcal{P}$.

The first property is that in each SCA step, $\mathcal{P}_1$ is approximated by a convex optimization problem $\mathcal{P}''_1$ that can be solved efficiently. The second property is that this approximation produces a non-decreasing sequence of objective values such that $\mathbb{F}(\boldsymbol{\Pi}_{i_1+1,i_2}) \geq \mathbb{F}(\boldsymbol{\Pi}_{i_1,i_2})$, which ensures that the inner loop converges to a stationary point of $\mathcal{P}_1$ [133]. The third property is that the convex sub-problem $\mathcal{P}_2$ is solved optimally at every AO iteration, which guarantees further improvement in the objective. The fourth property is that the sequence $\mathbb{F}(\boldsymbol{\Pi}_{i_2}) \triangleq \mathbb{F}(\boldsymbol{\Pi}_{i_1,i_2})$ over AO iterations is bounded and monotonic, which ensures convergence to a stationary point of the original problem $\mathcal{P}$ [134].

The inner SCA loop terminates when $|\mathbb{F}(\boldsymbol{\Pi}_{i_1,i_2}) - \mathbb{F}(\boldsymbol{\Pi}_{i_1-1,i_2})| \leq \epsilon$, and the outer AO loop terminates when $|\mathbb{F}(\boldsymbol{\Pi}_{i_2}) - \mathbb{F}(\boldsymbol{\Pi}_{i_2-1})| \leq \epsilon$, where ϵ determines the sensitivity of the convergence criterion. The convergence behavior of $\mathcal{P}_1$ and $\mathcal{P}_2$ is further validated numerically in Fig. 4.8 of Section 4.6.4.

Complexity Analysis of Algorithm 4.8

Assuming that the interior point method is used to solve convex problems (4.15) and (4.16), in the worst case, the complexity is equal to the cube of the number of real variables [79]. Since there are $2U$ real variables in (4.15) and (4.16), the complexity for solving Algorithm 4.8 is $\mathcal{O}(I_{\max}\left(I_{\max}\,(2U)^3 + (2U)^3\right))$.

4.5 Task Offloading and Delivery Strategies during HO Periods

This section focuses on the HO events considering different beam management strategies. The beam management techniques play an important role in designing the LEO satellite's radio resource management algorithm, which can be determined based on the satellite's on-board antenna system and on-board processor [9, 135]. We consider two beam management techniques that have been prevalent in the literature:

i. *Earth-moving beam:* The satellite beam moves across the Earth's surface (see Fig. 4.5(a)), maintaining a constant geometry relatively to the satellite at a constant velocity [135]. The benefit of *Earth-moving beam* includes the simplest steering operation at the LEO's antennas. When the serving LEO satellite reaches the edge of the current footprint, it is handed off to the approaching adjacent LEO satellite whose footprint is entering that area [9]. In this context, HOs are asynchronous and frequent, leading to increased challenges in managing radio resources. This architecture is only optimal when the UEs also move in the same direction as the LEOs, e.g., a fleet of cruise ships [136].

ii. *Earth-fixed beam:* In this case, the LEO satellite's beam can be steered to cover fixed coverage areas throughout a service period (see Fig. 4.5(b)). Once the service period of the current LEO satellite concludes, the approaching adjacent satellite takes over, resulting in a synchronous HO, which occurs less frequently compared with the former case. This scenario is suitable for both mobile and static UEs and requires the satellites to be equipped with an electronically and mechanically steerable array antenna [123].

Depending on the beam management techniques, the nature and frequency of HOs can vary, as shown in Fig. 4.6. It illustrates the impact of HO during the delivery phase of completed on-demand EO service task results. Steps 1 to 7 present various cases of HO, namely *Case A*, *Case B*, and *Case C*, respectively, which are further discussed in the following.

4.5.1 Case A

We refer to this as the *full-HO* case, indicating that all the tasks at the configuration time slot t cannot be delivered within the current service period of the current LEO satellite, i.e., $T_{\text{serve}} - t\Delta < l_{i,t}^{\text{rtn}}\ \forall i, t$. In this situation, all task results should be returned to the respective

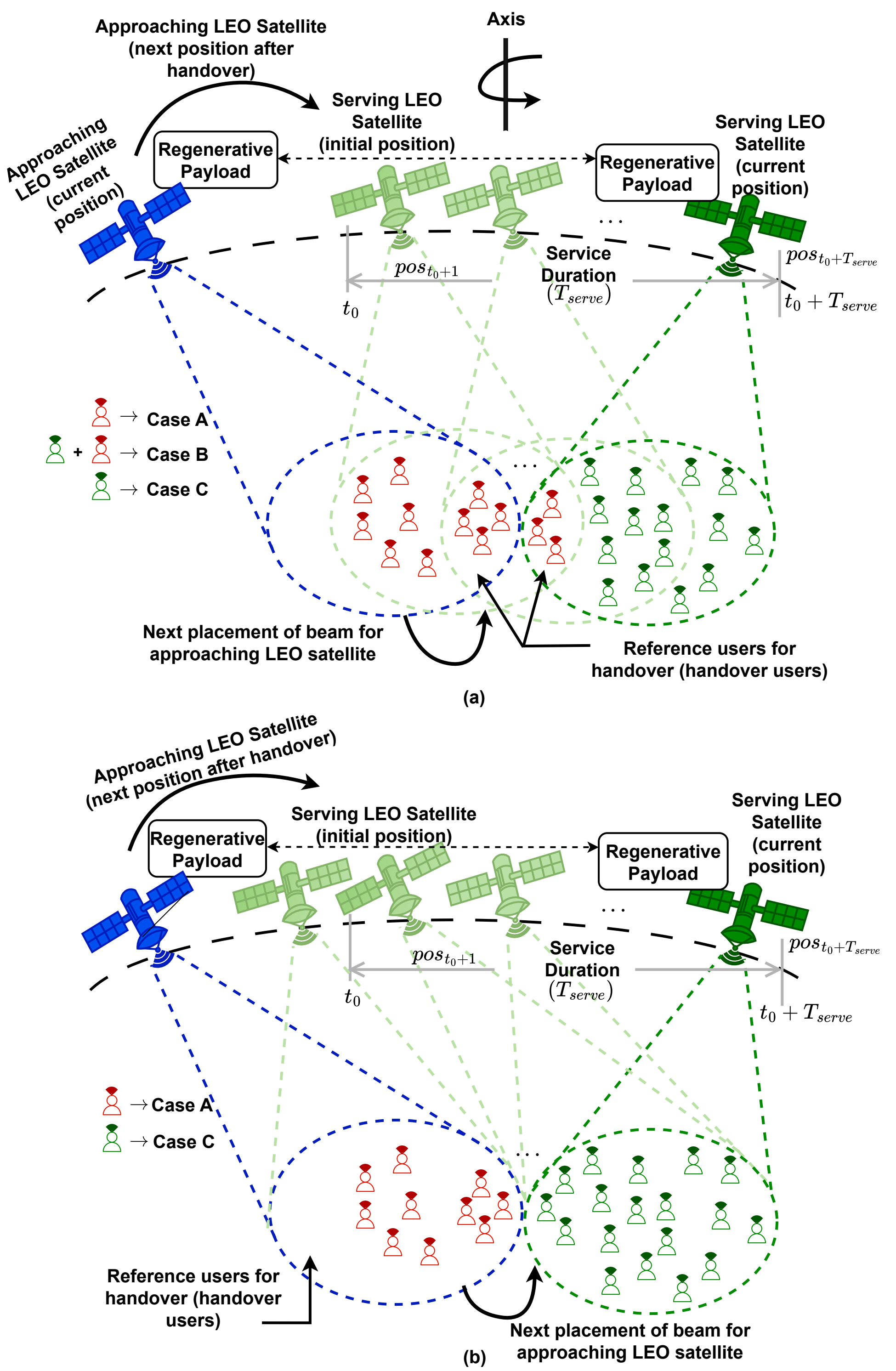

Figure 4.5: HO and non-HO scenarios in (a) *Earth-moving beam* scenario and (b) *Earth-fixed beam* scenario.

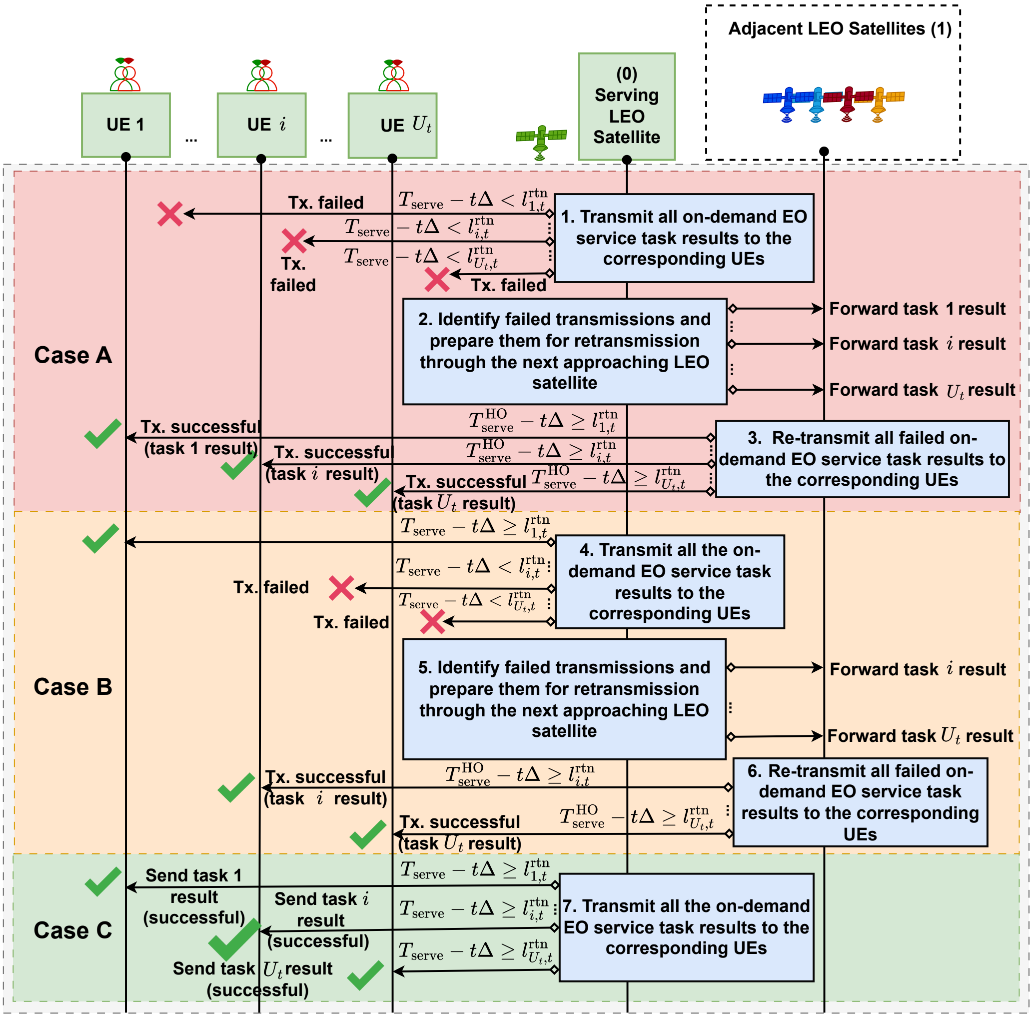

Figure 4.6: A key procedure for handling HO & non-HO scenarios in *Earth-moving* and *Earth-fixed beam* contexts during the delivery of on-demand EO services.

UEs via the nearest adjacent approaching LEO satellite, whose footprint is likely to overlap with the historical footprint of the current LEO satellite, as shown in Steps 1 to 3 of Fig. 4.6. The transmission from the approaching LEO must be optimized based on the number of completed tasks while fulfilling the QoS requirements. The bandwidth optimization problem for the downlink delivery to minimize the worst-case delivery latency, by optimizing the total available bandwidth, can be formulated as follows:

$$\mathcal{P}^{\text{deli}}: \min_{\{b_{i,t}^{(1)}\}} \left(\max_{\{i\}} \left(l_{i,t}^{\text{rtn}}\right)\right) \tag{4.17a}$$

$$\text{s.t. } r_{i,t}^{(1)} \geq \eta_{\text{req}}^{(1)}, \quad \forall i, t, \tag{4.17b}$$

$$\sum\nolimits_{i \in \mathcal{U}_t} b_{i,t}^{(1)} \leq B_{\text{UE},\sum}^{(1)}, \quad \forall t, \tag{4.17c}$$

where $r_{i,t}^{(1)} \triangleq b_{i,t}^{(1)} \log_2 (1 + \frac{p_{i,t}^{(1)}|h_{i,t}^{(1)}|^2}{b_{i,t}^{(1)} N_0})$ represents the downlink delivery rate from the adjacent LEO satellite to the task requesting UE, wherein $|h_{i,t}^{(1)}|^2$ is the downlink channel coefficient from the spot beam of the adjacent LEO satellite to the UE i, $p_{i,t}^{(1)}$ denotes the transmit power of the adjacent LEO satellite to communicate with UE i, and $b_{i,t}^{(1)}$ is the optimized downlink bandwidth from the adjacent LEO satellite to the UE i; $\eta_{\text{req}}^{(1)}$ is the minimum QoS requirement for the adjacent LEO satellite; and $B_{\text{UE},\sum}^{(1)}$ is the total downlink bandwidth for the adjacent LEO satellite.

The problem $\mathcal{P}^{\text{deli}}$ is convex because the rate function containing the logarithm term in (4.17b) is concave and can therefore be solved directly using the interior-points method [79]. This scenario is only applicable under the *Earth-fixed beams* configuration and the UEs are uniformly distributed within the coverage area of the spot beam during any given time slot t. The service period of an *Earth-fixed beam* is around 10 to 15 minutes during one orbital [42].

4.5.2 Case B

This can also be referred to as a *partial-HO* case, indicating that the task requests from some UEs cannot be completed within the serving period of the current LEO satellite, while the remaining UEs can be served within the serving period of the current LEO satellite, i.e., $T_{\text{serve}} - t\Delta \geq l_{i,t}^{\text{rtn}}$. To address this, the computed tasks that cannot be returned to the corresponding UEs must be forwarded to the nearest approaching adjacent LEO satellite, as illustrated in Steps 4 to 6 of Fig. 4.6. The delivery latency is further minimized by optimizing the bandwidth, as per (4.17).

This situation is only experienced in the case of an *Earth-moving beam* scenario. The service period of an *Earth-moving beam* is much shorter than that of an *Earth-fixed beam* and depends on the size of the beam and the velocity at which the beam is sweeping. *To illustrate*[4.3], if the velocity of the LEO satellite is 7.6 km/sec and the radius of the circular

[4.3] It is one of the possible cases, not the only one.

beam is 124 km, then the service period of the *Earth-moving beam* at a particular location is around 32.63 seconds during one orbital.

4.5.3 Case C

This can be referred as *non-HO* case, indicating that all the tasks at time slot t can be successfully delivered within the serving period, i.e., $T_{\text{serve}} - t\Delta \geq l_{i,t}^{\text{rtn}}$, $\forall i, t$, which is illustrated in Step 7 of Fig. 4.6. This scenario can only be experienced in the *Earth-fixed beam* scenario, assuming that the UEs are uniformly distributed within the footprint of the serving LEO satellite at any time slot t.

4.6 Performance Evaluation on Realistic System Parameters

In this section, we evaluate the performance of the proposed framework based on realistic LEO satellite parameters.

4.6.1 LEO Satellite Footprint

The serving Starlink LEO satellite 4280 is assumed to be in orbit just above Mt. Everest, Nepal at time t_0 [137]. It is at an altitude (H_s) of 545 km just above Earth's surface and is considered to be $d_{0,t}^{(1)}$ *distance apart*[4.4] from the $M = 4$ adjacent LEO satellites, i.e., 2 in the same orbit and the remaining 2 in the adjacent orbits. The FoV of one of the spot beams of the serving LEO satellite is shown in Fig. 4.7.

4.6.2 Synthetic Dataset Generated from Realistic Parameters

The frequency and location of task requests are synthetically generated based on realistic information due to the lack of suitable open-source datasets in this domain. According to [138, 139], during the suitable window period, at least 500 tourists daily journey to the Everest Base Camp (EBC), with 208 individuals aiming to climb the peak of Mt. Everest. The primary locations encountered during the ascent of Mt. Everest include EBC, Camp-I, Camp-II, Camp-III, Camp-IV, and Mt. Everest [140]. Requests to the LEO satellite are expected to originate from users residing within the vicinity of these locations. At any

[4.4] The inter-orbit topology is considered to be stable using ISL communication, automated maneuvers, and collision avoidance system.

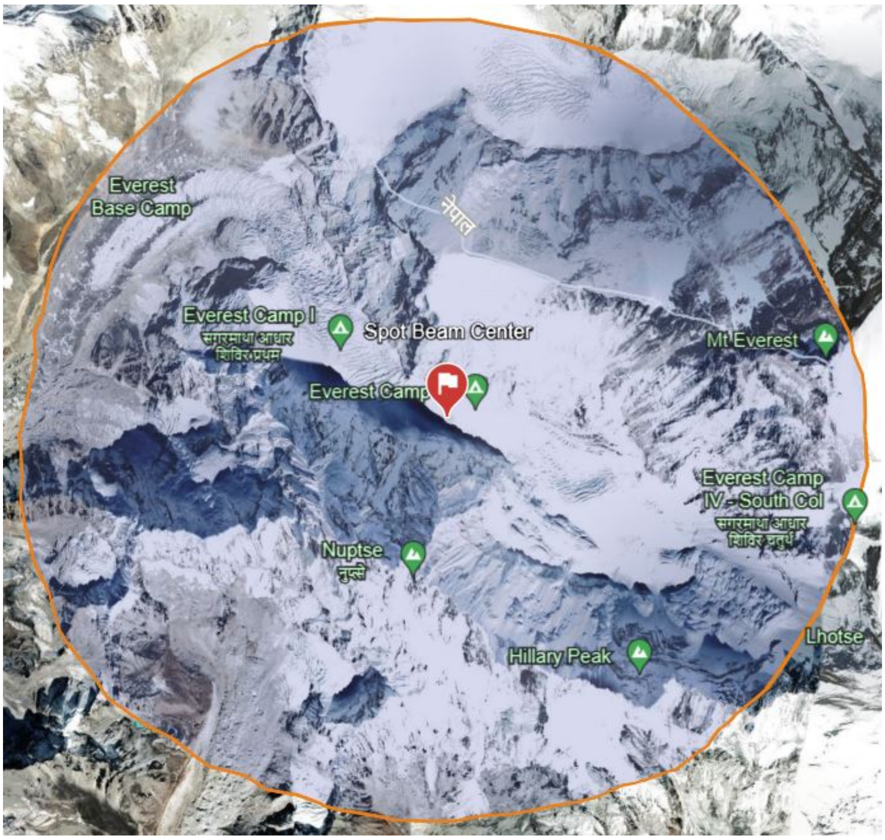

Figure 4.7: FoV of the serving LEO satellite's spot beam.

given time, there are 292 users at EBC, and 208 users are evenly distributed in Camp-I, Camp-II, Camp-III, Camp-IV, and Mt. Everest. Considering 2% of active users every minute at both EBC and ascending Mt. Everest, the task arrival rate vector is denoted as $\boldsymbol{\lambda} = [5.8, 0.8, 0.8, 0.8, 0.8, 1]\text{min}^{-1}$.

4.6.3 Network Implementation and Simulation Setup

The LEO satellite network is implemented using orbital data (two-line element, TLE) from Starlink LEO satellite 4280 [137], with its orbital path rendered in MATLAB using `satelliteScenario` object. The path loss between the satellite and ground UEs is computed based on the satellite's instantaneous location relative to fixed UE positions within its FoV (see Fig. 4.7).

The positions of UEs making task requests are distributed based on a task arrival rate vector $\boldsymbol{\lambda}$ across six major locations within the coverage area. The positions of UEs within each major location vary within a uniform range of 0.1 km to 1 km. Consequently, the radius of the adaptive spot is uniformly varied within the boundary [24, 124] km for which the spot

Table 4.2: Simulation Parameters

Parameters	**Value**
$\tau_{i,t}$, Δ	60 sec, 60 sec
$S_{i,t}$, $s_{i,t}$	U [10 MB, 20 MB] [141], 0.6 $\times$ $S_{i,t}$ [142]
$c_{i,t}$	$\{0.05, 0.075\}$ Kcycles/bit [143,144]
$F^{(0)}_{\sum}$, $F^{(1)}_{\sum}$, $F^{(2)}_{\sum}$	3 GHz [116], 3 $\times$ 4 GHz, 12 GHz
$E^{(0)}_{\sum}$, $E^{(1)}_{\sum}$, $\kappa^{(n)}$	5000 MJ [116], 4 $\times$ 5000 MJ, 10^{-28} [116]
R_{ISL}, R_{GW}	10 Gbps [116], 300 Mbps [116]
$d^{(1)}_{0,t}$, $d^{(2)}_{0,t}$	1000 km [116], 3000 km
$P^{(0)}_{\text{ISL},\sum}$, $P^{(0)}_{\text{GW},\sum}$	30 dBW [116], 20 dBW [116]
$P^{(0)}_{\text{UE},\sum}$	25 dBW [116]
$B^{(0)}_{\text{UE},\sum}$, f_c	100 MHz [123], 2 GHz [123]
$\eta^{(0)}_{\text{req}}$, N_0	50 Mbps [75,123], -203.9772 dBW/Hz

beam gain is between [22, 36] dBi [123].

The wireless channel for the non-terrestrial network is modeled using Rician fading, which captures both deterministic line-of-sight (LoS) and non-LoS (NLoS) components, as mentioned in [75], [123]. The large-scale fading is represented by free-space path loss, which varies dynamically based on the satellite's orbital position relative to the ground UE. In alignment with the standards of 3rd Generation Partnership Project (3GPP) Release 18 [24] for direct-to-device services based on LEO, we adopt an S-band carrier frequency (f_c = 2 GHz) for the service link. This choice is justified by the fact that atmospheric losses such as rain fade and cloud attenuation are negligible below 6 GHz [145], allowing us to exclude these effects from the channel model. All remaining simulation parameters related to tasks, system configuration, and channel characteristics are summarized in Table 4.2.

4.6.4 Performance Evaluation

In this part, we present numerical results for the satellite scenario described in Section VI-C. The simulation results are averaged over 200 random channel realizations. We compare the proposed framework with the following references:

- *Equal Task Portion Offloading:* This is a proposed task offloading scenario that involves an equal distribution of resources without optimizing them.
- *Binary Offloading:* This is a binary offloading method optimizing computational resources and downlink delivery bandwidth [112–115].

- *Fixed Computation and Communication:* This is a fractional task offloading scenario without optimizing computational resources and downlink bandwidth [111].
- *Without Offloading:* No task offloading occurs; all computations are performed on the serving LEO satellite.

The comparison with the terrestrial task offloading solution is not considered, as satellite systems are designed to complement the terrestrial networks, which are assumed to be unavailable in such challenging terrain.

Convergence and Complexity of the Proposed AO Framework

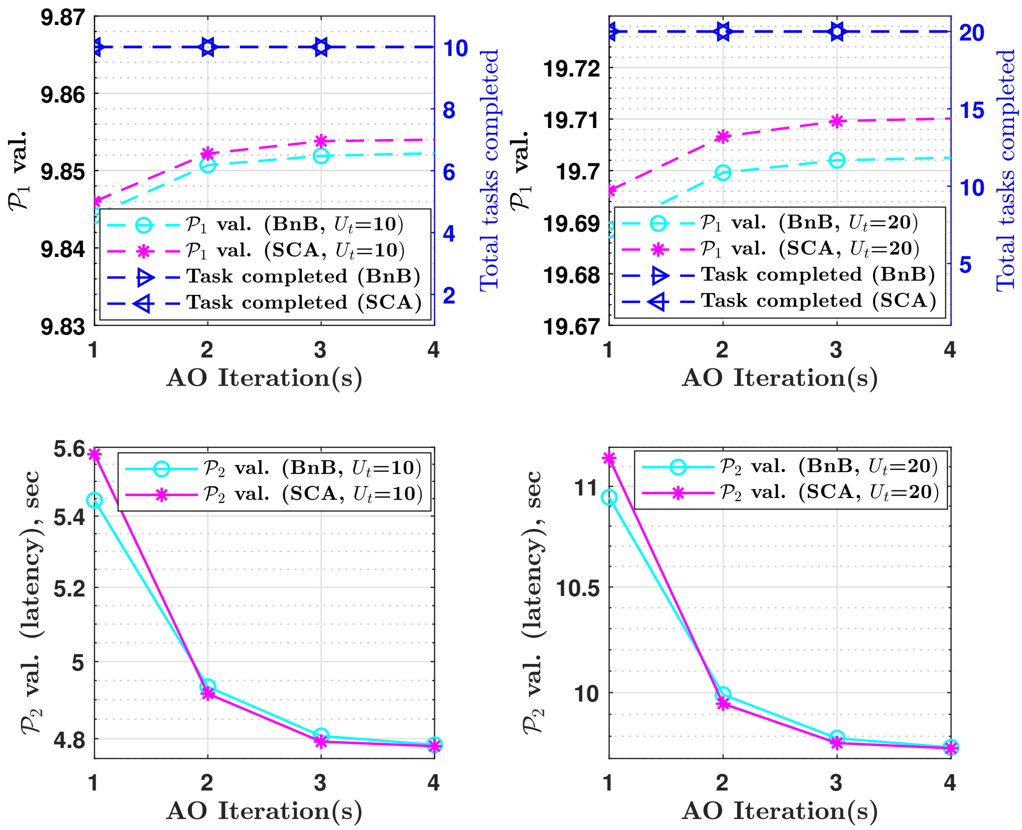

Figure 4.8: Convergence of the AO sub-problems $\mathcal{P}_1$ and $\mathcal{P}_2$ for the non-convex problem $\mathcal{P}$.

Fig. 4.8 depicts the convergence behavior of the proposed AO framework, highlighting its effectiveness in solving the non-convex problem $\mathcal{P}$. The figure tracks the objective values of the sub-problems $\mathcal{P}_1$ and $\mathcal{P}_2$ across successive AO iterations, comparing our SCA-based

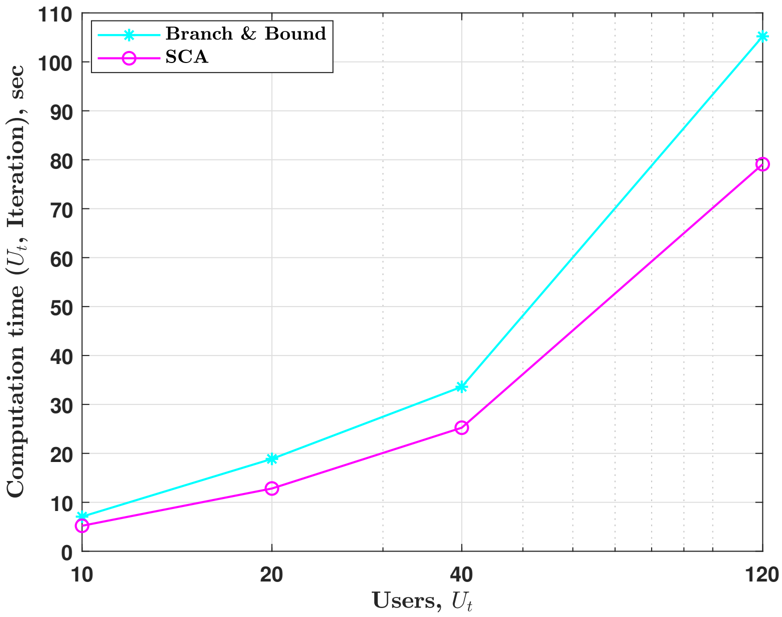

Figure 4.9: Time-complexity of the proposed AO algorithm.

approach for $\mathcal{P}_1$ against a global optimality benchmark obtained via the BnB method. Both approaches incorporate convex optimization for $\mathcal{P}_2$. The results show that the SCA-based method converges rapidly to a solution of comparable quality to the BnB benchmark, stabilizing in fewer than five iterations. Additionally, it meets the termination criterion of a relative error (ϵ) below 10^{-2}, confirming the efficiency and practical effectiveness of the proposed AO framework in minimizing the maximum task completion time while successfully completing all requested tasks U_t. The figure also shows that total task completion is maximized, i.e., equal to the number of requests (U_t), and that completion time improves consistently across AO iterations until convergence. This improvement is attributed to the choice of the weight parameter w_1, which balances multiple objectives in $\mathcal{P}$. The impact of w_1 will be analyzed in the next sub-section.

To analyze the computational complexity of our proposed AO algorithms, Fig. 4.9 presents the simulation time for both the SCA (SCA for $\mathcal{P}_1$ + convex optimization for $\mathcal{P}_2$) and BnB (BnB for $\mathcal{P}_1$ + convex optimization for $\mathcal{P}_2$) methods using the same CVX solver. For small problem sizes, the SCA method slightly outperforms BnB in computation time. As the problem size grows with the number of tasks, computation time increases exponentially; however, the SCA method remains faster than BnB while approaching BnB's performance. Given its comparatively lower computational cost, we will be using the SCA method for the remaining simulations.

In this context, an increase in the problem size corresponds to a rise in the number of service-requesting UEs within a time slot. To ensure practical applicability under these conditions, we propose setting an upper limit on the number of UEs that a LEO satellite can process within a single time slot. The remaining UEs should then be scheduled based on priority metrics such as signal strength, fairness, task size, task characteristics, tolerance time limits, and service duration.

Impact of the Weight on the Objective Function

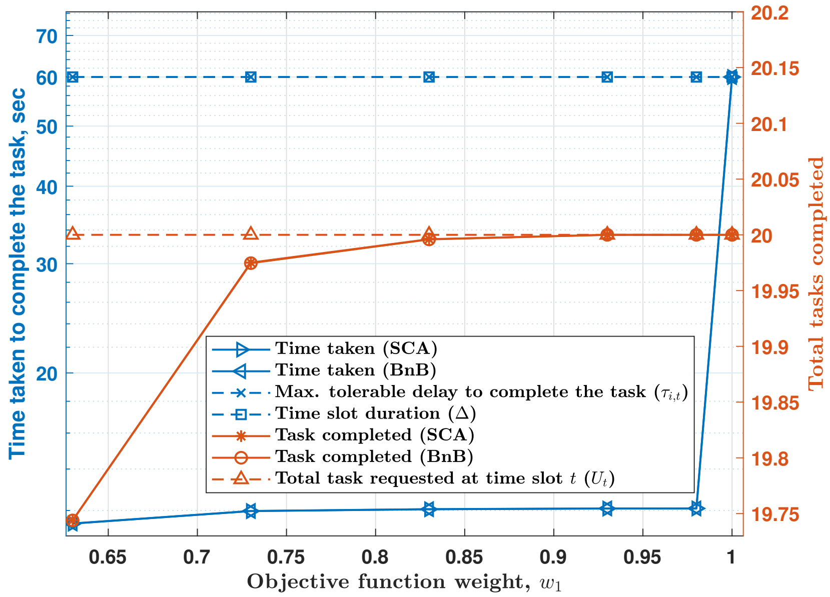

Figure 4.10: Weight vs. multi-objective optimization function value.

Since problem (4.10) simultaneously maximizes the executed tasks and minimizes the task delivery latency, the weight w_1 directly impacts the priority of the objective function in (4.10). Meanwhile, we have already fixed the weight, i.e., $w_2 = 10^{-4}$, to control the penalty function $\Psi^{(n)}$ for the SCA method. The effect of w_1 on the objective function is depicted in Fig. 4.10, provided that it is solved using SCA and BnB methods and $w_2 = 10^{-4}$ for the penalty function $\Psi^{(n)}$ in problem (4.15). It is shown that at $w_1 = 0.63$, incomplete fulfillment of requested tasks at time slot t occurs, resulting in a lower total task completion time. As w_1 exceeds 0.63, both the total completed tasks and the task completion time increase. Optimal performance is achieved when $w_1 = 0.93$, completing all tasks. However, at $w_1 = 1$, while all tasks are completed, the time taken equals the time slot duration Δ as there will be no intention to minimize the total task completion time. The SCA method employing the penalty function performs comparably to the BnB method in terms of achieving optimal values.

Note: SCA methods are well-suited for managing large numbers of concurrent service requests from UEs, thanks to their fast convergence, low computational complexity, and consistent performance, as shown in figures 4.8– 4.10. They enhance numerical convergence in non-convex optimization by using a strongly convex surrogate and offering a flexible framework. However, their performance heavily depends on step size selection (or weight w_2 in our case), which includes exact or successive line search methods and diminishing step sizes. While successive line search is simpler, it incurs computational costs from repeated evaluations of the nonsmooth function. Diminishing step sizes are sensitive to decay rates and are challenging to implement [38]. To address this, a fixed value for w_2 is employed in this chapter, ensuring faster convergence with minimal tuning. The value of w_2 is determined through trial and error.

Trend of Maximum Task Completion Time

Fig. 4.11 presents the trend of the proposed SCA method for selected time slots, where the number of tasks per time slot follows a Poisson distribution. The arrival rate for the Poisson distribution is derived from realistic observations. The figure illustrates that the proposed method outperforms other reference schemes in task completion time, ensuring efficient computation offloading, load balancing, and enhanced parallelism. Additionally, the trends of both the proposed method and reference schemes are significantly influenced by the number of task arrivals per time slot. Furthermore, the outage scenario is not shown in the simulation results but may occur if the system is overloaded and any of the constraints in optimization problem $\mathcal{P}_1$ is not satisfied. This can be mitigated through proper implementation of the scheduling algorithm [131].

Impact of Computational Cycles on Task Completion Time

Fig. 4.12 portrays the mean time required to fulfill the requested tasks for all methods, taking into account diverse computational requirements for processing data bits. With a computational requirement of 50 cycles per bit, the proposed method achieves a mean completion time

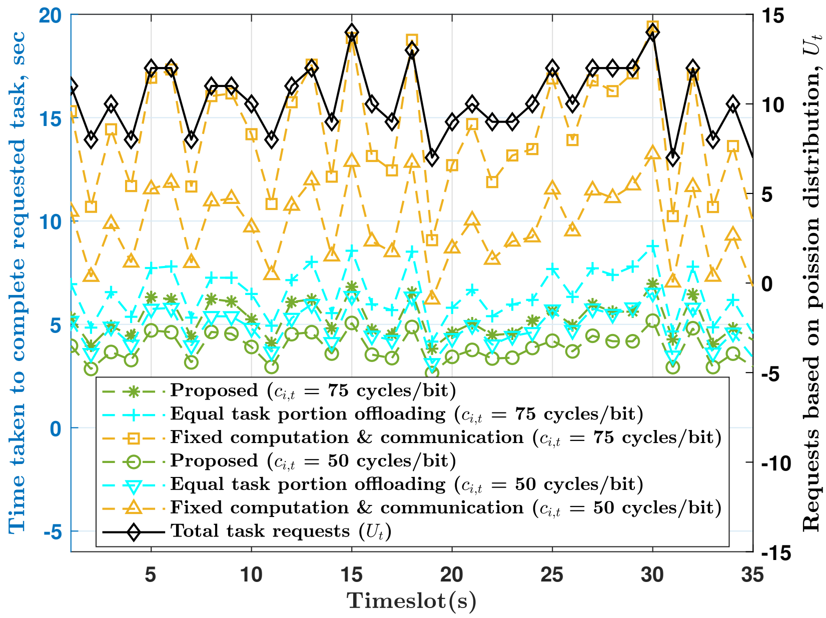

Figure 4.11: Trend of task completion time for different time slots.

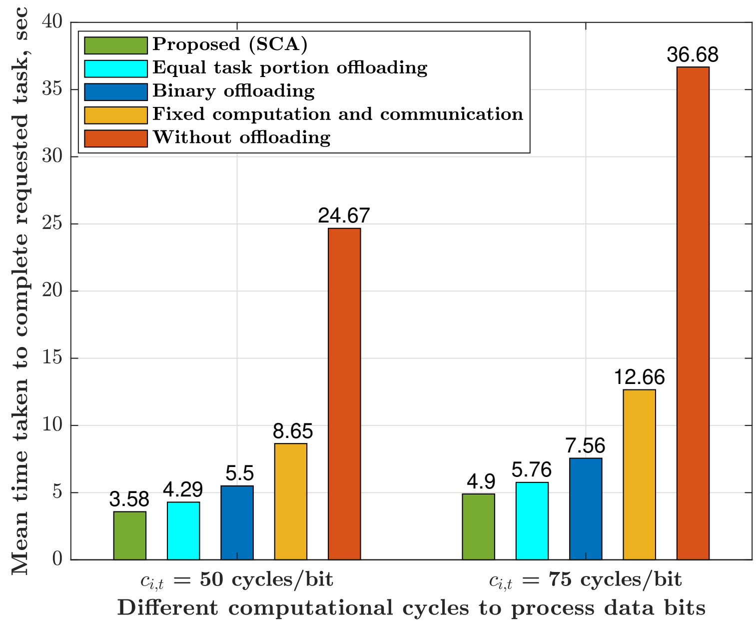

Figure 4.12: Mean task completion time for different values of computational cycles.

of 3.58 seconds, surpassing equal task portion offloading, binary offloading, fixed computation and communication, and without offloading by 16.55%, 34.90%, 58.61%, and 85.48%, respectively. As the computational requirement increases to 75 cycles per bit, the mean completion time for the proposed method increases by 36.87%, while equal task portion offloading, binary offloading, fixed computation and communication, and without offloading increase by 34.27%, 37.45%, 46.36%, and 48.68%, respectively. This indicates a sub-linear increase in the time taken to complete the requested task for both the proposed method and reference methods with the increment in computational requirements for processing data bits.

Impact of ISL and Feeder Link Capacity

Table 4.3: Different ISL (uplink, downlink) and Feeder-link Rate (uplink, downlink) Conditions

Condition	**Rates** $([R_{\text{ISL}}, R_{\text{GW}}])$
R1	[102.4, 30] Mbps
R2	[409.6, 120] Mbps
R3	[716.8, 210] Mbps
R4	[1024, 300] Mbps
R5	[10240, 3000] Mbps

Fig. 4.13 presents the impact of the ISL and feeder-link capacities on the offloading decision, assuming an equal size of 15 MB for all tasks. In condition 'R1', as indicated in Table 4.3, a substantial portion of the task is performed at the serving LEO satellite, followed by the adjacent LEO satellite, and then the cloud-aided GW, ultimately completing the entire requested task. This is attributed to the lower ISL and feeder link rates (uplink, downlink), with the feeder link rate being 3.4 times smaller than the ISL rate. As the ISL and feeder-link capacities increase, while maintaining the previous rate ratio of 3.4, it is observed that under conditions 'R2' to 'R4', a large portion of tasks are offloaded to adjacent LEO satellites, followed by the GW, and then the serving LEO satellite. In condition 'R5', where both ISL and feeder-link rates are improved, nearly equal proportions of tasks are offloaded to adjacent LEO satellites and GW due to energy requirement constraints in the serving and adjacent LEO satellites. The smallest portion of tasks is offloaded to the serving LEO satellite.

Fig. 4.14 shows the impact of ISL and feeder-link rates on computational resource utilization depicted for the SAGINs, influencing mean task completion time. In condition 'R1',

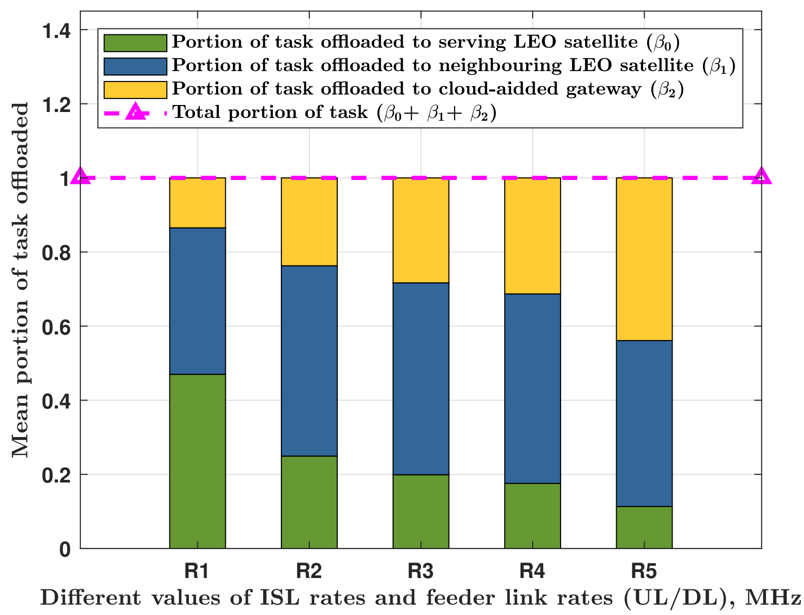

Figure 4.13: Impact of ISL rate on the portion of the task offloaded to the SAGINs.

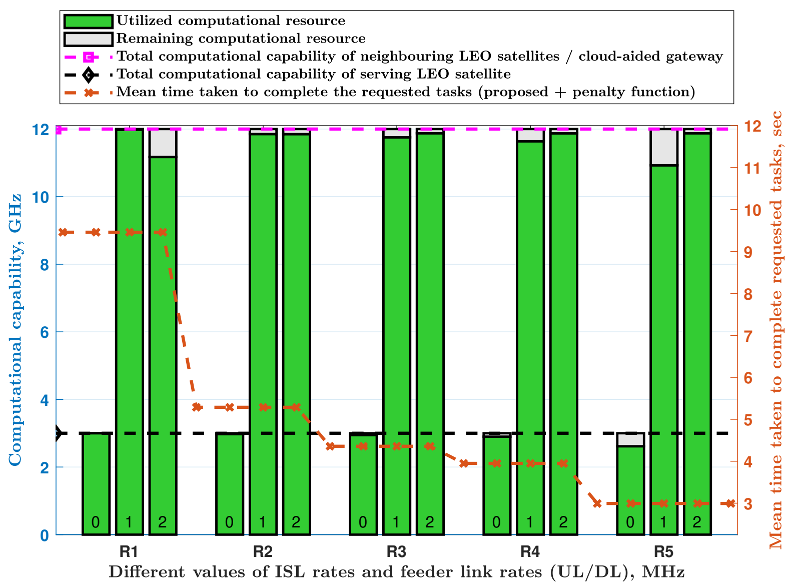

Figure 4.14: Impact of ISL rate on the utilization of the available computational resources of the SAGINs.

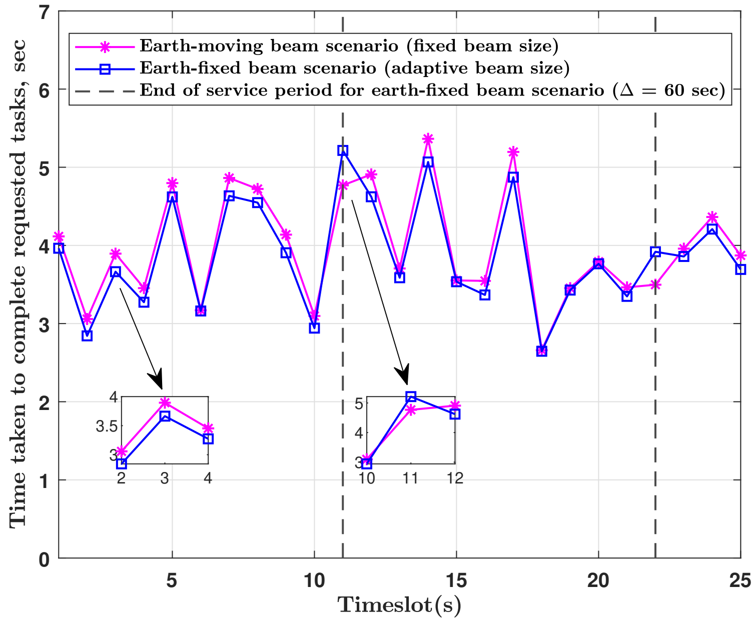

Figure 4.15: Trends of task completion time due to different HO scenarios.

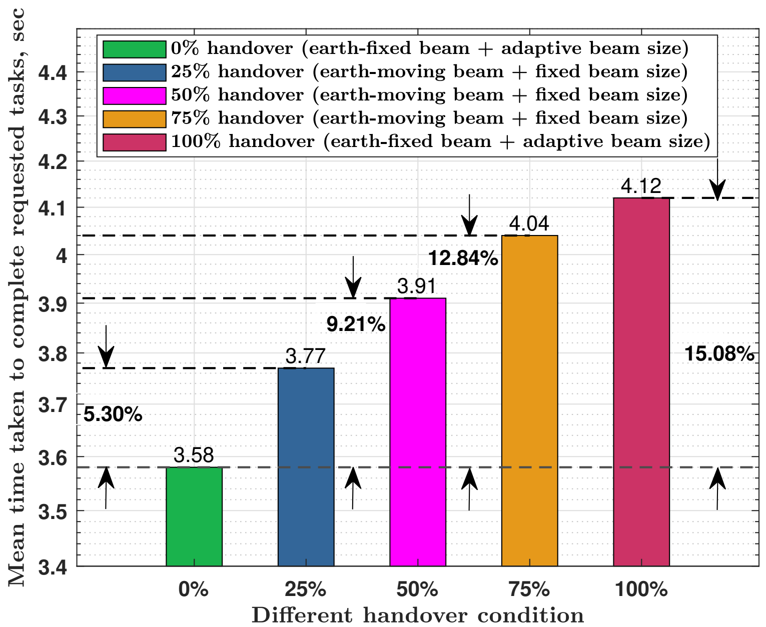

Figure 4.16: Mean task completion time for different HO scenarios.

tasks are predominantly assigned to the serving LEO satellite, leading to full utilization of its computational resources. Adjacent LEO satellites show slight under-utilization relative to their full capacity, while the GW remains comparatively under-utilized. The average task completion time in this scenario is approximately 9.46 seconds. As ISL and feeder link rates increase from 'R2' to 'R5', the unused computational capacity in the serving LEO satellite and adjacent LEO satellites also increases. However, the unused computational capacity in the GW decreases significantly due to a substantial improvement in the feeder-link rate and the absence of energy constraints for the GW. This results in a sudden 68.4% improvement in mean task completion time in 'R5', showcasing the massive parallel offloading (see Fig.4.13) of the requested tasks with the enhancement of the ISL and feeder link rates.

4.6.5 Impact of Beam Management Techniques

In this section, we demonstrate the impact of the *Earth-moving beam* and *Earth-fixed beam* schemes on task completion time. In the latter, we consider adaptive spot beams, which are only possible with a fully regenerative payload [123].

Fig. 4.15 depicts the impact of the *Earth-moving beam* and *Earth-fixed beam* on the HO scenario, along with their influence on the trend of task completion time. The *Earth-moving beam* exhibits frequent asynchronous partial HO, while the *Earth-fixed beam* undergoes synchronous *full-HO*. As shown in the figure, toward the end of the service period (indicated by the dashed vertical line), the task completion time is generally higher for the *Earth-fixed beam* (blue line) compared to the *Earth-moving beam* (magenta line). Conversely, during the service period, the *Earth-moving beam* typically exhibits longer task completion times, indicating more frequent HOs. This behavior arises because the *Earth-moving beam* sweeps across the Earth at 7.6 km/sec, while the *Earth-fixed beam* can continuously serve a given location for approximately 10 to 15 minutes.

Fig. 4.16 illustrates the impact of various HO scenarios on the mean task completion time. When *non-HO* conditions are considered, the mean task completion time is 3.58 seconds. In the cases of 25%, 50%, 75%, and 100% HOs, the mean task completion time worsens by 5.3%, 9.21%, 12.84%, and 15.08%, respectively.

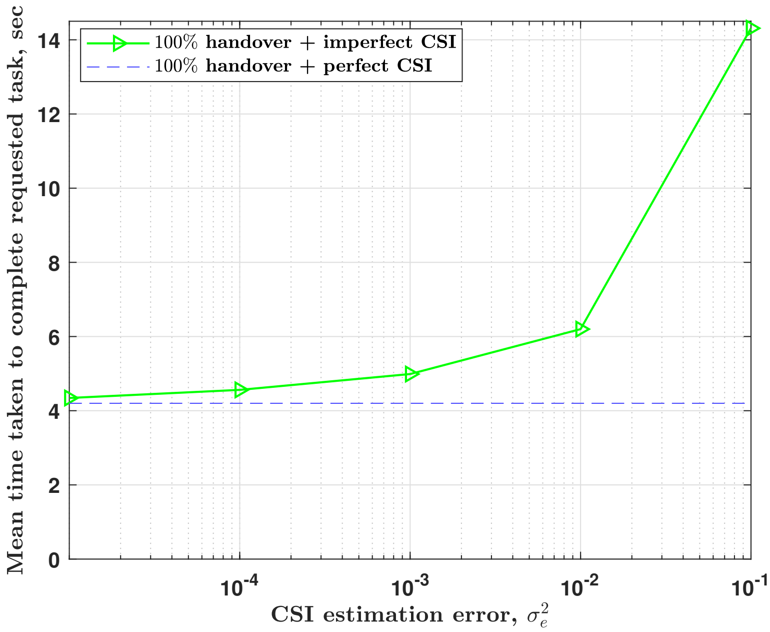

Figure 4.17: Mean task completion time versus error variance (σ_e^2) on the imperfect CSI for HO scenario.

4.6.6 Impact of Imperfect CSI

Having perfect CSI is practically challenging, especially in LEO systems. In realistic conditions, more likely in HO scenarios, the satellite operates based on imperfect channel estimation, in which the estimation error is modeled as a statistically independent variable from the estimated channel. Let σ_e^2 as the CSI estimation error variance, the achievable downlink rate under the robust receiver can be given as [123]: $r_{i,t}^{\text{imp}} = b_{i,t} \log_2 \left(1 + \frac{p_{i,t}^{(1)} \hat{g}_{i,t}^{(1)}}{p_{i,t}^{(1)} \sigma_e^2 + b_{i,t} N_0}\right)$.

Fig. 4.17 presents the impact of imperfect CSI on the proposed task offloading design using $r_{i,t}^{\text{imp}}$. The robustness of our design is demonstrated by its close performance to the perfect CSI case, even for estimation errors up to 10^{-3}. When the estimation error is large, the mean task completion time increases dramatically, as expected.

4.7 Open Discussion

In this section, we present unresolved research questions that require further clarification, such as: *How can a serving LEO satellite identify its adjacent neighboring LEO satellites?* and *How can the formulated problem be practically implemented?*

4.7.1 Selection of Adjacent Neighboring LEO Satellites

A serving LEO satellite can identify multi-orbital adjacent neighboring satellites using ephemeris data and ISLs. However, the selection of neighboring LEO satellites depends on the following factors:

- Availability of free computing resources in adjacent LEO satellites.
- Stability of ISLs, as maintaining stable links is challenging and requires specialized on-board hardware [123].
- Dynamic nature of the LEO topology.

The design of a robust algorithm that considers all these factors for selecting adjacent LEO satellites in our proposed SaaS platform remains an open discussion point and is left for future work. However, it is worth noting that incorporating dynamic availability of adjacent LEO satellites, based on real-time resource feedback during the monitoring window δ^{mon}, would primarily affect the upper bounds of constraints (4.10f)–(4.10h), i.e., $F_{\sum}^{(1)}$ and $E_{\sum}^{(1)}$. This modification does not alter the structure of the optimization problem, allowing the proposed framework to be extended to more realistic, dynamic scenarios without significant reformulation.

4.7.2 Practical Implementation Process

To practically implement the formulated design, one possible solution is to adopt a Kubernetes-based framework, as illustrated in Fig. 4.18. Kubernetes is a lightweight, open-source container orchestrator [124–127] with the capability to dynamically allocate computational resources across SAGINs. Upon implementation, Kubernetes will operate across all SAGINs, with core components like the application programming interface (API) server, scheduler, and pods, while OpenStack manages the virtual infrastructure [146]. SAGIN 0 will serve as both the master and worker node, while other SAGINs will function as worker nodes. The master node will host the control plane, optimizing task distribution and load balancing through horizontal and vertical auto-scaling [124–127], ensuring efficient processing of EO service requests, including image/video pre-processing and AI-based EO applications [147].

Although a Kubernetes-based approach is a viable framework for realizing this work in practice, other robust approaches may also exist, and this remains an open discussion point.

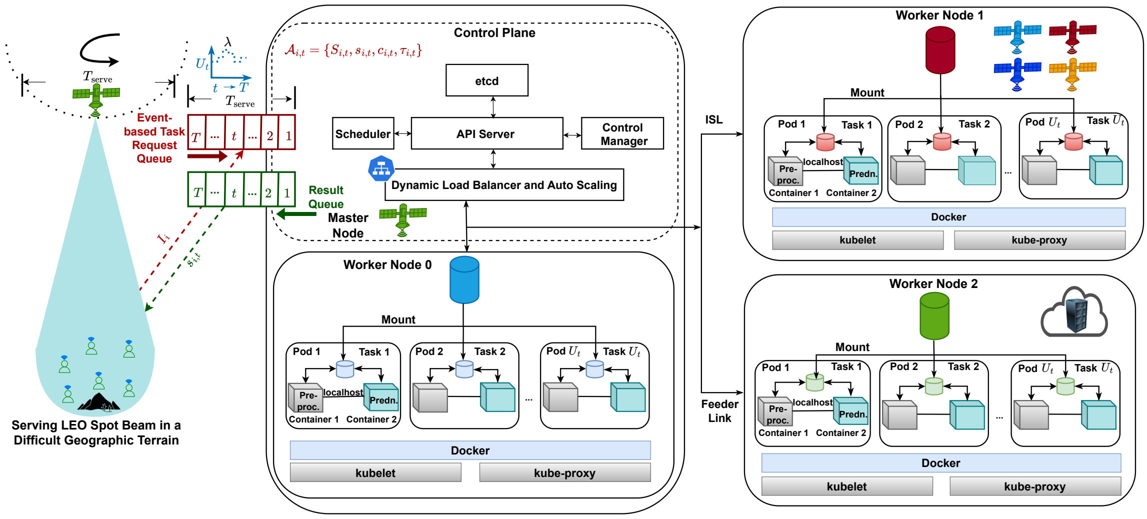

Figure 4.18: Schematic diagram of an open-source, very lightweight, and widely adopted container orchestrator (Kubernetes)-based SaaS platform designed for allocating computing resources of the SAGINs.

4.8 Summary

In this chapter, we proposed a novel framework for optimizing computational and communication resources in access SAGINs, addressing a realistic multi-objective optimization problem with practical implementation approaches using Kubernetes. Given the non-convex nature of the joint optimization problem, we employed an AO framework to decouple it into two sub-problems, which are solved iteratively: one for optimizing task offloading decisions, and the other for optimizing computation and service link bandwidth allocation. Simulations based on realistic parameters showed that our proposed approach outperforms other reference offloading approaches by at least 16%. Furthermore, we analyze the impact of the ISL and feeder link capacity on offloading decisions and task completion time. We also examined the influence of HOs showing that the *Earth-moving beam* scenario yields a better average task completion time than the *Earth-fixed beam* scenario, although with frequent and asynchronous HOs.

Chapter 5

ISAC-Enabled Handover Design in LEO Satellite Networks

Mega-constellations of low Earth orbit (LEO) satellites are envisioned to deliver global broadband and direct-to-cell services, requiring seamless handovers (HOs) to ensure uninterrupted connectivity. Conventional break-before-make HO protocols, although supported by inter-satellite links, suffer from beamforming delays due to the high orbital velocities of LEO satellites. To address these limitations, we propose an integrated sensing and communication (ISAC)-assisted HO (ISAC-HO) protocol that enables make-before-break HOs via ISAC-enabled ground terminals (GTs). A novel ISAC design capable of generating a three-dimensional (3D) beampattern under realistic conditions allows GTs to sense approaching LEO satellites without significantly compromising communication with the currently serving LEO satellite. The design problem is highly challenging due to its non-convexity and mixed-integer nature. To tackle these challenges, we propose an approach that leverages Riemannian manifold optimization and closed-form solutions. For multi-GT scenarios, we introduce a multi-agent deep reinforcement learning (MADRL) framework that mitigates sensing collisions and ensures quality-of-service under shared spectrum constraints. Numerical results confirm that the proposed ISAC design significantly improves the communication–sensing trade-off, enables smooth HO, and remains robust in the presence of imperfect channel state information.

5.1 Introduction

Extensive research is underway on beyond fifth-generation (B5G) and sixth-generation (6G) communications to provide seamless services and fulfill the growing and multifaceted demands of applications such as high-resolution video on demand, online gaming, and mission-critical Earth observation. These demands are caused due to the proliferation of high-end connected devices and rapid advancements in software applications, which have made the radio spectrum an increasingly scarce and valuable resource [42, 148]. B5G/6G networks aim to deliver high-throughput, low-latency communication services and are expected to support sensing and cognition capabilities [149], enabled by massive multiple-input multiple-output (MIMO) transceivers designed for dual purposes such as ISAC [150].

Meanwhile, the space industry is experiencing massive disruption [151] as tech giants race to deploy LEO satellite constellations to meet the unprecedented demand for mobile data traffic and to bridge the digital divide caused by geographic, economic, and technological barriers. Companies such as SpaceX, Eutelsat OneWeb, Amazon's Project Kuiper, Telesat, and Shanghai Spacecom Satellite Technology are competing to launch large constellations of LEO satellites that offer global broadband and direct-to-cell services [44].

The large constellations of LEO satellites are necessary to cover almost all parts of the Earth due to their smaller field of view (FoV) and non-geostationary nature. Because LEO satellites are non-geostationary, they can only provide service for a few minutes in an Earth-fixed-beam scenario during a single orbit [123]. To maintain continuous uninterrupted service between LEO satellites and GTs, seamless HO is required.

Based on 3GPP Release 17 [27], the HO procedure for LEO satellites follows conventional protocols used in terrestrial networks, as illustrated in Fig. 5.1. Specifically, it employs a *break-before-make conditional HO*[5.1], requiring the GT to detach from the serving LEO satellite before connecting to the approaching LEO (target). The serving LEO initiates this HO when triggered by factors such as radio resource management (RRM) measurements, service duration, and geographical location. However, the execution time of this procedure is non-deterministic. Empirical studies on Starlink LEO networks indicate that HO events occur approximately every 15 seconds, causing link interruptions ranging from tens to several hundreds of milliseconds, depending on satellite dynamics and GT responsiveness [152], [153].

[5.1] The dual active protocol stack (DAPS) HO, which is a make-before-break HO protocol, is not supported for non-terrestrial networks (NTN) according to 3GPP Release 17 [27].

These disruptions result in packet loss bursts and degraded throughput, which are inherent limitations of break-before-make HO protocols.

This challenge originates from the GT's requirement to rapidly beamform toward the fast-moving approaching LEO after receiving the `RRC Reconfiguration` message, which provides synchronization parameters such as ephemeris data, frequency, bandwidth, and timing. Synchronization with the approaching LEO can only begin after detachment from the current LEO, and only if the GT first verifies that HO conditions are met. This verification involves measuring the received signal strength (reference signal received power, RSRP) and quality (reference signal received quality, RSRQ) of the approaching LEO. The GT performs these measurements while still latched to the serving LEO, using periodic communication link interruptions to rapidly steer its directional beam toward the approaching LEO, guided by ephemeris data. To mitigate the uncertainty associated with this process, we propose an ISAC-HO mechanism as an alternative to the conventional HO procedure in LEO satellite systems.

To realize ISAC-HO, transceivers must be equipped with massive MIMO technology and ISAC functionalities. This enables a GT to maintain the communication link with the serving LEO while using a sensing beam to receive the downlink reference signals continuously broadcast by the approaching LEO. Conceptually, the ISAC-HO process in Fig. 5.2 operates as follows: When a conditional HO is triggered, the serving LEO initiates the process by transmitting an `RRC-ISAC Reconfiguration` message to the GT and an `HO Request` message to the approaching LEO. The `RRC-ISAC Reconfiguration` message instructs the GT to switch to ISAC mode without waiting for an HO acknowledgment, enabling it to proactively evaluate the link by measuring RSRP and RSRQ through its sensing beam while maintaining the active connection.

The HO acknowledgment received after the approaching LEO completes admission control serves purely as network-level validation. Upon receiving this message, the GT detaches only after sensing confirms that HO conditions are met, and immediately redirects its communication link using the pre-established beamforming vector. This mechanism enables seamless HO by eliminating transmission discontinuity. In the conventional scheme (Fig. 5.1), the GT detaches first and then evaluates the approaching LEO, resulting in a transmission halt. In contrast, the ISAC-HO[5.2] scheme allows the GT to evaluate the link through sensing first

[5.2] This work focuses on validating the core enabling capability of the proposed ISAC-HO protocol, i.e., ISAC beamforming. Full integration with measurement and decision protocols is part of the overall system concept.

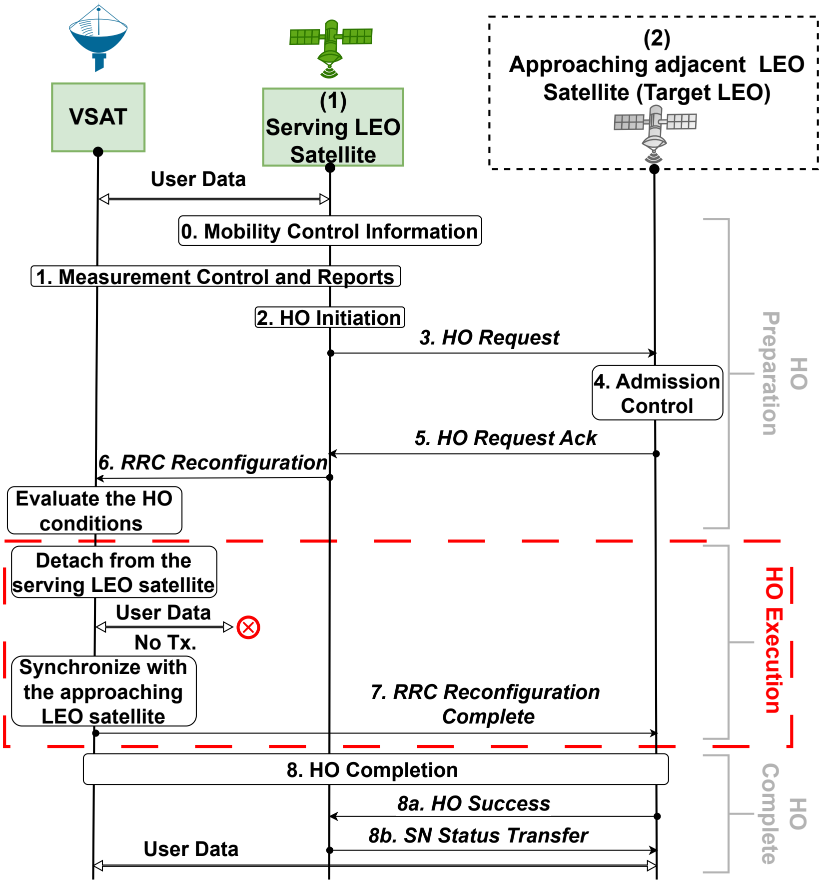

Figure 5.1: The conventional HO protocol for LEO satellites [27]

and detach only after confirmation, realizing a *make-before-break* approach that significantly reduces interruption time.

Building on the proposed ISAC-HO protocol illustrated in Fig. 5.2, we recognize that although seamless HO is achievable, its impact on joint communication and sensing performance must be rigorously evaluated under practical operational constraints. To achieve this, we model GTs as very small aperture terminals (VSATs) for their hardware advantages and investigate ISAC beamforming techniques to facilitate the development of an efficient ISAC design tailored to our practical use-case scenario.

5.1.1 Related Works

Early ISAC studies focused on interference management techniques to control the mutual interference between the radar and communications subsystems in single-carrier systems

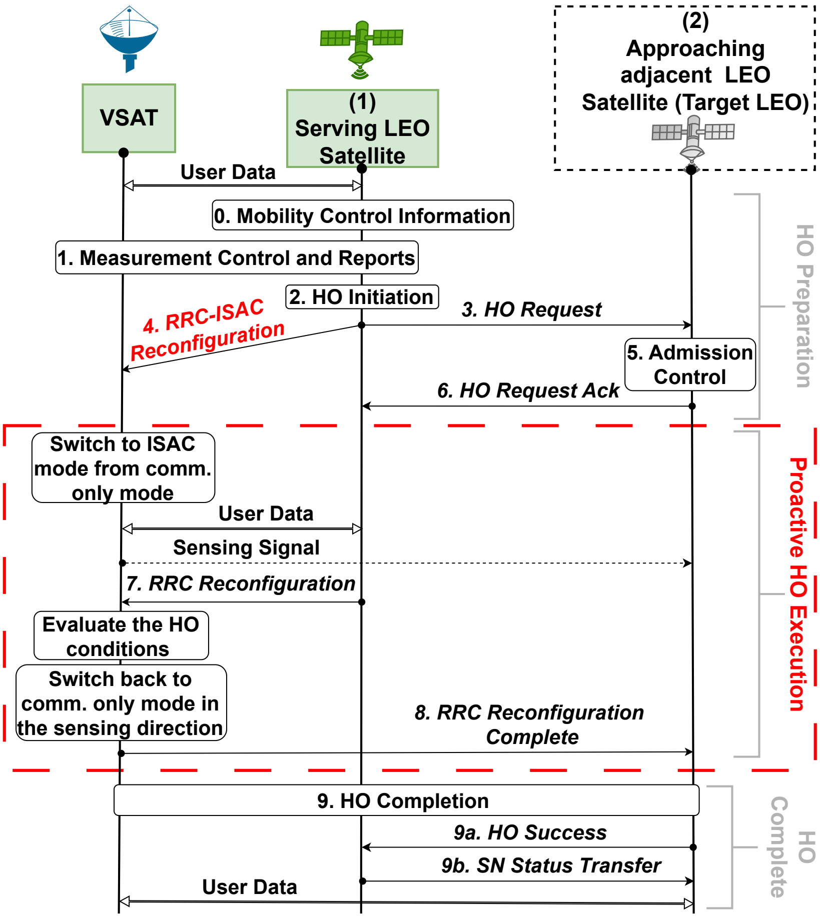

Figure 5.2: The proposed ISAC-HO protocol for LEO satellites.

[39, 154, 155]. In [39], two ISAC strategies for multi-user (MU) MIMO systems with monostatic radar sensing are proposed, involving either two separate sub-arrays for dedicated sensing and communication (DSAC) or a shared array at the base station. In [154], the transmit covariance matrix is designed to minimize the effective interference power at the radar receiver. In [155], it is shown that large-scale arrays can substantially mitigate mutual interference between radar and communications.

Recent works [156–159] have shifted focus towards ISAC in MU-MIMO systems, with [158, 159] employing orthogonal frequency-division multiplexing (OFDM) to enhance both radar and communication performance. In particular, [156] and [157] prioritize radar performance under communication constraints. In contrast, [158] targets the joint optimization of radar and communication performance using a weighted objective function, but without offering subcarrier-level flexibility. To address this limitation, the works in [159] introduce flexibility in subcarrier selection to improve the communication–sensing tradeoff in MIMO-OFDM ISAC systems.

While terrestrial ISAC has been extensively studied, its application in LEO satellite networks is still relatively unexplored. In [160], the authors provide an overview of LEO satellite systems and ISAC, proposing their integration in massive MIMO LEO systems for beam training, tracking, and resource allocation to achieve ubiquitous and seamless coverage. The work in [161] further explores ISAC designs based on rate-splitting in MU-MIMO LEO systems, aiming to maximize the minimum communication energy efficiency across all users while ensuring sensing performance. However, unlike [159], this approach lacks subcarrier selection flexibility, and the ISAC is exploited at the LEO side.

5.1.2 Motivation and Contributions

This work is motivated by the subcarrier-sharing techniques proposed in our previous work [159] for the terrestrial base stations. However, unlike [159], we consider a 3D single-user (SU) MIMO and MU-MIMO ISAC system for the uplink (UL) scenario to assist ISAC-HO in the LEO satellites. We adopt a realistic system model by employing large uniform planar arrays (UPAs) [162] at both the transmitter and receiver, which is not the case in prior works [39, 154–161].

Note that the design and optimization of UL ISAC is more challenging than in the downlink case and remains a subject of significant attention and ongoing research [163]. This is primarily because, in the UL scenario, collaboration between multiple users is not feasible without incurring additional communication overhead. Even if such overhead is allowed, it remains uncertain whether UL ISAC can be effectively realized within the short serving period of an LEO satellite. In contrast, downlink ISAC allows for more manageable spectrum allocation, as decisions are centrally coordinated by a single entity (the LEO satellite, in our case). Therefore, the technique developed in [159] is not readily applicable to our considered system and design problem, and a novel approach is required to effectively address UL ISAC in the MU-MIMO setting.

Our specific contributions can be summarized as follows:

- We propose a *make-before-break* ISAC-HO strategy in LEO satellite networks enabled by ISAC capability at GTs. In particular, we formulate a joint 3D ISAC beamforming and subcarrier allocation design for intelligent ISAC-HO, which offers flexibility in selecting subcarriers for sensing toward the approaching LEO satellite, while simultaneously maximizing communication spectral efficiency (SE) with the serving LEO satellite, enabling proactive ISAC-HO execution without service interruption. This design is subject to sensing, power, and minimum quality of service (QoS) constraints. The formulated problem belongs to the difficult class of non-linear mixed-integer non-convex optimization problems, which is generally non-deterministic polynomial-time hard.

- We propose to solve the optimization problem by decomposing it into two sub-problems. In the first sub-problem, we focus on designing an optimal covariance matrix to match the target sensing beampattern. In the second sub-problem, we adopt a three-stage sequential optimization scheme to evaluate the ISAC precoders and combiners. Specifically, subcarriers are first allocated for communication and sensing. Next, the ISAC precoders are optimized using Riemannian manifold optimization [39, 159]. Finally, a singular value decomposition (SVD)-based method is employed to obtain the combiner at the receiver. The proposed optimization is robust against imperfect channel state information (CSI) conditions.

- We extend the proposed ISAC design using a distributed MADRL approach to address practical multi-VSAT scenarios in which multiple VSATs are simultaneously served by an LEO satellite. Inspired by [35], our MADRL method enables decentralized spectrum allocation under uncertain dynamics, eliminating prior coordination while minimizing sensing collisions. Compared to *distributed ISAC optimization*, it improves SE and reduces both power consumption and collision probability.

- Finally, the advantages of the proposed framework are demonstrated with a realistic 3D beampattern and Starlink's LEO satellite scenario. For a 16×64 MIMO system with 8 subcarriers, the proposed ISAC scheme using 2 subcarriers achieves up to 17% improvement over the conventional ISAC scheme and up to 10% over DSAC within a given signal-to-noise (SNR) range. Its performance relative to time-division sensing and communication (TD-SAC) [164] depends on HO duration and frequency, and switching complexities. Furthermore, the robustness of the proposed ISAC design under imperfect CSI is also demonstrated.

The remainder of this chapter is organized as follows. Section 5.2 describes the system model and parameters. Section 5.3 presents the problem formulation. Section 5.4 introduces the decomposition and sequential optimization approach to address the proposed problem. Section 5.5 shows how imperfect CSI impacts our solution. Section 5.6 provides the solution for the MU-MIMO ISAC scenario based on MADRL. Section 5.7 demonstrates the effectiveness of the proposed schemes using numerical results. Finally, Section 5.8 concludes the chapter.

5.2 System Model

We focus on the UL in the fully regenerative LEO satellite networks, where the LEO satellite acts as a satellite access node and communicates with the VSAT via MIMO-OFDM technology. Due to the fast movement of LEO satellites, the VSAT experiences frequent HOs between the currently serving LEO and an approaching one. To enhance communication during these HOs, the VSAT is equipped with ISAC capability, which allows it to proactively sense the approaching LEO while maintaining communication with the currently active one, as illustrated in Fig. 5.3.

5.2.1 Communication System Model

The VSAT is equipped with a UPA consisting of N_{tx}^{x} elements along the x-axis and N_{tx}^{y} elements along the y-axis, forming a total of $N_{\mathrm{tx}} = N_{\mathrm{tx}}^{x} N_{\mathrm{tx}}^{y}$ antenna elements. It communicates with a UPA of $N_{\mathrm{rx}} = N_{\mathrm{rx}}^{x'} N_{\mathrm{rx}}^{y'}$ antenna elements on the LEO satellite, supporting N_s data streams, where $N_s \leq \min\{N_{\mathrm{tx}}, N_{\mathrm{rx}}\}$ and $N_{\mathrm{tx}} \ll N_{\mathrm{rx}}$. For ease of presentation, let $m \in \mathcal{M} = \{1, 2\}$ denote the LEO satellite access nodes in the ISAC-assisted VSAT system, where $m = 1$ refers to the serving LEO satellite, and $m = 2$ refers to the nearest approaching LEO satellite with a comparatively high service duration, which may reside in the same or adjacent orbits relative to the serving LEO. The position of LEO satellite m relative to the VSAT is represented by the triplet $\{d_m, \theta_m, \phi_m\}$, where d_m is the slant distance between the LEO satellite m and the VSAT, $\theta_m \in [-90°, 90°]$ and $\phi_m \in [-180°, 180°]$ are the elevation and azimuth angles seen from VSAT, respectively.

Let $\mathcal{K}_m = \{1, 2, \ldots, k, \ldots, K_m\}$ be the set of all UL subcarriers from the VSAT to LEO satellite m. Denote $\boldsymbol{s}_{1,k} \in \mathbb{C}^{N_s \times 1}$ as the N_s streams of the data symbols transmitted from

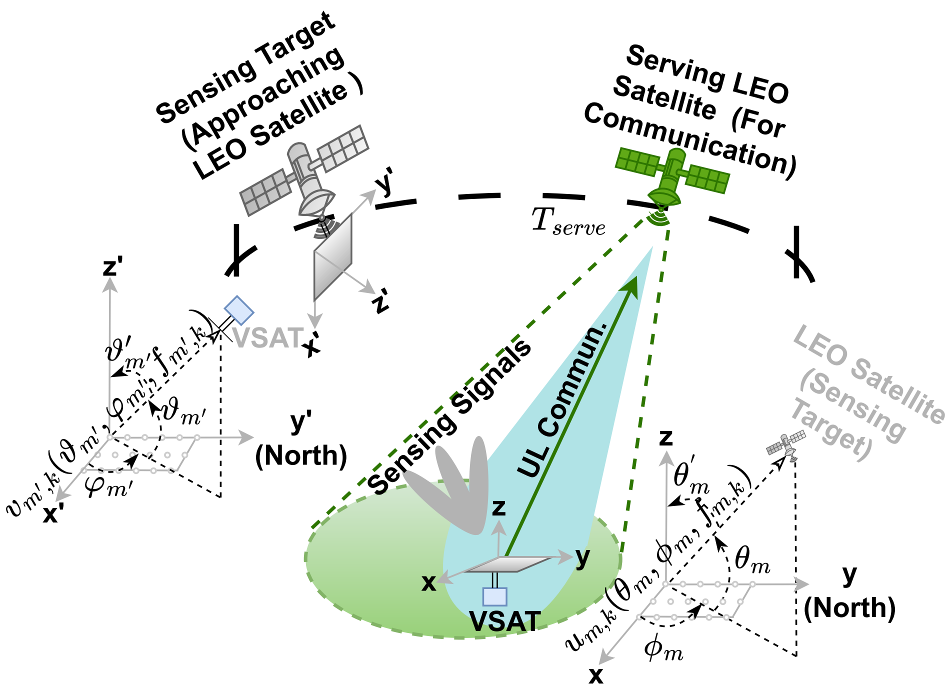

Figure 5.3: ISAC-assisted VSAT communicates with the serving LEO satellite, while beamforming towards an approaching LEO satellite to facilitate the *make-before-break* HO.

VSAT to serving LEO via subcarrier k with $\mathbb{E}\{\boldsymbol{s}_{1,k}\boldsymbol{s}_{1,k}^H\} = \boldsymbol{I}_{N_s}$. The received UL signal at the serving LEO (i.e., $m = 1$) via k-th subcarrier from the VSAT during its service period $T_{\texttt{serve}}$ is given as:

$$\boldsymbol{y}_{1,k} = \boldsymbol{F}_{1,k}^H \boldsymbol{H}_{1,k} \boldsymbol{W}_{1,k} \boldsymbol{s}_{1,k} + \boldsymbol{F}_{1,k}^H \boldsymbol{n}_{1,k}, \tag{5.1}$$

where $\boldsymbol{F}_{1,k} \in \mathbb{C}^{N_{\rm rx}\times N_s}$ is the combining matrix at the serving LEO, $\boldsymbol{H}_{1,k} \in \mathbb{C}^{N_{\rm rx}\times N_{\rm tx}}$ is the UL channel matrix from VSAT to serving LEO, $\boldsymbol{W}_{1,k} \in \mathbb{C}^{N_{\rm tx}\times N_s}$ is the precoding matrix at the VSAT, and $\boldsymbol{n}_{1,k} \sim \mathcal{CN}(0, \boldsymbol{I}_{N_s}\sigma_n^2)$ is the additive white Gaussian noise. After Doppler compensation [75], $\boldsymbol{H}_{1,k} = g_{1,k}\boldsymbol{v}\boldsymbol{u}^T$, where $g_{1,k}$ is the channel gain for the subcarrier k, incorporating both large-scale fading (dominated by free-space path loss) and small-scale fading components characterized by the Rician model in [42]; $\boldsymbol{v} \triangleq \{\boldsymbol{v}_{1,k}(\vartheta_1, \varphi_1, f_{1,k})\}_{\forall k} \in \mathbb{C}^{N_{\rm rx}\times 1}$ is the receive array response vector, wherein ϑ_1 and φ_1 are the elevation and azimuth angle of the arrival angle pairs at the serving LEO; and $\boldsymbol{u} \triangleq \{\boldsymbol{u}_{1,k}(\theta_1, \phi_1, f_{1,k})\}_{\forall k} \in \mathbb{C}^{N_{\rm tx}\times 1}$ is the transmit array response vector, wherein θ_1 and ϕ_1 are the elevation and the azimuth

Table 5.1: Summary of Main Notations

Notations	**Description**
$N_{\text{tx}}^{x} \times N_{\text{tx}}^{y}$	Number of planar antenna elements along the x- and y-axes of the VSAT antenna array
$N_{\text{rx}}^{x'} \times N_{\text{rx}}^{y'}$	Number of planar antenna elements along the x- and y-axes of the LEO satellite antenna array
$\mathcal{K}_m$	Set of all UL subcarriers from VSAT to m-th LEO
$\mathcal{J}_2$	Set of UL subcarriers for sensing approaching LEO
$\boldsymbol{H}_{1,k}$	UL channel matrix for the k-th subcarrier
$\boldsymbol{W}_{1,k}$	Precoder at the VSAT for the k-th subcarrier
$\boldsymbol{F}_{1,k}$	Combiner at the serving LEO for the k-th subcarrier
P_{V}, $\boldsymbol{N}_{1,k}$	Each VSAT total power and noise covariance matrix
$\boldsymbol{u}$, $\boldsymbol{v}$	Transmit and receive array response vectors
θ_1, ϕ_1	Elevation and azimuth angle of the departure angle pairs at the VSAT
ϑ_1, φ_1	Elevation and azimuth angle of the arrival angle pairs at the serving LEO

angle of the departure angle pairs at the VSAT. The symbol $f_{1,k}$ denotes the frequency of the k-th subcarrier used for this UL transmission.

Assuming a UPA with half-wavelength antenna spacing in both the x- and y-dimension, the steering vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ for communication with serving LEO using subcarrier k, where $\mathtt{X} \triangleq \{1, k\}$ denotes shorthand indexes, are defined as [165]:

$$\boldsymbol{u} = \frac{1}{\sqrt{N_{\text{tx}}}}[1, \dots, e^{j\lambda_{\mathtt{X}}(n^{x} cos\phi_1 + n^{y} sin\phi_1)cos\theta_1}, \dots, e^{j\lambda_{\mathtt{X}}((N_{\text{tx}}^{x}-1)cos\phi_1+(N_{\text{tx}}^{y}-1)sin\phi_1)cos\theta_1}]^T; \tag{5.2}$$

$$\boldsymbol{v} = \frac{1}{\sqrt{N_{\text{rx}}}}[1, \dots, e^{j\lambda_{\mathtt{X}}(n^{x'} cos\varphi_1 + n^{y'} sin\varphi_1)cos\vartheta_1}, \dots, e^{j\lambda_{\mathtt{X}}((N_{\text{rx}}^{x'}-1)cos\varphi_1+(N_{\text{rx}}^{y'}-1)sin\varphi_1)cos\vartheta_1}]^T, \tag{5.3}$$

where $\lambda_{\mathtt{X}} \triangleq \frac{\pi f_{\mathtt{X}}}{f_c}$. The indices $0 \leq n^{x} < N_{\text{tx}}^{x}$ and $0 \leq n^{y} < N_{\text{tx}}^{y}$ denote the x and y position of the UPA elements on the VSAT, while $0 \leq n^{x'} < N_{\text{rx}}^{x'}$ and $0 \leq n^{y'} < N_{\text{rx}}^{y'}$ represent the x' and y' positions of the UPA elements on the LEO satellite.

Assuming perfect CSI at the VSAT, based on (5.1), the SE at subcarrier k of the VSAT is given as:

$$\eta_{1,k} = \log_2\left(\det\left(\boldsymbol{I}_{N_s} + \frac{P_{\text{V}}}{N_s}\boldsymbol{N}_{1,k}^{-1}(\boldsymbol{F}_{1,k}^{H}\boldsymbol{H}_{1,k}\boldsymbol{W}_{1,k}\boldsymbol{W}_{1,k}^{H}\boldsymbol{H}_{1,k}^{H}\boldsymbol{F}_{1,k})\right)\right), \tag{5.4}$$

where P_{V} is the total power of the VSAT, $\boldsymbol{N}_{1,k} = \sigma_n^2\boldsymbol{F}_{1,k}^{H}\boldsymbol{F}_{1,k}$ is the noise covariance matrix after combining, and $\sigma_n^2 = N_0 b_{1,k}$. Here, N_0 is the noise spectral density and $b_{1,k} \triangleq \frac{B_\sum}{K_1}$ is the subcarrier k-th bandwidth, wherein $B_\sum$ is the total bandwidth.

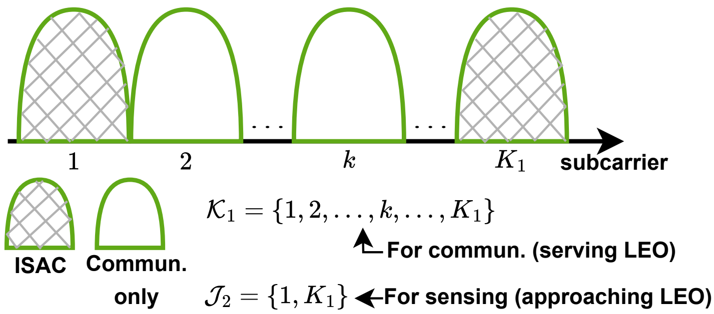

Figure 5.4: An example of flexibility in selection of ISAC subcarriers.

5.2.2 Sensing Operation

In addition to communication, the proposed ISAC framework allocates an optimized subset of subcarriers $\mathcal{J}_2 \subset \mathcal{K}_1$, with $|\mathcal{J}_2| = J_2$, as illustrated in Fig. 5.4, to simultaneously sense the approaching LEO satellite ($m = 2$) via 3D beamforming. This sensing operation proactively facilitates a *make-before-break* HO, as shown in Fig. 5.3, by confirming the presence of the target and ensuring a reliable communication link during the transition.

Assuming the spatial parameters $\{d_2, \theta_2, \phi_2\}$ of the approaching LEO are proactively known from network ephemeris data transmitted from the serving LEO ($m = 1$) to the VSAT as part of the `RRC-ISAC Reconfiguration` HO message (see Fig. 5.2), the VSAT optimizes a sensing covariance matrix $\boldsymbol{R}_{2,k}$ for each subcarrier $k \in \mathcal{J}_2$. The procedure for this optimization is explained in the next section. The matrix is designed to maximize the beamforming gain toward the approaching LEO based on the desired beampattern gain, thereby ensuring the sensing QoS required for seamless HO within the proposed ISAC framework with flexible subcarrier allocation.

5.3 Problem Formulation

Here, we formulate a joint optimization problem that maximizes the SE per subcarrier of the communication signal for the serving LEO while designing the optimal sensing covariance matrix based on the 3D steering vector of the approaching LEO, subject to a sensing QoS requirement. The proposed ISAC optimization problem is formulated as follows:

$$\mathcal{P}: \max_{\{\boldsymbol{R}_{2,k},\boldsymbol{W}_{1,k},\boldsymbol{F}_{1,k}\}_{k\in\mathcal{K}_1},\mu,\Gamma,\mathcal{J}_2} \frac{1}{K_1}\sum_{k\in\mathcal{K}_1}\eta_{1,k} \tag{5.5a}$$

$$\text{s.t.} \sum_{k\in\mathcal{J}_2}\sum_{n_{\text{el}}=1}^{N_{\text{el}}}\sum_{n_{\text{az}}=1}^{N_{\text{az}}} |\mu\mathcal{B}_2 - \boldsymbol{u}_{2,k}^H\boldsymbol{R}_{2,k}\boldsymbol{u}_{2,k}| \le \Gamma, \tag{5.5b}$$

$$\text{diag}(\boldsymbol{R}_{2,k}) = \mathbf{1}_{N_{\text{tx}}\times 1}P_{\text{V}}/N_{\text{tx}}, \quad \forall k, \tag{5.5c}$$

$$\boldsymbol{R}_{2,k} \ge \mathbf{0}_{N_{\text{tx}}\times N_{\text{tx}}};\ \boldsymbol{R}_{2,k} = \boldsymbol{R}_{2,k}^H, \quad \forall k, \tag{5.5d}$$

$$\|\boldsymbol{W}_{1,k}\|^2 \le P_{\text{V}}, \quad \forall k, \tag{5.5e}$$

$$\frac{\sum_{k\in\mathcal{J}_2}\|\boldsymbol{W}_{1,k}\boldsymbol{W}_{1,k}^H - \boldsymbol{R}_{2,k}\|^2}{J_2} \le \epsilon_0, \tag{5.5f}$$

where $\boldsymbol{R}_{2,k}$ denotes the Hermitian positive semi-definite covariance matrix of the sensing signal, μ represents a scaling factor for desired sensing beampattern matching, and $\boldsymbol{W}_{1,k}$ and $\boldsymbol{F}_{1,k}$ are precoder and combiner, respectively, which may or may not possess ISAC capabilities for subcarrier k. The desired beampattern gain, i.e., $\mathcal{B}_2 \triangleq \mathcal{B}_2(\theta_2^{n_{\text{el}}}, \phi_2^{n_{\text{az}}}, f_{2,k})$, describes the 3D target space defined by the elevation angle $\theta_2^{n_{\text{el}}}$, azimuth angle $\phi_2^{n_{\text{az}}}$, and subcarrier $f_{2,k}$ for sensing approaching LEO. The transmit steering vector $\boldsymbol{u}_{2,k}^H \triangleq \boldsymbol{u}_{2,k}(\theta_2^t, \phi_2^l, f_{2,k})$ is used to design the covariance matrix $\boldsymbol{R}_{2,k}$ of the sensing signal based on the scaled desired beampattern, $\mu\mathcal{B}_2$. The quality of the actual sensing beampattern, i.e., $\boldsymbol{u}_{2,k}^H\boldsymbol{R}_{2,k}\boldsymbol{u}_{2,k}$, is governed by the mean absolute error tolerance Γ, while the sensing performance of the ISAC beampattern is governed by the mean squared error (MSE) tolerance ϵ_0. To detect the approaching LEO, the VSAT scans a $N_{\text{el}} \times N_{\text{az}}$ angular grid of elevation and azimuth angle pairs for the approaching LEO, i.e., $\{\theta_2^{n_{\text{el}}}\}_{n_{\text{el}}=1}^{N_{\text{el}}}$ and $\{\phi_2^{n_{\text{az}}}\}_{n_{\text{az}}=1}^{N_{\text{az}}}$, over the detection ranges of $[0°, 90°]$ and $[-180°, 180°]$, respectively.

In problem $\mathcal{P}$, constraint (5.5b) defines the *3D sensing beampattern*[5.3] toward the approaching LEO, ensuring the mean absolute error from the desired beampattern $\mathcal{B}_2$ remains within Γ. Constraint (5.5c) guarantees that the waveform transmitted by different antenna elements has the same average power. Constraint (5.5d) ensures that the designed covariance matrix is Hermitian positive semi-definite. Constraint (5.5e) limits the total power of a VSAT for sensing and communication. Constraint (5.5f) ensures that the MSE in designing the covariance matrix of a subcarrier for the ISAC, based on the covariance matrix of the 3D

[5.3] Note that the design of the benchmark covariance matrix in (5.5b) differs from existing approaches [39, 159, 166] by incorporating an additional angular dimension, which is necessary for 3D waveform design.

beampattern design for sensing, remains within the tolerable limit ϵ_0.

Difficulty to solve problem $\mathcal{P}$: The problem $\mathcal{P}$ cannot be solved directly due to the non-convexity of the objective function, high order of $\boldsymbol{W}_{1,k}$, & integer variables $\mathcal{J}_2$ in constraint (5.5f). Moreover, constraint (5.5b) cannot ensure optimality of $\boldsymbol{R}_{2,k}$ as Γ is unknown and lacks a tight upper bound.

5.4 Problem Decomposition and Solution

In this section, a step-by-step algorithm is proposed to solve the original optimization problem $\mathcal{P}$ in (5.5) by decomposing it into two sub-problems. The overall solution strategy is presented in Fig. 5.5, which provides a high-level roadmap of the proposed approach.

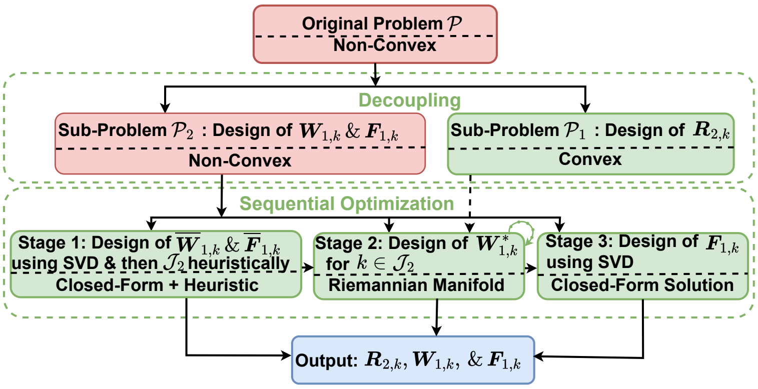

Figure 5.5: Roadmap to solve the original problem $\mathcal{P}$.

First, in the initial sub-problem, the optimal covariance matrix $\boldsymbol{R}_{2,k}$ is designed to achieve the desired target beampattern $\mathcal{B}_2$, which is essential for the next step. Once the optimal $\boldsymbol{R}_{2,k}$ is obtained, the second sub-problem focuses on evaluating sub-optimal values of $\boldsymbol{W}_{1,k}$, $\boldsymbol{F}_{1,k}$, and $\mathcal{J}_2$ to maximize the SE per subcarrier for serving the LEO satellite, while satisfying the sensing requirements.

5.4.1 Sub-Problem $\mathcal{P}_1$: Optimizing $\boldsymbol{R}_{2,k}$ under tighter constraints to better match the ISAC beampattern to $\mathcal{B}_2$

$$\mathcal{P}_1 : \min_{\{\boldsymbol{R}_{2,k},\mu,\Gamma\}} \Gamma; \quad \text{s.t. (5.5b)} - \text{(5.5d)}. \tag{5.6a}$$

The problem $\mathcal{P}_1$ is convex and can be efficiently solved using standard convex optimization tools such as CVX [167].

5.4.2 Sub-Problem $\mathcal{P}_2$: Optimizing $\boldsymbol{W}_{1,k}$, $\boldsymbol{F}_{1,k}$, and $\mathcal{J}_2$ using the previously optimized $\boldsymbol{R}_{2,k}$

$$\mathcal{P}_2 : \max_{\{\boldsymbol{W}_{1,k}, \boldsymbol{F}_{1,k}\}_{k \in \mathcal{K}_1}, \mathcal{J}_2} \frac{1}{K_1} \sum_{k \in \mathcal{K}_1} \eta_{1,k}; \quad \text{s.t. } (5.5\text{e}), (5.5\text{f}). \tag{5.7a}$$

The problem $\mathcal{P}_2$ is still non-convex due to its objective function and constraint (5.5f). Thus, to solve $\mathcal{P}_2$, we decompose it into three sub-problems and apply a sequential optimization approach, as described below.

Solving for $\mathcal{J}_2$

Without (5.5f), the problem (5.7) reduces to a communication-only problem. Consequently, communication combiners and precoders, i.e., $\overline{\boldsymbol{F}}_{1,k}$ and $\overline{\boldsymbol{W}}_{1,k}$, are derived using an eigenmode beamforming approach, due to the low rank of the channel, as satellite channels typically exhibit high correlation. More specifically, using an SVD-based approach, $\overline{\boldsymbol{F}}_{1,k}$ and $\overline{\boldsymbol{W}}_{1,k}$ are assigned to the left and right principal singular vectors of $\boldsymbol{H}_{1,k}$, respectively, such that $\eta_{1,k}$ is maximized. To satisfy the total power constraint at the VSAT, $\overline{\boldsymbol{W}}_{1,k}$ is normalized using an optimal power allocation strategy based on the water-filling approach [159], as follows:

$$\overline{\boldsymbol{W}}_{1,k} = \boldsymbol{V}_{1,k} \boldsymbol{P}^{1/2}, \tag{5.8}$$

where $\boldsymbol{V}_{1,k}$ contains the dominant right principal singular vectors of $\boldsymbol{H}_{1,k}$ and $\boldsymbol{P}^{1/2}$ is a diagonal matrix that satisfy the power constraint given in (5.5e). Based on $\overline{\boldsymbol{F}}_{1,k}$ and $\overline{\boldsymbol{W}}_{1,k}$, we evaluate the per-subcarrier SE from (5.4) and obtain the set $\mathcal{J}_2 \in \mathcal{K}_1$ of J_2 *subcarriers with the lowest SE, which will be used for ISAC design*[5.4], while the remaining subcarriers are used solely for communication.

[5.4] Note that the set of J_2 subcarriers are heuristically selected to ensure that the overall sensing and communication performance is not compromised.

Solving for optimal values of $\boldsymbol{W}_{1,k}$

Once $\mathcal{J}_2$ is determined, the next step is to design the precoding vectors for ISAC subcarriers within $\mathcal{J}_2$. Note that for subcarriers only used for communication, the precoding vectors are designed in (5.8). Consequently, this sub-problem is formulated as follows:

$$\begin{aligned}\mathcal{P}_2': \min_{\{\boldsymbol{W}_{1,k}\}_{k\in\mathcal{J}_2}} & \frac{\rho}{J_2}\sum_{k\in\mathcal{J}_2}\|\boldsymbol{W}_{1,k}\boldsymbol{W}_{1,k}^H-\boldsymbol{R}_{2,k}\|^2 \\ &+\frac{(1-\rho)}{J_2}\sum_{k\in\mathcal{J}_2}\|\boldsymbol{W}_{1,k}-\overline{\boldsymbol{W}}_{1,k}\|^2;\ \text{s.t. (5.5e)},\end{aligned} \tag{5.9a}$$

where ρ is a weighting factor that balances the sensing and communication performance, corresponding to the first and second terms of the objective function in (5.9a), respectively. Since constraints (5.5f) for different $k\in\mathcal{J}_2$ are independent, problem (5.9) can be solved across k independently using Riemannian manifold minimization to efficiently find a near-optimal solution [159]. Then, the final precoders for the proposed ISAC system are given as:

$$\boldsymbol{W}_{1,k}=\begin{cases}\overline{\boldsymbol{W}}_{1,k}, & k\notin\mathcal{J}_2\\ \boldsymbol{W}_{1,k}^*, & \text{otherwise},\end{cases} \tag{5.10}$$

where $\overline{\boldsymbol{W}}_{1,k}$ is given in (5.8) and $\boldsymbol{W}_{1,k}^*$ is the optimal solution of problem (5.9) solved by using the Riemannian manifold optimization [159].

Solving for optimal values of $\boldsymbol{F}_{1,k}$

After $\boldsymbol{W}_{1,k}$ is known for the proposed ISAC system, the solution for $\boldsymbol{F}_{1,k}$ can be obtained as the left principal singular vectors of $\boldsymbol{H}_{1,k}\boldsymbol{W}_{1,k}$ [159]. However, since $\boldsymbol{W}_{1,k}$ is computed at the VSAT and not immediately available to the serving LEO, we instead use the left singular vectors of $\boldsymbol{H}_{1,k}$ alone for the solution of $\boldsymbol{F}_{1,k}$. This approximation has minimal impact on average SE, as both methods employ unitary combiners that preserve the overall rate, though per-stream rates may differ. Moreover, transmitting $\boldsymbol{W}_{1,k}$ would introduce a one-hop propagation delay of approximately 47 channel uses (c.u.) within a 300 c.u. block. This estimate is based on a typical maximum slant distance of around 934 km, corresponding to a minimum working ground elevation angle of 25° for a LEO satellite [93] operating at an average orbital altitude of approximately 550 km above the Earth's surface. Considering that each channel uses (c.u.) lasts one symbol duration of 66.7 μs according to Long-Term Evolution (LTE) specifications [123], the resulting one-way propagation time is approximately 3.1 ms, which corresponds to roughly 47 c.u. ($\frac{3.1 \text{ ms}}{0.0667 \text{ ms}} \approx 47$). Note that such a one-hop delay may vary slightly depending on the satellite's instantaneous position and the elevation angle at the GT.

5.4.3 Initialization, Convergence, and Complexity Analysis

Initialization and Convergence Analysis

The sequential optimization framework for $\mathcal{P}_2$ comprises three stages (see Fig. 5.5). While stages 1 and 3 employ closed-form solutions, Stage 2 requires an iterative Riemannian manifold optimization procedure [159]. This solver is initialized with a random point on the complex hypersphere that satisfies the power constraint specified in (5.5e). For each sub-carrier k, it converges to a feasible solution for (5.9) within a fraction of a second, with the optimization terminating when the convergence criterion is met, that is, when the relative change in the objective function falls below 10^{-4}, as demonstrated in the convergence graph in Fig. 5.6.

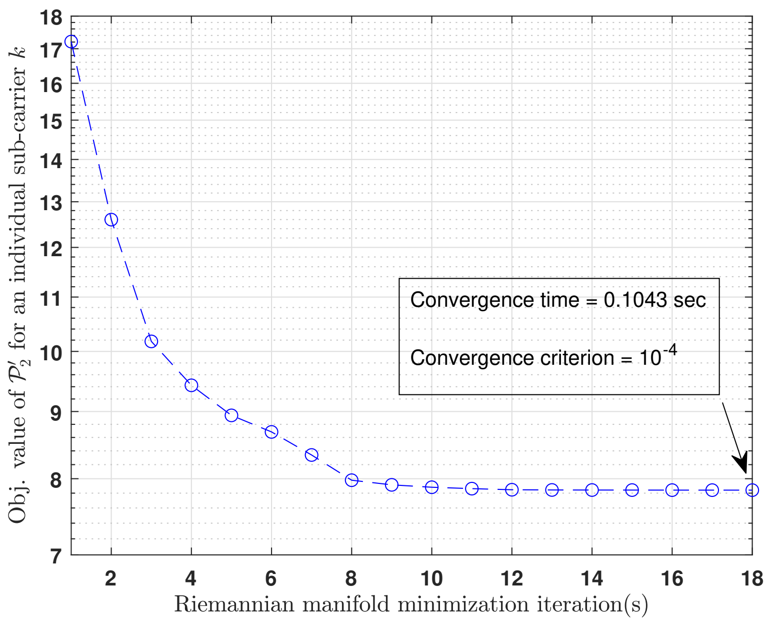

Figure 5.6: Convergence of Riemannian manifold minimization algorithm.

Complexity Analysis

Assuming that the interior point method is used to solve the convex problem in (5.6), the worst-case computational complexity is proportional to the cube of the number of real variables [123]. Since (5.6) involves N_s^2 real variables, its complexity is $N_{\mathrm{el}} N_{\mathrm{az}} \mathcal{O}(N_s^6)$. The complexity of computing $\overline{\boldsymbol{W}}_{1,k}$ in (5.8) and $\boldsymbol{F}_{1,k}$ is $K_1 \mathcal{O}(N_{\mathrm{tx}} N_{\mathrm{rx}}^2)$, while the complexity of the Riemannian manifold minimization is $J_2 \mathcal{O}(4N_{\mathrm{tx}}^2 N_s + 3N_{\mathrm{tx}})$ [39], [159]. Therefore, the overall complexity of the proposed method is $\mathcal{O}(N_{\mathrm{el}} N_{\mathrm{az}} N_s^6 + 4J_2 N_{\mathrm{tx}}^2 N_s + K_1 N_{\mathrm{tx}} N_{\mathrm{rx}}^2)$.

5.4.4 Extension to Multi-VSAT Scenarios

The solution to $\mathcal{P}$ presented in Sections 5.4.1 and 5.4.2, which leverages closed-form SVD and Riemannian manifold optimization, demonstrates effective convergence for the single-VSAT scenario (see Fig. 5.6). In practice, however, multiple VSATs often reside within the FoV of a single LEO satellite. In these multi-VSAT settings, each VSAT independently executes the ISAC-HO protocol to beamform sensing signals toward an approaching LEO satellite, using orbital information provided by the serving LEO via a `RRC-ISAC Reconfiguration` message (see Fig. 5.2).

A key challenge in this distributed environment is the selection of subcarriers for sensing ($\mathcal{J}_2$) purpose. Unlike centrally managed communication subcarriers ($\mathcal{K}_1$), sensing subcarriers are chosen independently by each VSAT from a shared spectrum pool through the proposed ISAC-HO algorithm, creating a risk of sensing collisions. Furthermore, applying the $\mathcal{P}$ optimization distributively at each VSAT leads to several issues including uncoordinated optimization, high computational complexity ($\mathcal{O}(N_{\mathrm{el}} N_{\mathrm{az}} N_s^6 + 4J_2 N_{\mathrm{tx}}^2 N_s + K_1 N_{\mathrm{tx}} N_{\mathrm{rx}}^2)$ per VSAT), and poor adaptability to dynamic NTN channels. Similarly, a restrictive approach where each VSAT is limited to its own communication subcarriers for sensing is also suboptimal, as it compromises both sensing accuracy and communication efficiency, thereby limiting the achievable communication–sensing performance tradeoff.

While $\mathcal{P}$ is effective for single-VSAT operation, it lacks mechanisms for inter-VSAT collaboration, leading to potential collisions and degraded performance. Although centralized coordination via the satellite is possible, it introduces significant signaling overhead and complexity. To address this challenge, we propose a MADRL-based approach [36, 168, 169]. This learning-based approach is well-suited to the stochastic radio environment and the physical separation of VSATs. It enables learning in unknown environments through trial-and-error while mitigating prohibitive computational demands, and it efficiently handles complex dynamics and large state-action spaces [35, 36]. Crucially, it allows for decentralized execution while incorporating limited coordination during training via global feedback, enabling agents to learn collaborative, collision-avoidant policies. This provides a scalable and adaptive solution that effectively complements the proposed optimization framework, as detailed in Section 5.6.

5.4.5 Solution Quality and Feasibility

Although the proposed method leverages SVD and Riemannian manifold optimization to efficiently solve sub-problems within $\mathcal{P}$ (see Fig. 5.5), the overall sequential optimization approach remains heuristic due to the non-convex nature of the original problem, whose global solution is computationally intractable. Nevertheless, the method yields a feasible, high-quality sub-optimal solution whose performance approaches the theoretical upper bounds (communication-only case), as illustrated in Fig. 5.13 of Section 5.7.2. Furthermore, all constraints of $\mathcal{P}$ are guaranteed to be satisfied if the sensing threshold ϵ_0 in (5.5f) is sufficiently large. Having established the solution's quality under ideal channel conditions, we now analyze its robustness to imperfect CSI in the next section.

5.5 Impact of Imperfect CSI

The previous sections assume perfect CSI for ISAC system design, which is optimistic in practice. Here, we propose a robust ISAC design that accounts for potential CSI estimation errors. The imperfect channel can be modeled as [123]:

$$\hat{\boldsymbol{H}}_{1,k} = \boldsymbol{H}_{1,k} + \boldsymbol{E}, \tag{5.11}$$

where $\hat{\boldsymbol{H}}_{1,k}$ is the imperfect channel estimation and $\boldsymbol{E}$ is the estimation error that is independent of the true channel and its elements follow the distribution $\mathcal{CN}(0, \boldsymbol{I}\sigma_e^2)$. Based on (5.1) and (5.11), the UL signal received at the serving LEO satellite under imperfect CSI can be expressed as:

$$\begin{aligned}\tilde{\boldsymbol{y}}_{1,k} =& \hat{\boldsymbol{F}}_{1,k}^{H}(\hat{\boldsymbol{H}}_{1,k} - \boldsymbol{E})\hat{\boldsymbol{W}}_{1,k}\boldsymbol{s}_{1,k} + \hat{\boldsymbol{F}}_{1,k}^{H}\boldsymbol{n}_{1,k} \\ =& \hat{\boldsymbol{F}}_{1,k}^{H}\hat{\boldsymbol{H}}_{1,k}\hat{\boldsymbol{W}}_{1,k}\boldsymbol{s}_{1,k} - \hat{\boldsymbol{F}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{W}}_{1,k}\boldsymbol{s}_{1,k} + \hat{\boldsymbol{F}}_{1,k}^{H}\boldsymbol{n}_{1,k},\end{aligned} \tag{5.12}$$

where $\hat{\boldsymbol{F}}_{1,k}^{H}$ and $\hat{\boldsymbol{W}}_{1,k}$ are imperfect combiner and precoder, respectively, derived from $\hat{\boldsymbol{H}}_{1,k}$ using the SVD-based method described in Section 5.4.2.

Let the SVD of the estimated channel be given by $\hat{\boldsymbol{H}}_{1,k} = \hat{\boldsymbol{U}}_{1,k}\hat{\boldsymbol{\Lambda}}_{1,k}\hat{\boldsymbol{V}}_{1,k}^{H}$. Based on this decomposition, the imperfect precoder and combiner are constructed as $\hat{\boldsymbol{W}}_{1,k} = \hat{\boldsymbol{V}}_{1,k}\sqrt{P_{\mathrm{V}}/N_s}$ and $\hat{\boldsymbol{F}}_{1,k}^{H} = \hat{\boldsymbol{U}}_{1,k}$, respectively. By substituting these into (5.12), the received signal can be expressed as:

$$\tilde{\boldsymbol{y}}_{1,k} = \underbrace{\hat{\boldsymbol{U}}_{1,k}^{H}\hat{\boldsymbol{U}}_{1,k}\hat{\boldsymbol{\Lambda}}_{1,k}\hat{\boldsymbol{V}}_{1,k}^{H}\hat{\boldsymbol{V}}_{1,k}\boldsymbol{s}_{1,k}\sqrt{P_{\mathrm{V}}/N_s}}_{\tilde{P}_{\mathrm{D}}} - \underbrace{\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{V}}_{1,k}\boldsymbol{s}_{1,k}\sqrt{P_{\mathrm{V}}/N_s}}_{\tilde{P}_{\mathrm{I}}} + \underbrace{\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{n}_{1,k}}_{\tilde{P}_{\mathrm{N}}}, \tag{5.13}$$

where $\tilde{P}_{\mathrm{D}}$, $\tilde{P}_{\mathrm{I}}$, and $\tilde{P}_{\mathrm{N}}$ represent the desired signal power, interfering signal power due to imperfect CSI, and noise power, respectively.

Accordingly, the achievable SE under imperfect CSI can be expressed as:

$$\tilde{\eta}_{1,k} = \log_2(\det(\boldsymbol{I}_{N_s} + \tilde{P}_{\mathrm{D}}/(\tilde{P}_{\mathrm{I}} + \tilde{P}_{\mathrm{N}}))). \tag{5.14}$$

By substituting $\tilde{P}_{\mathrm{D}}$ and $\tilde{P}_{\mathrm{I}}$ from Appendix B.1 into (5.14), we get:

$$\tilde{\eta}_{1,k} = \log_2\Big(\det\Big(\boldsymbol{I}_{N_s} + \frac{P_{\mathrm{V}} \cdot \mathrm{Diag}(\hat{\boldsymbol{\lambda}}_{1,k} \odot \hat{\boldsymbol{\lambda}}_{1,k})}{N_s(P_{\mathrm{V}}\sigma_e^2 + \sigma_n^2)}\Big)\Big), \tag{5.15}$$

where $\hat{\boldsymbol{\lambda}}_{1,k}$ denotes the singular values of $\hat{\boldsymbol{H}}_{1,k}$.

Since a decomposition and sequential optimization approach is used to solve problem $\mathcal{P}$, the performance of the sensing component in the ISAC system is highly dependent on the accurate covariance matrix, i.e., $\boldsymbol{R}_{2,k}$, while the performance of the communication subsystem relies on the optimal precoding design based on perfect CSI, i.e., $\overline{\boldsymbol{W}}_{1,k}$ (described in (5.8)).

However, when the estimated CSI is imperfect, the problem $\mathcal{P}_2^{'}$ is reformulated as:

$$\mathcal{P}_2^{''}: \min_{\{\tilde{\hat{\boldsymbol{W}}}_{1,k}\}_{k\in\mathcal{J}_2}} \frac{\rho}{J_2}\sum_{k\in\mathcal{J}_2}\|\tilde{\hat{\boldsymbol{W}}}_{1,k}\tilde{\hat{\boldsymbol{W}}}_{1,k}^H-\boldsymbol{R}_{2,k}\|^2+ \frac{(1-\rho)}{J_2}\sum_{k\in\mathcal{J}_2}\|\tilde{\hat{\boldsymbol{W}}}_{1,k}-\hat{\boldsymbol{W}}_{1,k}\|^2;\ \text{s.t. (5.5e)}, \tag{5.16a}$$

where $\tilde{\hat{\boldsymbol{W}}}_{1,k}$ is the power-scaled beamforming matrix for the ISAC system under an estimated imperfect CSI condition.

In the optimization problem $\mathcal{P}_2^{''}$, we aim to optimize $\tilde{\hat{\boldsymbol{W}}}_{1,k}$ with the objective function being the weighted sum of both sensing and communication performance metrics. Although $\tilde{\hat{\boldsymbol{W}}}_{1,k}$ is optimized to match the sensing covariance matrix, the resulting communication beamformer may not align correctly due to the imperfect reference precoding vectors, i.e., $\hat{\boldsymbol{W}}_{1,k}$. Since, in this case, we use $\hat{\boldsymbol{F}}_{1,k}=\hat{\boldsymbol{U}}_{1,k}$ as the combiner, the achievable SE expression for ISAC subcarrier k modifies to:

$$\tilde{\eta}_{1,k}=\log_2\Big(\det\Big(\boldsymbol{I}_{N_s}+\frac{\hat{\boldsymbol{H}}_{1,k}^H\hat{\boldsymbol{H}}_{1,k}\tilde{\hat{\boldsymbol{W}}}_{1,k}\tilde{\hat{\boldsymbol{W}}}_{1,k}^H}{N_s(\sigma_e^2\|\tilde{\hat{\boldsymbol{W}}}_{1,k}\|^2+\sigma_n^2)}\Big)\Big). \tag{5.17}$$

Note that the SE expressions in (5.15) and (5.17) assume that the estimation error $\boldsymbol{E}$ is uncorrelated with $\hat{\boldsymbol{H}}_{1,k}$, which is a reasonable approximation when σ_e^2 is not too large.

5.6 MADRL-based ISAC for Multi-VSAT Scenario

In this section, we consider a distributed MADRL-based ISAC (MADRL-ISAC) approach to solve $\mathcal{P}$ when multiple VSATs within the footprint of the serving LEO satellite are active simultaneously. Unlike the *distributed ISAC optimization*[5.5], the distributed MADRL-ISAC enables control over sensing decisions while maintaining decentralized execution. Specifically, during the offline training phase (which is generally executed on high-performance cloud servers near the gateway), VSATs served by the same LEO satellite inform the central node (see Fig. 5.7) of their sensing decisions by activating ISAC functionality at each time step t. Based on these reports, the central node provides the respective VSAT with global reward/penalty feedback. This limited exchange enables the proposed MADRL framework to

[5.5] The VSATs within the footprint of the serving LEO individually solve the ISAC optimization problem formulated in Section 5.3, using the solutions provided in Section 5.4, without any form of collaboration.

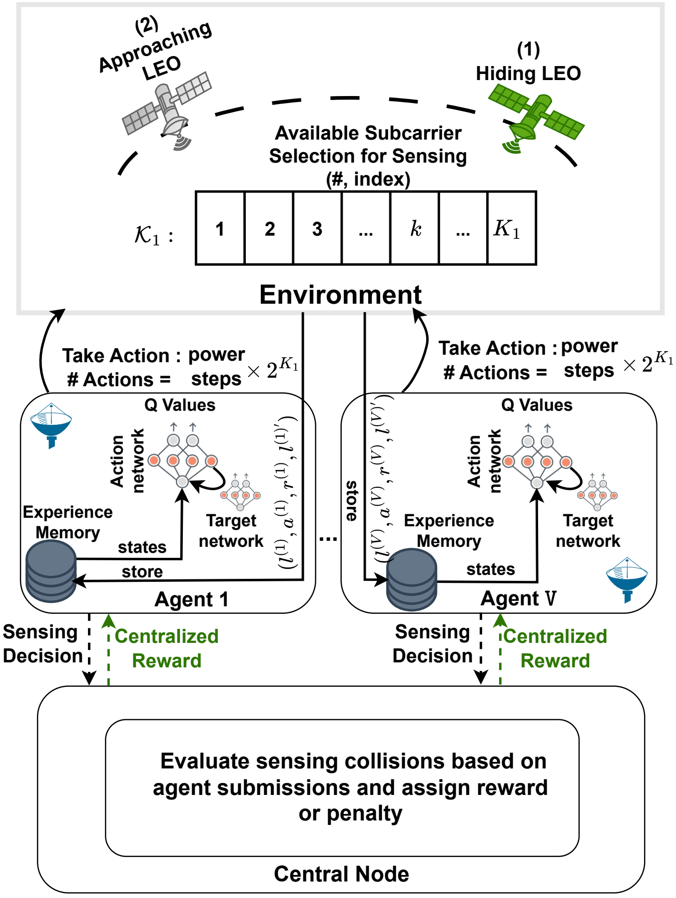

Figure 5.7: Distributed MADRL-ISAC for subcarrier selection in the multi-VSAT scenario.

be trained to proactively mitigate inter-VSAT sensing collisions while fulfilling the minimum QoS requirements (i.e., minimum achievable SE $\eta_{\min}$ and maximum tolerable sensing error ε_{th}) under dynamic environmental conditions.

The MADRL-ISAC problem is formulated as a partially observable Markov decision process, represented by the tuple $\langle \mathcal{V}, s_t, \{a_t^{(\mathtt{v})}\}_{\mathtt{v}=1}^{\mathtt{V}}, \{r_t^{(\mathtt{v})}\}_{\mathtt{v}=1}^{\mathtt{V}}, \{l_t^{(\mathtt{v})}\}_{\mathtt{v}=1}^{\mathtt{V}}, p, O \rangle$, where $\mathcal{V} = \{1, 2, \ldots, , \mathtt{v}, , \ldots, \mathtt{V}\}$ denotes the set of $\mathtt{V}$ agents, $s_t \in \mathcal{S}$ represents the global state at time step t, $a_t^{(\mathtt{v})}$ is the action taken by agent $\mathtt{v}$, and $r_t^{(\mathtt{v})}$ is agent $\mathtt{v}$'s hybrid reward function, which will be defined in Section 5.6.4. The term $l_t^{(\mathtt{v})}$ denotes agent $\mathtt{v}$'s local observation, p is the state transition probability function, and $O(\cdot)$ is the observation mapping function. At each time step t, each agent $\mathtt{v}$ receives a local observation $l_t^{(\mathtt{v})} = O(s_t, \mathtt{v})$ and selects an action $a_t^{(\mathtt{v})} \in \mathcal{A}^{(\mathtt{v})}$ based on its policy $\pi^{(\mathtt{v})}$. After decentralized action execution, each agent $\mathtt{v}$ receives a hybrid reward $r_t^{(\mathtt{v})}$. The environment (subcarrier selection) then transitions from state s_t to s_{t+1} according-ing to the transition probability $p(s_{t+1} \mid s_t, \{a_t^{(\mathtt{v})}\}_{\mathtt{v}=1}^{\mathtt{V}})$, which reflects the joint impact of all agents' actions. Subsequently, each agent $\mathtt{v}$ obtains a new observation of the environment, given by $l_{t+1}^{(\mathtt{v})} = O(s_{t+1}, \mathtt{v})$. Each episode simulation will terminate when a maximum number of time steps $I_{\max}$ is reached. The training continues for a total of $E_{\max}$ episodes.

Under partial observability, the state-value function for agent $\mathtt{v}$ is defined over its local observation $l_t^{(\mathtt{v})}$ as:

$$V^{(\mathtt{v})}(l_t^{(\mathtt{v})}) = \mathbb{E}_{\pi^{(\mathtt{v})},p}\left[\sum_{z=0}^{Z} \gamma^z r_{t+z+1}^{(\mathtt{v})} \mid l_t^{(\mathtt{v})}\right], \tag{5.18}$$

where $\mathbb{E}_{\pi^{(\mathtt{v})},p}$ denotes the expectation under policy $\pi^{(\mathtt{v})}$ and state transition probability p; $Z \triangleq \min(I_{\max} - t, T - t) - 1$, with $T \leq I_{\max}$ defines *early termination when the QoS requirement is not met*[5.6]; and $\gamma \in (0, 1]$ is the discount factor for future rewards.

Similarly, the action-value function (or Q-function) for agent $\mathtt{v}$ is defined with respect to the local observation-action pair $(l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})})$ as:

$$Q^{(\mathtt{v})}(l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})}) = \mathbb{E}_{\pi^{(\mathtt{v})},p}\left[\sum_{z=0}^{Z} \gamma^z r_{t+z+1}^{(\mathtt{v})} \mid l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})}\right]. \tag{5.19}$$

The goal of agent $\mathtt{v}$ is to learn an optimal policy $\pi^{(\mathtt{v})\star}$ that maximizes this Q-function. The corresponding Bellman optimality equation [36] under partial observability is:

$$Q^{(\mathtt{v})\star}(l, a) = r^{(\mathtt{v})}(l, a) + \gamma \mathbb{E}_{l'|l,a}[\max_{a'} Q^{(\mathtt{v})\star}(l', a')], \tag{5.20}$$

where $l \triangleq l_t^{(\mathtt{v})}$, $a \triangleq a_t^{(\mathtt{v})}$, l' and a' denote the next local observation and next action for agent $\mathtt{v}$, respectively. The optimal policy for agent $\mathtt{v}$ is given by $\pi^{(\mathtt{v})\star} = \arg\max_a Q^{(\mathtt{v})\star}(l, a)$.

To balance exploration and exploitation, we employ the Epsilon-Greedy policy [168], which is defined as:

$$\pi_\xi^{(\mathtt{v})} = \begin{cases} \arg\max_a Q^{(\mathtt{v})\star}(l, a), & \text{with prob. } 1\text{-}\xi, \\ \text{a random action } a, & \text{with prob. } \xi, \end{cases} \tag{5.21}$$

where $\xi \in [0, 1]$ is the epsilon rate, which is responsible for the trade-off between exploration

[5.6] A termination due to QoS violation occurs when the achievable SE $\eta^{(\mathtt{v})}$ falls below $\eta_{\min}$ or the sensing deviation $\varepsilon^{(\mathtt{v})}$ exceeds ε_{th}, where ε_{th} represents the maximum allowable average MSE between the desired and the designed sensing beampatterns [159].

and exploitation. Here, $\xi = 0$ indicates pure exploitation, while $\xi = 1$ pure exploration. Therefore, for effective learning, ξ is typically initialized close to 1 and gradually decayed to a small positive $\xi_{\min} > 0$.

5.6.1 Deep Q Network and Loss Function

Each agent $\mathtt{v}$ utilizes the deep neural network (DNN) to approximate the optimal Q-value function $Q^{(\mathtt{v})\star}$ in (5.20). In the MADRL-ISAC model, the DNN part consists of five fully connected layers, including input and output layers.

For a given experience tuple $\langle l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})}, r_t^{(\mathtt{v})}, l_{t+1}^{(\mathtt{v})} \rangle$, the temporal difference error $\delta_t^{(\mathtt{v})}$ can be written as:

$$\delta_t^{(\mathtt{v})} = \begin{cases} r_t^{(\mathtt{v})} - Q^{(\mathtt{v})}(l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})}; \Theta), & \text{if } l_{t+1}^{(\mathtt{v})} \text{ is } \textit{terminal}^{5.5}, \\ r_t^{(\mathtt{v})} + \gamma \max_{a'} Q^{(\mathtt{v})}(l_{t+1}^{(\mathtt{v})}, a'; \Theta^-) & \\ -Q^{(\mathtt{v})}(l_t^{(\mathtt{v})}, a_t^{(\mathtt{v})}; \Theta), & \text{otherwise}, \end{cases} \tag{5.22}$$

where Θ and Θ^- are the deep Q network (DQN) and target network parameters for the agent $\mathtt{v}$.

To train the DQN model for agent $\mathtt{v}$, the overall loss function $\mathcal{L}$ is set as:

$$\mathcal{L}^{(\mathtt{v})}(\Theta) = \mathbb{E}_{\{l,a,r,l'\} \in \mathcal{D}^{(\mathtt{v})}}[L(\delta_t^{(\mathtt{v})})], \tag{5.23}$$

where $\mathcal{D}^{(\mathtt{v})}$ denotes the mini-batch of experiences sampled from agent $\mathtt{v}$'s *experience replay memory*, and $L(\delta_t^{(\mathtt{v})})$ is based on the Huber loss function and is defined as [170]:

$$L(\delta_t^{(\mathtt{v})}) = \begin{cases} \frac{1}{2}(\delta_t^{(\mathtt{v})})^2, & \text{if } |\delta_t^{(\mathtt{v})}| \leq \delta_{\text{th}}, \\ \delta_{\text{th}} \cdot (|\delta_t^{(\mathtt{v})}| - \frac{1}{2}\delta_{\text{th}}), & \text{otherwise}, \end{cases} \tag{5.24}$$

where δ_{th} defines the threshold between quadratic and linear regimes. The Huber loss is used for its robustness to outliers, like the mean absolute error, while retaining sensitivity to small errors, similar to the MSE [170].

5.6.2 States

The global state at time step t is given by $s_t = \{\mathcal{F}_t^{(1)}, \mathcal{F}_t^{(2)}, \ldots, \mathcal{F}_t^{(\mathtt{V})}\}$, representing the joint subcarrier selections of all $\mathtt{V}$ agents. Each agent $\mathtt{v}$ maintains a binary selection vector $\mathcal{F}_t^{(\mathtt{v})} \in \{0,1\}^{K_1}$, where element $\mathtt{f}_{t,k}^{(\mathtt{v})} = 1$ indicates that agent $\mathtt{v}$ selected subcarrier k in time step $t-1$. This results in a global state space of cardinality $|\mathcal{S}|= 2^{K_1\mathtt{V}}$, as each agent independently selects among K_1 subcarriers. It is important to note that channel conditions and subcarrier collisions are part of the hidden state space, as they are not observable to any agent, and that power levels are excluded from the state space because they are instead considered within the action space.

5.6.3 Actions

For each time step t, each agent $\mathtt{v}$ selects a discrete transmit power level $P^{(\mathtt{v})}$ (in dB scale) from the range $[P_{\min}, P_{\max}]$, where $P_{\min}$ is the minimum operational power (in dB scale) for VSAT $\mathtt{v}$, $P_{\max}$ is the maximum allowable power (in dB scale), and Δ is the fixed step size (in dB scale) between adjacent power levels. Thus, the cardinality of the power-level action space is $|\mathcal{A}_{P^{(\mathtt{v})}}^{(\mathtt{v})}|= \frac{P_{\max}-P_{\min}}{\Delta} + 1$. Likewise, the agent also selects a preferred combination of subcarriers from the available set of K_1 subcarriers. The cardinality of the subcarrier selection action space is $|\mathcal{A}_{K_1}^{(\mathtt{v})}|= 2^{K_1}$. Therefore, the total action space cardinality for agent $\mathtt{v}$ is $|\mathcal{A}^{(\mathtt{v})}|= |\mathcal{A}_{P^{(\mathtt{v})}}^{(\mathtt{v})}|\times|\mathcal{A}_{K_1}^{(\mathtt{v})}|= (\frac{P_{\max}-P_{\min}}{\Delta} + 1) \times 2^{K_1}$.

5.6.4 Reward Function

The reward function for agent (VSAT) $\mathtt{v}$ is given as:

$$
\begin{aligned}
r^{(\mathtt{v})} = &\underbrace{\eta^{(\mathtt{v})}(1-\varepsilon^{(\mathtt{v})})}_{\text{Part I}} - \underbrace{w_1 \times 10^{P^{(\mathtt{v})}/10}}_{\text{Part II}} \\
&\underbrace{- w_2 \sum_{\mathtt{v}' \in \mathcal{V}\backslash\{\mathtt{v}\}} \mathbb{I}[\mathcal{F}^{(\mathtt{v})} \cap \mathcal{F}^{(\mathtt{v}')} \neq \emptyset]}_{\text{Part III}} \\
&\underbrace{- w_3[\max(0, \eta_{\min} - \eta^{(\mathtt{v})})] - w_4[\max(0, \varepsilon^{(\mathtt{v})} - \varepsilon_{\text{th}})]}_{\text{Part IV}} \\
&\underbrace{+ w_5 \prod_{\mathtt{v}' \in \mathcal{V}\backslash\{\mathtt{v}\}} \mathbb{I}[\mathcal{F}^{(\mathtt{v})} \cap \mathcal{F}^{(\mathtt{v}')} = \emptyset]}_{\text{Part V}},
\end{aligned}
\tag{5.25}
$$

where $\mathbb{I}[\cdot]$ denotes the indicator function, which returns 1 if the condition is true, and 0 otherwise; Part I presents the reward based on achieved SE ($\eta^{(\mathtt{v})}$) and sensing error ($\varepsilon^{(\mathtt{v})}$) for proposed ISAC design whose solution is provided in Section 5.4; Part II penalizes excessive power usage; Part III imposes a penalty for sensing collisions to actively discourage multiple agents from selecting the same sensing spectrum simultaneously; Part IV enforces QoS constraints for communication and sensing; and Part V introduces a collision-mitigation bonus that rewards an agent only when all agents, including itself, select completely non-overlapping sensing spectrum. This bonus directly incentivizes the multi-agent system to learn a globally coordinated, collision-free policy. The weight factors w_1 to w_5 balance the contributions of rewards, penalties, and bonuses in the overall reward function and can be determined through a trial-and-error mechanism.

5.6.5 Training Complexity

The computational complexity to train the proposed MADRL-ISAC algorithm is given by $\mathcal{O}(E_{\max} I_{\max} \sum_{\mathtt{v}=1}^{\mathtt{V}} D^{(\mathtt{v})} \sum_{l=1}^{l_c^{(\mathtt{v})}} n_{l-1}^{(\mathtt{v})} n_l^{(\mathtt{v})})$, where $n_l^{(\mathtt{v})}$ is the number of neurons in the l-th layer of agent $\mathtt{v}$'s neural network, with $n_0^{(\mathtt{v})}$ representing the input dimension and $n_{l_c^{(\mathtt{v})}}^{(\mathtt{v})}$ representing the output dimension. The output dimension depends on the size of action space $|\mathcal{A}^{(\mathtt{v})}|$, which *increases exponentially with the increase in subcarriers* K_1[5.7]. The term $D^{(\mathtt{v})} \triangleq |\mathcal{D}^{(\mathtt{v})}|$ denotes the batch size for agent $\mathtt{v}$, and $l_c^{(\mathtt{v})}$ is the number of DNN layers in its network. While larger $D^{(\mathtt{v})}$ linearly increases the cost per iteration, it often yields more stable gradient estimates and can reduce the required episodes $E_{\max}$ for convergence, representing a key training trade-off.

5.6.6 Inference Deployment for Testing Phase

Once the agents of the MADRL-ISAC model are well-trained in offline mode, their corresponding inference models are deployed to the respective VSATs via the serving LEO satellite. During the testing phase, the central node used solely for computing the global reward during training is no longer required. Each agent makes decisions locally based on its pre-trained policy. The exploration parameter (ξ) is set to zero, disabling exploration and enforcing purely exploitative behavior.

[5.7]This exponential relationship makes the training process computationally expensive for large K_1.

5.7 Performance Evaluation On Realistic Simulation Parameters

5.7.1 Network Modeling and System Parameters

The LEO satellite network is modeled in MATLAB using the `satelliteScenario` object, based on orbital parameters of the Starlink satellites LEO-31666 and LEO-5352 located in adjacent orbital planes [171]. In this setup, a VSAT in New York City, within the FoV of LEO-31666, communicates with it (green link) while simultaneously transmitting a sensing signal (grey link) toward the approaching LEO-5352, in order to establish a communication link with LEO-5352 at time t_0 for ISAC-HO, as illustrated in Fig. 5.8.

As described in Section 5.2.1, the wireless channel for the NTN is modeled using Rician fading with large-scale fading [123], capturing both deterministic line-of-sight (LoS) and non-line-of-sight (NLoS) components. The channel gain for subcarrier k of the NTN link, denoted by $g_{1,k}$, is expressed as $g_{1,k} = \sqrt{\beta_{1,k}}\,\hat{g}_{1,k}$, where the large-scale fading term $\beta_{1,k} = \frac{G_1 G_{\mathrm{V}} \lambda_{1,k}^2}{(4\pi d_1)^2}$ accounts for the free-space path loss and antenna gains, wherein, G_1 and G_{V} denote the antenna gains of the serving LEO and the VSAT terminal, respectively, d_1 is the instantaneous slant range between them, and $\lambda_{1,k}$ is the wavelength corresponding to subcarrier k. The small-scale fading component $\hat{g}_{1,k}$ is modeled as $\hat{g}_{1,k} = \sqrt{\Omega_{1,k}}\left(h_{\mathrm{LoS}}\sqrt{\frac{\kappa_1}{\kappa_1+1}} + h_{\mathrm{NLoS}}\sqrt{\frac{1}{\kappa_1+1}}\right)$, where $\Omega_{1,k} = \mathbb{E}\{|\hat{g}_{1,k}|^2\}$, κ_1 represents the Rician factor, h_{LoS} is the deterministic LoS component, and h_{NLoS} captures the NLoS scattering. Consequently, the real and imaginary parts of $\hat{g}_{1,k}$ are independently and identically distributed as $\mathcal{N}\left(\sqrt{\frac{\kappa_1 \Omega_{1,k}}{2(\kappa_1+1)}}, \frac{\Omega_{1,k}}{2(\kappa_1+1)}\right)$.

The simulations are performed for 16×64 MIMO-OFDM ISAC system. In accordance with 3GPP Release 18 [24] for direct-to-device services, an S-band carrier frequency ($f_c = 2$ GHz) is used for the service link, and atmospheric losses are neglected, as they are negligible below 6 GHz [145]. The VSAT, equipped with UPA, utilizes $K_1 = 8$ subcarriers. The frequency of the k-th subcarrier is given by $f_{1,k} \triangleq f_c + (k-1)b_{1,k}$, where f_c is the carrier frequency and $b_{1,k}$ is the subcarrier spacing. The array elements are spaced at $\frac{c}{2.4 f_{1,K_1}}$, where $c \approx 3\times 10^8$ m/s is the speed of light. This configuration generates a directional 3D beampattern focused on LEO-5352 (79° elevation, $-82°$ azimuth) for sensing at t_0. All remaining simulation parameters related to tasks, system configuration, and channel characteristics are summarized in Table 5.2.

5.7.2 Single VSAT Scenario: Optimization Based Approach

Figure 5.8: Realistic simulation scenario under single-VSAT condition.

Table 5.2: Simulation Parameters for ISAC

Parameters	**Value**
Serving and approaching LEOs, M	2
VSAT antennas, N_{tx}	4×4
LEO satellite antennas, N_{rx}	8×8
Serving LEO subcarriers, K_1	8
Approaching LEO target, $[\theta_2, \phi_2]$	$[79°, -82°]$
Carrier frequency, f_c	2 GHz
Total UL bandwidth, $B_{\sum}$	100 MHz
SNR	$[6 - 14]$ dB
Noise spectral density, N_0	-203.9772 dBW/Hz

Using the simulation parameters summarized in Table 5.2, we average results over 100 channel realizations and compare with the following baselines:

- *DSAC Design:* Allocates separate subsets of antenna elements and system power for simultaneous sensing and communication [39], enabling the application of the proposed ISAC-HO protocol.

- *TD-SAC Design:* Fully allocates all antenna elements and system power in alternating time slots for either sensing or communication [164], restricting the system to conventional HO protocol.

- *Conventional ISAC Design:* Utilizes all antenna elements and system power jointly for simultaneous sensing and communication across all subcarriers [158, 166], enabling the application of the proposed ISAC-HO protocol.

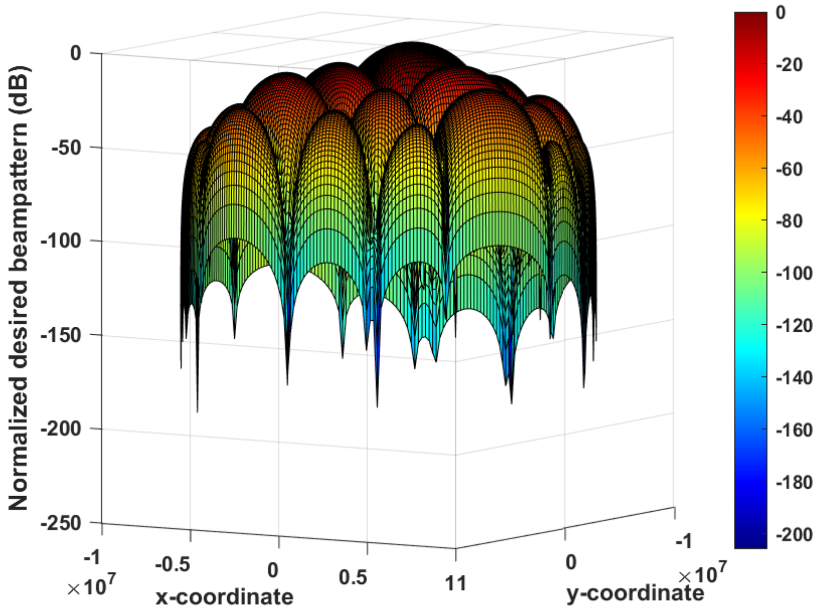

Figure 5.9: 3D desired beampattern design for ISAC scheme.

Figures 5.9 and 5.10 show the 3D and 2D views, respectively, of the desired target beampattern $\mathcal{B}_2$, generated based on the above-mentioned antenna element configuration, designed for sensing an approaching LEO satellite. As illustrated in Fig. 5.10, the half-power beamwidth (HPBW) of the target beampattern for the proposed design, denoted as $\delta_{\text{Prop.}}$, is 16°, achieved through the full utilization of antenna elements for sensing. In contrast, the HPBW for the DSAC design, δ_{DSAC}, is nearly twice as large, as the DSAC architecture partitions the antenna elements between sensing and communication. The resulting broader beamwidth reduces the suitability of the DSAC design for approaching LEO satellites, given the inverse relationship between beamwidth and directivity, where a narrower beam is essential for high-precision target sensing. It is also worth noting that both the TD-SAC and conventional ISAC designs can achieve an HPBW comparable to that of $\mathcal{B}_2$, provided all antenna elements are allocated for sensing.

Fig. 5.11 presents the 2D beampatterns on a linear scale, generated by the proposed ISAC design with $J_2 = \{2, 6\}$ for the intelligent ISAC-HO scenario, compared to the conventional ISAC design with $J_2 = K_1$, for $\rho = \{0.25, 0.5\}$. Despite utilizing a reduced number of subcarriers, the beampattern generated by the proposed ISAC scheme effectively approximates the desired beampattern and performs comparably to the conventional ISAC approach. The main lobes are successfully steered towards the direction of the approaching LEO satellite,

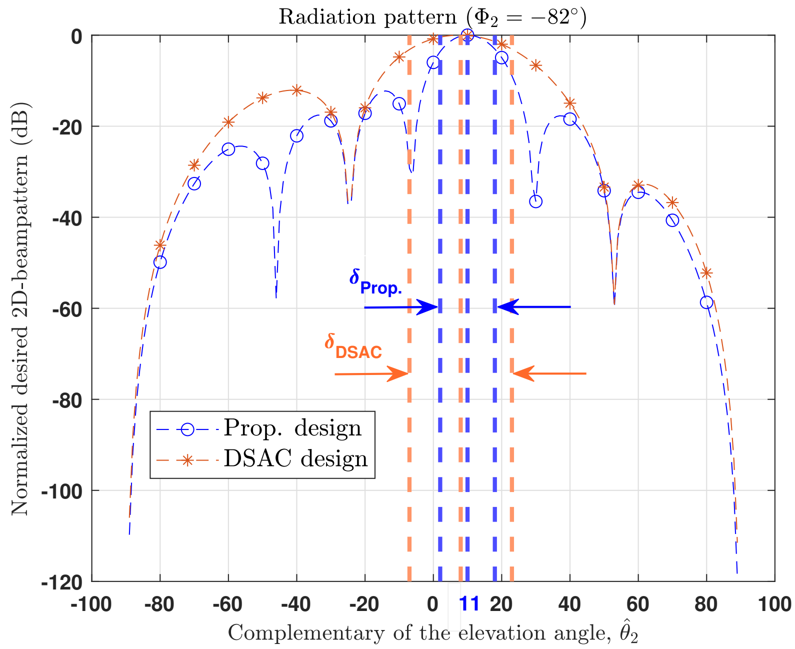

Figure 5.10: 2D desired beampattern design for ISAC and DSAC schemes.

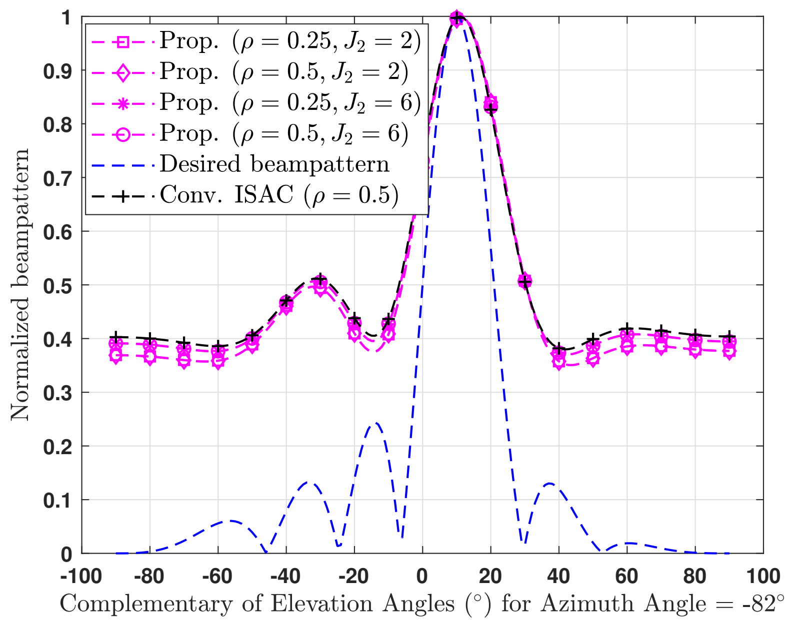

Figure 5.11: 2D beampatterns from proposed and conventional ISAC schemes.

while the sidelobes remain within acceptable deviation from the desired pattern. Although the proposed design shows slightly higher normalized mean-squared error deviation than the 2D beamforming methods in [39, 159, 166], this is attributed to the increased complexity of

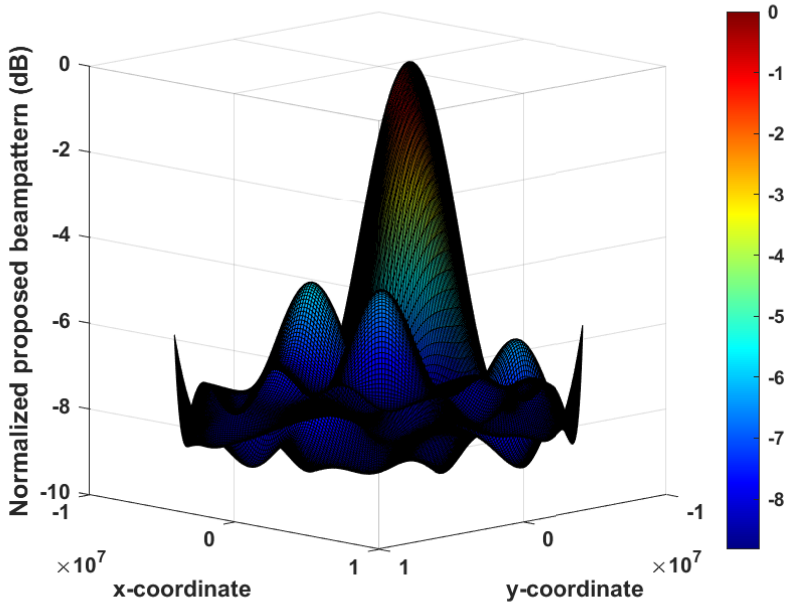

Figure 5.12: 3D beampattern with proposed ISAC scheme ($\rho = 0.5$, $J_2 = 6$).

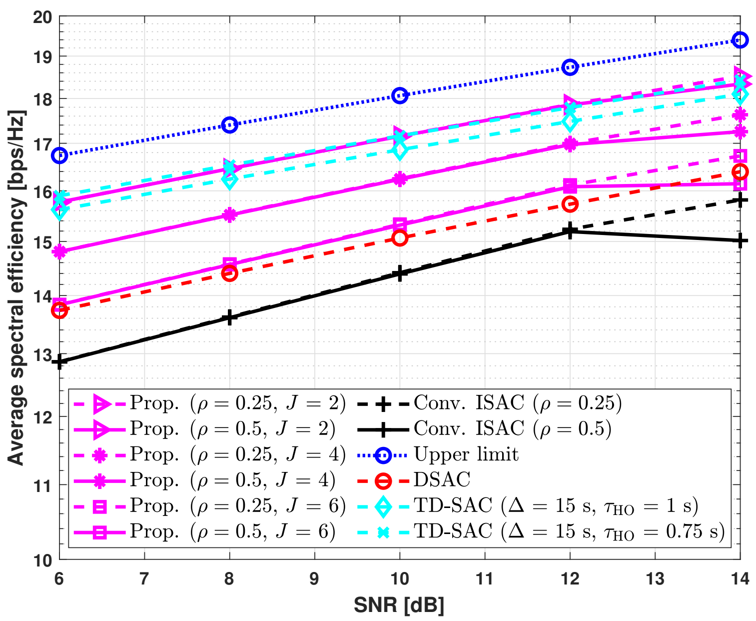

Figure 5.13: Achievable spectral efficiencies of the proposed & reference schemes with $\rho = \{0.25, 0.5\}$ & $J_2 = \{2, 4, 6\}$.

designing $\boldsymbol{R}_{2,k}$ for 3D beamforming that simultaneously satisfies both elevation and azimuth dimensions. Consequently, the impact of ρ is not clearly visible in Fig. 5.11 because the 2D projection cannot represent the complete 3D beamforming optimization performed by $\boldsymbol{R}_{2,k}$.

While ρ jointly influences both elevation and azimuth dimensions in the full spatial design, its effects become more distinguishable in conventional 2D beamforming as in [159].

Fig. 5.12 depicts the 3D beampattern of the proposed ISAC design for $\rho = 0.5$ and $J_2 = 6$. The figure confirms that the main lobe is accurately directed toward the approaching LEO satellite, and the resulting beam pattern closely approximates the desired reference in Fig. 5.9, despite noticeable sidelobe deviations. Although the radiation pattern to sense approaching LEO specifies around 20 dB sidelobe suppression requirement, the implemented design achieves only 4 dB suppression. This performance gap is attributed to the inherent trade-off between communication and sensing beamforming objectives, as well as the limitation introduced by employing a 2D covariance matrix in the synthesis of the 3D ISAC radiation pattern.

Fig. 5.13 illustrates the SE of the proposed ISAC scheme with baseline methods for various SNRs with $\rho = \{0.25, 0.5\}$ and $J_2 = \{2, 4, 6\}$. The proposed design approaches the SE upper limit (theoretical communication-only limit) with minimal subcarrier allocation for sensing, outperforming both DSAC and TD-SAC. In particular, TD-SAC, which supports break-before-make HOs, is highly sensitive to HO dynamics and performs reliably only when HO interruptions are short (e.g. $\tau_{\mathrm{HO}} < 1$ s) compared to the periodic network reconfiguration interval (e.g., $\Delta = 15$ s in LEO systems such as Starlink) [152], [153]. However, this is often not the case in large-scale NTN deployments, and its performance deteriorates under longer or more frequent HOs. As reported in [153], throughput measurements from the Starlink Standard dish show that the network reconfiguration process alone can cause approximately 9% packet loss. In contrast, the proposed ISAC-HO design employs a make-before-break strategy, eliminating packet loss during successful HOs.

The proposed system operates dynamically, using the ISAC framework only during the brief HO transition to ensure seamless connectivity. Once the HO is complete, it switches to a communication-only mode, as illustrated by the proposed ISAC-HO protocol in Fig. 5.2. As expected, the proposed scheme's SE degrades as the number of subcarriers increases, eventually converging with the conventional ISAC design when all subcarriers are utilized for both communication and sensing ($J_2 = K_1$). Furthermore, the proposed design achieves better SE at low ρ, but it degrades as ρ increases. At higher ρ, more emphasis is given to the sensing in the proposed ISAC scheme, and the achieved 3D beampattern shown in Fig. 5.12 exhibits only 4 dB sidelobe suppression. This leads to power spillage at high SNR, which reduces SE. This effect is more noticeable in the conventional ISAC design.

5.7.3 Multiple VSAT Scenario: MADRL Based Approach

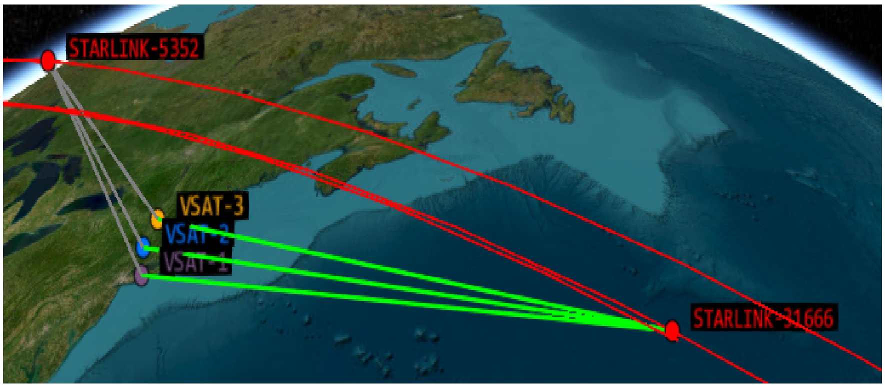

Figure 5.14: Realistic ISAC simulation scenario under multi-VSAT conditions.

To evaluate the performance of the MADRL-ISAC model introduced in Section 5.6, we consider a multi-VSAT scenario illustrated in Fig. 5.14. In this setup, multiple ISAC-assisted VSATs are concurrently served by the same LEO satellite, i.e., LEO-31666, while simultaneously transmitting sensing signals to the approaching satellite, i.e., LEO-5352, which is located in an adjacent orbit, to enable intelligent ISAC-HO.

We consider a total of $K = 6$ *available subcarriers*[5.8] , with three VSATs served by the same LEO satellite (LEO-31666), namely, VSAT-1, VSAT-2, and VSAT-3. Each of the three VSATs (or agents) is orthogonally allocated a distinct set of subcarriers by the current serving LEO to establish a communication link without interference. For simplicity, subcarrier indices $\{1, 2\}$, $\{3, 4\}$, and $\{5, 6\}$ are allocated to VSAT-1, VSAT-2, and VSAT-3, respectively, with equal bandwidth. The subcarrier indices to be allocated for sensing jointly, based on our proposed ISAC design for the multi-VSAT scenario, are determined by the MADRL-ISAC model. The remaining simulation parameters for the proposed MADRL-ISAC model are summarized in Table 5.3. The performance of the model in terms of convergence, as well as system-level metrics such as collision rate, SE, sensing accuracy, and power consumption, is presented in the following subsections.

Table 5.3: Simulation Parameters for MADRL-ISAC

Parameters	**Value**
Sensing direction (elevation) from $\mathtt{V}$ VSATs	[79°, 88°, 76°]
Sensing direction (azimuth) from $\mathtt{V}$ VSATs	[−82°, −75°, −75°]
Training episodes, testing episodes	1–500, 501–600
Agent $\mathtt{v}$ total actions, $\lvert\mathcal{A}^{\mathtt{v}}\rvert$	320
Operating power of VSAT, $[P_{\min}, P_{\max}]$	[6, 14] dBW
Power step, Δ	2 dBW
Max. iteration ($I_{\max}$), discounted factor (γ)	100, 0.9
Min. SE ($\eta_{\min}$), max. allowable sensing error (ε_{th})	8 [bps/Hz], 0.06
DQN learning rate, α	0.0001
DQN activation functions, optimizer	ReLU, Linear, Adam
Reward weight factors, $[w_1 - w_5]$	[0.5, 5, 10, 10, 15]
Epsilon rate (ξ), min. epsilon ($\xi_{\min}$)	0.995, 0.15

Convergence

We herein present the training and testing performance of the trained MADRL-ISAC model. The first 500 random channel realizations are used during the training phase, while the random channels with indices from 501 to 600 are used for testing.

Fig. 5.15 presents the total reward as a function of the number of training episodes for all three agents (or VSATs). The moving average reward for each agent increases as the number of training episodes increases. This suggests that the number of iterations within each episode is sufficiently high, indicating that QoS violations occur only after several iterations, which signifies that agents are effectively learning. After a certain number of training episodes, the moving average of the total reward for all agents stabilizes, following a similar pattern. However, it does not converge to a single value due to the fluctuating nature of the radio channel and potential subcarrier collisions, which are influenced by the distributed nature of the training. The moving average reward curves for all agents are nearly identical, as they operate within the footprint of the same serving LEO, and the collision information is incorporated into the reward function, as described in (5.25).

Fig. 5.16 shows the test performance of the trained MADRL-ISAC model. During the testing phase, the exploration parameter ξ is set to 0, enabling pure exploitation based on the trained policy (i.e., no exploration). Accordingly, the value of $I_{\max}$ is set to 1, meaning that each episode corresponds to a single time step t. The total reward versus episode

[5.8] The number of subcarriers K is not a practical assumption due to the massive computational cost involved in training MADRL-ISAC. According to 3GPP standard Rel. 18 [24], there can be a total of $132 \times 12 = 1584$ subcarriers in a 100 MHz bandwidth for NTN, assuming each subcarrier has a bandwidth of 60 kHz.

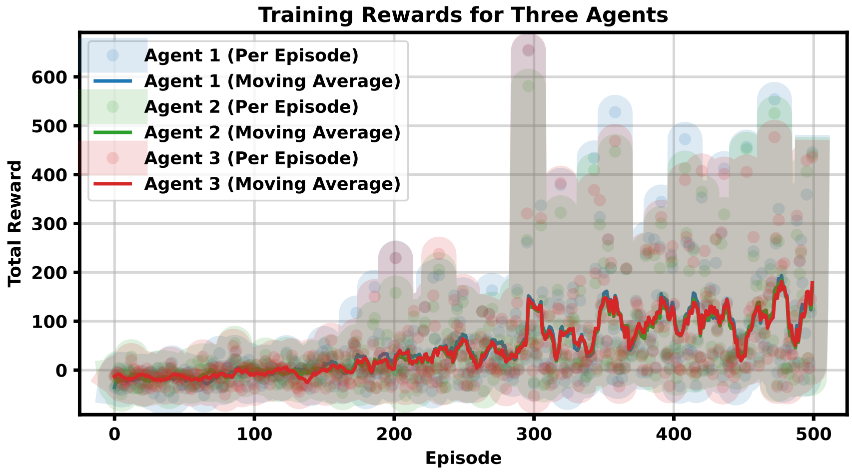

Figure 5.15: MADRL-ISAC model performance during training.

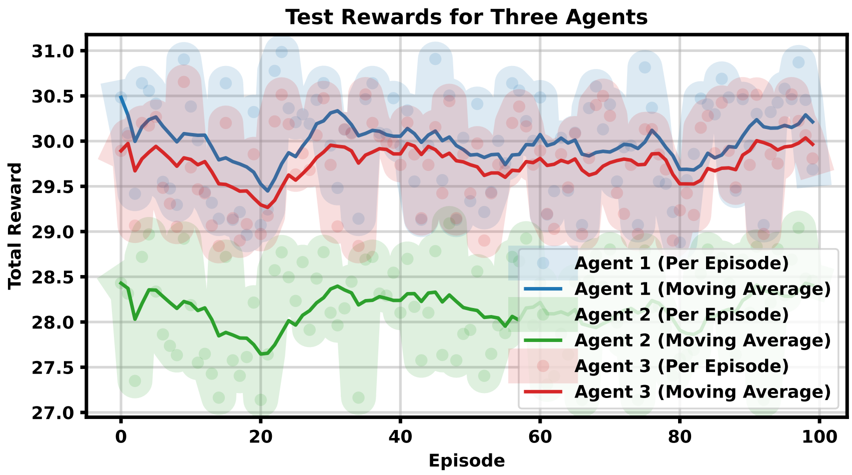

Figure 5.16: MADRL-ISAC model performance during testing.

graph in Fig. 5.16 reflects consistent state decisions that maximize system-level performance. *The selected subcarriers for sensing by VSAT-1, VSAT-2, and VSAT-3 are* $\{5\}$, $\{6\}$, *and* $\{1,\ 2\}$, *respectively*[5.9]. Despite identical state outputs, the total rewards vary across test episodes due to the stochastic nature of the radio channel. The evaluation of each agent's test performance employs the same formulation as the training reward function in (5.25), where the calculated metric depends on individual SE, sensing accuracy, power consumption, and QoS satisfaction. The coordinated subcarrier selection emerges from the trained policy, which inherently avoids subcarrier collisions through the learning process. The variation in the evaluated test performance across different VSAT agents reflects their distinct channel

[5.9]This is because the proposed ISAC system model in (5.5) does not account for interference between communication and sensing signals, and a similar realistic simulation environment is illustrated in Fig. 5.14.

conditions and operational constraints while maintaining effective coordination. The impact of the model on overall system performance is further discussed in the following subsections.

Collision Rate

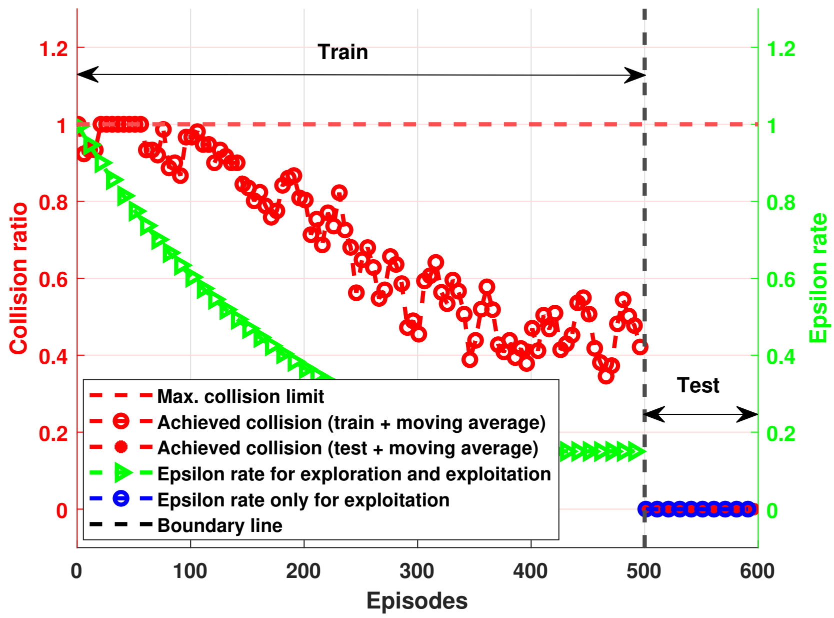

Figure 5.17: Collision rate achieved by the MADRL-ISAC model.

Fig. 5.17 depicts the collision performance of the MADRL-ISAC model during the training and testing episodes. In the early training episodes, the collision rate is high, i.e., between 80% and 100%. However, as the model learns through the Epsilon-Greedy policy over more training episodes, the collision rate reduces to less than 50%. Notably, the simulation results for the testing phase show that the fully trained policy achieves a zero-collision rate. This outcome occurs because each agent has learned to select a distinct set of subcarrier indices for sensing. Specifically, the test results indicate that the sensing subcarriers for VSAT agents 1–3 are $\{5\}$, $\{6\}$, and $\{1, 2\}$, while their communication subcarriers are $\{1, 2\}$, $\{3, 4\}$, and $\{5, 6\}$, respectively, ensuring no overlap and thus eliminating collisions.

Spectral Efficiency and Sensing Accuracy

Fig. 5.18 illustrates the performance of the trained MADRL-ISAC model in terms of the sum SE and the sum sensing error. As seen, both metrics improve with an increase in training episodes, fulfilling the QoS bounds. The test results indicate that the sum SE achieved by

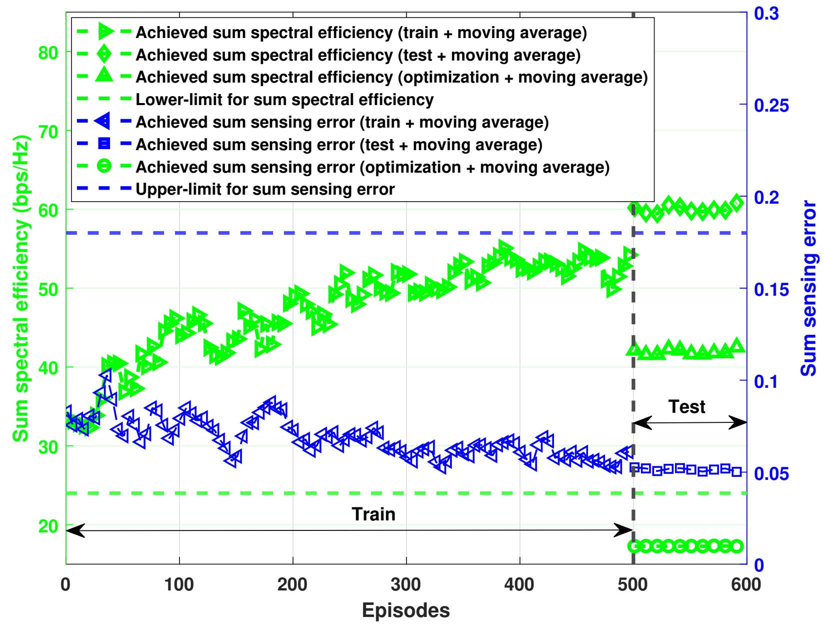

Figure 5.18: Sum SE and sensing error achieved by the MADRL-ISAC model.

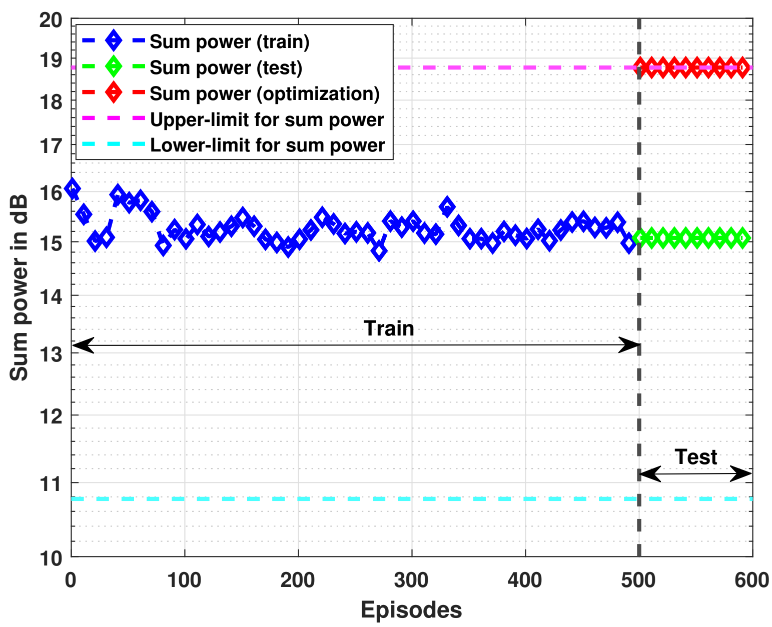

Figure 5.19: Sum power achieved by the MADRL-ISAC model.

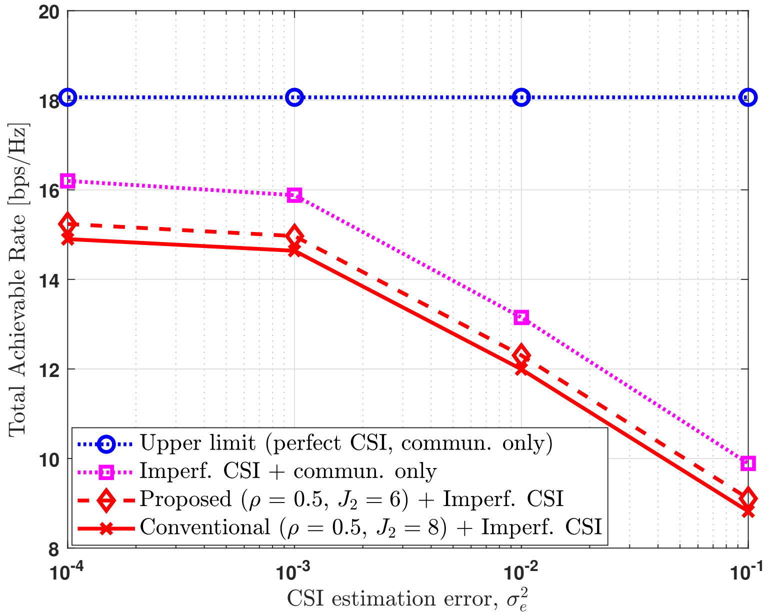

Figure 5.20: Total achievable data rate vs. CSI estimation error (σ_e^2) for the proposed and conventional ISAC schemes, as well as the communication-only design, when $P_{\mathrm{V}} = 10$ dBW.

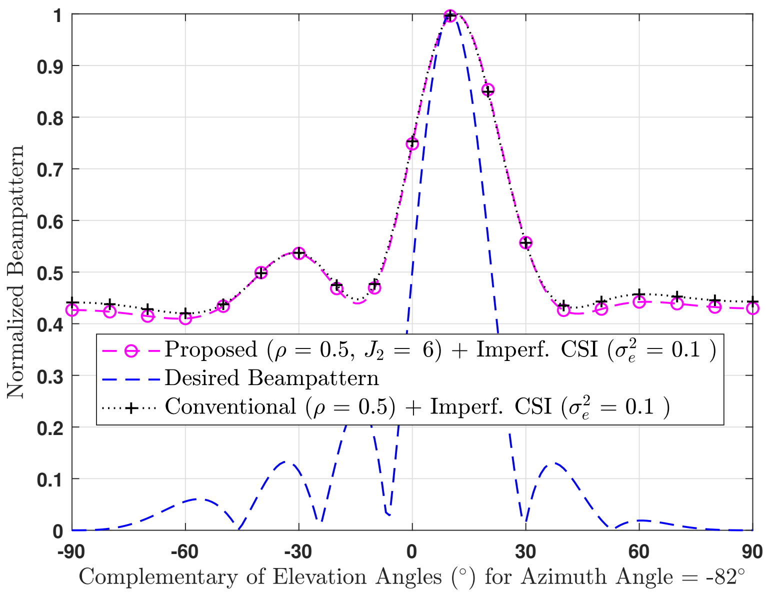

Figure 5.21: 2D beampatterns from proposed and conventional ISAC schemes under imperfect CSI.

the MADRL-ISAC approach is approximately 40% higher than that of the *distributed ISAC optimization*-based approach. In the latter, each VSAT selects sensing subcarriers from its own communication subcarrier set to avoid collisions. We allocate 50% of each VSAT's communication subcarriers for sensing to ensure minimal degradation of the communication link. In terms of sensing performance, the *distributed ISAC optimization*-based approach significantly outperforms MADRL-ISAC, achieving approximately 82% performance improvement. However, this improvement is largely attributed to higher transmit power. Specifically, the MADRL-ISAC model used a sum power level of 15.073 dBW during testing episodes, whereas the *distributed ISAC optimization*-based approach used the maximum available power budget of 18.77 dBW (see Fig. 5.19).

Power Consumption

Fig. 5.19 portrays the sum power used by the MADRL-ISAC model during both training and testing episodes. In the early training episodes, a high sum power of 16.06 dBW is consumed. However, as the model learns in the later stages of training, power consumption decreases to a comparatively lower level of 14.97 dBW, which represents a reduction of approximately 22% (in linear scale). During the test phase, the MADRL-ISAC model utilized a total VSAT transmit power of 15.073 dBW, with VSAT agent 1 consuming 8 dBW, VSAT agent 2 consuming 12 dBW, and VSAT agent 3 consuming 10 dBW. The variation in power levels among VSAT agents resulted from their individual training outcomes based on the reward function in (5.25). Each agent made independent decisions without collaboration, guided by its learned policy, to achieve better joint communication–sensing performance while using an appropriate power level compared to the distributed ISAC optimization-based approach, which required a total VSAT power budget of 18.77 dBW, where each VSAT utilized the maximum available power of 14 dBW. This reduction of approximately 20% in VSAT transmit power demonstrates the efficiency of the MADRL-ISAC approach, which not only conserves energy in GTs but also supports scalability and sustainability for large-scale NTNs.

5.7.4 Impact of Imperfect CSI

Based on the analysis in Section 5.5, this section presents simulation results that evaluate the impact of imperfect CSI on the achievable average SE and the 2D sensing beampattern, for both the communication-only and ISAC scenarios.

Fig. 5.20 depicts the impact of CSI estimation error variance (σ_e^2) on the average SE of both communication-only and various ISAC designs. For $\sigma_e^2 > 10^{-3}$, a significant degradation in SE is observed across all schemes. As expected, the proposed ISAC design performs slightly worse than the communication-only scenario under imperfect CSI, with additional degradation as the number of sensing subcarriers increases due to the complexity of joint communication and sensing. Among the ISAC schemes, the conventional design exhibits the largest performance loss. Despite this SE degradation, Fig. 5.21 shows that the sensing performance remains nearly identical to the perfect CSI case in Fig. 5.11, showing negligible variation in the main lobe's HPBW, accurate beam pointing toward the target (approaching LEO), and comparable sidelobe suppression.

5.8 Summary

In this chapter, an ISAC design for 3D beamforming was proposed to support ISAC-HOs in LEO satellite networks. It demonstrated flexibility in subcarrier selection, ensuring accurate sensing of approaching LEO satellite while utilizing all available subcarriers to maximize communication performance with the serving LEO satellite. Simulation results validated that reliable sensing beampatterns could be achieved with a reduced number of subcarriers, while communication performance was significantly improved compared to baseline approaches, provided that HOs are frequent.

Moreover, the MADRL-ISAC design was proposed to enable ISAC in a multi-GT environment, where multiple GTs concurrently request ISAC services. Simulation results demonstrated that the proposed design outperforms the distributed ISAC approach by achieving approximately 40% higher sum SE, while consuming 20% less total power and satisfying sensing QoS requirements without any collision incidents during testing. Furthermore, the robustness of the proposed ISAC design was demonstrated under imperfect CSI.

Chapter 6

Conclusions and Future Work

This chapter presents the main conclusions of the thesis and suggests potential directions for future research.

6.1 Main conclusions

In recent years, there has been a massive deployment of NGSO satellites, mostly driven by the goal of providing global direct-to-cell and broadband services, with a particular focus on serving remote and challenging geographical terrains, such as isolated islands, open ocean areas, and mountainous regions, as well as supporting rescue and planning operations in disaster-prone or conflict-prone zones. Unlike airborne NTNs, NGSO satellites experience minimal atmospheric drag, and compared to GEO systems, they provide lower propagation latency and reduced path loss due to their closer proximity to Earth. However, NGSO satellites face several challenges, such as brief serving durations per orbital pass at the specific location, frequent HO requirements for service continuity, and so on, due to their inherent characteristics, such as high orbital velocity, non-geostationary nature, and relatively smaller FoV. To support the IMT-2030 vision of ubiquitous connectivity and meet its KPIs for 6G networks, this thesis has focused on LEO satellite networks, a critical subset of spaceborne NTNs that offer superior propagation latency and path loss compared to MEO and GEO satellites. The work addresses key challenges associated with LEO systems while aiming to meet IMT-2030's performance targets by leveraging emerging 6G technologies, including ISAC and AI. In addition, a mechanism for evaluating the near-real-time KPIs of a deployed system has been proposed. The main contributions of this work are summarized as follows:

- **Chapter 2** developed a framework to automatically evaluate near real-time KPIs in terms of the received data rate. This framework leverages modern software tools and was validated using PCAP files containing UDP/IP streams generated from SES's DVB-NIP system, for both unicast and multicast scenarios. It can also directly operate on live UDP/IP streams. The KPI results generated by this tool closely matched those obtained through manual packet analysis tools such as Wireshark, confirming its accuracy and reliability. This framework significantly facilitates the debugging process in near real-time and streamlines the R&D process.

- **Chapter 3** proposed the FLARE-LEO framework to address RRM in LEO satellites by leveraging their advanced fully regenerative payload capabilities. This framework jointly designs spot beam coverage, adaptive beamforming, caching, multicasting, multiuser precoding, and dynamic bandwidth/power allocation, which significantly enhances user-centric data rates, reducing end-to-end downlink content delivery latency. To address the limited serving time of LEO satellites, novel HO strategies were devised with low communication and signaling overhead. These strategies utilize ISLs and operate independently of GWs during live transmission, ensuring seamless integration with the proposed framework. The key innovations include adaptive beamforming and an optimal precoding technique that outperforms conventional methods, along with collaborative HO mechanisms that improve the effective data rate compared to non-HO scenarios. Moreover, the robustness of the proposed framework was validated under imperfect CSI. Simulation results demonstrated that the proposed framework can deliver a total of 45 Gb to 45 users within a LEO footprint of 578 km radius in 20 seconds, using a 100 MHz bandwidth, under a non-HO scenario and assuming perfect CSI with all requested content pre-cached on-board. This corresponds to a remarkable system-level SE of 22.5 bps/Hz, closely approaching the projected IMT-2030 6G benchmarks.

- **Chapter 4** presented a conceptual three-tier SaaS framework to support on-demand critical EO services such as disaster forecasting, 2D/3D scene observation, route finding, and rescue operations using satellite imagery/video data. By distributing tasks partially and in parallel across SAGIN nodes, the framework significantly reduced the worst-case service completion time through optimal utilization of computational and communication resources, outperforming baseline approaches. The design incorporated core task offloading parameter tuple, such as EO data size, output size, computational requirements, and maximum tolerance time for service delivery. Moreover, the chapter outlined methods for delivering task results during HOs across different Earth beam configuration scenarios and demonstrated the robustness of the proposed framework under imperfect CSI.

- **Chapter 5** tackled the challenge of break-before-make HO scenario in LEO satellite networks, by introducing an intelligent make-before-break ISAC-HO scheme to support the IMT-2030 vision of seamless connectivity. This scheme utilizes flexible subcarrier allocation for joint communication and sensing, allowing ISAC-assisted GT to perform 3D beamforming toward an approaching LEO satellite for sensing, without significantly degrading communication with the currently serving LEO. To ensure the scalability of the proposed scheme, it was further extended to support multiple GTs using a MADRL framework, enabling efficient operation in dynamic multi-user environments. The chapter also demonstrated the robustness of the proposed HO scheme under imperfect CSI.

6.2 Future Works

This dissertation addressed some of the fundamental challenges in LEO satellite systems, including path loss, propagation delays, channel estimation, and short serving periods. It did this through a mix of theoretical frameworks, algorithm development, and simulation-based evaluations. The research explored several advanced techniques, such as caching, task offloading, multicasting, adaptive beamforming, MIMO, precoding design, ISAC, DL, and MADRL, to enhance the performance of the LEO satellite system. Moreover, key optimization approaches such as SCA, BnB, and Riemannian manifold optimization were also applied to efficiently exploit the resources of the LEO satellite system. To evaluate the performance of a deployed system (DVB-NIP), an automated tool was developed to assess the mean data rate in near real-time. The results in this thesis show that the suggested architectures and resource management strategies work well. They also highlight opportunities for future studies to build upon these contributions. The next section outlines several potential directions for future research.

- **Full-Fledged Automated DVB-NIP Packet Analyzer Tool Development:** In Chapter 2, an automated framework is developed to evaluate the DVB-NIP system in real-time, which currently supports only the assessment of the received mean data rate. To extend this work into a full-fledged automated packet analyzer tool, it is essential to incorporate features that can automatically display a comprehensive list of announced and received files, transmission characteristics, carousel cycles, streams, periods, live stream latency, multicast transport sessions, bootstrap streams, signaling files, the validity of parameters in the manifests, and the achieved data rate. This tool should offer more advanced monitoring capabilities than those currently provided by Wireshark, including automated visualizations and alerts, in near real-time, accessible via a dynamic web application or Android software.

- **Efficient On-board Cache Content Placement and Delivery Strategy in Mega Constellations of LEO Satellite Networks:** In Chapter 3, only the cache delivery strategy from a fully regenerative payload-enabled LEO satellite was considered. A possible future extension of this work could be the joint optimization of cache content placement and delivery strategies, leveraging recent advancements in AI and novel optimization techniques. This approach would account for the dynamic topology of LEO constellations, time-varying user demand (in terms of both data volume and content type), and the on-board computational and radio resource constraints inherent in mega constellations of LEO satellite networks. The ultimate goal would be to maintain content freshness dynamically in response to evolving user requirements.

- **Interplay of AI, Blockchain/Quantum, Optimization Techniques, and Open-Source Container Orchestrators for Resource Management in SAGINs to Enable a SaaS Platform for On-Demand Task Computation and Delivery:** In Chapter 4, computational resources across SAGIN were allocated to perform on-demand EO-based tasks on-board and to optimize downlink user-centric delivery bandwidth while returning task results. A possible future extension of this work could involve the dynamic allocation of radio resources such as power, bandwidth, and coverage areas, as discussed in Chapter 3, as well as computational resources across SAGIN nodes, using AI and novel optimization techniques. This would also include the optimization of ISL and feeder links, taking into account backlogs of uncompleted tasks. Furthermore, emerging technologies such as blockchain or quantum computing could be explored to enhance EO data protection and privacy. The practical validation of the proposed framework could be undertaken using a testbed built on an open-source container orchestration platform, such as Kubernetes, capable of both horizontal and vertical scaling.

- **Communication-Efficient Federated Learning for LEO Constellations to Train EO Imagery:** As discussed in Chapter 4, the practical benefits of a SaaS platform can be realized through EO-based services, including scene observation, route planning, and object detection. To support real-time inference for these services, AI model training must be conducted on-board satellites while taking into account the communication, computation, and non-geostationary constraints of LEO constellations. In this context, one of the promising research directions is on-board federated learning, which enables heterogeneous devices to collaboratively train AI models without sharing raw data, thereby preserving data privacy and reducing communication overhead.

- **AI and Reconfigurable Intelligent Surfaces (RIS)-Assisted ISAC for Intelligent HO Design in Mega-LEO Constellations:** In Chapter 5, the intelligent ISAC-HO design was presented for the single-target case. To build upon this work, future research should investigate multi-target tracking and ISAC beamforming optimization, enabling GTs to simultaneously track multiple LEO satellites while minimizing mutual interference between communication and sensing functions. The integration of RIS could enhance weak echo signals from distant LEO targets, thereby improving sensing reliability. Additionally, more scalable learning-based approaches should be investigated to reduce the computational complexity of training the MADRL-ISAC model, especially in large-scale network settings.

Appendix A

A.1 Proof of Proposition 3.1 in Chapter 3

Consider a function $\mathrm{g}(u, v)$ in $\mathbb{R}_+^2$. The Hessian of $\mathrm{g}(u, v) = u \log(1 + v/u)$ is given as follows:

$$\mathcal{H}_\mathrm{g} = \begin{bmatrix} \dfrac{-v^2}{u(u+v)^2} & \dfrac{v}{(u+v)^2} \\ \dfrac{v}{(u+v)^2} & -\dfrac{u}{(u+v)^2} \end{bmatrix} \tag{A.0.1}$$

For arbitrary vector $\mathbf{x} = [p\ q]^T$, we can calculate $\mathbf{x}^T \mathcal{H}_\mathrm{g} \mathbf{x} = -\frac{(pv-qu)^2}{u^2(u+v)^2}$, which is always non-positive. This implies that the function $u \log(1 + v/u)$ is concave in its supports. From (3.29), we can write $R_{1,k,a,m}^{DC,ZF} = \Phi \mathrm{g}(\hat{I}_{agg}^{HO} b_{1,a,m}, x_{1,k,a,m}) / \hat{I}_{agg}^{HO}$, which completes the proof of Proposition 3.1.

Appendix B

B.1 Signal and Interference Power Analysis for Imperfect CSI (for Chapter 5)

B.1.1 Desired Signal Power

Following (5.13), the desired signal power $\tilde{P}_{\mathrm{D}}$ is derived as:

$$\begin{aligned}\tilde{P}_{\mathrm{D}} &= \mathbb{E}_s\{||\sqrt{\frac{P_{\mathrm{V}}}{N_s}}\hat{\boldsymbol{U}}_{1,k}^H\hat{\boldsymbol{U}}_{1,k}\hat{\boldsymbol{\Lambda}}_{1,k}\hat{\boldsymbol{V}}_{1,k}^H\hat{\boldsymbol{V}}_{1,k}\boldsymbol{s}_{1,k}||^2\} \\ &= (P_{\mathrm{V}}/N_s)\cdot\mathrm{tr}(\mathbb{E}_s\{(\hat{\boldsymbol{\Lambda}}_{1,k}\boldsymbol{s}_{1,k})\times(\hat{\boldsymbol{\Lambda}}_{1,k}\boldsymbol{s}_{1,k})^H\}) \\ &= (P_{\mathrm{V}}/N_s)\cdot\mathrm{tr}(\hat{\boldsymbol{\Lambda}}_{1,k}\mathbb{E}_s\{\boldsymbol{s}_{1,k}\boldsymbol{s}_{1,k}^H\}\hat{\boldsymbol{\Lambda}}_{1,k}^H\}) \\ &= (P_{\mathrm{V}}/N_s)\cdot\mathrm{tr}(\hat{\boldsymbol{\lambda}}_{1,k}\odot\hat{\boldsymbol{\lambda}}_{1,k})\end{aligned} \tag{B.0.1}$$

where, $\hat{\boldsymbol{\lambda}}_{1,k} \triangleq \{\hat{\lambda}_{1,k}^i\}_{i=1}^{N_s} = \mathrm{diag}(\hat{\boldsymbol{\Lambda}}_{1,k})$ denotes the singular values of $\hat{\boldsymbol{H}}_{1,k}$, which are sorted in descending order, i.e., $\hat{\lambda}_{1,k}^1 \geq \hat{\lambda}_{1,k}^2 \geq \ldots \geq \hat{\lambda}_{1,k}^{N_s}$.

B.1.2 Interfering Signal Power

The interfering signal power $\tilde{P}_{\mathrm{I}}$ is derived as:

$$\begin{aligned}
\tilde{P}_{\mathrm{I}} &= \mathbb{E}_{e,s}\{||\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{V}}_{1,k}\mathbf{s}_{1,k}\sqrt{P_{\mathrm{V}}/N_s}||^2\} \\
&= P_{\mathrm{V}}/N_s \cdot \mathrm{tr}(\mathbb{E}_{e,s}\{(\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{V}}_{1,k}\mathbf{s}_{1,k})(\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{V}}_{1,k}\mathbf{s}_{1,k})^{H}\}) \\
&= P_{\mathrm{V}}/N_s \cdot \mathrm{tr}(\mathbb{E}_{e}\{\hat{\boldsymbol{U}}_{1,k}^{H}\boldsymbol{E}\hat{\boldsymbol{V}}_{1,k}\mathbb{E}_{s}\{\mathbf{s}_{1,k}\mathbf{s}_{1,k}^{H}\}\hat{\boldsymbol{V}}_{1,k}^{H}\boldsymbol{E}^{H}\hat{\boldsymbol{U}}_{1,k}\}) \\
&= P_{\mathrm{V}}/N_s \cdot \mathrm{tr}(\hat{\boldsymbol{U}}_{1,k}^{H}\mathbb{E}_{e}\{\boldsymbol{E}\boldsymbol{E}^{H}\}\hat{\boldsymbol{U}}_{1,k}) \\
&= \sigma_e^2 P_{\mathrm{V}}/N_s \cdot \mathrm{tr}(\hat{\boldsymbol{U}}_{1,k}^{H}\hat{\boldsymbol{U}}_{1,k}) \\
&= \sigma_e^2 P_{\mathrm{V}}
\end{aligned} \tag{B.0.2}$$

Bibliography

[1] SpaceWatch.GLOBAL. *Apple Launches emergency SOS via satellite on iPhone-14 and iphone-14-pro*. Accessed: Sept 30, 2024. [Online]. Available: https://spacewatch.global/2022/11/apple-launches-emergency-sos-via-satellite-on-iphone-14-and-iphone-14-pro/

[2] M. C. Weinzierl, K. Lucas, and M. Sarang, "SpaceX, Economies of Scale, and a Revolution in Space Access," *Harvard Business School Case 720-027*, 2021.

[3] Internet Society. *Perspectives on LEO satellites - using low Earth orbit Satellites for Internet Access*. Accessed: Sept 30, 2024. [Online]. Available: https://www.internetsociety.org/wp-content/uploads/2022/11/Perspectives-on-LEO-Satellites.pdf

[4] Starlink. *SECOND GENERATION STARLINK SATELLITES*. Accessed: Aug 8, 2025. [Online]. Available: https://www.starlink.com/public-files/Gen2StarlinkSatellites.pdf?srsltid=AfmBOorOGGAhGOsBK1WcFjNDVI8_IyHOMhaBeRyNBSvyMMc0scX1XtQL

[5] ——. *Starlink Business: Direct To Cell*. Accessed: Aug 8, 2025. [Online]. Available: https://www.starlink.com/business/direct-to-cell?srsltid=AfmBOopQ7avDh9pwPJGXo8fNgrG5Yr8rvCRWD7LYTR_M90jhSWBZucjT

[6] *Beam-hopping JoeySat launched*. Accessed: Jul. 01, 2023. [Online]. Available: https://www.esa.int/Applications/Connectivity_and_Secure_Communications/Beam-hopping_JoeySat_launched

[7] ESA. *Galileo Second Generation developing at full speed*. Accessed: Aug 8, 2025. [Online]. Available: https://www.esa.int/Applications/Satellite_navigation/Galileo/Galileo_Second_Generation_developing_at_full_speed

[8] S.-Y. Kang, M.-S. Jo, J.-Y. Choi, and S. S. Yang, "Cost Effectiveness of Reusable Launch Vehicles Depending on the Payload Capacity," *Aerospace*, vol. 12, no. 5, p. 364, 2025.

[9] *Solutions for NR to support non-terrestrial networks (NTN)*, document Technol. Report (TR) 38.821, Rel. 16, 3rd Gener. Partnership Project (3GPP), 2019.

[10] ITU Radiocommunication Sector, *Framework and overall objectives of the future development of IMT for 2030 and beyond*, International Telecommunication Union, Recommendation ITUR M.2160-0, 2023. [Online]. Available: https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2160-0-202311-I%21%21PDF-E.pdf.

[11] D. Wen, Y. Zhou, X. Li, Y. Shi, K. Huang, and K. B. Letaief, "A Survey on Integrated Sensing, Communication, and Computation," *IEEE Communications Surveys Tutorials*, 2024, doi:10.1109/COMST.2024.3521498.

[12] C. T. Nguyen *et al.*, "Emerging Technologies for 6G Non-Terrestrial-Networks: From Academia to Industrial Applications," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 3852–3885, 2024.

[13] *History*. Accessed: Aug. 03, 2025. [Online]. Available: https://dvb.org/about/history/

[14] O. B. Yahia *et al.*, "Evolution of High-Throughput Satellite Systems: A Vision of Programmable Regenerative Payload," *IEEE Communications Surveys Tutorials*, vol. 27, pp. 1565–1597, 2025.

[15] *Digital Transparent Processor (DTP)*. Accessed: Aug. 03, 2025. [Online]. Available: https://connectivity.esa.int/projects/digital-transparent-processor-dtp

[16] *DVB-S/S2/S2X*. Accessed: Aug. 03, 2025. [Online]. Available: https://www.etsi.org/technologies/dvb-s-s2

[17] *5G Non-Terrestrial Networks in 3GPP Rel-19 - Ericsson*. Accessed: Aug. 03, 2025. [Online]. Available: https://www.ericsson.com/en/blog/2024/10/ntn-payload-architecture

[18] S. S. S. G. Seeram, L. Feltrin, M. Ozger, S. Zhang, and C. Cavdar, "Feasibility Study of Function Splits in RAN Architectures with LEO Satellites," in *2024 Joint European Conference on Networks and Communications 6G Summit (EuCNC/6G Summit)*, Antwerp, Belgium, 2024, pp. 622–627.

[19] SES. *Showcasing Satellite Innovation with a New IP Broadcast Standard*. Accessed: Jan. 03, 2025. [Online]. Available: https://www.ses.com/blog/showcasing-satellite-innovation-new-ip-broadcast-standard

[20] TELESAT. *Real-Time Latency: Rethinking Remote Networks*. Accessed: Sept 30, 2024. [Online]. Available: https://www.telesat.com/wp-content/uploads/2022/11/Real-Time-Latency.pdf

[21] A. K. Meshram, S. Kumar, J. Querol, S. Andrenacci, and S. Chatzinotas, "Reduced Complexity Initial Synchronization for 5G NR Multibeam LEO-Based Non-Terrestrial Networks," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 1528–1551, 2025.

[22] *STARLINK-4798/2022-114AS Low Earth Orbit (LEO)*. Accessed: Sept 30, 2024. [Online]. Available: https://orbit.ing-now.com/satellite/53858/2022-114as/starlink-4798/

[23] Aerospace. *Popular Orbits 101*. Accessed: Sept 30, 2024. [Online]. Available: https://aerospace.csis.org/aerospace101/earth-orbit-101/

[24] *5G; NR; User Equipment (UE) radio transmission and reception; Part 5: Satellite access Radio Frequency (RF) and performance requirements*, document Tech. Specification (TS) 38.101-5, Ver. 18.5.0, Rel. 18, 3rd Gener. Partnership Project (3GPP), 2024.

[25] Starlink. *STARLINK DIRECT TO CELL SERVICE NOW AVAILABLE*. Accessed: June 27, 2025. [Online]. Available: https://www.starlink.com/public-files/DIRECT_TO_CELL_SERVICE_FEB_25.pdf?srsltid=AfmBOoqr96_x0etl8ATIHblhTdE8wB6gkzS2-rpM80W1ofhgTPUZByuu

[26] *Digital Video Broadcasting (DVB) and Native IP Broadcasting*, document Technol. Spec. (TS) 103 876, Version 1.1.1, European Telecommunications Standards Institute (ETSI), 2024.

[27] *NR and NG-RAN Overall description*, document Tech. Specification (TS) 38.3-5, Ver. 17.2.0, Rel. 17, 3rd Gener. Partnership Project (3GPP), 2022.

[28] C. Daehnick, I. Klinghoffer, B. Maritz, and B. Wiseman, "Large LEO satellite constellations: Will it be different this time?" *McKinsey Company Technical Report (Aerospace Defese Practice)*, May 2020.

[29] Y. Chen, S. Zhang, M. Xiao, Z. Qian, J. Wu, and S. Lu, "Multi-user Edge-assisted Video Analytics Task Offloading Game based on Deep Reinforcement Learning," in *Proc. IEEE 26th Int. Conf. Parallel and Distributed Syst.*, Hong Kong, 2020, pp. 266–273.

[30] H. Ha and S. Song, "Semantic Abstraction: Open-World 3D Scene Understanding from 2D Vision-Language Models," in *Proc. 6th Annual Conf. Robot Learn.*, Cornell University, Ithaca, 2022.

[31] *Satellite-as-a-service: The future of satellite network communications: Eutelsat.* Accessed: Jan. 29, 2024. [Online]. Available: https://www.eutelsat.com/en/blog/satellite-as-a-service-the-future-of-satellite-network-communications.html

[32] *EO Software Applications.* Accessed: Dec. 03, 2024. [Online]. Available: https://eop-cfi.esa.int/index.php/applications

[33] *Satellite Imagery for Emergency Management.* Accessed: Dec. 03, 2024. [Online]. Available: https://www.euspaceimaging.com/blog/2024/07/01/satellite-imagery-for-emergency-management/

[34] S. Bhandari, N. Ranjan, P. Khan, H. Kim, and Y.-S. Hong, "Deep Learning-Based Content Caching in the Fog Access Points," *Electronics*, vol. 10, no. 4, p. 512, 2021.

[35] S. Bhandari, N. Ranjan, Y.-C. Kim, and H. Kim, "Deep Reinforcement Learning for Dynamic Spectrum Access in the Multi-Channel Wireless Local Area Networks," in *Proc. IEEE Int. Conf. Electron., Info., and Commun. (ICEIC)*, Jeju, Korea, Republic of, 2022, pp. 1–4.

[36] Y. S. Nasir and D. Guo, "Multi-Agent Deep Reinforcement Learning for Dynamic Power Allocation in Wireless Networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2239–2250, Oct. 2019.

[37] J. Clausen, "Branch and bound algorithms-principles and examples," *Department of Computer Science, University of Copenhagen*, pp. 1–30, 1999.

[38] Y. Yang, M. Pesavento, S. Chatzinotas, and B. Ottersten, "Successive Convex Approximation Algorithms for Sparse Signal Estimation With Nonconvex Regularizations," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 6, pp. 1286–1302, Dec. 2018.

[39] F. Liu, C. Masouros, A. Li, H. Sun, and L. Hanzo, "MU-MIMO communications with MIMO radar: From co-existence to joint transmission," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2755–2770, 2018.

[40] *Using mosek with CVX*. Accessed: Jan. 26, 2024. [Online]. Available: http://cvxr.com/cvx/doc/mosek.html

[41] Ericsson. *Traffic by Application – Mobility Report*. Accessed: Jan. 03, 2025. [Online]. Available: https://www.ericsson.com/4adb7e/assets/local/reports-papers/mobility-report/documents/2024/ericsson-mobility-report-november-2024.pdf

[42] S. Bhandari, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Efficient Content Caching for Delivery Time Minimization in the LEO Satellite Networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Rome, Italy, 2023, pp. 1246–1252.

[43] Eutelsat Group. *DVB Native IP: The Gamechanger for Content Distribution*. 2023 White Paper, Accessed: Jan. 03, 2025. [Online]. Available: https://www.idirect.net/wp-content/uploads/2023/09/Whitepaper-DVB-Native-IP-1.pdf

[44] Internet Society. *Perspectives on LEO Satellites - Using Low Earth Orbit Satellites for Internet Access*. Accessed: Jan. 03, 2025. [Online]. Available: https://www.internetsociety.org/wp-content/uploads/2022/11/Perspectives-on-LEO-Satellites.pdf

[45] Wireshark. *The World's Most Popular Network Protocol Analyzer*. Accessed: Jan. 07, 2025. [Online]. Available: https://www.wireshark.org/

[46] *Generic Token Bucket Algorithm*, document Technol. Spec. MEF 41, The Metro Ethernet Forum (MEF), 2013.

[47] GitHub. *Scapy: The Python-Based Interactive Packet Manipulation Program & Library*. Accessed: Jan. 07, 2025. [Online]. Available: https://github.com/secdev/scapy

[48] ——. *FLUTE - File Delivery over Unidirectional Transport*. Accessed: Jan. 07, 2025. [Online]. Available: https://github.com/ypo/flute

[49] Django Project. *Django documentation: Django documentation.* Accessed: Jan. 07, 2025. [Online]. Available: https://docs.djangoproject.com/

[50] S. Bhandari *et al.*, "An Automatic Data Completeness Check Framework for Open Government Data," *Appl. Sci.*, vol. 11, no. 19, p. 9270, 2021.

[51] I. Leyva-Mayorga *et al.*, "LEO Small-Satellite Constellations for 5G and Beyond-5G Communications," *IEEE Access*, vol. 8, pp. 184 955–184 964, 2020.

[52] S. Liu *et al.*, "LEO Satellite Constellations for 5G and Beyond: How Will They Reshape Vertical domains?" *IEEE Commun. Mag.*, vol. 59, no. 7, pp. 30–36, Jul. 2021.

[53] O. Kodheli *et al.*, "Satellite Communications in the New Space Era: A Survey and Future Challenges," *IEEE Commun. Surveys and Tuts.*, vol. 23, no. 1, pp. 70–109, 1st Quart., 2021.

[54] T. X. Vu, N. Maturo, S. Chatzinotas, J. Grotz, T. Christophory, and B. Ottersten, "Dynamic Bandwidth Allocation and Edge Caching Optimization for Nonlinear Content Delivery through Flexible Multibeam Satellites," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Seoul, South Korea, May 2022, pp. 1143–1148.

[55] G. Thomas, "Enabling Technologies for flexible HTS payloads," in *Proc. 33rd AIAA Int. Commun. Satell. Syst. Conf. Exhib.*, Queeensland, Australia, Sept. 2015, pp. 2015–4348.

[56] N. Mazzali, M. R. B. Shankar, and B. Ottersten, "On-board signal predistortion for digital transparent satellites," in *Proc. IEEE 16th Int. Workshop Signal Process. Adv. in Wireless Commun. (SPAWC)*, Stockholm, Sweden, Jun. 2015, pp. 535–539.

[57] G. Shacham, "On board processing payload," in *Proc. 24th KA Broadband Commun. Conf.*, Niagara Falls, ON, Canada, 2018, pp. 1–6.

[58] T. M. Braun and W. R. Braun, *PROCESSING PAYLOAD AND FLEXIBLE PAYLOAD.* Hoboken, NJ, USA: Wiley, 2021, pp. 270–296.

[59] T. S. Abdu, E. L. S. Kisseleff, J. Grotz, S. Chatzinotas, and B. Ottersten, "Demand-Aware Onboard Payload Processor Management for High Throughput NGSO Satellite

Systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 5, pp. 4883–4899, Oct. 2023.

[60] T. X. Vu, Y. Poirier, S. Chatzinotas, N. Maturo, J. Grotz, and F. Roelens, "Modeling and Implementation of 5G Edge Caching over Satellite," *Int. J. Satell. Commun. Netw.*, vol. 38, no. 5, pp. 395–406, Sep. 2020.

[61] M. Luglio, S. P. Romano, C. Roseti, and F. Zampognaro, "Satellite multi-beam multicast support for an efficient community-based CDN," *Comput. Netw.*, vol. 217, p. 109352, Nov., 2022.

[62] P. J. Honnaiah, N. Maturo, S. Chatzinotas, and J. Krause, "Demand-Based Adaptive Multi-Beam Pattern and Footprint Planning for High Throughput GEO Satellite Systems," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 1526–1540, 2021.

[63] Q. T. Ngo, K. T. Phan, W. Xiang, A. Mahmood, and J. Slay, "Two-Tier Cache-Aided Full-Duplex Hybrid Satellite–Terrestrial Communication Networks," *IEEE Trans. Aerosp. Elecron. Syst.*, vol. 58, no. 3, pp. 1753–1765, Jun. 2022.

[64] X. Zhu, C. Jiang, L. Kuang, and Z. Zhao, "Cooperative Multilayer Edge Caching in Integrated Satellite-Terrestrial Networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 5, pp. 2924–2937, May 2022.

[65] K. An, Y. Li, X. Yan, and T. Liang, "On the Performance of Cache-Enabled Hybrid Satellite-Terrestrial Relay Networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1506–1509, Oct. 2019.

[66] D. Han, W. Liao, H. Peng, H. Wu, W. Wu, and X. Shen, "Joint Cache Placement and Cooperative Multicast Beamforming in Integrated Satellite-Terrestrial Networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 3131–3143, Mar. 2022.

[67] T. X. Vu, S. Chatzinotas, and B. Ottersten, "Dynamic Bandwidth Allocation and Precoding Design for Highly-Loaded Multiuser MISO in Beyond 5G Networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 3, pp. 1794–1805, Mar. 2022.

[68] A. Guidotti and A. Vanelli-Coralli, "Clustering strategies for multicast precoding in Multibeam Satellite Systems," *Int. J. Satell. Commun. Netw.*, vol. 38, no. 2, pp. 85–104, Mar. 2020.

[69] A. Guidotti, A. Vanelli-Coralli, and C. Amatetti, "Federated Cell-Free MIMO in Non-Terrestrial Networks: Architectures and Performance," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 60, no. 3, pp. 3319–3347, Jun. 2024.

[70] *Movielens 1M Dataset*. Accessed: May. 01, 2023. [Online]. Available: https://grouplens.org/datasets/movielens/1m/

[71] *SES-14 Satellite*. Accessed: Jul. 01, 2023. [Online]. Available: https://www.ses.com/our-coverage#/explore/satellite/369

[72] *Electronic Warfare and Radar Systems Engineering Handbook*, 4th ed., Avionics Dept. NAWCWD, Nav. Air Warfare Center Weapons Div., St. Point Mugu Naws, CA, USA, 2013.

[73] *5G NR: Multiplexing and Channel Coding*, document Tech. Spec. (TS) 138.211, Version 16.2.0, 3rd Gener. Partnership Project (3GPP), 2020.

[74] T. X. Vu, S. Chatzinotas, and B. Ottersten, "Edge-Caching Wireless Networks: Performance Analysis and Optimization," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2827–2839, Apr. 2018.

[75] K. X. Li *et al.*, "Downlink Transmit Design for Massive MIMO LEO Satellite Communications," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 1014–1028, Feb. 2022.

[76] D. Bharadia and S. Katti, "FastForward: Fast and constructive full duplex relays," in *Proc. ACM Conf. SIGCOMM*, Chicago, IL, USA, Aug. 2014, pp. 199–210.

[77] D. Arthur and S. Vassilvitskii, "K-means++: the advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, 2007, pp. 127–135.

[78] R. Apu and M. Gavrilova, *Generalized Voronoi Diagram: A Geometry- Based Approach to Computational Intelligence*. Berlin, Germany: Springer, 2009, vol. 158.

[79] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[80] M. K. Pakhira, "A Linear Time-Complexity k-Means Algorithm Using Cluster Shifting," in *Int. Conf. Compt. Intell. Commun. Netw.*, Bhopal, India, Nov. 2014, pp. 1047–1051.

[81] S. Loffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Int. Conf. Mach. Learn.*, Lille, France, 2015, pp. 448–456.

[82] H. Wu and X. Gu, "Towards dropout training for convolutional neural networks," *Neural Netw.*, vol. 71, pp. 1–10, Nov. 2015.

[83] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022.

[84] S. Ruder, "An overview of gradient descent optimization algorithms," 2016, *arXiv:1609.04747*.

[85] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. on Learn. Represent.*, San Diego, CA, USA, 2015, pp. 1–15.

[86] N. Ranjan, S. Bhandari, P. Khan, Y.-S. Hong, and H. Kim, "Large-scale road network congestion pattern analysis and prediction using deep convolutional autoencoder," *Sustainability*, vol. 13, no. 9, p. 5108, 2021.

[87] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 5353–5360.

[88] M. Y. Abdelsadek, G. K. Kurt, and H. Yanikomeroglu, "Distributed Massive MIMO for LEO Satellite Networks," *IEEE Open J. Commun. Soc*, vol. 3, pp. 2162–2177, Nov. 2022.

[89] M. C. M. S. F. Riera-Palou, G. Femenias and A. I. Pérez-Neira, "Scalable Cell-Free Massive MIMO Networks With LEO Satellite Support," *IEEE Access*, vol. 10, pp. 37 557–37 571, Apr. 2022.

[90] J. A. Nanzer, S. R. Mghabghab, S. M. Ellison, and A. Schlegel, "Distributed Phased Arrays: Challenges and Recent Advances," *IEEE Trans. Microw. Theory and Techn.*, vol. 69, no. 11, pp. 4893–4907, Nov. 2021.

[91] D. Tuzi, T. Delamotte, and A. Knopp, "Satellite Swarm-Based Antenna Arrays for 6G Direct-to-Cell Connectivity," *IEEE Access*, vol. 11, pp. 36 907–36 928, Mar. 2023.

[92] N2YO. *Technical details for satellite starlink-4798*. Accessed: Jul. 01, 2023. [Online]. Available: https://www.n2yo.com/satellite/?s=53858

[93] S. Cakaj, "The parameters comparison of the 'starlink' leo satellites constellation for different orbital shells," *Frontiers Commun. and Netw.*, vol. 2, May 2021.

[94] H. W. Lee, S. Shimizu, S. Yoshikawa, and K. Ho, "Satellite Constellation Pattern Optimization for Complex Regional Coverage," *J. Spacecraft and Rockets*, vol. 57, no. 6, pp. 1309–1327, 2020.

[95] W. C. Jakes and D. C. Cox, *Microwave mobile communications.* Hoboken, NJ, USA: Wiley, 1994.

[96] *LTE in a nutshell: The physical layer*, Telesystem Innov., Northwood, OH, USA, 2010.

[97] T. X. Vu *et al.*, "Machine Learning-Enabled Joint Antenna Selection and Precoding Design: From Offline Complexity to Online Performance," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 3710–3722, Jun. 2021.

[98] A. Islam, A. Debnath, M. Ghose, and S. Chakraborty, "A Survey on Task Offloading in Multi-access Edge Computing," *J. Syst. Architecture*, vol. 118, p. 102225, 2021.

[99] J. Wang, J. Hu, G. Min, A. Y. Zomaya, and N. Georgalas, "Fast Adaptive Task Offloading in Edge Computing Based on Meta Reinforcement Learning," *IEEE Trans. Parallel and Distributed Syst.*, vol. 32, no. 1, pp. 242–253, 2021.

[100] Z. Quadir, K. N. Le, N. Saeed, and H. S. Munawar, "Towards 6G internet of things: Recent advances, use cases, and open challenges," *ICT Express*, vol. 9, no. 3, pp. 296–312, 2023.

[101] A. Mahmood, T. X. Vu, W. U. Khan, S. Chatzinotas, and B. Ottersten, "Optimizing Computational and Communication Resources for MEC Network Empowered UAV-RIS Communication," in *Proc. IEEE Global Commun. Workshops (GLOBECOM Workshops)*, Rio de Janeiro, Brazil, 2022, pp. 974–979.

[102] X. Dai, Z. Xiao, H. Jiang, and J. C. S. Lui, "UAV-Assisted Task Offloading in Vehicular Edge Computing Networks," *IEEE Trans. Mobile Computing*, vol. 23, no. 4, pp. 2520–2534, 2024.

[103] N. Lin, H. Tang, L. Zhao, S. Wan, A. Hawbani, and M. Guizani, "A PDDQNLP Algorithm for Energy Efficient Computation Offloading in UAV-Assisted MEC," *IEEE Trans. Wireless Commun.*, vol. 22, no. 12, pp. 8876–8890, 2023.

[104] Q. Ren, O. Abbasi, G. K. Kurt, H. Yanikomeroglu, and J. Chen, "Caching and Computation Offloading in High Altitude Platform Station (HAPS) Assisted Intelligent Transportation Systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 11, pp. 9010–9024, 2022.

[105] ——, "High Altitude Platform Station (HAPS) Assisted Computing for Intelligent Transportation Systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Madrid, Spain, 2021, pp. 1–6.

[106] D. S. Lakew, A. T. Tran, N. N. Dao, and S. Cho, "Intelligent Offloading and Resource Allocation in HAP-Assisted MEC Networks," in *Proc. Int. Conf. Info. and Commun. Tech. Convergence (ICTC)*, Jeju, Republic of Korea, 2021, pp. 1582–1587.

[107] *HAPS Technological Assessment Report.* Accessed: Jan. 25, 2024. [Online]. Available: https://www.frontex.europa.eu/assets/EUresearchprojects/2023/FX_HAPS_WP2_-_Technological_Assessment_Consolidated.pdf

[108] I. Leyva-Mayorga *et al.*, "Satellite Edge Computing for Real-Time and Very-High Resolution Earth Observation," *IEEE Trans. Commun.*, vol. 71, no. 10, pp. 6180–6194, 2023.

[109] *SpaceX Non-Geostationary Satellite System.* Accessed: Jan. 29, 2024. [Online]. Available: https://fcc.report/IBFS/SAT-MOD-20181108-00083/1569860.pdf

[110] *OneWeb Non-Geostationary Satellite System.* Accessed: Jan. 29, 2024. [Online]. Available: https://fcc.report/IBFS/SAT-MPL-20200526-00062/2379565.pdf

[111] Q. Tang, Z. Fei, B. Li, and Z. Han, "Computation Offloading in LEO Satellite Networks With Hybrid Cloud and Edge Computing," *IEEE Internet of Things J.*, vol. 8, no. 11, pp. 9164–9176, 2021.

[112] M. Tong, X. Wang, S. Li, and L. Peng, "Joint offloading decision and Resource Allocation in mobile edge computing-enabled satellite-terrestrial network," *Symmetry*, vol. 14, no. 3, p. 564, 2022.

[113] M. Tong, S. Li, X. Wang, and P. Wei, "Inter-satellite cooperative offloading decision and resource allocation in Mobile Edge Computing-enabled satellite–terrestrial networks," *Sensors*, vol. 23, no. 2, p. 668, 2023.

[114] Y. Hao, Z. Song, Z. Zheng, Q. Zhang, and Z. Miao, "Joint Communication, Computing, and Caching Resource Allocation in LEO Satellite MEC Networks," *IEEE Access*, vol. 11, pp. 6708–6716, 2023.

[115] C. Mei *et al.*, "Joint Task Offloading and Resource Allocation for space–Air–Ground Collaborative Network," *Drones*, vol. 7, no. 7, p. 482, 2023.

[116] H. Zhang, R. Liu, A. Kaushik, and X. Gao, "Satellite Edge Computing With Collaborative Computation Offloading: An Intelligent Deep Deterministic Policy Gradient Approach," *IEEE Internet of Things J.*, vol. 10, no. 10, pp. 9092–9107, 2023.

[117] N. Cheng *et al.*, "Space/Aerial-Assisted Computing Offloading for IoT Applications: A Learning-Based Approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, 2019.

[118] F. Tang, H. Hofner, N. Kato, K. Kaneko, Y. Yamashita, and M. Hangai, "A Deep Reinforcement Learning-Based Dynamic Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN)," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 276–289, Jan. 2022.

[119] C. Zhou *et al.*, "Deep Reinforcement Learning for Delay-Oriented IoT Task Scheduling in SAGIN," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 911–925, Feb. 2021.

[120] D. Han *et al.*, "Two-Timescale Learning-Based Task Offloading for Remote IoT in Integrated Satellite–Terrestrial Networks," *IEEE Internet of Things J.*, vol. 10, no. 12, pp. 10 131–10 145, Jun. 2023.

[121] Q. Chen, Z. Guo, W. Meng, S. Han, C. Li, and T. S. Quek, "A Survey on Resource Management in Joint Communication and Computing-Embedded SAGIN," *IEEE Commun. Surveys and Tuts.*, 2024, doi: 10.1109/COMST.2024.3421523.

[122] Z. Yin *et al.*, “A comprehensive survey of orbital edge computing: Systems, applications, and algorithms,” *Chinese J. Aeronautics*, 2024, doi: https://doi.org/10.1016/j.cja.2024.11.026.

[123] S. Bhandari, T. X. Vu, and S. Chatzinotas, “User-Centric Flexible Resource Management Framework for LEO Satellites With Fully Regenerative Payload,” *IEEE J. Sel. Areas Commun.*, vol. 42, no. 5, pp. 1246–1261, 2024.

[124] E. Carmona-Cejudo and F. Iadanza, “Demo: A Decision Support System for Task Offloading Optimization in Cloud-to-Far-Edge Kubernetes Networks,” in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Seoul, Republic of Korea, 2022, pp. 1–2.

[125] E. Carmona-Cejudo, F. Iadanza, and M. S. Siddiqui, “Optimal Offloading of Kubernetes Pods in Three-Tier Networks,” in *Proc. IEEE Wireless Commun. and Networking Conf. (WCNC)*, Austin, TX, USA, 2022, pp. 280–285.

[126] Q. M. Nguyen, L. A. Phan, and T. Kim, “Load-Balancing of Kubernetes-Based Edge Computing Infrastructure Using Resource Adaptive Proxy,” *Sensors*, vol. 22, no. 8, p. 2869, 2022.

[127] H. Wang, Y. Wang, G. Liang, Y. Gao, and W. Zhang, “Research on load balancing technology for microservice architecture,” in *Proc. MATEC Web of Conf.*, 2021, p. 8002.

[128] G. Wang, F. Yang, J. Song, and Z. Han, “Can TDD Be Employed in LEO SatCom Systems? Challenges and Potential Approaches,” *https://doi.org/10.48550/arXiv.2502.08179*, 2025.

[129] A. U. Chaudhry and H. Yanikomeroglu, “L]aser Intersatellite Links in a Starlink Constellation: A Classification and Analysis,” *IEEE Veh. Technol. Mag.*, vol. 16, no. 2, pp. 48–56, 2021.

[130] G. Wang, F. Yang, J. Song, and Z. Han, “Free Space Optical Communication for InterSatellite Link: Architecture, Potentials and Trends,” *arXiv preprint arXiv:2310.17505v1*, 2023.

[131] F. Wang, H. Yao, W. He, H. Chang, X. Xin, and S. Guo, “Time-Sensitive Scheduling Mechanism Based on End-to-End Collaborative Latency Tolerance for Low-Earth-Orbit Satellite Networks,” *IEEE Trans. Network Science and Engineering*, vol. 11, no. 6, pp. 5149–5162, 2024.

[132] F. Tang, C. Wen, L. Luo, M. Zhao, and N. Kato, “Blockchain-Based Trusted Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN): A Federated Reinforcement Learning Approach,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3501–3516, Dec. 2022.

[133] A. Beck, A. Ben-Tal, and L. Tetruashvili, “A sequential parametric convex approximation method with applications to nonconvex truss topology design problems,” *J. Glob. Optim.*, vol. 47, no. 1, pp. 29–51, 2010.

[134] S. Lu, I. Tsaknakis, and M. Hong, “Block Alternating Optimization for Non-convex Min-max Problems: Algorithms and Applications in Signal Processing and Communications,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, UK, 2019, pp. 4754–4758.

[135] O. Korcak and F. Alagoz, “Optimal Beam Management in Earth-fixed Satellite Systems,” in *Proc. Int. Workshop on Satell. and Space Commun.*, Salzburg, Austria, 2007, pp. 43–47.

[136] SES. *Connected Cruises: Cruise Internet Solution — SES*. Accessed: Jan. 17, 2024. [Online]. Available: https://www.ses.com/find-service/cruise

[137] N2YO.com. *Technical details for satellite starlink-4280*. Accessed: Jan. 10, 2024. [Online]. Available: https://www.n2yo.com/satellite/?s=53178

[138] Ian Taylor Trekking. *How many people have climbed Mount Everest?* Accessed: Jan. 17, 2024. [Online]. Available: https://iantaylortrekking.com/blog/how-many-people-have-climbed-mount-everest/

[139] N. Baral, S. Kaul, J. T. Heinen, and S. B. Ale, “Estimating the value of the World Heritage Site designation: A case study from Sagarmatha (Mount Everest) National Park, Nepal,” *Protected Areas, Sustainable Tourism and Neo-liberal Governance Policies*, pp. 80–95, 2020.

[140] GeographyPods. *Reaching the top - word - geography for 2020 & beyond.* Accessed: Jan. 17, 2024. [Online]. Available: https://www.geographypods.com/uploads/7/6/2/2/7622863/f8reachingthetop.pdf

[141] Y. Jiang, Y. Mao, G. Wu, Z. Cai, and Y. Hao, "A collaborative optimization strategy for computing offloading and resource allocation based on multi-agent deep reinforcement learning," *Comput. and Elec. Eng.*, vol. 103, p. 108278, 2022.

[142] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Upper Saddle River: Pearson, 2019.

[143] Z. Li, X. Zhou, Y. Liu, C. Fan, and W. Wang, "A computation offloading model over collaborative cloud-edge networks with Optimal Transport theory," in *Proc. IEEE 19th Int. Conf. Trust, Security and Privacy in Comput. and Commun.)*, Guangzhou, China, 2020, pp. 1006–1011.

[144] T. Deng, Y. Chen, G. Chen, M. Yang, and L. Du, "Task offloading based on edge collaboration in MEC-enabled IoV networks," *J. Commun. and Netw.*, vol. 25, no. 2, pp. 197–207, 2023.

[145] *Study on New Radio (NR) to support non-terrestrial networks*, document Tech. Report (TR) 38.811, Ver. 15.1.0, Rel. 15, 3rd Gener. Partnership Project (3GPP), 2019.

[146] *The most widely deployed open source cloud software in the world.* Accessed: Jan. 29, 2024. [Online]. Available: https://www.openstack.org

[147] AWS. *Processing satellite imagery with serverless architecture — AWS Compute Blog.* Accessed: Jan. 21, 2024. [Online]. Available: https://aws.amazon.com/blogs/compute/processing-satellite-imagery-with-serverless-architecture/

[148] S. Bhandari, T. X. Vu, and S. Chatzinotas, "LEO Satellite-assisted Task Offloading for a Near Real-time Earth Observation Service," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Cape Town, South Africa, 2024, pp. 3033–3038.

[149] M. Giordani, M. Polese, M. Mezzavilla, S. Rangan, and M. Zorzi, "Toward 6G networks: Use cases and technologies," *IEEE Commun. Mag.*, vol. 58, no. 3, pp. 55–61, Mar. 2020.

[150] F. Liu *et al.*, "Integrated Sensing and Communications: Toward Dual-Functional Wireless Networks for 6G and Beyond," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 6, pp. 1728–1767, Jun. 2022.

[151] M. C. Weinzierl, K. Lucas, and M. Sarang, "SpaceX, Economies of Scale, and a Revolution in Space Access," *Harvard Business School Case 720-027*, 2021.

[152] D. Zhao, X. Zhang, and M. Lee, "SatPipe: Deterministic TCP Adaptation for Highly Dynamic LEO Satellite Networks," in *IEEE INFOCOM 2025 - IEEE Conference on Computer Communications*, London, United Kingdom, 2025, pp. 1–10.

[153] D. Laniewski, E. Lanfer, and N. Aschenbruck, "Measuring Mobile Starlink Performance: A Comprehensive Look," *IEEE Open Journal of the Communications Society*, vol. 6, pp. 1266–1283, 2025.

[154] B. Li, A. P. Petropulu, and W. Trappe, "Optimum co-design for spectrum sharing between matrix completion based mimo radars and a mimo communication system," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4562–4575, 2016.

[155] S. Buzzi, C. D'Andrea, and M. Lops, "Using massive MIMO arrays for joint communication and sensing," in *Proc. Annual Asilomar Conf. Signals, Syst., Comp.)*, Pacific Grove, CA, USA, 2019, pp. 5–9.

[156] C. Qi, W. Ci, J. Zhang, and X. You, "Hybrid beamforming for millimeter wave MIMO integrated sensing and communications," *IEEE Commun. Lett.*, vol. 26, no. 5, pp. 1136–1140, May 2022.

[157] X. Wang, Z. Fei, J. A. Zhang, and J. Xu, "Partially-connected hybrid beamforming design for integrated sensing and communication systems," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6648–6660, Oct. 2022.

[158] Z. Cheng, Z. He, and B. Liao, "Hybrid beamforming design for ofdm dual-function radar-communication system," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 6, pp. 1455–1467, 2021.

[159] N. T. Nguyen, N. Shlezinger, K. H. Ngo, V. D. Nguyen, and M. Juntti, "Joint Communications and Sensing Design for Multi-Carrier MIMO Systems," in *Proc. IEEE*

Statistical Signal Process. Workshop (SSP Workshop), Hanoi, Vietnam, 2023, pp. 110–114.

[160] L. You *et al.*, "Ubiquitous Integrated Sensing and Communications for Massive MIMO LEO Satellite Systems," *IEEE Internet of Things Mag.*, vol. 7, no. 4, pp. 30–35, Jul. 2024.

[161] Z. Liu, L. Yin, W. Shin, and B. Clerckx, "Rate-Splitting Multiple Access for Quantized ISAC LEO Satellite Systems: A Max-Min Fair Energy-Efficient Beam Design," *IEEE Trans. Wireless Commun.*, vol. 23, no. 10, pp. 15 394–15 408, Oct. 2024.

[162] A. A. Lu, X. Gao, X. Meng, and X. G. Xia, "Omnidirectional Precoding for 3D Massive MIMO With Uniform Planar Arrays," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2628–2642, Apr. 2020.

[163] M. Temiz, E. Alsusa, and M. W. Baidas, "A Dual-Function Massive MIMO Uplink OFDM Communication and Radar Architecture," *IEEE Trans. Cognitive Commun. and Networking*, vol. 8, no. 2, pp. 750–762, Jun. 2022.

[164] Q. Zhang, H. Sun, X. Gao, X. Wang, and Z. Feng, "Time-Division ISAC Enabled Connected Automated Vehicles Cooperation Algorithm Design and Performance Evaluation," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 7, pp. 2206–2218, Jul. 2022.

[165] O. E. Ayach, S. Rajagopal, S. Abu-Surra, Z. Pi, and R. W. Heath, "Spatially Sparse Precoding in Millimeter Wave MIMO Systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1499–1513, 2014.

[166] F. Liu, L. Zhou, C. Masouros, A. Li, W. Luo, and A. Petropulu, "Toward dual-functional radar-communication systems: Optimal waveform design," *IEEE Trans. Signal Process.*, vol. 66, no. 16, pp. 4264–4279, 2018.

[167] CVX. *Introduction.* Accessed: Jan. 29, 2025. [Online]. Available: https://cvxr.com/cvx/doc/intro.html

[168] K. Xu, N. V. Huynh, and G. Y. Li, "Distributed-Training-and-Execution Multi-Agent Reinforcement Learning for Power Control in HetNet," *IEEE Trans. Commun.*, vol. 71, no. 10, pp. 5893–5903, Oct. 2023.

[169] A. A. Khan and R. S. Adve, "Centralized and Distributed Deep Reinforcement Learning Methods for Downlink Sum-Rate Optimization," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8410–8426, 2020.

[170] G. P. Meyer, "An Alternative Probabilistic Interpretation of the Huber Loss," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Nashville, TN, USA, 2021, pp. 5257–5265.

[171] N2YO. *Starlink satellites*. Accessed: Jan. 10, 2025. [Online]. Available: https://www.n2yo.com/satellites/?c=52&p=A