3D Channel Tracking in Space-Air-Ground Integrated Networks

Jiadong Yu

Doctor of Philosophy

School of Electronic Engineering and Computer Science

Queen Mary, University of London

Queen Mary
University of London

October 5, 2021

Abstract

The space-air-ground integrated network (SAGIN) aims to provide seamless wide-area connections, high throughput and strong resilience for beyond the fifth generation (B5G) and future communications. As a multidimensional network, SAGIN adopts different communication links across three segments: the space segment with satellite networks, the air segment with aerial networks, and the ground segment with territorial networks. Apart from Ka-band millimetre wave (mmWave) frequencies being utilized for low earth orbit (LEO) satellites and medium earth orbit (MEO) satellites communications, with emerging smart devices brought online and crowded under-6GHz spectrum, mmWave frequencies have also been widely considered to support both aerial networks and territorial networks. To ensure stable wireless communications and tackle the severer propagation loss of mmWave transmission, massive multiple input and multiple output (MIMO) and intelligent reflecting surfaces (IRSs), which can configure directional beams and bring huge improvements of radiated energy efficiency, are two technologies to be employed in SAGIN.

Conventionally, perfect channel state information (CSI) is the fundamental knowledge to enable building reliable communication connections. With massive antenna arrays installed on orbiting satellites, navigation unmanned aerial vehicles (UAVs), and base stations, it's very challenging to acquire real-time mmWave CSI in SAGIN due to the heavy overheads and the dynamic environment. Most existing mmWave channel estimation work proposed compressive sensing (CS) based algorithms to reduce the heavy overheads with the assumption that the environment is in two-dimensional (2D) space and static without any movement. However, in SAGIN, 2D and static assumptions are not practical. Hence, tracking the dynamic three-dimensional (3D) CSI using small training overheads becomes a crucial and demanding task.

In this thesis, 3D channel tracking algorithms are proposed based on unique characteristics of air-ground and space-air links. For IRS-assisted air-ground links, we propose a 3D geometry dynamic channel model with both UAV navigation and mobile user movement. We further develop a deep learning (DL)-based channel tracking algorithms with two modules: deep neural network (DNN) channel pre-estimation for denoising and stacked bi-directional long short term memory (Stacked Bi-LSTM) for channel tracking. For space-air links, we exploit the on-grid and off-grid single user (SU) and multi-user (MU) UAV-satellite communications. Two statistical spatial and temporal correlation sparsity of the dynamic channel models called 3D two-dimensional Markov model (3D-2D-MM) and multi-dimensional Markov model (MD-MM) are developed by introducing the more realistic 3D movement in the system. Based on the message passing rule and the proposed Markov structures, 3D dynamic turbo approximate message passing algorithm (3D-DTAMP) and multi-dimensional dynamic turbo approximate message passing (MD-DTAMP) are derived for channel tracking. Our proposed algorithms can achieve better channel estimation accuracy with comparable complexity and smaller training overheads.

Acknowledgments

Foremost, I would like to express my deepest gratitude to my primary supervisor Prof. Yue Gao for his prudent guidance, continuous support and patience during my PhD journey, and to my co-supervisor Dr. Yuanwei Liu for his kind help and continuous support in the final year of my PhD. With their professional supervision, I could figure out research problems, deal with challenges, and then contribute a list of publications.

Secondly, I would like to thank Prof. Sherman Shen (University of Waterloo), and Prof. Wei Zhang (University of New South Wales), for their technical support and research feedback. Through our collaboration, I gained special research experience and a thorough understanding of my research work. Moreover, I want to thank my colleague and friend Dr. Xiaolan Liu for all her patient revision of my research paper and I appreciate the moment that we share with each other on not only research but also life inspiration.

I would also like to express my thanks to supportive teams at QMUL, especially Ms. Melissa Yeo and the EECS staff, for their support of the student community. I also sincerely thank colleagues in the antenna group including Dr. Zhijin Qin, Dr. Xingjian Zhang, Wei Su, Dr. Wenqiang Yi, Dr. Xiao Liu, Dr. Shaoqing Hu, Dr. Shaker Alkaraki, Haoyang Zhang, Joseph Nannim Mamvong, for their insightful discussion and kind help.

I am deeply grateful to my dear friend Dr. Yusha Liu for always being able to motivate me to make progress together and enjoy every moment of life together. I would also like to thank my friends Dr. Luping Xiang, Dr. Haoran Qi, Dr. Yan Liu, Dr. Xiaomiao Li, Zihang Song, for their continuous encouragement and accompany during my PhD research.

Finally, I would like to express my deepest gratitude to my beloved family especially my parents and my boyfriend Dr. Yunpeng Zhan, who are always there giving me enormous love, encouragement, and supporting me to be myself. At the same time, I would like to thank my British family Ms. Elaine Povey, Ms. Angie Bromfield, and Mr. Edward Roy Lewis. Their gentleness and caring give me a sense of belonging in the UK all these years. To them, I dedicate this thesis.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

2D-MM	two-dimensional Markov model
3D	three-dimensional
3D-2D-MM	three-dimensional two-dimensional Markov model
3D-DTAMP	three-dimensional dynamic turbo approximate message passing
3GPP	third Generation Partnership Project
5G	the fifth generation
AMP	approximate message passing
ANN	artificial neural network
AoA	angles of arrival
AoD	angles of departure
B5G	beyond the fifth-Generation
BS	base station
CoSaMP	compressive sampling matching pursuit
CS	compressive sensing
CSI	channel state information
DFT	discrete Fourier transform
DL	deep learning
DNN	deep neural network
D-OMP	differential OMP
EM	expectation-maximization
eMBB	enhanced mobile broadband

Gbps	gigabit-per-second
GEO	geostationary orbit
HMM	hidden Markov model
IEEE	Institute of Electrical and Electronics Engineers
IoT	internet of things
IRSs	intelligent reflecting surfaces
LASSO	least absolute shrinkage and selection operator
LEO	low earth orbit
LMMSE	linear minimum mean-square error
LoS	line-of-sight
LS	least square
LSTM	long short term memory
mmWave	Millimetre Wave
MANET	mobile ad hoc network
MD-DTAMP	multi-dimensional dynamic turbo approximate message passing
MD-MM	multi-dimensional Markov model
MEO	medium earth orbit
MIMO	multiple-input multiple-output
ML	machine learning
MMSE	minimum mean square error
MP	matching pursuit
MS	mobile station
MU	multi-user
OMP	orthogonal matching pursuit
Rel-15	Release 15
RNN	Recurrent neural network
SAGIN	space-air-ground integrated network
SBL	sparse Bayesian learning
SNR	signal-to-noise ratio

SP	subspace pursuit
Stacked Bi-LSTM	stacked bi-direction long short term memory
SU	single-user
TNMSE	time-averaged normalized mean square error
UAV	unmanned aerial vehicle
UPA	uniform plane array
V2V	vehicle-to-vehicle
WLANs	wireless local area networks

Chapter 1

Introduction

Space-air-ground integrated network (SAGIN) has attracted intensive attention from both academia and industry for beyond the fifth generation (B5G) communication and future wireless communications. The advantages from three segments (i.e., space, air, and ground) can be exploited to support multifarious services and scenarios in an efficient and cost-effective manner. Territorial networks in the ground segment can serve mobile users through high data rates in urban or suburban areas. Aerial networks in the air segment can be utilized to boost capacity in areas with congested or disaster terrestrial infrastructure. Satellite networks in the space segment can help reach ubiquitous coverage in rural and remote areas. Moreover, both satellites and unmanned aerial vehicles (UAVs) can assist terrestrial networks and relieve the overload data demands [1].

With the rapid increase of mobile data growth and the use of smart devices, the bandwidth shortage has become an unprecedented challenge for wireless services. The less-congested millimetre-wave (mmWave) frequency spectrum, offering the potentiality of high-capacity wireless transmission of multi gigabit-per-second(Gbps) data rates, has been widely considered as a promising technology solution to such significantly increase network capacity [2]. Moreover, for low earth orbit (LEO) satellite and medium earth orbit (MEO) satellite communication in SAGIN, Ka-band mmWave frequency has been widely deployed [3]. Therefore, multiple mmWave frequencies will be operated across three segments in SAGIN. However, there are two major characteristics of mmWave propagation that should be drawn special attention: atmospheric attenuation and shadowing by objects. The atmospheric oxygen absorption is especially severe at the frequency around 60GHz and 120GHz, and the rain attenuation is also greater than that of sub-6GHz [4]. Furthermore, due to the small wavelength, mmWave signal can be easily interrupted by obstacles, such as human bodies, doors, buildings, and vegetation [5].

To mitigate the vulnerability of the mmWave transmission, two advanced technologies have been broadly studied: massive multiple-input multiple-output (MIMO) and intelligent reflecting surfaces (IRSs). Although the short wavelength is the main reason behind the strong path loss of the mmWave transmission, the large antenna arrays can be integrated into a smaller form, due to the principle that the distance between any two antenna array elements is basically half of the wavelength. With the help of extra antenna elements in the massive MIMO system, highly directional beams can be generated and bring huge improvements of the radiated energy efficiency [6]. Similarly, an IRS is an artificial surface that can be implemented with large arrays or metamaterial elements to customize the propagation of the radio waves impinging upon it [7]. In SAGIN, satellites in the space segment, UAV relays in the air segment, and base stations in the ground segment can all be installed with large antenna arrays. At the same time, IRSs can be installed on buildings, walls, street lamps, and so on. Conventionally, to enable both MIMO and IRSs on configuring directional beams, beamforming matrices should be designed based on the knowledge of the channel state information (CSI). However, in SAGIN, the large number of antenna arrays/ reflection elements and the dynamic propagation environment caused by satellite orbiting, UAV navigation and ground user movement impose more difficulty to obtain accurate real-time channels. Moreover, state-of-art studies are not practical, as they focus on 2D static channel estimation without considering the time-varying environment.

Different from channel estimation in static case, channel tracking can provide more reliable dynamic CSI to support communications. This is because it considers the adjacent observations correlation across time domain in dynamic environment. This thesis will present several mmWave dynamic 3D channel tracking algorithms based on specific characteristics in air-ground and space-air links.

1.1 Motivation and Contribution

The detailed motivations and contributions of my PhD research are summarised in the following.

1.1.1 Deep Learning Based 3D Channel Tracking for IRS-assisted Air-Ground Communications

As wireless communication networks move towards B5G, mmWave is a promising technology to accommodate explosive increment of users online with wide spectral efficiency. To mitigate the high penetration loss and short wireless paths of mmWave transmission, UAVs aided communications have drawn dramatically attention due to its flexibility in establishing line-of-sight (LoS) communications in SAGIN. However, with the blockage in the environment, and due to the movement of UAVs and mobile users, the directional paths are very vulnerable. IRSs that can reflect signals to generate virtual LoS paths are capable of providing stable communications and serving wider coverage. From our knowledge, state-of-art air-ground communications haven't considered the help of IRSs in the system. Hence, for the channel estimation problem in IRS-assisted UAV communications systems, there are still many challenges to tackle: good estimation performance, the small pilot overheads, and the dynamic channel caused by the mobility of both UAV and mobile users. Therefore, in such a complex dynamic environment of the IRSs-assisted air-ground links in SAGIN, efficient channel tracking algorithms should be designed to support stable wireless communications.

In this thesis, we exploit a three-dimensional (3D) geometry dynamic channel model in the mmWave IRS-assisted UAV-enabled communication system which considers the velocities of UAV and mobile users, and the blockage parameter of LoS link. Moreover, we develop a novel deep learning (DL) based channel tracking algorithm consisting of two modules: channel pre-estimation and channel tracking. A deep neural network (DNN) with offline training is designed for denoising in the pre-estimation module. Furthermore, for channel tracking, a stacked bi-directional long short term memory (Stacked Bi-LSTM) is developed based on a framework that can trace back historical time sequence together with a bidirectional structure over multiple stacked layers. Simulations have shown that the proposed channel tracking algorithm requires fewer epochs to convergence compared to benchmark algorithms, such as least square (LS) and DNN. It also demonstrates that the proposed algorithm is superior to different benchmarks (DNN+LSTM, DNN+Bi-LSTM) with small pilot overheads and comparable computation complexity.

1.1.2 3D Channel Tracking for SU Space-Air Communications

Acting as another crucial link segment of the SAGIN, space-air communication has drawn much attention. Due to the LEO satellite orbiting and 3D UAV trajectory navigation, it is a key challenge to track real-time channel information in such dynamic environment. Fortunately, different from the conventional channel estimation, there are some distinct properties that can be considered to support the channel tracking in space-air links: the high non-stationary and the dominant LoS rays.

In this thesis, we explore the 3D channel tracking for a Ka-band space-air communication systems. We firstly propose a statistical dynamic channel model called the 3D two-dimensional Markov model (3D-2D-MM) for the UAV-satellite communication system by exploiting the probabilistic insight relationship of both hidden value vector and joint hidden support vector. Specifically, the hidden value vector can be treated as the amplitude of the channel gain and the joint hidden support vector can be treated as the sparsity of the directions in angular domain. For the joint hidden support vector, we consider a more realistic 3D support vector in both azimuth and elevation directions. Moreover, for each direction, the spatial sparsity structure and the time-varying probabilistic relationship between degree patterns named the spatial and temporal correlation are studied, respectively. Furthermore, we derive a novel 3D dynamic turbo approximate message passing (3D-DTAMP) algorithm to recursively track the dynamic channel through the message passing rule based on the proposed 3D-2D-MM structure. Numerical results show that our proposed algorithm achieves superior channel tracking performance named time-averaged normalized mean square error (TNMSE) to the state-of-the-art algorithms with lower pilot overheads and comparable complexity.

1.1.3 3D on-grid and off-grid Channel Tracking for MU Space-Air Communications

To further explore the more practical scenario of space-air communications in SAGIN, the spatial relationship between multiple UAVs should be further considered. Moreover, as conventional mmWave channel models are proposed based on 3D on-grid angles, it is inevitable to study the true 3D off-grid cases[1] with improved channel tracking accuracy.

In this thesis, we further propose a statistical dynamic channel model called the multi-dimensional Markov model (MD-MM), which investigates the more realistic spatial and temporal correlation in the sparse UAVs-satellite channel. Specifically, the spatial and temporal probabilistic relationships of multi-user (MU) hidden support vector, single-user (SU) joint hidden support vector, and SU hidden value vector are investigated. The specific transition probabilities that connect the SU and MU hidden support vectors for both azimuth and elevation directions are defined. Moreover, we derive a novel multi-dimensional dynamic turbo approximate message passing (MD-DTAMP) algorithm through the message passing rule based on the proposed MD-MM structure for tracking the 3D dynamic channel in the multiple UAVs system. Furthermore, we also develop a gradient update scheme to recursively find the azimuth and elevation offset

[1] Specifically, for on-grid case, the angle range is divided into multiple average grids. The true angle will be approximated as the nearest on-grid degree. Differently, off-grid can be treated as the true value.

for 3D off-grid estimation. Numerical results verify that the proposed algorithm shows superior 3D channel tracking performance called TNMSE with smaller pilot overheads and comparable complexity.

1.2 Publications

The publications during my PhD study are listed below. Part of [J2] and [C1] are included in Chapter 2. The work in [J4] is discussed in Chapter 3. The work in Chapter 4 has been published as [J1] and [C2]. Moreover, the work in Chapter 5 has been accepted as [J3].

Journal papers:

[J1]. **J. Yu**, X. Liu, Y. Gao and X. Shen, "3D Channel Tracking for UAV-Satellite Communications in Space-Air-Ground Integrated Networks," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 12, pp. 2810-2823, Dec. 2020, doi: 10.1109/JSAC.2020.3005490.

[J2]. **J. Yu**, X. Liu, H. Qi and Y. Gao, "Long-term Channel Statistic Estimation for Highly-Mobile Hybrid MmWave Multi-User MIMO Systems," in IEEE Transactions on Vehicular Technology, doi: 10.1109/TVT.2020.3040794.

[J3]. **J. Yu**, X. Liu, Y. Gao and X. Shen, " 3D On and Off-Grid Dynamic Channel Tracking for Multiple UAVs and Satellite Communications," in IEEE Transactions on Wireless Communications, accepted with minor revision, 2021.

[J4]. **J. Yu**, X. Liu, Y. Gao ,C. Zhang, W. Zhang, " Channel Tracking with Deep Learning in Intelligent Reflection Surface Assisted UAV Communication System," in IEEE Transactions on Wireless Communications, major revision, 2021.

[J5]. **J. Yu**, R. Zhang, Y. Gao and L. Yang, "Modularity-Based Dynamic Clustering for Energy Efficient UAVs-Aided Communications," in IEEE Wireless Communications Letters, vol. 7, no. 5, pp. 728-731, Oct. 2018, doi: 10.1109/LWC.2018.2816649.

[J6]. X. Liu, **J. Yu**, Z. Feng and Y. Gao, "Multi-agent Reinforcement Learning for Resource Allocation in IoT Networks with Edge Computing," in China Communications, vol. 17, no. 9, pp. 220-236, Sept. 2020, doi: 10.23919/JCC.2020.09.017.

[J7]. X. Liu, **J. Yu**, J. Wang and Y. Gao, "Resource Allocation with Edge Computing in IoT Networks via Machine Learning," in IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3415-3426, April 2020, doi: 10.1109/JIOT.2020.2970110.

[J8]. X. Liu, **J. Yu**, et al., "Learning to Predict the Mobility of Users in Mobile mmWave Networks," in IEEE Wireless Communications, vol. 27, no. 1, pp. 124-131, February 2020, doi: 10.1109/MWC.001.1900241.

Conference papers:

[C1]. **J. Yu**, X. Liu, H. Qi, W. Zhang and Y. Gao, "Spatial Channel Covariance Estimation for Hybrid mmWave Multi-User MIMO Systems," 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 2019, pp. 1-6.

[C2]. **J. Yu**, X. Liu, and Y. Gao, "On-Grid 3D Dynamic Channel Tracking forSpace-Air Communications with Multiple UAVs," IEEE/CIC International Conference on Communications in China (ICCC), accepted.

[C3]. H. Qi, **J. Yu**, A. Cavallaro and Y. Gao, "Robust Compressive Sensing of Multiband Spectrum with Partial and Incorrect Priors," ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6.

1.3 Thesis Outline

Chapter 2 provides an overview of the mmWave channel estimation and channel tracking in SAGIN. The structure of the SAGIN is firstly introduced. The reason why our research focusing on mmWave channel estimation in SAGIN is addressed. Conventional CS-based and DL-based mmWave channel estimation algorithms are explained. Moreover, due to the dynamic links in SAGIN, two basic mmWave channel tracking algorithms are further reviewed.

Chapter 3 proposes a 3D geometry dynamic channel model in mmWave IRS-assisted air-ground communications. A two modules DL-based channel tracking scheme is developed: channel pre-estimation for denoising using DNN and channel tracking using Stacked Bi-LSTM.

Chapter 4 proposes an on-grid 3D-2D-MM for the SU space-air communications. For the joint hidden support vector, a more realistic 3D support vector in both azimuth and elevation directions is explored. The spatial and temporal correlation are all considered in each direction. A novel 3D-DTAMP algorithm is proposed to track the dynamic channel based on the 3D-2D-MM priors.

Chapter 5 proposes a statistical dynamic channel model called MD-MM for the MU space-air communications. The spatial and temporal probabilistic relationships of MU hidden support vector, SU joint hidden support vector, and SU hidden value vector are explored. A novel MD-DTAMP algorithm is proposed to track the 3D dynamic channel. A gradient update scheme is further developed to find the 3D offset for the off-grid estimation.

Chapter 6 draws the conclusions of this thesis and potential future research work.

Chapter 2

Background

2.1 Overview of SAGIN Networks

In order to handle the dramatically increased traffic demands of different communication services, such as the Internet of Things (IoT), big data, cloud and edge computing, it's imperative to exploit the technology to provide wireless communication services with high data rate at any place on the earth. The SAGIN[8] has been drawn dramatic attention for its advantages such as seamless wide-area connections, high throughput, and strong resilience for B5G communications [1]. In recent years, several announcements have been published to deploy satellite constellations into LEO which containing several hundred to thousands of rather small-sized satellites to construct the SAGIN, such as OneWeb, Norwegian STEAM network [9], and SpaceX[10], etc.

2.1.1 SAGIN Architecture

There are mainly three segments in the SAGIN multidimensional network: the space segment with satellite network, the air segment with aerial network and the ground segment with territorial network [11]. The architecture of the SAGIN can be found in Fig. 2.1.

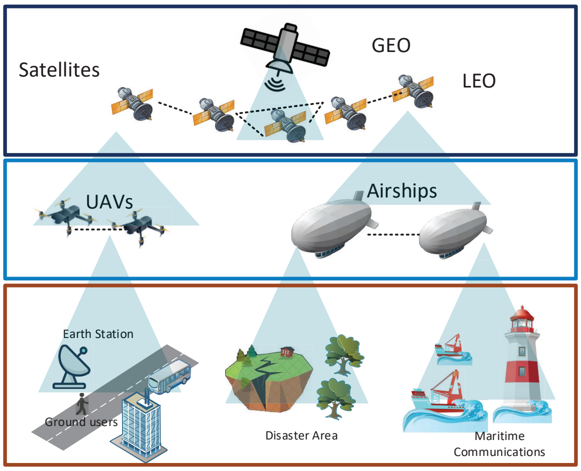


Figure 2.1: The illustration of the Space-Air-Ground Integrated Networks

- Space segment: The space network consists of satellite constellations which are orbiting at different altitude. Based on the altitude, satellites can be categorized into geostationary orbit (GEO), MEO, and LEO. Moreover, satellite constellations can construct various satellite networks with different characteristics, such as narrowband satellite networks, broadband satellite networks, and multi-layered satellite networks with inter-satellite links and inter-layer links.

- Air segment: The air network is mainly composed of low and high altitude platforms such as UAVs, airships, and balloons. These aerial mobile systems are acting as base stations and relays to provide larger coverage wireless access services. Compared to ground base stations, the aerial base stations in the air network can be low-cost, easy-deployment and can provide wider coverage of communication service.

- Ground segment: The ground network includes different terrestrial system networks such as cellular networks, mobile ad hoc network (MANET)[12], wireless local area networks (WLANs), and so on. However, the coverage of ground networks is limited especially in rural, remote, maritime and disaster areas.

2.1.2 Why mmWave Channel Estimation in SAGIN?

As more and more devices are brought online, the mobile data demand grows dramatically, and the spectrum bands below 6 GHz are becoming increasingly crowded. The underutilized mmWave frequency spectrum is a promising technology to accommodate explosive increment of users online with wide spectral efficiency and multiple Gigabit data transmission speeds [2], [13].

Recently, a number of candidate frequency bands in the mmWave range from 24 - 100 GHz, specifically, the following bands were approved for study (in GHz: 24.25 -27.5; 31.8 - 43.5; 45.5 - 50.2; 50.4 - 52.6; 66 - 76;81 - 86)[14], [15]. The initial commercial deployments of the fifth generation (5G) system based on both sub-6 GHz and at mmWave frequencies (up to 52GHz) are already underway during 2019, focusing on enhanced mobile broadband (eMBB) based on the Release 15 (Rel-15) and Rel-16 of the 3rd Generation Partnership Project (3GPP) specifications [16]. 3GPP has already nearly completed a study of use cases, deployment scenarios and requirements for the full frequency range 52.6-114.25 GHz, and this is expected to lead to a detailed study of technical considerations in Rel-17 and specifications in Rel-18 [16]. Apart from 3GPP, the Institute of Electrical and Electronics Engineers (IEEE) also has recently defined specifications in the 60 GHz mmWave band in 802.11 ay[17]. Moreover, Ka-band which is the mmWave spectrum range between 26.5 GHz to 36 GHz is widely used in MEO and GEO satellites system to achieve high-speed data transmission [3], [18].

Hence, for future wireless communications, multiple mmWave frequencies will be deployed across different communication links in SAGIN, such as space-air links[19], space-ground links [14], and air-ground links[20].

2.1.2.1 Propagation Characteristics of mmWave Communications

Although mmWave can provide wider available bandwidth, due to the characteristics of the mmWave transmission such as weather dependent and sensitivity to obstacles (as shown in Fig. 2.2), there are some fundamental technical challenges to tackle [2].

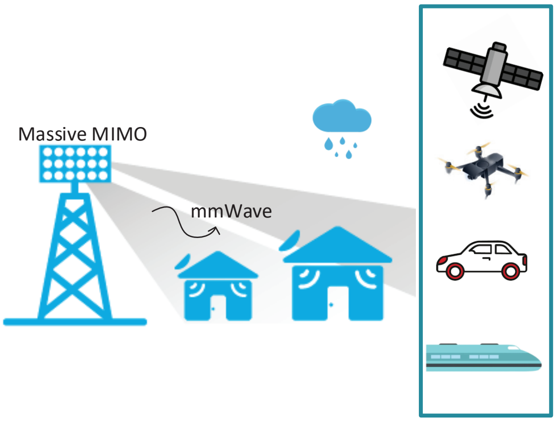


Figure 2.2: The illustration of the mmwave transmission

- Atmospheric attenuation[21], [22]: This kind of attenuation of mmWave transmission is caused particularly by the absorption of oxygen, water vapour gas, and of course rain. Moreover, there are some other inter-related factors such as temperature, altitude, pressure and the most important one, operating carrier frequency that determines the intensity of gaseous absorption[23].

- Shadowing by objects: Since mm-waves do not effectively penetrate or diffract around, static and dynamic obstacles such as foliage, buildings, walls, cars, and even human bodies can shadow the signal [5].

To mitigate the vulnerability of mmWave links caused by atmospheric attenuation effect and shadowing effect, advanced technologies have been widely studied and developed.

2.1.2.2 Enable Technologies and Technical Challenge

Due to the mmWave propagation characteristics, some advanced technologies such as massive MIMO, and IRSs are developed to enable the feasibility of mmWave communications.

- Massive MIMO (also known as large-scale antenna systems)[6], [24]: As the distance between any two antenna elements in the antenna array is normally half of the wavelength, with frequency increases, the large number of antenna elements can be packed into a smaller form to support massive MIMO system. With the support of large antenna arrays, multiple directional beams can be generated to support spatial multiplexing. The fundamental physics of generating sharpness energy for directivity communication is a coherent superposition of wave-front [6] as shown in Fig.2.3. By appropriately shaping the signal sent out from each antenna element, different phase shifts of the signal emitted can lead to a constructive wave-front with focused sharpness regions heading to the specific terminals. Thanks to such directional communications, atmospheric attenuation can be mitigated with improved radiated energy efficiency. At the same time, the narrow beams generated implies securer links can be constructed against eavesdropping and jamming [25].

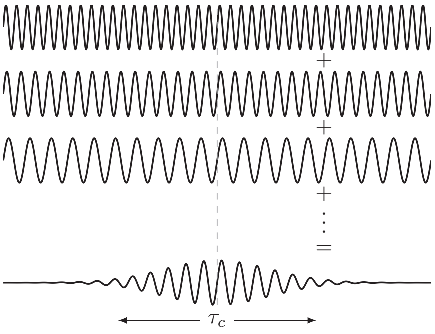


Figure 2.3: The illustration of coherence superposition.

- IRSs (also known as large intelligent surfaces and re-configurable intelligent surfaces) [7], [26], [27]: An IRS is a software-defined surface with low cost and low energy consumption that can be installed in building facades, ceilings, indoor walls, road signs, as well as pedestrians' wearable devices [28]. There are two different kinds of implementation: a large array of reflecting elements whose inter-distance is of the order of half the wavelength and the use of metamaterial elements whose size and inter-distance is much smaller than the wavelength [27]. As mentioned before, mmWave suffers severer shadowing by obstacles, by re-configuring the phase shifts of elements on the IRS, the blocked signals can be reflected and form virtual LoS links between mobile users and base stations with enhanced transmission signal power.

The technologies mentioned above support the severe path-loss of mmWave links with directional communication beams that are generated from both massive antenna arrays at the terminal and reflection elements on IRSs. Normally, the design of the beamforming matrices to configure directional beams is based on perfect knowledge of channel state information. In SAGIN, large antenna arrays can be installed in satellites[29], [30], UAV relays[31], [32], and base stations[6], [24] across three segments. Moreover, as a compensating for the vulnerability of mmWave transmission, the deployment of IRSs in the ground segment is inevitable [28]. Therefore, due to the large number of antenna/reflection elements and the dynamic propagation environment with satellite orbiting, UAV navigation, and ground user movement in SAGIN, it is even challenging to acquire precise channel state information [33].

2.2 Channel Estimation in mmWave Communication

2.2.1 Conventional mmWave Channel Model

In a conventional mmWave MIMO system[34], the BS with N_{BS} antenna elements communicates with a mobile station (MS) with N_{MS} antenna elements. The channel model of each MS for the k^{th} subcarrier can be written as

$$\boldsymbol{H}[k] = \sqrt{N_{\mathrm{BS}} N_{\mathrm{MS}}} \sum_{c=0}^{N_{\mathrm{cluster}}} \sum_{r=1}^{N_{\mathrm{ray}}} \rho_{c,r} \boldsymbol{a}_{\mathrm{MS}} \left(\theta_{c,r}^{\mathrm{MS}}\right) \boldsymbol{a}_{\mathrm{BS}}^{*} \left(\phi_{c,r}^{\mathrm{BS}}\right) e^{j2\pi f_{c,r} k T_s}, \tag{2.1}$$

where N_{cluster} and N_{ray} denote the number of propagation clusters and the number of rays in each cluster, $\rho_{c,r}$ represents the gain of the r^{th} ray in the c^{th} cluster, $f_{c,r}$ and T_s denote the Doppler shift and sampling period, respectively. Besides, $\boldsymbol{a}_{\mathrm{BS}}\left(\phi_{c,r}^{\mathrm{BS}}\right)$ and $\boldsymbol{a}_{\mathrm{MS}}\left(\theta_{c,r}^{\mathrm{MS}}\right)$ represent the receive and transmit array response vectors, where ϕ and θ are the azimuth of angles of arrival (AoAs) and departure (AoDs), respectively.

The channel model for MS can be briefly written as

$$\boldsymbol{H} = \boldsymbol{A}_{\mathrm{BS}} \left(\mathrm{diag}\left(\boldsymbol{g}\right)\right) \left(\boldsymbol{A}_{\mathrm{MS}}\right)^{*}, \tag{2.2}$$

where $\mathrm{diag}\left(\boldsymbol{g}\right) \in \mathbb{C}^{L \times L}$ is a diagonal matrix of the channel gains, which means there are $L = N_{\mathrm{cluster}} \times N_{\mathrm{ray}}$ channel paths between the MS and the BS. The channel gain of the MS incorporating the effect of the Doppler shift vector is

$$\boldsymbol{g} = \left[g_{1,1}, ..., g_{1,N_{\mathrm{ray}}}, ..., g_{N_{\mathrm{cluster}},1} ... g_{N_{\mathrm{cluster}},N_{\mathrm{ray}}}\right]^{\mathrm{T}}, \tag{2.3}$$

with time-varying Doppler shift gain as $g_{c,r} = \rho_{c,r} e^{j2\pi f_{c,r} k T_s}$, $\boldsymbol{A}_{\mathrm{BS}} \in \mathbb{C}^{N_{\mathrm{BS}} \times L}$ is the AoA vector at the BS side, $\boldsymbol{A}_{\mathrm{MS}} \in \mathbb{C}^{N_{\mathrm{MS}} \times L}$ is the AoD vector at MS. Here, $\boldsymbol{A}_{\mathrm{BS}} \in \mathbb{C}^{N_{\mathrm{BS}} \times L}$ is defined as

$$\boldsymbol{A}_{\mathrm{BS}} = \left[\boldsymbol{a}_{\mathrm{BS}}\left(\phi_{1,1}\right), ..., \boldsymbol{a}_{\mathrm{BS}}\left(\phi_{1,N_{\mathrm{ray}}}\right), ..., \boldsymbol{a}_{\mathrm{BS}}\left(\phi_{N_{\mathrm{cluster}},N_{\mathrm{ray}}}\right)\right] \tag{2.4}$$

and $\boldsymbol{A}_{\mathrm{MS}}$ can be written in a similar way. As we take the uniform linear array (ULA) into consideration and the array response vector at BS can be written as

$$\boldsymbol{a}_{\mathrm{BS}}\left(\phi_{c,r}\right) = \frac{1}{\sqrt{N_{\mathrm{BS}}}} \left[1, ..., e^{j\frac{2\pi}{\lambda} d \sin \phi_{\mathrm{c,r}}}, ..., e^{j\frac{2\pi}{\lambda} d (N_{\mathrm{BS}}-1) \mathrm{sin} \phi_{\mathrm{c,r}}}\right]^{\mathrm{T}}, \tag{2.5}$$

where ϕ is azimuth of AoA. This model only consider the quantized on-grid angles.

Due to spatial channel sparsity, the channel model can be extended to a virtual model and then processed by compressive sensing techniques, which is presented as

$$\boldsymbol{H} = \tilde{\boldsymbol{A}}_{\mathrm{BS}} \left(\mathrm{diag}\left(\tilde{\boldsymbol{g}}\right)\right) \left(\tilde{\boldsymbol{A}}_{\mathrm{MS}}\right)^{*}, \tag{2.6}$$

where $\mathrm{diag}\left(\tilde{\boldsymbol{g}}\right) \in \mathbb{C}^{D \times D}$ is a sparse matrix with the number of non-zero elements as L, $\tilde{\boldsymbol{A}}_{\mathrm{BS}} \in \mathbb{C}^{N_{\mathrm{BS}} \times D}$, $\tilde{\boldsymbol{A}}_{\mathrm{MS}} \in \mathbb{C}^{N_{\mathrm{MS}} \times D}$ are the array response vectors dictionary matrices with spatial grids D at BS and MS, respectively.

2.2.2 Compressive Sensing Based Approaches

In mmWave wireless communication systems that have a large number of array elements installed at the antenna of base stations, due to the limited number of scatterers in the environment, the channel can be quite sparse under appropriate spatial domain [34]. Hence, compared to the traditional Nyquist sampling theorem, compressive sensing (CS) techniques have been widely leveraged for mmWave channel estimation to reduce the number of measurements.

Mathematically, CS sampling process can be written as

$$\boldsymbol{y} = \boldsymbol{\Psi}\boldsymbol{h} + \boldsymbol{n}, \tag{2.7}$$

where $\boldsymbol{y} \in \mathbb{C}^{N\times 1}$ denotes the measurement vector, $\boldsymbol{h} \in \mathbb{C}^{M\times 1}$ denotes the signal with length $M \gg N$, $\boldsymbol{\Psi} \in \mathbb{C}^{N\times M}$ represents the linear transform of signal $\boldsymbol{h}$, and $\boldsymbol{n} \in \mathbb{C}^{N\times 1}$ is the noise. The signal $\boldsymbol{h}$ can be written as a sparse format with a suitable basis, i.e.,

$$\boldsymbol{h} = \boldsymbol{A}\boldsymbol{g}, \tag{2.8}$$

where $\boldsymbol{A} \in \mathbb{C}^{M\times M}$ is the basis matrix, and $\boldsymbol{g} \in \mathbb{C}^{M\times 1}$ is the sparse representation of $\boldsymbol{h}$ in the $\boldsymbol{A}$ domain. The sparse $\boldsymbol{g}$ has the non-zero support $P = \{m : \boldsymbol{g}[m] \neq 0\}$ with $|P| \ll M$. Hence, by substituting $\boldsymbol{h}$ into the linear measurement process in (2.8), the measurement vector $\boldsymbol{y}$ in (2.7) can be formulated as

$$\boldsymbol{y} = \underbrace{\boldsymbol{\Psi}\boldsymbol{A}}_{\boldsymbol{\Phi}}\boldsymbol{g} + \boldsymbol{n}, \tag{2.9}$$

with $\boldsymbol{\Phi} \in \mathbb{C}^{N\times M}$ as the measurement matrix. By leveraging CS algorithm, the high dimensional sparse signal $\boldsymbol{g}$ can be recovered from significantly smaller size of measurements vector $\boldsymbol{y}$ based on the model in (2.9) and the known measurement matrix $\boldsymbol{\Phi}$.

By observing the signal model in (2.9), there are two aspects that can affect the recovery accuracy of the CS algorithms: the design of the measurement matrix and the sparse structure of the signal [35]. Conventionally, different CS algorithms and different applications have their unique requirements for the measurement matrix. Moreover, in mmWave wireless communication systems, the physical scattering structure of the propagation environment has a burst structure of the channel support in angular domain [36] and a clustered structure in the spatial domain [37]. Therefore, in mmWave channel estimation, it is necessary to design efficient CS algorithms based on the requirement of measurement matrix and specific sparsity structures under various applications and scenarios.

For mmWave wireless communications channel estimation, the measurement vector $\boldsymbol{y}$ in (2.9) is the received signal, the sparse signal $\boldsymbol{g}$ is the sparse channel information in an appropriate domain that needed to be estimated. Several common CS methods that are widely applied for mmWave channel estimation are introduced as follows.

2.2.2.1 Orthogonal Matching Pursuit

Greedy approaches can iteratively improve the estimation by choosing the column of the measurement matrix that has the most correlation with the residual. As the basic and most commonly adopted greedy algorithm, orthogonal matching pursuit (OMP) implements the heuristic by incrementally pursuing a new support candidate and finding the optimal solution that describes the detected supports [38].

Algorithm 2.1: OMP based channel estimation

```
Input: $\boldsymbol{\Phi}$, received signal $\boldsymbol{y}$, and activated path number $P$.
Output: Recovered path gain $\boldsymbol{g}$.
Initialization: $\boldsymbol{g} \leftarrow \emptyset$, $\mathcal{S} \leftarrow \emptyset$, $\boldsymbol{r} \leftarrow \boldsymbol{y}$.
for $iter = 1 : P$ do
    $j \leftarrow \arg\max_i \|[\boldsymbol{\Phi}]_{:,i}^H \boldsymbol{r}\|_2$
    $\mathcal{S} \leftarrow \mathcal{S} \cup j$.
    $\boldsymbol{r} \leftarrow \left(\boldsymbol{I} - \boldsymbol{\Phi}_{\mathcal{S}} \boldsymbol{\Phi}_{\mathcal{S}}^{\dagger}\right) \boldsymbol{y}$
end for
$\boldsymbol{g} = \boldsymbol{\Phi}_{\mathcal{S}}^{\dagger} \boldsymbol{y}$
```

The details of the OMP based channel estimation are introduced as follows. To begin with, the initialization gives the support $\mathcal{S} = \emptyset$ and recovered path gain $\boldsymbol{g} = \emptyset$. The next iteration updates the residual

$$\boldsymbol{r} = \left(\boldsymbol{I} - \boldsymbol{\Phi}_{\mathcal{S}}\boldsymbol{\Phi}_{\mathcal{S}}^{\dagger}\right)\boldsymbol{y}, \tag{2.10}$$

where $(\cdot)^{\dagger}$ represents pseudo inverse. To be noticed, the sparse level which is the activated total paths P in wireless communications can be defined as the overall iteration. The support position is updated based on the correlation of the sensing matrix and the residual through

$$\mathcal{S} = \mathcal{S} \cup \arg\max_{i} \|[\boldsymbol{\Phi}]_{:,i}^{H}\boldsymbol{r}\|_2, \tag{2.11}$$

where $(\cdot)^{H}$ represents conjugate transpose and $\|\cdot\|$ is the l_2 norm. Finally, the recovered channel gain $\boldsymbol{g}$ can be updated based on least-squares:

$$\boldsymbol{g} = \boldsymbol{\Phi}_{\mathcal{S}}^{\dagger}\boldsymbol{y}. \tag{2.12}$$

The pseudo-code for OMP based channel estimation can be found in **Algorithm 2.1**.

Besides the OMP algorithm, there are many variants of OMP such as compressive sampling matching pursuit (CoSaMP)[39] and subspace pursuit (SP)[40]. The difference between OMP and these two algorithms is that the OMP cumulatively detects the support in the desired signal, while CoSaMP, SP update more columns to the active set, and they allow removal of elements from the active set as well. However, these greedy algorithms do not incorporate more complicated sparsity structures based on various applications and environment.

2.2.2.2 Sparse Bayesian Learning

Sparse Bayesian learning (SBL) as a Sparse-Bayesian-inference-based algorithm has been deployed to solve the sparse channel estimation CS problem[41], [42]. It can deal with the unknown parameters involved in the CS model, such as the uncertain parameters induced by the priors, or the uncertain parameters involved in the measurement matrix.

The details of the SBL based channel estimation are introduced as follows. Given the CS model in (2.9), SBL assumes a Gaussian prior model of the signal to be sensed:

$$p\left(\boldsymbol{g};\gamma\right)=\mathcal{CN}\left(\boldsymbol{g}\mid\mathbf{0},diag\left(\gamma\right)\right), \tag{2.13}$$

where $\boldsymbol{\gamma}$ is defined as a vector of prior hyper-parameters. To be noticed that $\boldsymbol{g}$ is modelled as a statistical signal, where $\mathcal{CN}\left(\boldsymbol{g}\mid\mathbf{0},diag\left(\boldsymbol{\gamma}\right)\right)$ denotes the probability density function of Gaussian distribution of variables $\boldsymbol{g}$ with means $\mathbf{0}$ and covariance matrix $diag\left(\boldsymbol{\gamma}\right)$. By considering the Gaussian noise $\boldsymbol{n}$ which has the variance σ^2, the probability density function of the received measurement vector $\boldsymbol{y}$ conditioned on $\boldsymbol{g}$ can be written as

$$p(\boldsymbol{y}\mid\boldsymbol{g};\sigma^2)=\mathcal{CN}\left(\boldsymbol{y}\mid\boldsymbol{g},\sigma^2\boldsymbol{I}\right). \tag{2.14}$$

The purpose of SBL is to maximize the marginal likelihood (ML) with σ^2 and hyper-parameters $\boldsymbol{\gamma}$. This can be expressed by

$$(\sigma^2_{\mathrm{ML}},\gamma_{\mathrm{ML}})=\arg\max_{\sigma^2,\gamma}p\left(\boldsymbol{y};\gamma,\sigma^2\right)=\arg\max_{\sigma^2,\gamma}\int p(\boldsymbol{y}\mid\boldsymbol{g};\sigma^2)p\left(\boldsymbol{g};\gamma\right)d\boldsymbol{g}. \tag{2.15}$$

Since the above problem cannot be solved in closed form, iterative estimators such as the Expectation-Maximization (EM) based SBL algorithm have to be employed [43]. The basic idea behind the EM framework is to find a convex surrogate function approximating the original objective, and an optimum of the original objective can be found by iteratively optimizing the surrogate function until convergence. The surrogate function is defined as

$$Q(\boldsymbol{\gamma},\boldsymbol{\gamma}^{(k)})=E_{\boldsymbol{g}\mid\boldsymbol{y};\sigma^2,\boldsymbol{\gamma}^{(k)}}p(\boldsymbol{y}\mid\boldsymbol{g};\sigma^2)p(\boldsymbol{g};\boldsymbol{\gamma}), \tag{2.16}$$

where $(\cdot)^{(k)}$ denotes the parameters in the k^{th} iteration and $E_{\boldsymbol{g}\mid\boldsymbol{y};\sigma^2,\boldsymbol{\gamma}^{(k)}}(\cdot)$ denotes the expectation of a random variable with the conditional probability distribution $f(\boldsymbol{g}\mid\boldsymbol{y};\sigma^2,\boldsymbol{\gamma}^{(k)})$. Based on the Bayesian rule, the posterior probability term in $p(\boldsymbol{g}\mid\boldsymbol{y};\sigma^2,\boldsymbol{\gamma}^{(k)})$

is accounted by

$$p(\boldsymbol{g} \mid \boldsymbol{y}; \sigma^2, \boldsymbol{\gamma}^{(k)}) = \frac{p(\boldsymbol{y} \mid \boldsymbol{g}; \sigma^2) p(\boldsymbol{g}; \boldsymbol{\gamma})}{p(\boldsymbol{y}; \boldsymbol{\gamma},^2\ )} = \mathcal{CN}(\boldsymbol{y} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}), \tag{2.17}$$

where $\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \boldsymbol{\Phi}^H \boldsymbol{y}$, and $\boldsymbol{\Sigma} = \left(\sigma^{-2} \boldsymbol{\Phi}^H \boldsymbol{\Phi} + \boldsymbol{\Gamma}^{-1}\right)^{-1}$ with $\boldsymbol{\Gamma} = diag\,(\boldsymbol{\gamma})$ [43]. Then in maximization step, $\sigma^{2(k+1)}, \boldsymbol{\gamma}^{(k+1)}$ are updated by maximizing the surrogate function $Q(\boldsymbol{\gamma}, \boldsymbol{\gamma}^{(k)})$. Details can be expressed as

$$\begin{aligned} \gamma_m^{(k+1)} &= \boldsymbol{\Sigma}_{m,m}^{(k)} + \mu_m^{2(k)}, \\ \sigma^{2(k+1)} &= \frac{\left\| \boldsymbol{y}_v - \boldsymbol{\Phi} \boldsymbol{\mu} + \left(\sigma^2\right)^{(k)} \left[M - \operatorname{tr}(\boldsymbol{\Sigma} \boldsymbol{\Sigma}_0)\right] \right\|}{N}, \end{aligned} \tag{2.18}$$

where $\boldsymbol{\Sigma}_{m,m}^{(k)}$ is the m^{th} diagonal element of $\boldsymbol{\Sigma}^{(k)}$, and $\mu_m^{2(k)}$ is the m^{th} element of $\boldsymbol{\mu}^{2(k)}$.

Upon convergence, $\boldsymbol{g}$ is then obtained as the expectation of the posterior distribution as

$$\boldsymbol{g} = \boldsymbol{\mu}_{\mathrm{ML}} = \left(\boldsymbol{\Phi}^H \boldsymbol{\Phi} + \sigma_{\mathrm{ML}}^2 diag(\gamma_{\mathrm{ML}})\right) \boldsymbol{\Phi}^H \boldsymbol{y}. \tag{2.19}$$

The pseudo-code for SBL based channel estimation can be found in **Algorithm 2.2**.

Although SBL is proved in general settings of numerical simulation to solve the CS problem with superior recovery performance, the limitations of the SBL are that the two-layer hierarchical prior can not handle more complicated sparse priors that may occur in practice, such as the Markov tree priors or the hidden Markov priors.

2.2.2.3 Approximate Message Passing

Approximate message passing (AMP) [44] applies Gaussian and quadratic approximations at the asymptotic region (i.e., $M \to \infty$) to loopy belief propagation, which is fast and highly accurate and admits to rigorous analysis. Because of its fast convergence, the iterative AMP algorithm has been used to recover the CS problem with low computational complexity. The detailed technical explanation can refer to [44].

Algorithm 2.2: SBL based channel estimation

```
Input: $\boldsymbol{\Phi}$, dictionary length $M$, maximum iteration $K$, received signal $\boldsymbol{y}$, and stop criteria $\epsilon$.
Output: $\boldsymbol{g}$
Input $\boldsymbol{y}$
while ($\Delta Q > \epsilon$ and $k < K$) do
E-Step:
    $\boldsymbol{\mu}^{(k)} \leftarrow \sigma^{-2}\boldsymbol{\Sigma}\boldsymbol{\Phi}^H\boldsymbol{y}$
    $\Sigma^{(k)} \leftarrow \left(\left(\sigma^{-2}\boldsymbol{\Phi}^H\right)\boldsymbol{\Phi}^H + \boldsymbol{\Gamma}^{-1}\right)^{-1}$
M-Step:
    $\gamma_m^{(k+1)} \leftarrow \boldsymbol{\Sigma}_{m,m}^{(k)} + \mu_m^{2(k)}$
    $\Delta Q \leftarrow Q - Q\text{old}$
end while
$\boldsymbol{g} \leftarrow \boldsymbol{\mu}$
```

Algorithm 2.3: AMP based channel estimation

```
Input: $\boldsymbol{\Phi}$, received signal $\boldsymbol{y}$, and total number of iterations $K$.
Output: Recovered path gain $\boldsymbol{g}$.
1: Initialization: $\boldsymbol{g}_0 \leftarrow \emptyset$, $\boldsymbol{v}_{-1} \leftarrow \boldsymbol{0}$, $b_0 \leftarrow 0$,$c_0 \leftarrow 0$.
2: for $k = 0 : (K-1)$ do
3:     $\boldsymbol{v}_t \leftarrow \boldsymbol{y} - \boldsymbol{\Phi}\boldsymbol{g}_k + b_k\boldsymbol{v}_{k-1} + c_t\boldsymbol{v}_{k-1}^H$
4:     $\sigma_k^2 \leftarrow \frac{1}{N}\|\boldsymbol{v}_k\|_2^2$.
5:     $\boldsymbol{r}_t \leftarrow \boldsymbol{g}_t + \boldsymbol{\Phi}^H\boldsymbol{v}_t$
6:     $\boldsymbol{g}_{k+1} \leftarrow \boldsymbol{\eta}_{st}(\boldsymbol{r}_t; \lambda_t, \sigma_t^2)$
7:     $b_{k+1} = \frac{1}{N}\Sigma_{i=1}^M \frac{\partial\eta_{st}(r_{k,i};\lambda_t,\sigma_t^2)}{\partial r_{k,i}}$
8:     $c_{k+1} = \frac{1}{N}\Sigma_{i=1}^M \frac{\partial\eta_{st}(r_{k,i};\lambda_t,\sigma_t^2)}{\partial r_{k,i}^H}$
9: end for
10: $\boldsymbol{g} = \boldsymbol{g}_K$
```

The pseudo-code details of the AMP based channel estimation are shown in **Algorithm 2.3**. The term $b_k\boldsymbol{v}_{k-1}$ and the term $c_k\boldsymbol{v}_{k-1}^H$ in step 3 are called Onsager Correction, which are introduced into the AMP algorithm to accelerate the convergence [45]. The most important step of this algorithm is step 6, in which the sparse information $\boldsymbol{g}_{k+1}$ is updated through the soft threshold shrinkage function $\boldsymbol{\eta}_{st}$. The shrinkage function is a non-linear element-wise operation that considers the sparsity structure of the vector $\boldsymbol{g}$ to approach a sparser result. In Step 7 and Step 8, the element-wise derivatives of the shrinkage function$\boldsymbol{\eta}_{st}$ at the input vector $\boldsymbol{r}_t$ and its conjure vector $\boldsymbol{r}_t^H$ are respectively calculated to obtain b_{t+1} and c_{t+1}.

However, with different applications and scenarios of the channel estimation, the general AMP algorithm cannot fully exploit the prior distribution of the signal. By exploring the various sparsity structure, the estimation can be more accurate.

2.2.3 Deep Neural Network Based Approach

DL has been successfully applied in a wide range of areas with significant performance improvement in different fields, such as, computer vision, natural language processing, speech recognition, and so on [46]. Generally, DNN is a deeper version of artificial neural networks (ANNs) with multiple layers (more than three hidden layers). The structure of the network of ANN and DNN can be found in Fig.2.4.

In both ANN and DNN, each layer of the network consists of multiple neurons, each of which has an output that is a nonlinear function such as, Sigmoid function or the Relu function. For DNN, the output of the overall network $\boldsymbol{o}$ is a cascade of the nonlinear transformation of the input data $\boldsymbol{i}$, mathematically as

$$\boldsymbol{o} = f(\boldsymbol{i}, \boldsymbol{\theta}) = f_{\mathcal{L}-1}(f_{\mathcal{L}-2}...f_1(\boldsymbol{i})), \tag{2.20}$$

where f is the nonlinear function, $\mathcal{L}$ denotes the total number of layers, and $\boldsymbol{\theta}$ represents the weights and bias in DNN.

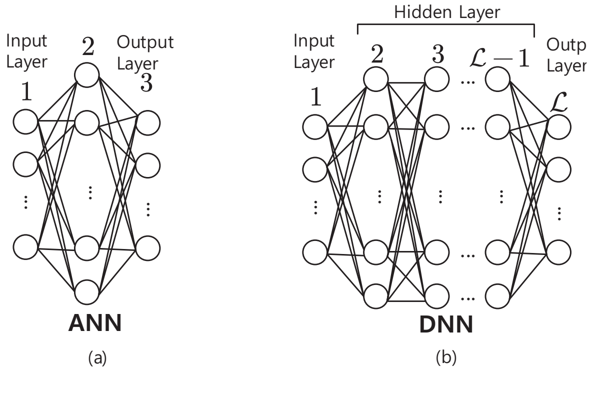


Figure 2.4: The structure of (a) ANN, (b)DNN.

During the training phase, the model is trained to minimize the difference between the output of the neural network and the supervision data, and the weights and bias can be learnt through back propagation. For channel estimation, DNN can be regarded as a black box, and the sparsity characteristics of the complicated mmWave channel information can be acquired through the training procedure [47], [48].

2.3 Channel Tracking in mmWave Communication

In SAGIN, the mobility of ground users and vehicular systems, the navigation of UAVs, and the orbiting satellites all need accurate alignment of narrow mmWave beams to avoid severe loss of signal power through transmission. Moreover, to establish a stable link, it is non-trivial to obtain the real-time channel information. Different from the static channel estimation, channel tracking can provide more accurate estimation in the dynamic environment as it considers both the correlation across time sequence and the observation at the current stage. However, this imposes a formidable challenge on modelling and tracking the time-varying channel with small training overheads.

2.3.1 Prior-Based CS Approaches

In sparse channel estimation, consecutive frames tend to share some multi-paths due to the slowly varying propagation environment between base stations and users [49]. As such, considering the temporal correlation as prior information is the main idea behind channel tracking. Utilizing temporal correlation as extra information for dynamic channel tracking together with conventional CS algorithms can be a robust solution. For example, in [49]–[51], both the prior information obtained from the temporal correlation and the channel sparsity have been exploited to design the prior-aided sparse channel tracking schemes. Moreover, [52], [53] considered the dynamic evolution of the channel, however, they failed to study the dynamic spatial sparsity structure while the channel evolves through time. Although both prior information obtained from the temporal correlation and the channel sparsity have been exploited to design the prior-aided sparse channel tracking schemes in [49]–[53], the spatial sparsity structure and the probabilistic temporal dependency of the channels jointly to track the dynamic channels are not considered.

2.3.2 Recurrent Neural Network

As recurrent neural network (RNN) is a class of machine learning that has been widely used for time-series tracking and prediction [54]. Different from DNN which only learns the network parameters from training data, RNN can utilize the memory of past states of the sequence input data. Hence, besides prior-based CS based approaches, such machine learning algorithm can also be utilized for channel tracking. The structure of the RNN for channel prediction can be found in Fig. 2.5 [55].

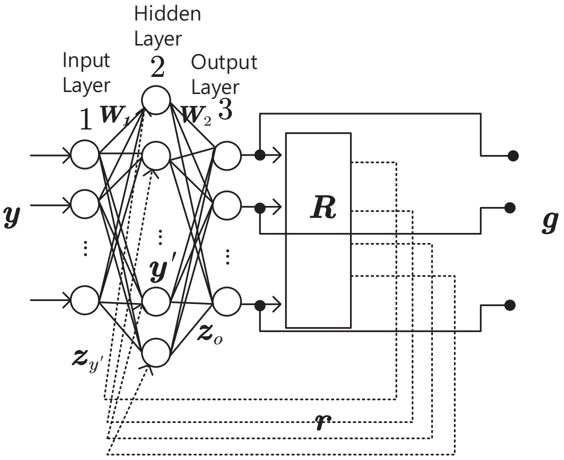


Figure 2.5: The structure of RNN.

Basically, a simple RNN consists of three layers: an input layer, one hidden layer, and an output layer. Each neuron connection between the activation of a neuron in depredecessor layer and the input of a neuron in the successor layer is assigned a weight. Hence there are two weight matrix $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ as shown in Fig. 2.5. The input for the hidden layer at time slot t can be expressed as

$$\boldsymbol{z}_{y'}(t) = \boldsymbol{W}_1\boldsymbol{y}(t) + \boldsymbol{r}(t) + \boldsymbol{b}_{y'}, \tag{2.21}$$

with $\boldsymbol{b}_{y'}$ denotes the bias vector in the hidden layer. The recurrent component $\boldsymbol{r}(t) = \boldsymbol{R}\boldsymbol{g}(t-1)$, which denotes the mapping of the output at the previous time slot with matrix $\boldsymbol{R}$. The activation vector of the hidden layer is $\boldsymbol{y}^{'}(t) = f(\boldsymbol{z}_{y'}(t))$, with $f(\cdot)$ as the activation function, normally Sigmoid function. Then, the input for the output layer is $\boldsymbol{z}_o(t) = \boldsymbol{W}_2\boldsymbol{y}^{'}(t) + \boldsymbol{b}_o$, with $\boldsymbol{b}_o$ as the biases in the output layer. Hence the result of the output can be written as $\boldsymbol{g}(t) = f(\boldsymbol{z}_o(t))$.

Similar to the operation of DNN, back propagation in combination with gradient descent learning is deployed for the training of the RNN model to update the weights and biases iteratively. Once the training process completed, the trained network can be used to process upcoming sequences.

2.4 Summary

This chapter presents the overview structure of the SAGIN and the importance of mmWave channel estimation in SAGIN. Some fundamental mmWave channel estimation algorithms are introduced, such as CS-based algorithms: OMP, SBL, AMP, and DNN. Moreover, some mmWave channel tracking algorithms: Prior-based CS algorithms and RNN are also introduced. Based on different links across three segments of SAGIN, different channel tracking algorithms can be proposed accordingly. For the links with a clearer sparse structure, temporal correlation and the dynamic sparsity structure of the information can be jointly considered. For the links with more complex structure, the DL-based approach can be considered thanks to its powerful learning ability and lower computation complexity.

Chapter 3

Deep Learning Based 3D Channel Tracking for IRS-assisted Air-Ground Communications

Millimeter wave (mmWave) communications have drawn significant research efforts for supporting high data rates and capacity to resolve the spectrum crunch crisis caused by emerging massive connectives.[56]. However, the small wavelength of mmWave with the high penetration loss mainly rely on the line of sight (LoS) communication paths [57]. The mobility unmanned aerial vehicles (UAVs) acting as aerial base stations can provide a solution in mmWave communications on serving wider coverage, supporting reliable connections, and providing energy efficient communications [58], [59]. The flexibility of UAVs has led to plenty of applications such as security surveillance, real-time monitoring, rescue, and emergency communications [60]. Although UAV-aided systems are considered as promising techniques for future wireless communications for smart city, the complex urban environment poses potential blockage problem on LoS links between navigation UAVs and ground users [61].

To address the blockage problem, intelligent reflection surfaces (IRSs) can be installed to assist the UAV-aided system to offer ubiquitous communication services [61]–[64]. IRSs that can construct virtual LoS paths to enhance the quality and coverage of wireless propagation has become an invaluable solution on overcoming signal pathloss of mmWave transmission and on securing communications [65], [66]. This is because that the low-cost IRS can intelligently adjust its phase shifts to steer signal power towards targeted directions and reduce information leakage. Additionally, deploying IRSs in the UAV-aided system can further help with the time- and energy- consuming problem caused by UAV navigation when some users are far away [62].

Moreover, to achieve high quality communications, it is indispensable to acquire accurate channel state information (CSI) for effective transmissions. Compared to the CSI estimation in the conventional communication systems, channel tracking in IRS-assisted UAV systems are more challenging in a time-varying environment with small overheads. Since IRS is unable to perform signal processing, the large number of reflecting elements of IRS, and the mobility of UAV and users all leads to high complexity. In this chapter, a DL-based channel tracking algorithm is proposed in IRS-assisted air-ground communications. This chapter is structured as follows. Section 3.1 outlines the introduction and contributions of the work in this chapter. The mmWave IRS-assisted air-ground wireless communication system is introduced in Section 3.2. In Section 3.3, a DL-based channel tracking algorithm is proposed. Then, the proposed channel tracking performance over a time-variant channel is quantified in Section 3.4. Finally, conclusions are drawn in Section 3.5.

3.1 Introduction

3.1.1 Related Work

Conventional channel estimation algorithms such as minimum mean square error (MMSE) and LS are not suitable for mmWave communication with massive MIMO system [67]. This is because of the substantial training overheads needed for accurate channel estimations when the total number of transmit antennas is very large. By considering the approximate sparse characteristic of the channel information in time, frequency, angular and Doppler domains, compressive sensing (CS) based estimation techniques, such as orthogonal matching pursuit (OMP) and compressive sampling matching pursuit (CoSaMP), have been broadly investigated and developed with further reduced pilot overheads [68]. However, since the demands and applications burst and more advanced technologies are installed in wireless communication systems, it is critical to develop higher dimensional channel estimation and tracking algorithms based on more complex channel models in different scenarios [69].

Compared to the terrestrial base stations, the UAV-based aerial base stations' advantage is the adjustable altitude and mobility. This flexibility enables UAVs to play a vital role in wireless communications. Particularly, UAVs can assit the communication devices such as sensors and monitors that are unable to transmit over a long distance due to the energy constraints [70]. Although the existing statistical MIMO channel models are suitable for most communication environment, the unique features, such as the three-dimensional(3D) space movement of UAVs at high altitude can not be captured [32]. Hence, recent work [32], [71]–[73] proposed geometry-based UAV channel models for air-to-ground communication environment. The first work that considered the UAV's mobility from the moving velocities to directions in the vertical plane is reported in [32]. Moreover, authors in [71]–[73] derived the time-varying parameters of the AoAs and AoDs to describe the nonstationarity of the dynamic channel caused by the movement of the UAV. Differently, authors in [71] ignored the scatterers' movement and contained a LoS link and only one non-LoS link. Authors in [72] investigated the cluster-based multiple propagation paths. Besides, the initial azimuth and elevation AoAs and AoDs were further considered in [73].

Apart from UAVs aided communications that can provide effective communication services, the deployment of IRSs in wireless communications has attracted remarkable attention [74]. The IRS that consists of an array of reflecting elements with low cost and low energy consumption, can be installed in building facades, ceilings, indoor walls, road signs, as well as pedestrians' wearable devices[28]. By re-configuring the phase shifts of elements on the IRS, the reflected signals can be propagated. They can form virtual LoS links between mobile users and base stations with enhanced transmission signal power[75], [76]. Also, an IRS-aided system can improve spectral efficiency and enhance the communication coverage capability [77]. However, the promised communication performance brought by an IRS relies on good knowledge of CSI. Hence, obtaining accurate CSI with low complexity channel estimation and tracking approaches is indispensable and more challenging for practical IRS implementation [28].

Recently, channel estimation in IRS-assisted systems has been widely studied based on both conventional[78]–[82] and specific [83], [84] communication scenarios. For conventional scenarios, [78] proposed a three-phase pilot-based channel estimation framework for IRS-assisted MU communications, in which different links are estimated in different phases, separately. The authors in [79] developed two customized schemes, simultaneous-user channel estimation and sequential-user channel estimation, by separately considering different dominant links in the real-world scenario. Moreover, [80]–[82] all considered the conventional scenario but with part of the reflecting elements switched on in IRS. For some specific communication scenarios, [83] considered an indoor IRS-assisted channel model which assumed that the LoS path being blocked by the obstacles and the virtual LoS path reflected by IRSs as the major link. Double-IRS cooperatively aided MIMO system that has cascaded single-reflection and double-reflection links have been further exploited in [84].

Data-driven deep learning (DL) techniques have shown remarkable effectiveness to revolutionize communication systems [28]. In contrast to the traditional model driven channel estimation approaches, DL with powerful learning capabilities can be adopted for estimating the CSI that has beyond linear correlations in emerging advanced communication environment [85]–[89]. A deep residual learning method for channel estimation in IRS-assisted multi-user communications by considering the channel model with both LoS links and virtual LoS links is proposed in[85]. The authors in [86] proposed a learned denoising-based approximate message passing network to learn the channel structure for beamspace mmWave massive MIMO systems. Similarly, a deep denoising neural network-assisted CS channel estimation framework for mmWave IRS systems with reduced training overheads has been proposed in [87]. Moreover, authors of [88] developed a DL-based solution that enables IRS to learn to interact with the signal optimally. [89] proposed a twin convolutional neural network (CNN) architecture to estimate both LoS and cascaded non-LoS paths in IRS-assisted mmWave massive MIMO systems.

3.1.2 Motivation and Contribution

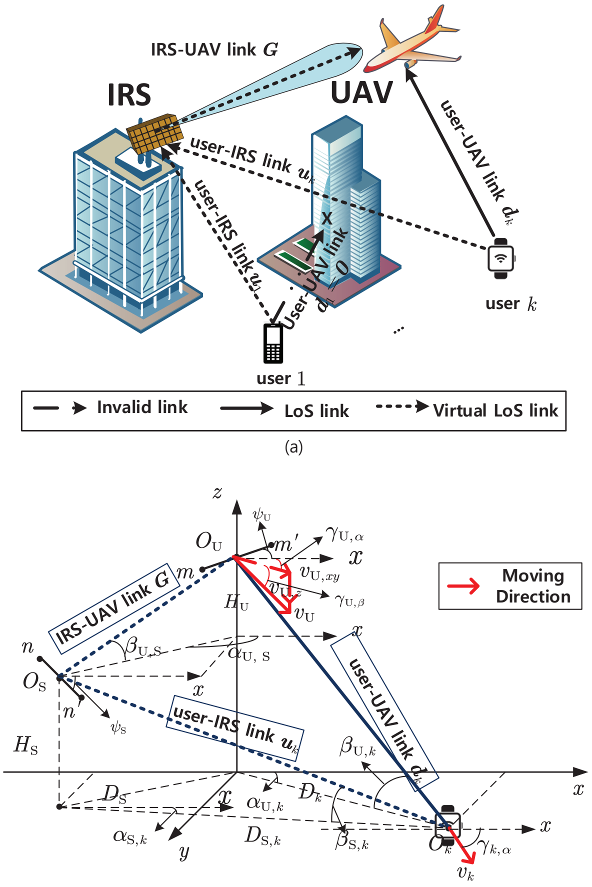
Although [32], [71]–[73] studied the UAVs aided MIMO system, the carrier frequencies were considered around 5.9GHz or below. With a considerable amount of devices being brought online, deploying mmWave frequency above 28GHz for wireless communication is the inevitable trend [23]. When considering mmWave UAVs aided communication, the LoS channel can be easily blocked by the obstacles and can suffer more vital penetration loss. Hence, it is necessary to leverage IRSs to reinforce the communication quality and extend coverage in the mmWave UAVs aided communication networks. Moreover, the environment and the mobile users are assumed to be static for simplicity in the existing work [78]–[89] for channel estimation in an IRS-assisted communication system. However, in practice, the mobility of users cannot be ignored. Hence, time-varying channel estimation or channel tracking is a potential research topic but full of challenge. Furthermore, the proposed DL channel estimation techniques in [85]–[89] simply considered off-line training. None of the mentioned work considered the time-sequence of historical information for channel tracking.

For the channel estimation problem in IRS-assisted UAV communications systems, there are mainly three challenges to tackle: good estimation performance, the small pilot overheads, and the time-varying channel caused by the mobility of both UAV and mobile users. Motivated by the aforementioned literature review, we propose a DL-based channel tracking algorithm in mmWave IRS-assisted UAV-enabled communication systems with lower training overheads and significantly improved tracking performance. The main contributions are summarized as follows:

- From our knowledge, this is the first work that develops a 3D geometry-based dynamic channel model in mmWave IRS-assisted UAV-enabled communication system with fixed IRS, navigation UAV, and mobile users. We define the mmWave time-variant channel model consisting of both dynamic LoS link (user-UAV) with blockage parameter and dynamic cascaded virtual LoS link (user-IRS-UAV) with activation parameter. The velocities, the max Doppler effects, the propagation delays and the time delays are specially included in the system model.

- We propose a DL-based channel tracking mechanism to track the time-varying channel in the developed system. The proposed algorithm consists of two modules: channel pre-estimation and channel tracking. The pre-estimation is achieved by using a deep neural network (DNN) to perform off-line training on pre-collected training dataset. The tracking module is designed as a stacked bi-directional long short term memory (Stacked Bi-LSTM) that can track the CSI over a time-varying channel in a data-driven manner. The Stacked Bi-LSTM is constructed by a framework that involves a certain historical trace-back period in the time sequence and bidirectional structure over multiple stacked layers.

- In comparison with the benchmark algorithms, the proposed channel tracking algorithm requires less epochs for the convergence of the loss function during the off-line training phase. Moreover, simulations demonstrate that the proposed scheme (DNN followed by Stacked Bi-LSTM) shows better channel tracking performance with smaller pilot overheads than the benchmark algorithms, and it has comparable complexity.



(a)

(b)

Figure 3.1: (a) The illustration of the IRS-assisted UAV-enabled communication system. (b) The system model of the uplink of the IRS-assisted UAV-enabled single-user communication.

3.2 System Model

This chapter considers the uplink IRS-assisted UAV-enabled multi-user system, which consists of M element antenna array equipped on UAV and N reflecting elements installed on IRS serving K users. The k^{th} user is assumed to be equipped with a single antenna. To enhance the quality of wireless communication services between UAV and mobile users, IRS is considered to be placed at the high-rise building to provide extended coverage and enables less movement of UAV[90], [91]. Important notations are summarized in Table 3-A.

As shown in Fig. 3.1 (a), there are mainly two different kinds of links in the system:

the user-UAV LoS link and the user-IRS-UAV virtual LoS link. The k^{th} user-UAV LoS link at time t is represented by $\boldsymbol{d}_k(t) \in \mathbb{C}^{M\times 1}$. The k^{th} user-IRS-UAV virtual LoS link at time t consists of two LoS sub-links: the user-IRS sub-link $\boldsymbol{u}_k(t) \in \mathbb{C}^{N\times 1}$ and the IRS-UAV sub-link $\boldsymbol{G}(t) \in \mathbb{C}^{M\times N}$. The illustration details of the dynamic 3D IRS-assisted channel model is presented in Fig. 3.1 (b). The moving speed of UAV and k^{th} user are represented by v_{U} and v_k.

The LoS link between k^{th} user and UAV at time t is $\boldsymbol{d}_k(t) = [d_{k,m}(t,\tau)]_{M\times 1}$ [71], [73] with

$$\begin{aligned} d_{k,m}(t,\tau) = \mathit{\Omega}_{k,m} e^{-j2\pi f_c \tau_{k,m}} \delta(\tau - \tau_{\mathrm{U},k,\mathrm{LoS}}) \\ e^{j2\pi f_{\mathrm{U}} t [\cos(\alpha_{\mathrm{U},k} - \gamma_{\mathrm{U},\alpha}) \cos\beta_{\mathrm{U},k} \cos\gamma_{\mathrm{U},\beta} + \sin\beta_{\mathrm{U},k} \sin\gamma_{\mathrm{U},\beta}]} \\ e^{j2\pi f_k t [\cos(\alpha_{\mathrm{U},k} - \gamma_{k,\alpha}) \cos\beta_{\mathrm{U},k}]}, \end{aligned} \tag{3.1}$$

where $\mathit{\Omega}_{k,m}$ is the attenuation factor between the m^{th} $(0 \leqslant m \leqslant m^{'} \leqslant M)$[1] antenna element on UAV and the single antenna on k^{th} user, f_c is the carrier frequency, $\tau_{k,m} = L_{k,m}/c$ is the propagation delay of the waves between the m^{th} antenna element on UAV and single antenna on k^{th} user at time t with $L_{k,m}$ as the distance between user and m^{th} antenna element, $\tau_{\mathrm{U},k,\mathrm{LoS}}$ is the time delay of this k^{th} user-UAV LoS link at time t, $f_{\mathrm{U}} = \frac{v_U}{\lambda}$ and $f_k = \frac{v_k}{\lambda}$ are the maximum Doppler frequency caused by the movement of both UAV and k^{th} user, $\alpha_{\mathrm{U},k}$ and $\beta_{\mathrm{U},k}$ are the azimuth and elevation angle between the k^{th} user and UAV, $\gamma_{\mathrm{U},\alpha}$ and $\gamma_{\mathrm{U},\beta}$ are the azimuth and elevation angle of the UAV's moving direction, $\gamma_{k,\alpha}$ is the azimuth angle of the k^{th} user's moving direction associate with the UAV location. The time delay of this LoS link is

$$\tau_{\mathrm{U},k,\mathrm{LoS}} = \frac{D_k}{c_0 \cos\beta_{\mathrm{U},k}}, \tag{3.2}$$

with D_k as the xy-plane antenna center distance between UAV and k^{th} user, c_0 as light

[1] The reason of having the $m^{'th}$ element presented in the illustration Fig. 3.1 is to show the antenna flat angle ψ_{U}. Similar to the existence of the $n^{'th}$ reflection element on IRS, the physical flat angle is represented by ψ_{S}.

speed.

To be noticed, for vitual LoS link, the main difference between $d_{k,m}(t,\tau)$ in (3.1) and each LoS sub-link is that the IRS is installed at a fixed place without moving at all i.e., $f_{\mathrm{S}}=0$. Hence, the k^{th} user-IRS LoS sub-link at time t is $\boldsymbol{u}_k(t)=[u_{k,n}(t,\tau)]_{N\times 1}$ with

$$\begin{aligned} u_{k,n}\left(t,\tau\right) = \mathit{\Omega}_{k,n} e^{-j2\pi f_c \tau_{k,n}} \delta\left(\tau-\tau_{\mathrm{S},k,\mathrm{LoS}}\right) \\ e^{j2\pi f_k t\left[\cos(\alpha_{\mathrm{S},k}-\gamma_{k,\alpha})\cos\beta_{\mathrm{S},k}\right]}, \end{aligned} \tag{3.3}$$

where $\mathit{\Omega}_{k,n}$ is the attenuation factor between the n^{th} $(0 \leqslant n \leqslant n^{'} \leqslant N)$ reflecting element on IRS and the single antenna on k^{th} user, $\tau_{\mathrm{S},k,\mathrm{LoS}}$ is the time delay of this k^{th} user-IRS LoS sub-link at time t, $\alpha_{\mathrm{S},k}$ and $\beta_{\mathrm{S},k}$ are the relative azimuth and elevation direction between k^{th} user and IRS, $\gamma_{k,\alpha}$ is the azimuth angle of the k^{th} user's moving direction, $\tau_{k,n}=L_{k,n}/c$ is the propagation delay of the waves between the n^{th} antenna element on IRS and single antenna on k^{th} user at time t with $L_{k,n}$ as the distance between user and n^{th} element on IRS.

The IRS-UAV LoS sub-link at time t is $\boldsymbol{G}_k(t)=[g_{m,n}(t,\tau)]_{M\times N}$ with

$$\begin{aligned} g_{m,n}(t,\tau) = \mathit{\Omega}_{m,n} e^{-j2\pi f_c \tau_{m,n}} \delta(\tau-\tau_{\mathrm{U,S,LoS}}) \\ e^{j2\pi f_{\mathrm{U}} t[\cos(\alpha_{\mathrm{U,S}}-\gamma_{\mathrm{U},\alpha})\cos\beta_{\mathrm{U,S}}\cos\gamma_{\mathrm{U},\beta}+\sin\beta_{\mathrm{U,S}}\sin\gamma_{\mathrm{U},\beta}]}, \end{aligned} \tag{3.4}$$

where $\mathit{\Omega}_{m,n}$ is the attenuation factor between the m^{th} antenna element on UAV and the n^{th} reflecting element on IRS, $\tau_{\mathrm{U,S,LoS}}$ is the time delay of this IRS-UAV LoS sub-link at time t, $\alpha_{\mathrm{U,S}}$ and $\beta_{\mathrm{U,S}}$ are the relative azimuth and elevation direction between UAV and IRS, $\tau_{m,n}=L_{m,n}/c_0$ is the propagation delay of the waves between the m^{th} antenna element on UAV and n^{th} element on IRS with $L_{m,n}$ as the distance.

The corresponding time delay for two sub-links at time t are

$$\tau_{\mathrm{S},k,\mathrm{LoS}} = \frac{D_{\mathrm{S},k}}{c_0 \cos\beta_{\mathrm{S},k}} \tag{3.5}$$

Table 3-A: Table of Important Symbols

Symbols	**Explanations**
$M, m, m^{'}$	Total number and index of antenna elements of UAV
$N, n, n^{'}$	Total number and index of elements of IRS
N_p	Total number of training pilot
$\alpha_{\mathrm{U},k}, \beta_{\mathrm{U},k}$	Azimuth and elevation AoAs/AoDs of user-UAV link
$\alpha_{\mathrm{S},k}, \beta_{\mathrm{S},k}$	Azimuth and elevation AoAs/AoDs of user-IRS link
$\alpha_{\mathrm{U,\ S}}, \beta_{\mathrm{U,\ S}}$	Azimuth and elevation AoA/AoD of IRS-UAV link
$\gamma_{\mathrm{U},\alpha}, \gamma_{\mathrm{U},\beta}$	Azimuth and elevation angle of UAV movement
$\gamma_{k,\alpha}$	Angle of k^{th} user movement
$\beta_{t,n}$	Amplitude at n^{th} element of IRS at time slot t
$\phi_{t,n}$	Phase shift at n^{th} element of IRS at time slot t
v_{U}	UAV movement speed
$v_{\mathrm{U},xy}, v_{\mathrm{U},z}$	UAV speed's horizontal and perpendicular components
v_k	Speed of k^{th} user movement
f_{U}	Maximum Doppler frequency of UAV
f_k	Maximum Doppler frequency of k^{th} user
ψ_{U}	Antenna physical flat angle on UAV
ψ_{S}	Reflection elements' physical flat angle on IRS
$\tau_{k,m}$	The propagation delay of the wave between UAV m^{th} antenna element and k^{th} user single antenna at time t
$\tau_{k,n}$	The propagation delay of the wave between IRS n^{th} reflection element and k^{th} user single antenna at time t
$\tau_{m,n}$	The propagation delay of the wave between UAV m^{th} antenna element and IRS n^{th} reflection element at time t
$\tau_{\mathrm{U},k,\mathrm{LoS}}$	The time delay of k^{th} user-UAV LoS link at time t
$\tau_{\mathrm{S},k,\mathrm{LoS}}$	The time delay of k^{th} user-IRS LoS link at time t
$\tau_{\mathrm{U,S,LoS}}$	The time delay of IRS-UAV LoS link at time t
D_k	The xy-plane center distance between UAV and k^{th} user
D_{S}	The xy-plane center distance between IRS and UAV
$D_{\mathrm{S,k}}$	The xy-plane center distance between IRS and k^{th} user
$\varOmega_{k,m}$	Attenuation factor between the UAV m^{th} antenna element and the k^{th} user single antenna
$\varOmega_{k,n}$	Attenuation factor between the IRS n^{th} reflection element and the k^{th} user single antenna
$\varOmega_{m,n}$	Attenuation factor between the UAV m^{th} antenna element and the IRS n^{th} reflection element
$L_{k,m}$	The distance between the UAV m^{th} antenna element and the k^{th} user single antenna
$L_{k,n}$	The distance between the IRS n^{th} reflection element and the k^{th} user single antenna
$L_{m,n}$	The distance between the UAV m^{th} antenna element and the IRS n^{th} reflection element

and

$$\tau_{\mathrm{U,S,LoS}} = \frac{D_{\mathrm{S}}}{c_0 \cos\beta_{\mathrm{U,S}}}. \tag{3.6}$$

The k^{th} user-IRS-UAV virtual LoS link at time t can be expressed as $\boldsymbol{G}(t)\boldsymbol{\Phi}(t)\boldsymbol{u}_k(t) \in \mathbb{C}^{M\times 1}$, in which $\boldsymbol{\Phi}(t) = \text{diag}(\boldsymbol{\varphi}(t)) \in \mathbb{C}^{N\times N}$ is the phase shift matrix of IRS with $\boldsymbol{\varphi}(t) = [\beta_{t,1}e^{j\phi_{t,1}}, ..., \beta_{t,n}e^{j\phi_{t,n}}, ..., \beta_{t,N}e^{j\phi_{t,N}}]^T \in \mathbb{C}^{N\times 1}$, where $\beta_{t,n} \in [0, 1]$ and $\phi_{t,n} \in [0, 2\pi]$ are the reflection amplitude and phase shift of the subsurface n of IRS at time slot t [83].

Hence, the IRS-assisted UAV uplink at time t is the superimposition of the vitual LoS link and the LoS link, which can be given as

$$\begin{aligned}\dot{\boldsymbol{H}}_k(t) &= \boldsymbol{G}(t)\boldsymbol{\Phi}(t)\boldsymbol{u}_k(t) + \boldsymbol{\eta}_k\boldsymbol{d}_k(t)\\ &= \underbrace{\boldsymbol{G}(t)\text{diag}(\boldsymbol{u}_k(t))}_{\boldsymbol{H}_{\text{C},k}(t)}\boldsymbol{\varphi}(t) + \boldsymbol{\eta}_k\boldsymbol{d}_k(t), k = 1, ...K,\end{aligned} \tag{3.7}$$

where $\boldsymbol{H}_{\text{C},k}(t) \in \mathbb{C}^{M\times N}$ denotes the cascaded k^{th} user-IRS-UAV channel at time slot t, $\boldsymbol{\eta}_k \in \{0, 1\}^{M\times 1}$ is the blockage parameter that is distributed according to a Bernoulli distribution [92] with the blockage probabilities for LoS link is p_k.

The received signal at the UAV from K users at time slot t can be expressed as

$$\begin{aligned}\boldsymbol{Y}(t)' &= \sum_{k=1}^{K}\dot{\boldsymbol{H}}_k(t)\boldsymbol{s}_k + \boldsymbol{N}_k'(t)\\ &= \sum_{k=1}^{K}\underbrace{[\boldsymbol{H}_{C,k}(t), \boldsymbol{\eta}_k\boldsymbol{d}_k(t)]}_{\boldsymbol{H}_k(t)}\underbrace{[\boldsymbol{\varphi}(t); 1]^T}_{\boldsymbol{r}(t)}\boldsymbol{s}_k + \boldsymbol{N}_k'(t),\end{aligned} \tag{3.8}$$

where $\boldsymbol{Y}(t)' \in \mathbb{C}^{M\times N_p}$ is the MU received signal, $\boldsymbol{H}_k(t) \in \mathbb{C}^{M\times(N+1)}$ is the channel matrix that need to be estimated, $\mathbf{r}(t) \in \mathbb{C}^{(N+1)\times 1}$, $\boldsymbol{s}_k \in \mathbb{C}^{1\times N_p}$ is the training pilot with $\boldsymbol{s}_k\boldsymbol{s}_k^H = \mathcal{P}N_p$ and $\boldsymbol{s}_{k_a}\boldsymbol{s}_{k_b} = 0$, where $\mathcal{P}$ is the power of each user, $k_a, k_b \in 1, 2, ..., K$, and $k_a \neq k_b$. $\boldsymbol{N}_k'(t) \sim \mathcal{CN}(0, \sigma^2_{\boldsymbol{N}'}\boldsymbol{I})$ is the AWGN noise matrix.

For MU case, since the pilot sequences of each two users are orthogonal, the received signal vectors for k^{th} user can be decoupled by multiplying a sequence $\boldsymbol{s}_k^H$ as

$$\begin{aligned}\underbrace{\frac{1}{\mathcal{P}N_p}\boldsymbol{Y}(t)^{'}\boldsymbol{s}_k^H}_{\boldsymbol{Y}_k(t)} &= \dot{\boldsymbol{H}}_k(t) + \frac{1}{\mathcal{P}N_p}\boldsymbol{N}_k^{'}(t)\boldsymbol{s}_k^H \\ &= \underbrace{[\boldsymbol{H}_{C,k}(t), \boldsymbol{\eta}_k\boldsymbol{d}_k(t)]}_{\boldsymbol{H}_k(t)}\underbrace{[\boldsymbol{\varphi}(t);1]^T}_{\boldsymbol{r}(t)} + \underbrace{\frac{1}{\mathcal{P}N_p}\boldsymbol{N}_k^{'}(t)\boldsymbol{s}_k^H}_{\boldsymbol{N}_k(t)},\end{aligned} \tag{3.9}$$

where $\boldsymbol{Y}(t)_k \in \mathbb{C}^{M\times N_p}$ is the received signal vector at the UAV from the k^{th} user, $\boldsymbol{H}_k(t) \in \mathbb{C}^{M\times(N+1)}$ is the channel matrix that need to be estimated, $\mathbf{r}(t) \in \mathbb{C}^{(N+1)\times 1}$, $\boldsymbol{N}_k(t) \sim \mathcal{CN}(0, \sigma_{\boldsymbol{N}}^2\boldsymbol{I})$.

The channel $\boldsymbol{H}_k(t)$ is estimated by a channel tracker, denoted by $\mathcal{F}(\cdot)$, which depends on the received signal $\boldsymbol{Y}(t)^{'}$, $\boldsymbol{r}(t)$, and the training pilot $\boldsymbol{s}_k$. The estimated channel $\tilde{\boldsymbol{H}}_k(t) = \mathcal{F}(\boldsymbol{Y}_k(t); \boldsymbol{r}(t), \boldsymbol{s}_k)$. Hence, the pilot-aided channel estimation problem can be written as

$$\begin{aligned}&\min_{\mathcal{F}} \mathbb{E}\left[\|\boldsymbol{H}_k(t) - \tilde{\boldsymbol{H}}_k(t)\|^2\right], \\ &\text{s.t. } \operatorname{tr}\left(\boldsymbol{s}_k\boldsymbol{s}_k^H\right) = E_s,\end{aligned} \tag{3.10}$$

where $E_s = N_pT_sP$ is the energy constraint with N_p as pilot overheads length, T_s as the length of one time slot, and P denotes the transmit power.

To be noticed, because of the mobility of both UAV and k^{th} user, the relative azimuth and elevation angle and the relative distance constantly change through time. We assume the initial position of the UAV is $(x_{\mathrm{U}}, y_{\mathrm{U}}, z_{\mathrm{U}}) = (0, 0, H_{\mathrm{U}})$, k^{th} user is $(x_k, y_k, z_k) = (D_k\cos\alpha_{\mathrm{U},k}, D_k\sin\alpha_{\mathrm{U},k}, 0)$, IRS is $(x_{\mathrm{S}}, y_{\mathrm{S}}, z_{\mathrm{S}}) = (D_{\mathrm{S}}\cos\alpha_{\mathrm{U},k}, D_{\mathrm{S}}\sin\alpha_{\mathrm{U},k}, H_{\mathrm{S}})$. The UAV speed v_{U} has the moving direction $\gamma_{\mathrm{U},\alpha}$ and $\gamma_{\mathrm{U},\beta}$. The user speed v_k has the moving direction $\gamma_{k,\alpha}$. Hence, the location should be updated for t^{th} time slot, i.e., UAV is $(0, 0, H_{\mathrm{U}} - v_{\mathrm{U}}t\sin\gamma_{\mathrm{U},\beta})$, k^{th} user is $(D_k\cos\alpha_{\mathrm{U},k} + v_kt\cos\gamma_{k,\alpha}, D_k\sin\alpha_{\mathrm{U},k} + v_kt\sin\gamma_{k,\alpha}, 0)$. Hence, the updated relative azimuth, elevation angles and distances in (3.1)-(3.6) can be calculated based on new locations. Furthermore, we assumed that the antenna on UAV

$$L_{k,m} = \sqrt{(|D_k \sin \alpha_{\mathrm{U},k}| - \mathit{\Delta}_m \sin \psi_{\mathrm{U}})^2 + (D_k \cos \alpha_{\mathrm{U},k} + \mathit{\Delta}_m \cos \psi_{\mathrm{U}})^2 + z_{\mathrm{U}}^2}, \tag{3.11}$$

$$L_{k,n} = \sqrt{(|D_{\mathrm{S},k} \sin \alpha_{\mathrm{S},k}| - \mathit{\Delta}_n \sin \psi_{\mathrm{S}})^2 + (D_{\mathrm{S},k} \cos \alpha_{\mathrm{S},k} + \mathit{\Delta}_n \cos \psi_{\mathrm{S}})^2 + z_{\mathrm{S}}^2}, \tag{3.12}$$

$$L_{m,n} = \sqrt{\begin{array}{c}(|D_{\mathrm{S}} \cos \alpha_{\mathrm{U,S}}| - \mathit{\Delta}_m \cos \psi_{\mathrm{U}} + \mathit{\Delta}_n \cos \psi_{\mathrm{S}})^2 + \\ (D_{\mathrm{S}} \sin \alpha_{\mathrm{U,S}} + \mathit{\Delta}_n \sin \psi_{\mathrm{S}} - \mathit{\Delta}_m \sin \psi_{\mathrm{U}})^2 + (z_{\mathrm{U}} - z_{\mathrm{S}})^2\end{array}}. \tag{3.13}$$

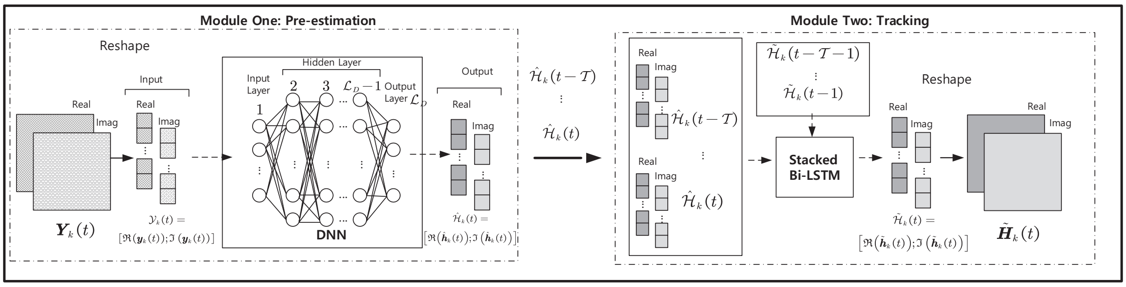

Module One: Pre-estimation
Reshape
Real
Imag
$\boldsymbol{Y}_k(t)$
Input
Real
Imag
$\mathcal{Y}_k(t) = [\Re(\boldsymbol{y}_k(t)); \Im(\boldsymbol{y}_k(t))]$
Hidden Layer
Input Layer
1
2
3
$\mathcal{L}_D - 1$
Output Layer $\mathcal{L}_D$
DNN
Output
Real
Imag
$\hat{\mathcal{H}}_k(t) = [\Re(\hat{\boldsymbol{h}}_k(t)); \Im(\hat{\boldsymbol{h}}_k(t))]$
$\hat{\mathcal{H}}_k(t-\mathcal{T})$
$\hat{\mathcal{H}}_k(t)$
Module Two: Tracking
Real
Imag
$\hat{\mathcal{H}}_k(t-\mathcal{T})$
Real
Imag
$\hat{\mathcal{H}}_k(t)$
$\tilde{\mathcal{H}}_k(t-\mathcal{T}-1)$
$\tilde{\mathcal{H}}_k(t-1)$
Stacked Bi-LSTM
Reshape
Real
Imag
$\tilde{\mathcal{H}}_k(t) = [\Re(\tilde{\boldsymbol{h}}_k(t)); \Im(\tilde{\boldsymbol{h}}_k(t))]$
Real
Imag
$\tilde{\boldsymbol{H}}_k(t)$

Figure 3.2: The illustration framework of the overall channel tracking algorithm.

has fixed physical flat position angle ψ_{U}. Similarly, the reflecting surface on IRS has fixed physical flat position angle ψ_{S}. The element distance on both antenna and reflecting surface is $\mathit{\Delta} = \lambda/2$. Therefore, the propagation delay of the waves $\tau_{k,m}, \tau_{k,n}, \tau_{m.n}$ can be calculated accordingly.

By taking the relative position in Fig.3.1(b) as an example, with $\mathit{\Delta}_m = \frac{1}{2}(M + 1 - 2m)\mathit{\Delta}$ as the distance between the m^{th} element and the center O_{U}, and $\mathit{\Delta}_n = \frac{1}{2}(N + 1 - 2n)\mathit{\Delta}$ as the distance between the n^{th} element and the center O_{S}, the distance between the k^{th} user and the m^{th} element on UAV antenna $L_{k,m}$, the distance between k^{th} user and n^{th} element on IRS is $L_{k,n}$, and the distance between m^{th} element on UAV antenna and n^{th} element on IRS is $L_{m,n}$ are calculated as in (3.11), (3.12), and (3.13).

3.3 Proposed Deep Learning Based Channel Tracking Algorithm

Data-driven DL based framework is now widely designed and employed for channel estimation [47], [93], [94]. This is because that DL can extract the characteristics of the complex environment information from received signals without the need of prior knowledge about the channel statistics [94]. Additionally, DL-based algorithms have low computational complexity with simple operations such as multiplications [93]. However, all these DL-based channel estimation work ignore the time sequence of the CSI. In another word, the adjacent observations of the time-varying CSI can be further utilized for more precise prediction.

To track the time varying channel, it is necessary to give neural networks the ability of learning the behavior of the correlation across time domain. There are two widely considered methods called recurrent neural network (RNN) and LSTM on solving the time-varying tasks, such as natural language processing. Both methods consider the information from the previously entered data and the currently entered data to predict. Specially, RNN has feedback loops to maintain information over time. However, it's difficult for RNN on learning long-term temporal dependencies due to the vanishing gradient problem. Differently, LSTM introduces input and forget gates for better preservation of long-term dependencies on dealing gradient flow [95].

By combining the advantages of DNN with multilayer perception mechanism on extracting characteristics of complex environment and LSTM with different input/output layers on passing information across time domain, in this section, DNN followed by Stacked Bi-LSTM framework is proposed to track the time sequence CSI in IRS-assisted UAV communication systems. The illustration of the overall structure of the proposed algorithm is shown in Fig. 3.2. There are mainly two modules in this framework: DNN for denoising pre-estimation based on off-line trained model and Stacked Bi-LSTM for tracking with the assistance of the history sequence information. To present the channel tracking problem in the vector form, we denote $\boldsymbol{y}_k(t) = \mathrm{vec}\left(\boldsymbol{Y}_k(t)\right) \in \mathbb{C}^{MN_p \times 1}$ and $\boldsymbol{h}_k(t) = \mathrm{vec}\left(\boldsymbol{H}_k(t)\right) \in \mathbb{C}^{M(N+1) \times 1}$.

3.3.1 Module One: DNN Channel Estimation Framework

DNN is an extension version of the artificial neural network with multiple hidden layers between the input and output layers [46]. To be specific, each hidden layer has multiple neurons and each output is the weighted sum of neurons operated by a nonlinear function. The Sigmoid function $f_{\text{Sigmoid}}(x) = \frac{1}{1+e^{-x}}$ and the ReLU function $f_{\text{ReLU}}(x) = \max(0, x)$ are the widely used nonlinear function for activation in the DNN.

The pre-estimation DNN is shown in Fig. 3.2(a) with $\mathcal{L}_D - 2$ hidden layers among total $\mathcal{L}_D$ layers. The l^{th} hidden layer of the network consists of N_e neurons where $2 \leqslant l \leqslant \mathcal{L}_D - 1$ and $1 \leqslant n_e \leqslant N_e$. The DNN input vector $\mathcal{Y}_k(t) = [\Re(\boldsymbol{y}_k(t)); \Im(\boldsymbol{y}_k(t))] \in \mathbb{R}^{MN_P \times 2}$ with the real and imaginary part of $\boldsymbol{y}_k(t)$ as $\Re(\boldsymbol{y}_k(t)) = [\Re(y_{k,1}(t)), ..., \Re(y_{k,MN_p}(t))]$ and $\Im(\boldsymbol{y}_k(t)) = [\Im(y_{k,1}(t)), ..., \Im(y_{k,MN_p}(t))]$, respectively. The total number of input layer neurons is $2MN_p$. Similarly, the DNN output vector $\hat{\mathcal{H}}_k(t) = [\Re(\hat{\boldsymbol{h}}_k(t)); \Im(\hat{\boldsymbol{h}}_k(t))] \in \mathbb{R}^{M(N+1) \times 2}$. The total number of output layer neurons is $2M(N+1)$. The total number of neurons on each hidden layer is defined as $16MN_P$.

To express the DNN transmission principle, we use $\boldsymbol{i}_l$ represents the input of the l^{th} layer neurons. o_{l,n_e} represents the output of the n_e^{th} neuron at l^{th} layer. $\boldsymbol{W}_l^{(\text{DNN})}$ and $\boldsymbol{b}_l^{(\text{DNN})}$ denote the weight matrix and the bias vector of the l^{th} layer. Hence, each neuron's output can be expressed as

$$o_{l,n_e} = f_{l,n_e}\left(b_{l,n_e}^{(\text{DNN})} + \boldsymbol{w}_{l,n_e}^{(\text{DNN})T} \boldsymbol{i}_l\right), \tag{3.14}$$

with f_{l,n_e} as the activation function for l^{th} layer and n_e^{th} neuron. For training stage, with total $\mathcal{B}$ batch size, the output of the DNN with $\dot{b}^{th}$ batch can be expressed as

$$\hat{\mathcal{H}}_k(\dot{b}, t) = \boldsymbol{f}_{\mathcal{L}_D}(...\boldsymbol{f}_2(\mathcal{Y}_k(\dot{b}, t); \boldsymbol{\theta}_2)...; \boldsymbol{\theta}_{\mathcal{L}_D}). \tag{3.15}$$

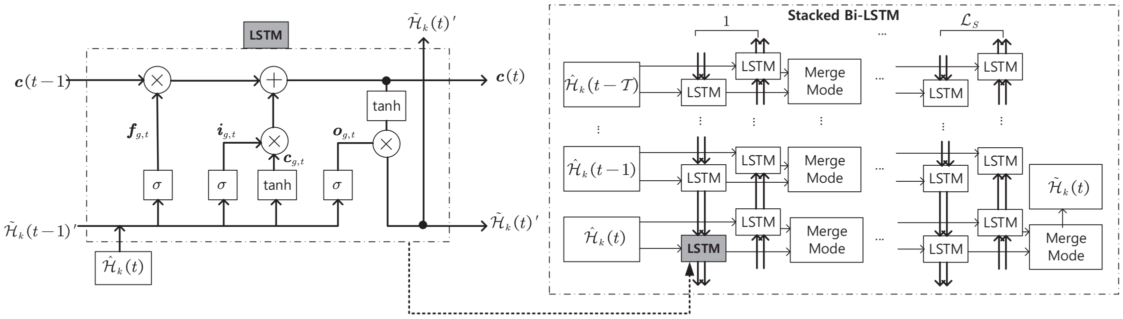


Figure 3.3: The structure of Stacked Bi-LSTM.

During the training phase of constructing DNN, the parameter set $\boldsymbol{\theta}_l = (\boldsymbol{W}_l^{(\mathrm{DNN})}, \boldsymbol{b}_l^{(\mathrm{DNN})})$ which represents the weights and biases of the DNN model at the l^{th} layer can be obtained through gradient descent by recursively minimize the loss function $\mathrm{Loss}(\boldsymbol{\theta})$ until convergence. The loss function across all the layers is defined as

$$\mathrm{Loss}(\boldsymbol{\theta}) = \frac{1}{2M(N+1)\mathcal{B}} \sum_{\dot{b}}^{\mathcal{B}} \left(\hat{\mathcal{H}}_k(\dot{b},t) - \mathcal{H}_k(\dot{b},t)\right)^2, \tag{3.16}$$

with $\boldsymbol{\theta}$ denotes all the parameter set across all the layers, $\mathcal{H}_k(\dot{b},t) = [\Re(\boldsymbol{h}_k(\dot{b},t)); \Im(\boldsymbol{h}_k (\dot{b},t))]$ denotes the true value of the channel in the $\dot{b}^{th}$ batch of the training process. The detail of the training process of the channel pre-estimation DNN can be found in **Algorithm 3.1**.

3.3.2 Module Two: Stacked Bi-LSTM Channel Tracking

As the second module of the overall algorithm framework, Stacked Bi-LSTM tracks the time sequence CSI based on pre-estimation denoised channel information (i.e., $\hat{\mathcal{H}}_k(t-\mathcal{T})$ to $\hat{\mathcal{H}}_k(t)$) and previous channel tracking output (i.e., $\tilde{\mathcal{H}}_k(t-\mathcal{T}-1)$ to $\tilde{\mathcal{H}}_k(t-1)$). Specifically, $\mathcal{T}$ denotes the total history time slot utilized in the model.

3.3.2.1 LSTM

As an enhancement of the recurrent neural network, LSTM is a gradient-based learning algorithm which is able to connect previous information to the current task [96], [97].

Algorithm 3.1: Training of Module One: Channel Pre-estimation DNN

```
Input:Training received signal $\mathcal{Y}_k(1), ..., \mathcal{Y}_k(T)$, training true channel information $\mathcal{H}_k(1), ..., \mathcal{H}_k(T)$.
Output:Trained pre-estimation DNN.
Initialization:Randomize initial weights $\boldsymbol{\theta}$.
Generate a set of training sequences $\mathcal{Y}_k(1), ..., \mathcal{Y}_k(T)$ and $\mathcal{H}_k(1), ..., \mathcal{H}_k(T)$ with selected SNRs and pilot overheads size $N_P$.
Design the pre-estimation DNN framework with $\mathcal{L}_D$ layers and $N_e$ neurons in each hidden layer. Set the learning rate and batch size.
while not convergence do
    Update weights $\boldsymbol{\theta}$ by minimizing loss function in (3.16).
end while
```

Normally, the time sequence data fed to the LSTMs along the chain-like structure in a forward direction. The illustration of a single LSTM cell is shown in Fig. 3.3 on the left hand side. (This illustration structure only takes the first layer forward LSTM as an example.)

The main difference between LSTM architecture and widely known RNN is the hidden layers in each LSTM cell. The first layer is called forget layer which is also known as the forget gate $\boldsymbol{f}_{g,t}$. It consists of the information passed from previous layer $\tilde{\mathcal{H}}(t-1)$ and current denoised input $\hat{\mathcal{H}}(t)$ with weights $\tilde{\boldsymbol{W}}_f, \hat{\boldsymbol{W}}_f$ and bias $\boldsymbol{b}_f$ along with activation function:

$$\boldsymbol{f}_{g,t} = \sigma\left(\hat{\boldsymbol{W}}_f \hat{\mathcal{H}}_k(t) + \tilde{\boldsymbol{W}}_f \tilde{\mathcal{H}}_k(t-1) + \boldsymbol{b}_f\right). \tag{3.17}$$

The second layer is the input gate $\boldsymbol{i}_{g,t}$ which can be written as

$$\boldsymbol{i}_{g,t} = \sigma\left(\hat{\boldsymbol{W}}_i \hat{\mathcal{H}}_k(t) + \tilde{\boldsymbol{W}}_i \tilde{\mathcal{H}}_k(t-1) + \boldsymbol{b}_i\right). \tag{3.18}$$

It has the similar structure as the forget gate $\boldsymbol{f}_{g,t}$ but with different weights $\tilde{\boldsymbol{W}}_i, \hat{\boldsymbol{W}}_i$ and bias $\boldsymbol{b}_i$. The third layer is the cell input state $\boldsymbol{c}_{g,t}$ that can be calculated as

$$\boldsymbol{c}_{g,t} = \tanh\left(\hat{\boldsymbol{W}}_c \hat{\mathcal{H}}_k(t) + \tilde{\boldsymbol{W}}_c \tilde{\mathcal{H}}_k(t-1) + \boldsymbol{b}_c\right), \tag{3.19}$$

through a tanh function. The final layer is called output gate $\boldsymbol{o}_{g,t}$, which can be calculated as

$$\boldsymbol{o}_{g,t} = \sigma\left(\hat{\boldsymbol{W}}_o \hat{\mathcal{H}}_k(t) + \tilde{\boldsymbol{W}}_o \tilde{\mathcal{H}}_k(t-1) + \boldsymbol{b}_o\right). \tag{3.20}$$

To summarize the aforementioned hidden layers structure, $\boldsymbol{W}$ and $\boldsymbol{b}$ are the weight matrix and bias vector of the corresponding parameters. σ represents the gate activation function which is normally sigmoid function.

Apart from the cell input state $\boldsymbol{c}_{g,t}$ in the hidden layers, there are two other cell states in the structure: the previous cell output state $\boldsymbol{c}(t-1)$ fed in the current LSTM cell, and the current cell output state $\boldsymbol{c}(t)$ passed to the next LSTM cell. The output state at current time t can be updated as

$$\boldsymbol{c}(t) = \boldsymbol{f}_{g,t} \otimes \boldsymbol{c}(t-1) + \boldsymbol{i}_{g,t} \otimes \boldsymbol{c}_{g,t}, \tag{3.21}$$

$\boldsymbol{A} \otimes \boldsymbol{B}$ denotes the Kronecker product. More importantly, the conventional input layer is the denoised time sequence $\hat{\mathcal{H}}_k(t)$ at time slot t passed from DNN pre-estimation. Finally, the output layer can be calculated as

$$\tilde{\mathcal{H}}_k(t)' = \boldsymbol{o}_{g,t} \otimes \tanh(\boldsymbol{c}(t)). \tag{3.22}$$

3.3.2.2 Stacked Bi-LSTM

The illustration of the Stacked Bi-LSTM is shown in Fig. 3.3 on the right hand. To overcome the drawback of single LSTM cell which can only capture the history information, bidirectional structure has been proposed to combine both forward and backward directions to be able to utilize both history and future information [98]. Hence the forward output $\tilde{\mathcal{H}}_k(t)'^{(f)}$ and backward output $\tilde{\mathcal{H}}_k(t)'^{(b)}$ of each LSTM cell is calculated based on the relative input and the output layer function in (3.22). The forward layer output is iteratively calculated based on the time slot $t - \mathcal{T}$ to $t - 1$. Similarly, the backward layer output is calculated based on the reversed time sequence $t-1$ to $t-\mathcal{T}$. The output of bidirectional structure can be expressed as

$$\tilde{\mathcal{H}}_k(t)^{'} = \sigma_{f,b}\left(\tilde{\mathcal{H}}_k(t)^{'(f)} \oplus \tilde{\mathcal{H}}_k(t)^{'(b)}\right), \tag{3.23}$$

which is the combination of the forward and backward output with the pre-defined merge mode $\sigma_{f,b}$, $\boldsymbol{A} \oplus \boldsymbol{B}$ denotes the direct sum. The $\sigma_{f,b}$ function can be concatenating, summation, average or multiplication.

It has been proved that by stacking multiple hierarchical models, the performance can be improved progressively[99]. Hence, we adopt a stacked structure where the output from the lower layer is then fed as the input to the upper layer with $\mathcal{L}_S \geqslant 2$ Bi-LSTM layers. The workflow of the Stacked Bi-LSTM considers both forward and backward directions and deeper structure with $\mathcal{T}$ time slots. The final tracking output after the $\mathcal{L}_S^{th}$ layer of the Stacked Bi-LSTM can be written as

$$\tilde{\mathcal{H}}_k(t) = \sigma_{f,b}\left(\tilde{\mathcal{H}}_k(t)^{'(f)}_{\mathcal{L}_S} \oplus \tilde{\mathcal{H}}_k(t)^{'(b)}_{\mathcal{L}_S}\right). \tag{3.24}$$

The training process of the channel tracking Stacked Bi-LSTM can be found in **Algorithm 3.2**. The proposed overall channel tracking algorithm can be found in **Algorithm 3.3**.

3.4 Numerical Results

In this section, the training loss of various algorithms, the channel tracking performance and the complexity are evaluated. Based on the framework with two modules, we specially compare the channel tracking performance of DNN, DNN followed by LSTM, DNN followed by Bi-LSTM and DNN followed by Stacked Bi-LSTM. Specifically, DNN followed by Stacked Bi-LSTM is the proposed algorithm we mainly focused on evaluated. Basic parameters are set as follows: carrier frequency $f_c = 28$GHz, the UAV speed

```
Algorithm 3.2: Training of Module Two: Channel Tracking Stacked Bi-LSTM
    Input:Pre-estimation $\hat{\mathcal{H}}_k(1), ..., \hat{\mathcal{H}}_k(T)$ training sequence, true channel
  information $\mathcal{H}_k(1), ..., \mathcal{H}_k(T)$ training sequence.
    Output:Trained tracking model for the current sequence.
    Initialization:Randomize initial weights $\boldsymbol{W}$ and bias $\boldsymbol{b}$.
    Generate a new set of training sequences $\hat{\mathcal{H}}_k(1), ..., \hat{\mathcal{H}}_k(T)$ and $\mathcal{H}_k(1), ..., \mathcal{H}_k(T)$
      with selected SNR, pilot overheads size $N_P$, and $\mathcal{T}$ historic time step.
    Design the bi-directional tracking framework with $\mathcal{L}_S$ stacked layers and $\mathcal{T}$ historic
      time step in each layer. Set the learning rate and batch size.
    while not convergence do
       Update weights $\boldsymbol{W}$ and bias $\boldsymbol{b}$ by minimizing loss function in (3.16).
    end while
```

```
Algorithm 3.3: Proposed Overall Channel Tracking Algorithm
    Input: Received signal $\mathcal{Y}_k(1), ..., \mathcal{Y}_k(T)$.
    Output: Channel tracking information $\tilde{\mathcal{H}}_k(T)$.
      % Module One Channel Pre-estimation:
    Construct the $\mathcal{L}_D$ layers DNN framework.
    Load the DNN optimized parameters that has been trained in Algorithm 3.1.
    Pre-estimate the channel information $\hat{\mathcal{H}}_k(1), ..., \hat{\mathcal{H}}_k(T)$.
      % Module Two Channel Tracking:
    Train the channel tracking model as in Algorithm 3.2 with $(T-1)$ sequence data
      and $\mathcal{T}$ historic time step.
    Use time sequence $T-\mathcal{T}$ to $T-1$ as input data to track $T$ channel information
      $\tilde{\mathcal{H}}_k(T)$ based on the trained tracking model.
```

$v_{\mathrm{U}} = 5m/s$ with azimuth and elevation angle of UAV speed $\gamma_{\mathrm{U},\alpha} = \pi/6$ and $\gamma_{\mathrm{U},\beta} = \pi/6$. The k^{th} user speed $v_k = 5m/s$ with azimuth angle of user speed $\gamma_{k,\alpha} = \pi/24$. To prove the superiority tracking performance of our proposed scheme, in the simulation, we consider one moving user in the system. Total number of IRS reflection elements $N = 8$ and total number of antenna elements on the k^{th} UAV $M = 8$. The antenna angle on UAV is $\psi_{\mathrm{U}} = \pi/6$, the reflecting surface on IRS has fixed physical flat angle $\psi_{\mathrm{S}} = \pi/6$. The height of UAV $H_{\mathrm{U}} = 2000m$, the height of IRS $H_{\mathrm{S}} = 100m$. The initial relative position of the UAV, IRS, and the k^{th} user is shown in Fig.3.1 (b) with IRS location as $(-600m, -600m, 100m)$ and UAV location as $(500m, 600m)$. The total time sequence is 200 time slots[2] with train test split rate set as 0.8 for channel tracking performance

[2] For simplicity, we assume that the tracking signal is received in every time slot. This means that $T = 200$ in the simulation.

evaluation. To be noticed, during the off-line learning, the adaptive moment (Adam) estimation optimizer is employed.

3.4.1 Loss function

Firstly, the loss function of the channel pre-estimation DNN with a different number of pilot overheads sizes N_P and hidden layers $\mathcal{L}_D$ are shown in Fig. 3.4. The total batch size $\mathcal{B}$ and learning rate ε are set as $512, 0.001$ for training the pre-estimation DNN. The total training data set is selected from many SNR$= -5, 0, 5, 10, 15, 20$dB and blockage probability from $p_k = 0.3, 0.4, 0.5, 0.6$ with $T = 200$ sequences. It can be observed that when $\mathcal{L}_D = 3$ for both $N_P = 10, 20$ require longer epochs to achieve convergence of the loss function. The difference for convergence of the loss function between $\mathcal{L}_D = 4, 5, 6$ is not significant. Besides, the converge loss for pilot overheads size $N_P = 10$ is higher than $N_P = 20$. The reason lies in that with N_P increases, the input size and neurons on each layer are increased accordingly, resulting in a more complex structure but more suitable model. However, with N_P increases, the network neuron size increases accordingly. Empirically, our model can achieve sufficient performance gain with relative small N_P. Hence, $\mathcal{L}_D = 4$ and $N_P = 10$ are adopted for the remaining simulations.

Secondly, the loss function of different channel tracking methods (LSTM, Bi-LSTM and proposed algorithm) are compared in Fig. 3.5. The proposed algorithm has total stacked layers $\mathcal{L}_S = 3$. The channel sequence for training difference tracking methods is set as SNR$= 20$dB and $p_k = 0.3$ with a total $T = 200$ sequence. In Fig.3.5 (a), different historical time step $\mathcal{T} = 2, 3, 4$ are compared based on $\mathcal{B} = 64$ and $\epsilon = 0.01$. The proposed algorithm can quickly converge to loss near 0 for selected dynamic channel sequence epochs compared to the other two methods. It's worth noticing that the suitable hyper-parameter $\mathcal{T}$ can be selected differently based on various scenarios, such as different v_{U} and v_k. In Fig.3.5 (b), different batch size $64, 128$ are compared based on $\mathcal{T} = 3$ and $\epsilon = 0.01$. The proposed algorithm shows a similar convergence performance.

Additionally, loss function of the proposed algorithm with different hyper-parameters

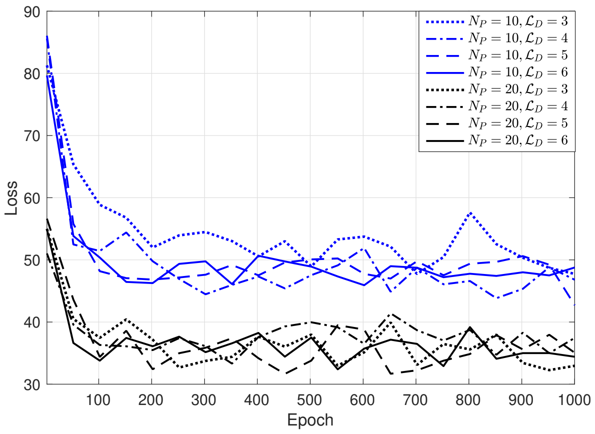


Figure 3.4: Loss function of the channel pre-estimation DNN with different number of pilot overheads sizes $N_P = 10, 20$ and DNN total layers $\mathcal{L}_D = 3, 4, 5, 6$.

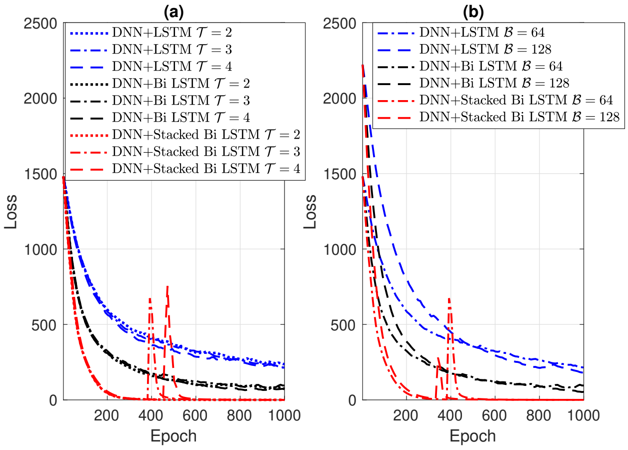


Figure 3.5: (a) Loss function of various channel tracking methods with different historical time step $\mathcal{T} = 3, 4, 5$. (b) Loss function of various channel tracking methods with different batch sizes $\mathcal{B}$.

are shown in Fig. 3.6 for the channel tracking of total $T = 200$ dynamic channel under SNR= 20dB, pilot overheads size $N_P = 10$. By comparing different learning rates in Fig. 3.6 (a), $\mathcal{T}$ and $\mathcal{L}_S$ in Fig. 3.6 (b), $\epsilon = 0.01$ with $\mathcal{B} = 64$, $\mathcal{T} = 3$ and $\mathcal{L}_S = 3$, the hyper-parameter set that require slightly less epochs to achieve convergence of the loss function are selected for channel tracking performance evaluation in the following subsection.

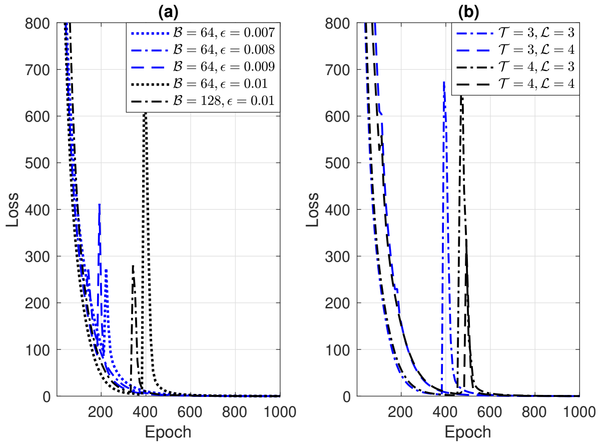


Figure 3.6: (a) Loss function of the proposed algorithm with various batch sizes $\mathcal{B}$ and learning rate ϵ. (b) Loss function of the proposed algorithm with various historical time steps $\mathcal{T} = 3, 4$ and stacked layers $\mathcal{L}_S = 3, 4$.

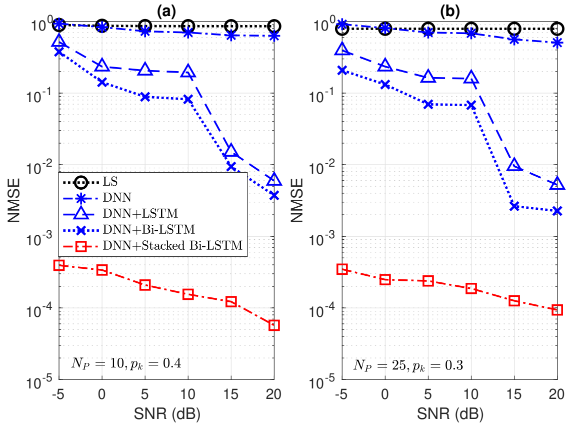


Figure 3.7: NMSE performance for different algorithms.

3.4.2 Channel tracking performance

To evaluate the tracking performance of the algorithms, we refer the performance metric [85] as

$$\text{NMSE} = \frac{E\left(\|\tilde{\boldsymbol{H}}_k - \boldsymbol{H}_k\|^2\right)}{E\left(\|\boldsymbol{H}_k\|^2\right)} \tag{3.25}$$

where $\tilde{\boldsymbol{H}}_k$ is the estimated tracking channel, $\boldsymbol{H}_k$ is the true channel. To evaluate the NMSE performance of the algorithms, the total $T = 200$ sequence is divided into train and test data set with time slot $1 - 160$ as training set (for LSTM, Bi-LSTM and proposed) and $161-200$ as testing tracking data set. The performance is evaluated over the average of the total 500 rounds. The conventional LS algorithm [100] is applied in our system as a benchmark.

In Fig. 3.7, it can be observed that all algorithms show down trends NMSE performance with SNR increases. The proposed algorithm maintains significantly superior performance compared to other algorithms with small pilot overheads. It's worth noticed that LS and DNN are static estimation methods that only take the received signal into consideration for estimation. Differently, the rest three algorithms shows performance gain as they also observe the correlation across time domain. Specially, with bi-directional and stacked structure, our proposed algorithm can not only extract information from past, but also from future and deeper network structure.

3.4.3 Complexity

The complexity and computation time of different algorithms are characterize in this subsection. The complexity of the pre-estimation DNN without considering the bias is $\mathcal{O}\left(\sum_{l=1}^{\mathcal{L}_D-1}(N_e^l N_e^{l+1})\right)$ with N_e^l represents the neuron numbers in each layer. To be specific, the input layer ($l=1$) has $2MN_P$ neurons, each hidden layer ($l=2,...,(\mathcal{L}_D-1)$)

Table 3-B: Table of Operation Time (ms) for Channel Pre-estimation and Tracking

Module One: Different Layers of Pre-estimation DNN				
	$M=8, N=8$			$M=8, N=4$
	$N_P=10$	$N_P=15$	$N_P=20$	$N_P=10$
DNN $\mathcal{L}_D=3$	0.209	0.343	0.327	0.068
DNN $\mathcal{L}_D=4$	0.334	0.558	0.575	0.111
DNN $\mathcal{L}_D=5$	0.421	0.711	0.722	0.148
DNN $\mathcal{L}_D=6$	0.551	0.885	0.920	0.185
Module Two: Different Algorithms for Channel Tracking				
	$M=8, N=8$		$M=4, N=8$	
$N_P=10$	$\mathcal{T}=2$	$\mathcal{T}=5$	$\mathcal{T}=2$	$\mathcal{T}=5$
LSTM	2.956	2.983	0.283	0.394
Bi-LSTM	4.630	5.215	0.457	0.719
Stacked Bi-LSTM $\mathcal{L}_S=2$	8.770	10.543	0.733	1.442
Stacked Bi-LSTM $\mathcal{L}_S=3$	12.813	15.477	1.220	2.165

has $16MN_P$ neurons, and the output layer ($l = \mathcal{L}_D$) has $2M(N+1)$ neurons. The complexity of a single LSTM cell can be calculated as $\mathcal{O}\left(N_c(4N_c + 4N_i + N_o + 3)\right)$ [101]. Specifically, N_c, N_i, N_o are the number of memory cell, the number of input units and the number of output units, respectively. The N_i and N_o are both $2M(N+1)$. Moreover, for LSTM $N_c = 1\mathcal{T}$, Bi-LSTM $N_c = 2\mathcal{T}$, Stacked Bi-LSTM $N_c = 2\mathcal{T}\mathcal{L}_S$. For Bi-LSTM, the forward and backward bi-directional estimation gives $N_c = 2\mathcal{T}$. For Stacked Bi-LSTM, with $\mathcal{L}_S$ multiple layers of Bi-LSTM stacked together, it's obvious that $N_c = 2\mathcal{T}\mathcal{L}_S$.

As can be seen from Table. 3-B that the Python Jupyter Notebook run-time of the multiple layers of pre-estimation DNN increases when layers $\mathcal{L}_D$, N_P, M increases. Besides, for channel tracking, Stacked Bi-LSTM requires a longer running time due to the complex structure, but achieves dramatically better NMSE performance over dynamic channel sequences, as proved in previous subsection.

3.5 Summary

In this chapter, we developed the 3D geometry dynamic channel model in mmWave IRS-assisted UAV-enabled communication system. The navigation of UAVs and the movement of mobile user are considered to construct the time-variant channel. There are mainly two dominant links in the channel model: dynamic LoS link (user-UAV) and dynamic virtual LoS link (user-IRS-UAV). We further proposed a novel DL-based channel tracking algorithm composed of two schemes: DNN channel pre-estimation for denoising and Stacked Bi-LSTM for channel tracking. Specifically, Stacked Bi-LSTM is a framework that can trace historical information in the time sequence based on a bidirectional structure over multiple stacked layers. Simulation results have shown that the proposed channel tracking algorithm dramatically outperforms different benchmarks with small pilot overheads and comparable computation complexity.

Chapter 4

3D Channel Tracking for SU Space-Air Communications

Although Chapter 3 discussed the 3D channel tracking in air-ground communications, it is not a complete communication scheme without the support of satellite in SAGIN. Furthermore, due to the satellite-ground links are very sensitive to terrestrial interference[102], UAVs can also act as relays to improve the performance of communications with enhanced capacity for covered areas. Hence, in this chapter, we further exploit the space-air links with UAV-satellite communications to complete the connections across three segments in SAGIN.

Compared to traditional ground or satellite networks, UAV-aided communication as a key part of SAGIN can be applied not only in the scenario that ground infrastructure cannot be accessed but also for emergency communications in crowded or disaster areas. Besides, with UAVs deployed in the middle of ground-space communications, shorter range LoS links provide a significant performance improvement over long-distance links [58]. For space-air links, to mitigate the problem of the spectrum crowding, mmWave Ka-Band (26.5-36GHz) offers additional frequency ranges with higher data rates [3], [18]. Although Ka-band is much more weather dependent due to the high path loss, an antenna array with a large number of antenna elements provides higher transmission gain and compensates the propagation losses [24]. In the above-mentioned scenarios, accurate channel estimation with a small pilot overheads is a crucial challenge to both UAV-satellite (space-air) links (e.g., LEO and MEO satellites [103]) and air-to-ground links over dynamic wireless channels due to the satellite orbiting and UAV 3D trajectory [104], [105]. This chapter is organized as follows. The related work and main contributions of this chapter are firstly reviewed in Section 4.1. The SU space-air link communications system model is introduced in Section 4.2. In Section 4.3, the 3D-2D-MM channel model is elaborated and the problem of channel tracking is formulated. The proposed algorithm 3D-DTAMP is presented in Section 4.4. Numerical results and conclusions are then provided in Section 4.5 and Section 4.6, respectively.

4.1 Introduction

4.1.1 Related Work

Apart from naively applying the conventional compressive sensing algorithms to solve the channel estimation problem, several structured sparse channel estimation algorithms have been developed by learning spatial correlation. The spatial correlation is a feature that non-zero elements of an angular domain channel are concentrated in several bursts [36], [37], [106], [107]. In [106], a joint orthogonal matching pursuit algorithm which exploited the hidden joint channel sparsity among multiple users was proposed with improved performance. Moreover, the sparsity of channel supports for the neighbouring antennas were similar as shown in [107]. Additionally, by obtaining the partial knowledge of the physical scattering structure channel, the proposed least absolute shrinkage and selection operator algorithm showed robust recovery performance [36]. In [37], the structured spatial domain sparsity was modeled as a Markov prior by considering the relationship of channel support vectors. However, all of these investigations were based on the static channel environment which could not be applied for the mobile scenario.

Different from the static sparse channel estimation, channel tracking was realized by considering prior information from the previously estimated channel. This temporal correlation can help to track the channel information with further reduced training overheads and dramatically improved real-time estimation performance [50], [108], [109]. In [108], by updating the channel variation based on the temporal correlation over time, the proposed differential OMP (D-OMP) tracked the dynamic sparse channels with improved accuracy. Similarly, [50] proposed a method called structured matching pursuit which simultaneously detected the common and the dynamic channel taps by looking at the insight of path delays in each time slot. By further excavating the physical direction between the base station and mobile user, and utilizing the obtained channel information from the previous time slot, a priori-aided tracking method was proposed in [109] to predict the information of beamspace channels for the next time slot.

In order to have better channel tracking performance, it is necessary to jointly consider both spatial correlation and temporal correlation of the dynamic channel [110]–[112]. The works in [110] and [111] decoupled the time-varying channel estimation procedure into AoAs/AoDs and path gains estimation with [110] utilized unscented Kalman filter, [111] explored the block-sparse problem of the angle, separately. Additionally, to accurately formulate the more realistic dynamic model of AoDs support in the flat surface over time, spatial sparsity structure and temporal transition probability during the dynamic evolution of the channel were captured in [112].

4.1.2 Motivation and Contribution

As discussed above, the channel estimation or tracking methods in [36], [37], [50], [106]–[112] that consider the local ground environment characteristics can be directly applied to the air-to-ground links. The most widely considered mobile scenario is vehicle-to-vehicle (V2V) communications. [113] has briefly summarized three different mmWave V2V channel modeling: stochastic channel models, geometry-based channel models, and deterministic channel models. Stochastic channel models detailed statistics such as delay, Doppler and angular domains. Geometry-based models [114] considered cluster number, position and nonisotropic scattering. Deterministic modeling addressed the issues of sparse multipath components. However, as another segment of the SAGIN, the space-air communication has some distinct properties: the high non-stationary channel condition and the dominant LoS ray [1]. Due to the limited scatterers of ground propagation environment, the angular domain structured sparsity of channels has been studied in [36], [37], [106], [107]. This property was similar to space-air communication with one concentrate ray. However, in practice, it is non-trivial to obtain the real-time channel information due to the orbiting of the satellite and 3D UAV trajectories [115], which impose a formidable challenge on modeling and tracking the dynamic channel. By utilizing the advantage of the prior information of the channel, time-varying channel tracking has been considered in [50], [108], [109]. However, all these tracking algorithms have ignored the dynamic angular domain spatial correlation. Although both spatial and temporal correlation were explored in [110]–[112] for channel tracking, the premise of the system was that the communication was transmitted over the same flat surface. Additionally, the tracking methods mentioned above [50], [108]–[111] lacks the consideration of the dynamic probability of the model. Although the spatial sparsity structure has been introduced in [112], the insight of a more complicated 3D dynamic spatial sparsity has not been investigated.

Motivated by the aforementioned literature review, to capture a more realistic space-air dynamic channel information, we consider a fading channel as dynamic probability model and explore the 3D dynamic angle domain sparsity structure over time in this paper. We study the space-air communication system with UAVs equipped with a uniform plane array (UPA) and propose a method to recursively track the 3D dynamic channel with lower pilot overheads, comparable complexity, and better recovery performance. The main contributions are summarized as follows:

- We develop a statistical 3D dynamic channel model for space-air communications called 3D-2D-MM. The 3D-2D-MM captures the structured sparsity in both azimuth and elevation spatial domains as well as probabilistic temporal correlation.

- The dynamic spatial domain transition probabilities are designed for real-world space-air communication channel modeling to replace the expression of the complex continuous relative displacement between the 3D UAV trajectory and orbital satellite.
- We propose a novel 3D-DTAMP algorithm which recursively tracks the 3D dynamic channel by exploiting the relationship of both azimuth and elevation angle supports.
- The proposed 3D-DTAMP algorithm achieves better reconstruction performance with lower pilot overheads and comparable complexity.

4.2 System model

The SAGIN and the proposed space-air communication models are in Fig. 4.1. Important notations are summarized in Table 4-A. We consider a narrow-band channel model with OFDM transmission through a space-air communication downlink. Since array antennas have been widely employed on commercial satellites to improve the spectral efficiency by generating directional beams [116], the assumption in our model is that the LEO satellite is installed with UPA $M = N_x \times N_y$ antennas. Additionally, similar as in [117], a single omnidirectional antenna is implemented on UAVs. For the training procedure, the symbols $\boldsymbol{S} \in \mathbb{C}^{M \times N_{\mathrm{S}}}$ are transmitted. Then, the received signal at the UAV can be expressed as

$$\boldsymbol{y}_t^{\mathrm{RX}} = \boldsymbol{H}_t \boldsymbol{S} + \boldsymbol{n}_t^{\mathrm{RX}}, \tag{4.1}$$

with the transmitted pilot $\boldsymbol{S}_k$ designed based on the partial discrete Fourier transform (DFT) random permutation measurement matrix $\boldsymbol{\Phi}_{k,t}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right)$ [37], [118]. Details are shown in **Appendix A**, and $\boldsymbol{n}_t^{\mathrm{RX}} \sim \mathcal{CN}(0, \sigma_n^2 \boldsymbol{I})$ is the additive white Gaussian noise.

For a fixed satellite communication system over Ka-band, the LoS ray is the dominant path [105]. Hence, the $\boldsymbol{H}_t$ denotes the frequency domain channel matrix with one LoS

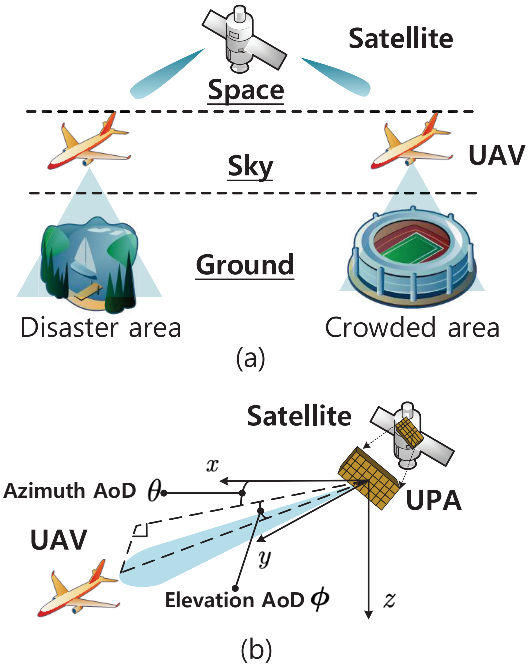


Figure 4.1: (a) The space-air-ground integrated network. (b) The illustration of the space-air communications system model.

path, which is given by,

$$\boldsymbol{H}_t = \alpha_t \boldsymbol{A}^{\mathrm{H}} \left(\theta_t, \phi_t\right), \tag{4.2}$$

where α_t is the complex channel gain and $\boldsymbol{A}\left(\theta_t, \phi_t\right) \in \mathbb{C}^{(N_x \times N_y) \times 1}$ represents the transmit array response vectors, with θ_t and ϕ_t as azimuth and elevation directions of AoD at time t. We take the UPA into consideration and the steering vector can be written as

$$\boldsymbol{A}\left(\theta_t, \phi_t\right) = \frac{1}{\sqrt{N_x \times N_y}} [1, ..., e^{j\frac{2\pi}{\lambda} d \sin\theta_t (n_x \sin\phi_t + n_y \cos\phi_t)}, ..., e^{j\frac{2\pi}{\lambda} d \sin\theta_t (N_x \sin\phi_t + N_y \cos\phi_t)}]^{\mathrm{T}}, \tag{4.3}$$

where λ is the signal wavelength, with array size as $N_x \times N_y$ and $0 \leqslant n_x < N_x$ and $0 \leqslant n_y < N_y$ are the indices of an antenna element. In this paper, we consider the case of quantized AoD pair, i.e., $\theta_t \in \{0, \theta_{\text{range}}/N_x, ..., \theta_{\text{range}}(N_x - 1)/N_x\}$ and $\phi_t \in \{0, \phi_{\text{range}}/N_y, ..., \phi_{\text{range}}(N_y - 1)/N_y\}$ [1] which means AoD pair is chosen from the $N_x \times N_y$

[1] Due to the long distance between the UAV and the satellite, the moving range of both azimuth and elevation angle would be limited.

Table 4-A: Table of Important Notations

Notations	Explanations
n_x, N_x	Index and total number of x-axis antenna/sparsity size
n_y, N_y	Index and total number of y-axis antenna/sparsity size
N_S	Total pilot number
m, M	Index and total number of the antenna and the sparse grid
t, T	Index and total number of the time slot
θ_t, ϕ_t	Azimuth and elevation directions of AoD at time slot t
θ_{range}	The range of azimuth directions of AoD
ϕ_{range}	The range of elevation directions of AoD
α_t	Complex channel gain at time slot t
$\boldsymbol{A}\left(\theta_t, \phi_t\right)$	Transmit array response vector at time slot t
$\boldsymbol{H}_t$	Frequency domain channel model at time slot t
$\boldsymbol{S}$	Transmit pilot symbols
$\boldsymbol{y}_t^{\mathrm{RX}}, \boldsymbol{y}_t$	Received signal at UAV at time slot t
$\boldsymbol{y}^{(t)}$	Total received signal at UAV for period t
$\boldsymbol{n}_t^{\mathrm{RX}}, \boldsymbol{n}_t$	Additive white Gaussian noise at time slot t
$g_{t,m}$	Channel gain element for the m^{th} AoD at time slot t
$\boldsymbol{g}_t$	Sparse matrix complex channel gain at time slot t
$\boldsymbol{g}^{(t)}, \boldsymbol{g}^{(T)}$	Total sparse matrix complex channel gain for period t or T
$b_{t,m}$	Hidden support element for the m^{th} AoD at time slot t
$\boldsymbol{b}_t$	Hidden support vector at time slot t
$\boldsymbol{b}^{(t)}, \boldsymbol{b}^{(T)}$	Hidden support vector for total time slot period t or T
$a_{t,m}$	Azimuth support element for the m^{th} AoD at time slot t
$e_{t,m}$	Elevation support element for the m^{th} AoD at time slot t
$\boldsymbol{a}_t, \boldsymbol{e}_t$	Azimuth and elevation support vector at time slot t
$\boldsymbol{a}^{(t)}, \boldsymbol{a}^{(T)}$	Azimuth support vector for total time slot period t or T
$\boldsymbol{e}^{(t)}, \boldsymbol{e}^{(T)}$	Elevation support vector for total time slot period t or T
$\vartheta_{t,m}$	Hidden value element for the m^{th} AoD at time slot t
$\boldsymbol{\vartheta}_t$	Hidden value vector at time slot t
$\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\vartheta}^{(T)}$	Hidden value vector for total time slot period t or T

uniform grid [2].

For large number of antenna elements at the LEO satellite, spatial resolution of the angular basis extends. Hence, the channel is treated sparse and the channel model can be written as a virtual model and then processed by compressive sensing techniques, which is presented as

$$\boldsymbol{H}_t = \boldsymbol{g}_t \tilde{\boldsymbol{A}}^{\mathrm{H}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}\right), \tag{4.4}$$

[2] In this paper, the quantized AoDs are considered, leave the continuous condition for future work.

where $\boldsymbol{g}_t \in \mathbb{C}^{1\times M}$ is a sparse matrix with one non-zero elements (which means only one dominant path), $\tilde{\boldsymbol{A}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}\right) \in \mathbb{C}^{(N_x \times N_y)\times M}$ is the array response vector dictionary matrix with spatial grids M.

4.3 Channel Model and Problem Formulation

In this section, the 3D-2D-MM channel model is introduced and the problem formulation of the proposed dynamic channel tracking is presented.

4.3.1 3D-2D-MM Channel Model

The dynamic angular domain channel $\boldsymbol{g}^{(T)} = \{\boldsymbol{g}_1, ..., \boldsymbol{g}_T\}$ can be modeled as a probabilistic signal model with two hidden random processes, $\boldsymbol{b}^{(T)} = \{\boldsymbol{b}_1, ..., \boldsymbol{b}_T\}$ and $\boldsymbol{\vartheta}^{(T)} = \{\boldsymbol{\vartheta}_1, ..., \boldsymbol{\vartheta}_T\}$. Here, $\boldsymbol{b}_t = \left[b_{t,1,1}, ..., b_{t,N_x,N_y}\right] = \left[b_{t,1}, ..., b_{t,M}\right] \in \{0,1\}^M$ represents the joint hidden support vector at time t, which indicates the channel sparsity with $b_{t,m} \in \{1,0\}$ [3]. The hidden value vector $\boldsymbol{\vartheta}_t = [\vartheta_{t,1}, ..., \vartheta_{t,M}] \in \mathbb{C}^M$ represents the temporal correlation of channel coefficients. The dynamic channel element for the m^{th} AoD direction in the time t can be written as

$$g_{t,m} = b_{t,m}\vartheta_{t,m}, \tag{4.5}$$

where $b_{t,m}$ denotes whether there is an active path or not (i.e., $b_{t,m} = 1$ means the m^{th} AoD path is activated) and $\vartheta_{t,m}$ represents the complex path gain.

To formulate the more realistic dynamic channel (i.e., the non-activated AoD path at the current time slot[4] t may be activated at time slot $t+1$), a probabilistic channel model with channel prior distribution can be formed as

$$p\left(\boldsymbol{g}^{(T)}, \boldsymbol{b}^{(T)}, \boldsymbol{\vartheta}^{(T)}\right) = p\left(\boldsymbol{g}^{(T)} \mid \boldsymbol{b}^{(T)}, \boldsymbol{\vartheta}^{(T)}\right) p\left(\boldsymbol{b}^{(T)}\right) p\left(\boldsymbol{\vartheta}^{(T)}\right), \tag{4.6}$$

[3] To be noticed, $m = (n_x - 1) \times N_x + n_y$

[4] In this paper, each time slot indicates a single estimation transmit frame.

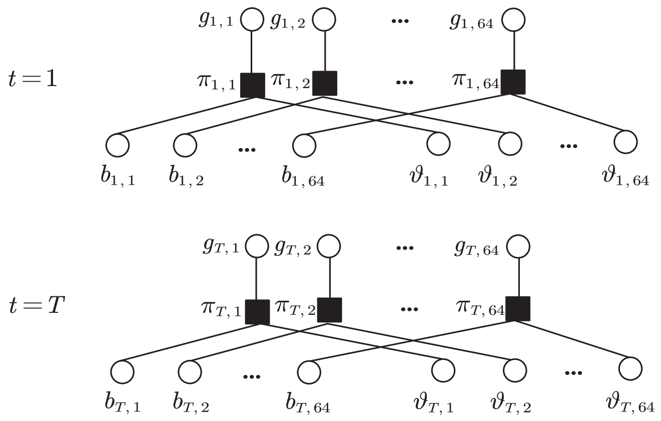


Figure 4.2: Factor graph of 3D-2D-MM of the channel when $N_x = 8$ and $N_y = 8$. The detailed model of the joint hidden support vector $\boldsymbol{b}_t$ and hidden value vector $\boldsymbol{\vartheta}_t$ are illustrated in Fig. 4.3 and Fig. 4.4, separately.

where the channel vector conditional prior is

$$p\left(\boldsymbol{g}^{(T)} \mid \boldsymbol{b}^{(T)}, \boldsymbol{\vartheta}^{(T)}\right) = \prod_{t=1}^{T} \prod_{m=1}^{M} p\left(g_{t,m} \mid b_{t,m}, \vartheta_{t,m}\right) = \prod_{t=1}^{T} \prod_{m=1}^{M} \delta\left(g_{t,m} - b_{t,m}\vartheta_{t,m}\right), \tag{4.7}$$

with $\delta(\cdot)$ is the Dirac delta function. The factor graph of the combined channel can be found in Fig. 4.2, with $\pi_{t,m}$ as the conditional prior $p\left(g_{t,m} \mid b_{t,m}, \vartheta_{t,m}\right)$.

The joint hidden support vector of azimuth and elevation directions and the hidden value vector are detailed as follows:

4.3.1.1 Joint Hidden Support Vector of Azimuth and Elevation

The relationship of the azimuth support $\boldsymbol{a}^{(T)}$ and elevation support $\boldsymbol{e}^{(T)}$ can be found in Fig. 4.3(a), where only one active path is considered.

In our model, the known LEO satellite orbiting and the uncertainty UAV mobility are considered as the relative angular variation rather than the certain location. To be specific, because of the 3D UAV trajectory [115], together with the fixed orbital voyage of the LEO satellite, the relative displacement of the UAV and LEO satellite is continuous. This means for a time-varying case, the azimuth and elevation angles would only change to adjacent degree in the angular domain with certain probabilities.

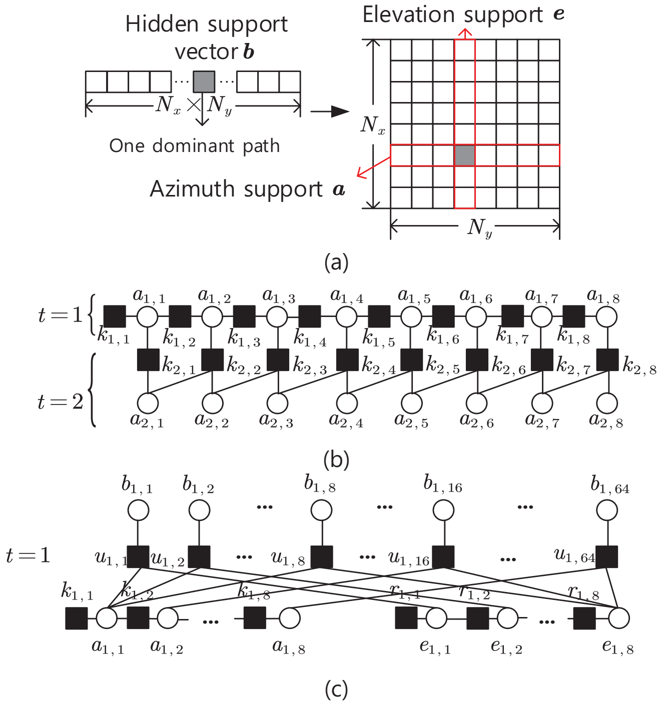

Hidden support vector b
One dominant path
Elevation support e
Azimuth support a
N_x
N_y
$N_x \times N_y$
(a)
$t=1$
$t=2$
$a_{1,1}$ … $a_{1,8}$
$k_{1,1}$ … $k_{1,8}$
$k_{2,1}$ … $k_{2,8}$
$a_{2,1}$ … $a_{2,8}$
(b)
$b_{1,1}$ $b_{1,2}$ $b_{1,8}$ $b_{1,16}$ $b_{1,64}$
$u_{1,1}$ $u_{1,2}$ $u_{1,8}$ $u_{1,16}$ $u_{1,64}$
$k_{1,1}$ $k_{1,2}$ $k_{1,8}$ $r_{1,1}$ $r_{1,2}$ $r_{1,8}$
$a_{1,1}$ $a_{1,2}$ $a_{1,8}$ $e_{1,1}$ $e_{1,2}$ $e_{1,8}$
(c)

Figure 4.3: (a) Hidden support matrix illustration. (b) Factor graph of 2D-MM of the azimuth support vectors when $T = 2$ and $N_x = 8$. (c) Factor graph of the 3D-2D-MM joint hidden support vector $\boldsymbol{b}_t$ when $T = 1$, $N_x = 8$, $N_y = 8$.

These probabilities can be treated as an expression of the mobility for both the UAV and satellite. Practically, parameters can be trained in advance and stored for real-time channel tracking based on the expectation maximization algorithm proposed in [53].

In other words, the spatial sparsity pattern will act highly related with the prior adjacent pattern, i.e., a_{t+1,n_x} has higher probability depends on the adjacent a_{t,n_x-1} in previous slot [36]. It is also verified that the support changes slowly over time [108], which indicates a_{t+1,n_x} has higher probability depends on the a_{t,n_x}. The spatial sparsity and the probabilistic relationship passing through time are called spatial correlation and temporal correlation, respectively. Combining both spatial and temporal correlation is the reason for naming the azimuth support vector model as 2D-MM. The model can be expressed as:

$$\begin{aligned} p\left(\boldsymbol{a}^{(T)}\right) = & \underbrace{p\left(a_{1,1}\right)}_{q_{a_{1,1}}} \prod_{n_x=2}^{N_x} \underbrace{p\left(a_{1,n_x} \mid a_{1,n_x-1}\right)}_{q^{(AS)}_{a_{1,n_x},a_{1,n_x-1}}} \\ & \times \prod_{t=2}^{T} \left(\underbrace{p\left(a_{t,1} \mid a_{t-1,1}\right)}_{q^{(AT)}_{a_{t,1},a_{t-1,1}}} \prod_{n_x=1}^{N_x} \underbrace{p(a_{t,n_x} \mid a_{t-1,n_x-1}, a_{t-1,n_x})}_{q^{(A)}_{a_{t,n_x},a_{t-1,n_x-1},a_{t-1,n_x}}} \right), \end{aligned} \tag{4.8}$$

where $q_{a_{1,1}}$, $q^{(AS)}_{a_{1,n_x},a_{1,n_x-1}}$, $q^{(AT)}_{a_{t,1},a_{t-1,1}}$ and $q^{(A)}_{a_{t,n_x},a_{t-1,n_x-1},a_{t-1,n_x}}$ are the transition probabilities that equal to the factor nodes $k_{1,1}$, k_{1,n_x}, $k_{t,1}$ and k_{t,n_x} in Fig. 4.3(b) and in Fig. 4.3(c) of the azimuth directions. To be noticed, the azimuth supports $a_{t,n_x} \in \{0,1\}$. The illustration of this 2D-MM is in the factor graph in Fig. 4.3(b). The support of the elevation direction of AoD $\boldsymbol{e}^{(T)}$ is modeled in the similar way.

Hence, the total 3D-2D-MM of the AoD supports, which combines both azimuth and elevation direction supports is in Fig. 4.3(c). To be noticed, each hidden support element $b_{t,m}$ in hidden support vector $\boldsymbol{b}_t$ is actually the combination of the n_x^{th} azimuth and n_y^{th} elevation support vectors. The joint conditional prior of channel support probability is given by

$$p\left(\boldsymbol{b}^{(T)}|\boldsymbol{a}^{(T)}, \boldsymbol{e}^{(T)}\right) = \prod_{t,n_x,n_y}^{T,N_x,N_y} p\left(b_{t,n_x,n_y}|a_{t,n_x}, e_{t,n_y}\right). \tag{4.9}$$

4.3.1.2 Gaussian Markov Model of Hidden Value Vector

The illustration of the factor graph of the hidden value vector $\boldsymbol{\vartheta}_t$ and the Gaussian-Markov model is in Fig. 4.4(a) and Fig. 4.4(b), respectively. Due to the path gains change smoothly over time [108], the hidden value vector can be formulated as Gauss-Markov processes [53]

$$\vartheta_{t,m} = (1-\beta)\left(\vartheta_{t-1,m} - \mu\right) + \beta\omega_{t,m} + \mu, \tag{4.10}$$

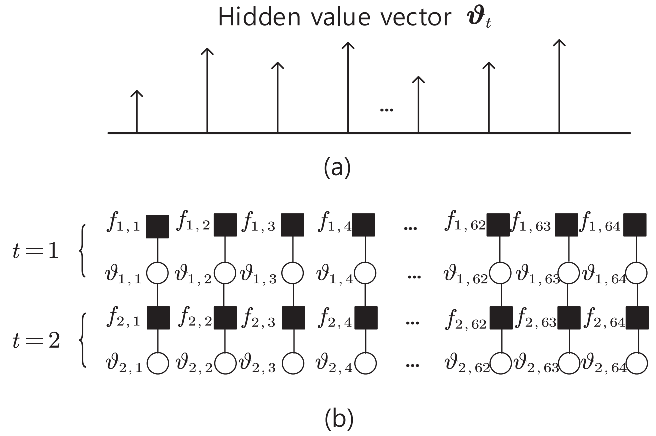

Hidden value vector $\boldsymbol{\vartheta}_t$
...
(a)
$t=1$
$f_{1,1}$ $f_{1,2}$ $f_{1,3}$ $f_{1,4}$ ... $f_{1,62}$ $f_{1,63}$ $f_{1,64}$
$\vartheta_{1,1}$ $\vartheta_{1,2}$ $\vartheta_{1,3}$ $\vartheta_{1,4}$ ... $\vartheta_{1,62}$ $\vartheta_{1,63}$ $\vartheta_{1,64}$
$t=2$
$f_{2,1}$ $f_{2,2}$ $f_{2,3}$ $f_{2,4}$ ... $f_{2,62}$ $f_{2,63}$ $f_{2,64}$
$\vartheta_{2,1}$ $\vartheta_{2,2}$ $\vartheta_{2,3}$ $\vartheta_{2,4}$ ... $\vartheta_{2,62}$ $\vartheta_{2,63}$ $\vartheta_{2,64}$
(b)

Figure 4.4: (a) Hidden value vector illustration at a single time. (b) Factor graph of Gauss-Markov model of the hidden value vectors with $T = 2$, $M = 64$.

where $\beta \in [0, 1]$, $\omega_{t,m} \sim \mathcal{CN}(0, \zeta)$, $\mu \in \mathbb{C}$ is the mean of the process. When $\beta = 0$, then $\vartheta_{t,m} = \vartheta_{t-1,m}$, which indicates that $\boldsymbol{\vartheta}_t$ remain unchanged over time. When $\beta = 1$, then $\vartheta_{t,m} = \beta\omega_{t,m} + \mu$, which means that the hidden value vector shows the i.i.d Gaussian distribution with mean μ over time. If $0 < \beta < 1$, the conditional probability can be written as

$$p\left(\vartheta_{t,m} \mid \vartheta_{t-1,m}\right) \sim \mathcal{CN}\left(\vartheta_{t,m}; (1-\beta)\,\vartheta_{t-1,m} + \beta\mu, \beta^2\zeta\right). \tag{4.11}$$

The distribution of the steady state of the process is $\vartheta_{t,m} \sim \mathcal{CN}(\mu, \sigma^2 = \frac{\beta\zeta}{2-\beta})$. And the joint distribution of $\boldsymbol{\vartheta}^{(T)}$ can be formulated as

$$p\left(\boldsymbol{\vartheta}^{(T)}\right) = \prod_{m=1}^{M} p\left(\vartheta_{1,m}\right) \prod_{t=2}^{T} p\left(\vartheta_{t,m} \mid \vartheta_{t-1,m}\right), \tag{4.12}$$

where $p\left(\vartheta_{1,m}\right)$ and $p\left(\vartheta_{t,m} \mid \vartheta_{t-1,m}\right)$ are the transition probabilities that equal to the factor nodes $f_{1,m}$ and $f_{t,m}$ in Fig. 4.4(b).

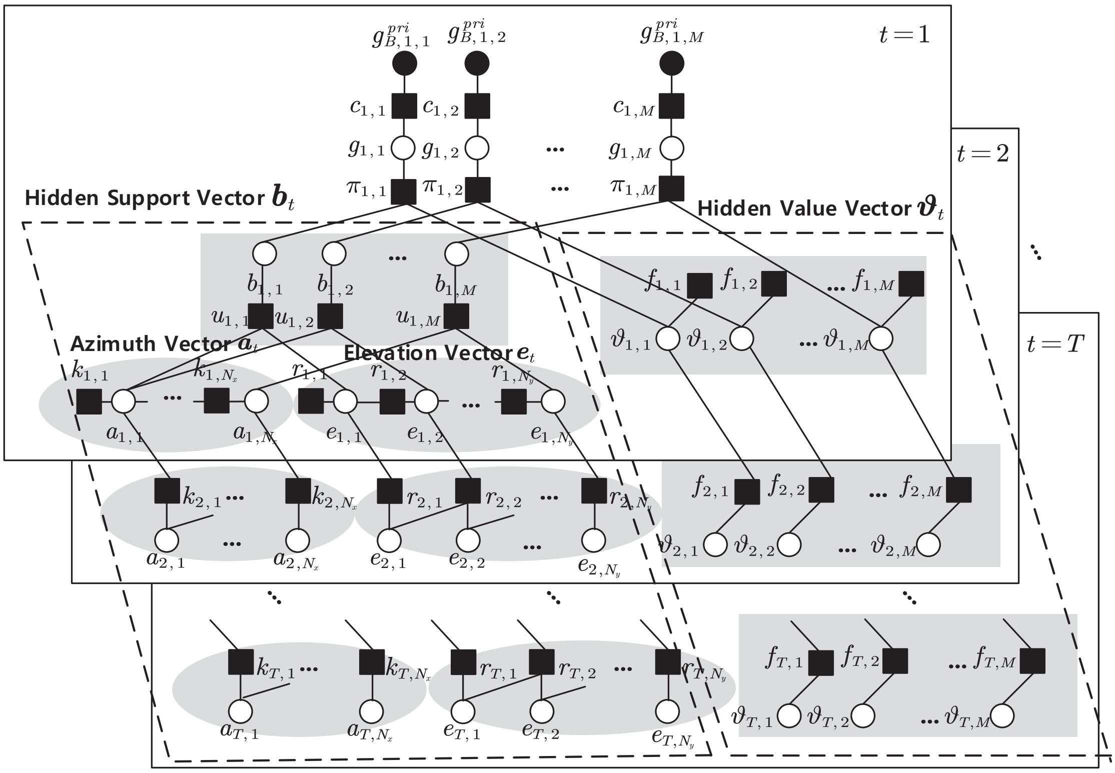


Figure 4.5: Factor graph representation of the proposed 3D-2D-MM.

4.3.2 Dynamic Sparse Channel Tracking Problem Formulation

By taking the angular domain channel (4.4) into consideration, the received signal can be written as

$$\boldsymbol{y}_t^{\mathrm{RX}} = \boldsymbol{g}_t \tilde{\boldsymbol{A}}^H(\boldsymbol{\theta}, \boldsymbol{\phi}) \boldsymbol{S} + \boldsymbol{n}_t^{\mathrm{RX}}. \tag{4.13}$$

with $\boldsymbol{y}_t^{\mathrm{RX}} \in \mathbb{C}^{1 \times N_S}$. Then, the received signal can be rewritten into the standard CS model

$$\boldsymbol{y}_t = \boldsymbol{\Phi} \boldsymbol{g}_t + \boldsymbol{n}_t, \tag{4.14}$$

with $\boldsymbol{y}_t \in \mathbb{C}^{N_S \times 1}$, $\boldsymbol{g}_t = \boldsymbol{g}_t^T$, and the sensing matrix $\boldsymbol{\Phi} = \boldsymbol{S}^T(\tilde{\boldsymbol{A}}^H)^T \in \mathbb{C}^{N_S \times M}$. To be noticed, the transmitted pilot is defined as in **Appendix A** (In this way, $\boldsymbol{\Phi}$ can be treated as an approximate partial orthogonal sensing matrix and provides good performance).

Given the received signals of the t time period $\boldsymbol{y}^{(t)}$, the purpose is to track the dynamic channel vector $\boldsymbol{g}_t$ at time t. The channel vector can be estimated as $\hat{g}_{t,m} = E\left(g_{t,m}|\boldsymbol{y}^{(t)}\right)$, where the expectation is over the marginal posterior

$$p\left(g_{t,m}|\boldsymbol{y}^{(t)}\right) \propto \sum_{\boldsymbol{b}^{(t)}} \int_{\boldsymbol{g}^{(t)}_{-(t,m)},\boldsymbol{\vartheta}^{(t)}} p(\boldsymbol{g}^{(t)}, \boldsymbol{b}^{(t)}, \boldsymbol{\vartheta}^{(t)}, \boldsymbol{y}^{(t)}) \tag{4.15}$$

with $\boldsymbol{g}^{(t)}_{-(t,m)}$ denotes the vector $\boldsymbol{g}^{(t)}$ without the element $g_{t,m}$. Our goal is to obtain the MMSE of the estimated $\boldsymbol{g}$.

4.4 Proposed 3D Dynamic Turbo Approximate Message Passing Algorithm

The factor graph of the total 3D-2D-MM distribution $p(\boldsymbol{g}^{pri}_{\mathcal{B},t}, \boldsymbol{b}^{(t)}, \boldsymbol{\vartheta}^{(t)}, \boldsymbol{y}^{(t)})$ is shown in Fig. 4.5. This factor graph contains two sub-graphs: the joint hidden support $\boldsymbol{b}^{(T)}$ (on the left-hand side of Fig. 4.5) and the hidden value vector $\boldsymbol{\vartheta}^{(T)}$ (on the right-hand side of Fig. 4.5) through time T. The structure of the total hidden support $\boldsymbol{b}^{(T)}$ can be further extended as a 3D-2D-MM factor graph which contains both azimuth support vector $\boldsymbol{a}^{(T)}$ and elevation support vector $\boldsymbol{e}^{(T)}$ through time T. The expression of function nodes is listed in Table 4-B. The details of all the nodes are elaborated in Section 4.3.1.

There are two reasons for utilizing the AMP framework in our model: rigorous analysis and extremely fast runtimes [119]. Conventional AMP runs with the static priors. However, for our special designed factor graph, the prior distribution knowledge for the AMP algorithm changes over time. Due to the complex structure in the total factor graph of the 3D-2D-MM, it is challenging to achieve the exact marginal posterior in (4.15). The extended version called turbo-AMP was proposed in [120] which offered a way to further improve the performance. Conventional turbo compressed sensing is an iterative algorithm which follows the two-step iteration framework [121]: linear minimum mean-square error (LMMSE) estimator and i.i.d. prior-based MMSE estimator. In this

Table 4-B: Table of Factors, Distributions and Functional Forms Associated with Our 3D-2D-MM

Factor	Distribution	Function Form
$c_{t,m}\left(g_{t,m}, g^{pri}_{\mathcal{B},t,m}\right)$	$p(g^{pri}_{\mathcal{B},t,m} \mid g_{t,m})$	$\mathcal{CN}\left(\boldsymbol{g}_t; \boldsymbol{g}^{pri}_{\mathcal{B},t}, v^{pri}_{\mathcal{B},t}\boldsymbol{I}\right)$
$\pi_{t,m}\left(g_{t,m}, b_{t,m}, \vartheta_{t,m}\right)$	$p\left(g_{t,m} \mid b_{t,m}, \vartheta_{t,m}\right)$	$\delta\left(g_{t,m} - b_{t,m}\vartheta_{t,m}\right)$
$f_{1,m}\left(\vartheta_{1,m}\right)$	$p\left(\vartheta_{1,m}\right)$	$\mathcal{CN}\left(\vartheta_{1,m}; \mu, \sigma^2\right)$
$f_{t,m}\left(\vartheta_{t,m}, \vartheta_{t-1,m}\right)$	$p\left(\vartheta_{t,m} \mid \vartheta_{t-1,m}\right)$	$\mathcal{CN}\left(\vartheta_{t,m}; \left(1-\beta\right)\vartheta_{t-1,m} + \beta\mu, \beta^2\zeta\right)$
$k_{1,1}(a_{1,1})$	$p(a_{1,1})$	$(1-\lambda)^{1-a_{1,1}}\lambda^{a_{1,1}}$
$k_{1,m}(a_{1,m}, a_{1,m-1})$	$p(a_{1,m} \mid a_{1,m-1})$	$\begin{cases}\left(q^{(AS)}_{0,1}\right)^{a_{1,m}}\left(1-q^{(AS)}_{0,1}\right)^{(1-a_{1,m})}, a_{1,m-1}=0\\ \left(1-q^{(AS)}_{1,0}\right)^{a_{1,m}}\left(q^{(AS)}_{1,0}\right)^{(1-a_{1,m})}, a_{1,m-1}=1\end{cases}$
$k_{t,1}(a_{t,1}, a_{t-1,1})$	$p(a_{t,1} \mid a_{t-1,1})$	$\begin{cases}\left(q^{(AT)}_{0,1}\right)^{a_{t,1}}\left(1-q^{(AT)}_{0,1}\right)^{(1-a_{t,1})}, a_{t-1,1}=0\\ \left(1-q^{(AT)}_{1,0}\right)^{a_{t,1}}\left(q^{(AT)}_{1,0}\right)^{(1-a_{t,1})}, a_{t-1,1}=1\end{cases}$
$k_{t,m}(a_{t,m}, a_{t,m-1}, a_{t-1,m})$	$p(a_{t,m} \mid a_{t-1,m-1}, a_{t-1,m})$	$\left(q^{(A)}_{a_{t-1,m-1},a_{t-1,m},1}\right)^{a_{t,m}}\left(1-q^{(A)}_{a_{t-1,m-1},a_{t-1,m},1}\right)^{(1-a_{t,m})}$, $\mathrm{a}_{t-1,m-1}, a_{t-1,m} \in \{0,1\}$
$r_{1,1}(e_{1,1})$	$p(e_{1,1})$	$(1-\lambda)^{1-e_{1,1}}\lambda^{e_{1,1}}$
$r_{1,m}(e_{1,m}, e_{1,m-1})$	$p(e_{1,m} \mid e_{1,m-1})$	$\begin{cases}\left(q^{(ES)}_{0,1}\right)^{e_{1,m}}\left(1-q^{(ES)}_{0,1}\right)^{(1-e_{1,m})}, e_{1,m-1}=0\\ \left(1-q^{(ES)}_{1,0}\right)^{e_{1,m}}\left(q^{(ES)}_{1,0}\right)^{(1-e_{1,m})}, e_{1,m-1}=1\end{cases}$
$r_{t,1}(e_{t,1}, e_{t-1,1})$	$p(e_{t,1} \mid e_{t-1,1})$	$\begin{cases}\left(q^{(ET)}_{0,1}\right)^{e_{t,1}}\left(1-q^{(ET)}_{0,1}\right)^{(1-e_{t,1})}, e_{t-1,1}=0\\ \left(1-q^{(ET)}_{1,0}\right)^{e_{t,1}}\left(q^{(ET)}_{1,0}\right)^{(1-e_{t,1})}, e_{t-1,1}=1\end{cases}$
$r_{t,m}(e_{t,m}, e_{t-1,m-1}, e_{t-1,m})$	$p(e_{t,m} \mid e_{t-1,m-1}, e_{t-1,m})$	$\left(q^{(E)}_{e_{t-1,m-1},e_{t-1,m},1}\right)^{e_{t,m}}\left(1-q^{(E)}_{e_{t-1,m-1},e_{t-1,m},1}\right)^{(1-e_{t,m})}$, $\mathrm{e}_{t-1,m-1}, e_{t-1,m} \in \{0,1\}$
$u_{t,m}(b_{t,m}, a_{t,n_x}, e_{t,n_y})$	$p(b_{1,m} \mid a_{t,n_x}, e_{t,n_y})$	$a_{t,n_x}e_{t,n_y}(\gamma_{\boldsymbol{b}_t})^{b_{t,m}}(1-\gamma_{\boldsymbol{b}_t})^{(1-b_{t,m})} + (1-a_{t,n_x}e_{t,n_y})\delta(b_{t,m})$

section, to perfectly exploit the dynamic channel structure, we elaborate the details of the proposed algorithm based on turbo iterative framework with the proposed 3D-2D-MM MMSE denoiser. The proposed algorithm will be introduced in three parts: Module $\mathcal{A}$ LMMSE Estimator, Module $\mathcal{B}$ message passing MMSE denoiser, and message passing through time.

4.4.1 Module $\mathcal{A}$: LMMSE Estimator

In Module $\mathcal{A}$, the angular channel vector $\boldsymbol{g}_t$ is estimated based on the received signal $\boldsymbol{y}_t$ with the prior distribution $\mathcal{CN}\left(\boldsymbol{g}_t; \boldsymbol{g}^{pri}_{\mathcal{A},t}, v^{pri}_{\mathcal{A},t}\boldsymbol{I}\right)$, where the extrinsic mean and variance are defined as [121]

$$\boldsymbol{g}^{post}_{\mathcal{A},t} = \boldsymbol{g}^{pri}_{\mathcal{A},t} + \frac{v^{pri}_{\mathcal{A},t}}{v^{pri}_{\mathcal{A},t} + \sigma_e^2}\boldsymbol{\Phi}_t^H\left(\boldsymbol{y}_t - \boldsymbol{\Phi}\boldsymbol{g}^{pri}_{\mathcal{A},t}\right) \tag{4.16}$$

and

$$v_{\mathcal{A},t}^{post} = v_{\mathcal{A},t}^{pri} - \frac{N_S}{M} \frac{\left(v_{\mathcal{A},t}^{pri}\right)^2}{v_{\mathcal{A},t}^{pri} + \sigma_e^2}, \tag{4.17}$$

respectively. The extrinsic distribution of $\boldsymbol{g}_t$ is given by

$$\mathcal{CN}\left(\boldsymbol{g}_t; \boldsymbol{g}_{\mathcal{A},t}^{post}, v_{\mathcal{A},t}^{post}\boldsymbol{I}\right) \propto \mathcal{CN}\left(\boldsymbol{g}_t; \boldsymbol{g}_{\mathcal{A},t}^{pri}, v_{\mathcal{A},t}^{pri}\boldsymbol{I}\right) \mathcal{CN}\left(\boldsymbol{g}_t; \boldsymbol{g}_{\mathcal{A},t}^{ext}, v_{\mathcal{A},t}^{ext}\boldsymbol{I}\right), \tag{4.18}$$

where the extrinsic mean and variance are respectively given by

$$\boldsymbol{g}_{\mathcal{A},t}^{ext} = v_{\mathcal{A},t}^{ext} \left(\frac{\boldsymbol{g}_{\mathcal{A},t}^{post}}{v_{\mathcal{A},t}^{post}} - \frac{\boldsymbol{g}_{\mathcal{A},t}^{pri}}{v_{\mathcal{A},t}^{pri}} \right) \tag{4.19}$$

$$v_{\mathcal{A},t}^{ext} = \left(\frac{1}{v_{\mathcal{A},t}^{post}} - \frac{1}{v_{\mathcal{A},t}^{pri}} \right)^{-1}. \tag{4.20}$$

4.4.2 Module $\mathcal{B}$: Message Passing MMSE Denoiser

In module $\mathcal{B}$, the message passing MMSE denoiser is executed by exploiting the spatial sparsity details of the proposed 3D-2D-MM.

As the core structure of the turbo compressed sensing, the extrinsic mean and variance are transmitted from the LMMSE estimator in module $\mathcal{A}$, which means $\boldsymbol{g}_{\mathcal{B},t}^{pri} = \boldsymbol{g}_{\mathcal{A},t}^{ext}$ and $v_{\mathcal{B},t}^{pri} = v_{\mathcal{A},t}^{ext}$ [121]. For message passing algorithm, the basic assumption is that $\boldsymbol{g}_{\mathcal{B},t}^{pri} = \boldsymbol{g}_t + \boldsymbol{z}_t$ where $\boldsymbol{z}_t \sim \mathcal{CN}(0, v_{\mathcal{B},t}^{pri}\boldsymbol{I})$ is independent of $\boldsymbol{g}_t$ [121]. Aiming to calculate the approximate posterior distributions of $p(g_{t,m}|\boldsymbol{g}_{\mathcal{B}}^{pri})$ instead of (4.15), the sum-product message passing rule is used, which carefully follows the message passing structure shown in Fig. 4.5. The formulation details can be found in **Appendix B** and the procedure is summarized as follows:

- Firstly, the message passing over the Gaussian hidden value vector $\boldsymbol{\vartheta}_t$ is given by (B.1) with μ and σ^2 as the input to update the message $\upsilon_{\vartheta_{t,m} \to \pi_{t,m}}$.

Algorithm 4.1: 3D-2D-MM Azimuth and Elevation Support Estimation

```
Input: $\rho_{t,n_x}^{(A)in}$, $\rho_{t,n_x}^{(A)dyna}$, $\rho_{t,n_y}^{(E)in}$, $\rho_{t,n_y}^{(E)dyna}$.
Output: $\rho_{t,n_x}^{(A)out}$, $\rho_{t,n_y}^{(E)out}$.
% Azimuth Estimation:
if $t = 1$ then
    Calculate $\upsilon_{u_{t,m} \to a_{t,n_x}}$ as in (B.7)
    Initialization: $\gamma_{t,1}^{(A)f} = \lambda$, $\gamma_{t,N_x}^{(A)b} = \frac{1}{2}$
    for $n_x = 2, ..., N_x$ do
        Use (B.8) to calculate $\gamma_{t,n_x}^{(A)f}$.
    end for
    for $n_x = 1, ..., N_x - 1$ do
        Use (B.9) to calculate $\gamma_{t,n_x}^{(A)b}$.
    end for
else if $t > 1$ then
    Initialization:$\gamma_{t,1}^{(A)f} = q_{0,1}^{(AT)}(1 - \rho_{t,1}^{ac}) + \left(1 - q_{1,0}^{(AT)}\right) \rho_{t,1}^{ac}$,
    $\gamma_{t,N_x}^{(A)b} = \frac{1}{2}$.
    for $n_x = 2, ..., N_x$ do
        Calculate the $\gamma_{t,n_x}^{(A)f}$ as in (B.10)
    end for
    for $n_x = 1, ..., N_x - 1$ do
        Calculate $\gamma_{t,n_x}^{(A)b} = \frac{1}{1+\gamma^{(A)}}$ use (B.11).
    end for
end if
Calculate AoD azimuth support vector $\rho_{t,n_x}^{(A)out}$ as in (B.12).
%Message Passing Over Azimuth to Elevation support:
Calculate individual message $\rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out}$ using (B.14).
Calculate message $\upsilon_{a_{t,n_x} \to u_{t,m}}(a_{t,n_x})$ and $\upsilon_{u_{t,m} \to e_{t,n_y}}(e_{t,n_y})$ as in (B.13) and (B.15),
respectively.
% Elevation Estimation:
if $t = 1$ then
    Calculate $\upsilon_{u_{t,m} \to e_{t,n_y}}$ as in (B.17)
    Initialization: $\gamma_{t,1}^{(E)f} = \lambda$, $\gamma_{t,N_y}^{(E)b} = \frac{1}{2}$
    for $n_x = 2, ..., N_y$ do
        Use (B.18) to calculate $\gamma_{t,n_y}^{(E)f}$.
    end for
    for $n_y = 1, ..., N_y - 1$ do
        Use (B.19) to calculate $\gamma_{t,n_y}^{(E)b}$.
    end for
else if $t > 1$ then
    Initialization:$\gamma_{t,N_y}^{(E)b} = \frac{1}{2}$,
    $\gamma_{t,1}^{(E)f} = q_{0,1}^{(ET)}(1 - \rho_{t,1}^{(E)dyna}) + \left(1 - q_{1,0}^{(ET)}\right) \rho_{t,1}^{(E)dyna}$.
    for $n_y = 2, ..., N_y$ do
        Calculate the $\gamma^{(E)f}_{t,n_y}$ as in (B.20)
    end for
    for $n_y = 1, ..., N_y - 1$ do
        Calculate $\gamma^{(E)b}_{t,n_y} = \frac{1}{1+\gamma^{(E)}}$ use (B.21).
    end for
end if
Calculate AoD elevation support vector $\rho^{(E)out}_{t,n_y}$ as in (B.22).
```

- Secondly, the message is passed over the path $g_{t,m} \rightarrow \pi_{t,m} \rightarrow b_{t,m} \rightarrow u_{t,m} \rightarrow a_{t,n_x}$ using (B.2) to (B.4), with the input $\boldsymbol{g}^{pri}_{\mathcal{B},t}$ and $v^{pri}_{\mathcal{B},t}$ from module $\mathcal{A}$.

- Then the forward-backward message passing is performed over the 3D-2D-MM with sequence three paths : the 2D-MM of the azimuth support $\boldsymbol{a}_t$, path $a_{t,n_x} \rightarrow u_{t,m} \rightarrow e_{t,n_y}$ and the 2D-MM of the elevation support $\boldsymbol{e}_t$. The details of the 3D-2D-MM azimuth and elevation support nonzero probability estimation are summarized in **Algorithm 4.1**.

- The message is finally passed back over the path $e_{t,n_y} \rightarrow u_{t,m} \rightarrow b_{t,m} \rightarrow \pi_{t,m} \rightarrow g_{t,m}$ using (B.23) to (B.26).

Based on the calculated updated messages $\upsilon_{\pi_{t,m} \rightarrow g_{t,m}}$, the posterior distributions can be written as

$$p\left(g_{t,m}|\boldsymbol{g}^{pri}_{\mathcal{B},t}\right) \propto \upsilon_{\pi_{t,m} \rightarrow g_{t,m}}\left(g_{t,m}\right) \upsilon_{ct,m \rightarrow g_{t,m}}\left(g_{t,m}\right), \tag{4.21}$$

where $\upsilon_{ct,m \rightarrow g_{t,m}}\left(g_{t,m}\right) = \mathcal{CN}\left(g_{t,m}; g^{pri}_{\mathcal{B},t,m}, v^{pri}_{\mathcal{B},t}\right)$. Then the posterior mean and variance of each element of $\boldsymbol{g}_t$ can be updated as

$$g^{post}_{\mathcal{B},t,m} = \int_{g_{t,m}} g_{t,m} p\left(g_{t,m}|\boldsymbol{g}^{pri}_{\mathcal{B},t}\right) \tag{4.22}$$

and

$$v_{\mathcal{B},t}^{post} = \frac{1}{M} \sum_{m=1}^{M} \mathrm{Var}\left(g_{t,m}|\boldsymbol{g}_{\mathcal{B},t}^{pri}\right), \tag{4.23}$$

where $\mathrm{Var}\left(g_{t,m}|\boldsymbol{g}_{\mathcal{B},t}^{pri}\right)$ denotes the conditional variance of $g_{t,m}$ given $\boldsymbol{g}_{\mathcal{B},t}^{pri}$. Then similar to (4.19) and (4.20), the extrinsic update mean and covariance can be written as

$$\boldsymbol{g}_{\mathcal{A},t}^{pri} = \boldsymbol{g}_{\mathcal{B},t}^{ext} = v_{\mathcal{A},t}^{pri} \left(\frac{\boldsymbol{g}_{\mathcal{B},t}^{post}}{v_{\mathcal{B},t}^{post}} - \frac{\boldsymbol{g}_{\mathcal{B},t}^{pri}}{v_{\mathcal{B},t}^{pri}} \right) \tag{4.24}$$

and

$$v_{\mathcal{A},t}^{pri} = v_{\mathcal{B},t}^{ext} = \left(\frac{1}{v_{\mathcal{B},t}^{post}} - \frac{1}{v_{\mathcal{B},t}^{pri}} \right)^{-1}. \tag{4.25}$$

4.4.3 Message Passing Through Time

For channel tracking, temporal correlation can be utilized for better recovery with prior information provided. Therefore, we also consider the message passing through time. There are mainly three paths passing through time: the path of the Gaussian hidden value vector $\boldsymbol{\vartheta}_t$, the path of the azimuth support $\mathbf{a}_t$ and the path of the elevation support $\mathbf{e}_t$. The sequence paths can be observed in Fig. 4.6.

4.4.3.1 The Gaussian Hidden Value Vector $\boldsymbol{\vartheta}_t$ Passing Through Time

The message passing from factor node $\pi_{t,m}$ to variable node $\vartheta_{t,m}$ should be

$$v_{\pi_{t,m}\rightarrow\vartheta_{t,m}}^{true}\left(\vartheta_{t,m}\right) = \rho_{t,m}^{out}\mathcal{CN}\left(\vartheta_{t,m}; \boldsymbol{g}_{\mathcal{B},t}^{pri}, v_{\mathcal{B},t}^{pri}\right) + \left(1-\rho_{t,m}^{out}\right)\mathcal{CN}\left(0; \boldsymbol{g}_{\mathcal{B},t}^{pri}, v_{\mathcal{B},t}^{pri}\right). \tag{4.26}$$

To be notice, if $b_{t,m} = 0$, $g_{t,m}$ would be 0 in (4.5), which makes $\vartheta_{t,m}$ unobservable. To solve this kind of problem, similar in [53], a threshold which is slightly less than 1 is

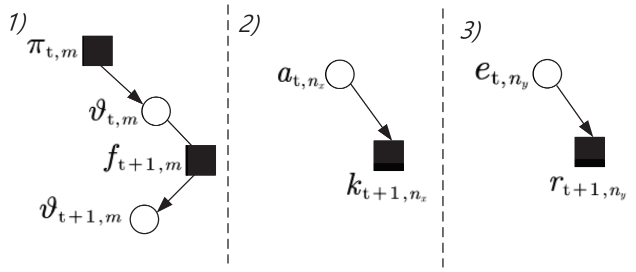


Figure 4.6: The illustration of the message passing sequence paths through time, which shows details of the equation (4.27) to (4.36).

introduced. Hence, the modified message passing is given by

$$\upsilon_{\pi_{t,m}\to\vartheta_{t,m}}\left(\vartheta_{t,m}\right) = \mathcal{CN}\left(\vartheta_{t,m};\mu_{t,m}^{in},(\sigma_{t,m}^{in})^2\right), \tag{4.27}$$

with

$$\left(\mu_{t,m}^{in},(\sigma_{t,m}^{in})^2\right) = \begin{cases} \left(\frac{1}{\epsilon}\boldsymbol{g}_{\mathcal{B},t}^{pri},\frac{1}{\epsilon^2}\upsilon_{\mathcal{B},t}^{pri}\right) & ,\rho_{t,m}^{out}\leqslant \mathrm{Th}, \quad (4.28) \\ \left(\boldsymbol{g}_{\mathcal{B},t}^{pri},\upsilon_{\mathcal{B},t}^{pri}\right) & ,\rho_{t,m}^{out} > \mathrm{Th}, \quad (4.29) \end{cases}$$

where ϵ is a small positive value close to 0. Then the message passing from factor node $f_{t+1,m}$ to variable node $\vartheta_{t+1,m}$ is formed as

$$\upsilon_{f_{t+1,m}\to\vartheta_{t+1,m}}\left(\vartheta_{t+1,m}\right) = \mathcal{CN}\left(\vartheta_{t+1,m};\mu_{t+1,m}^{dyna},\left(\sigma_{t+1,m}^{dyna}\right)^2\right), \tag{4.30}$$

where

$$\mu_{t+1,m}^{dyna} = (1-\beta)\left(\frac{\left(\sigma_{t,m}^{dyna}\right)^2\left(\sigma_{t,m}^{in}\right)^2}{\left(\sigma_{t,m}^{dyna}\right)^2+\left(\sigma_{t,m}^{in}\right)^2}\right)\times\left(\frac{\mu_{t,m}^{dyna}}{\left(\sigma_{t,m}^{dyna}\right)^2}+\frac{\mu_{t,m}^{in}}{\left(\sigma_{t,m}^{in}\right)^2}\right)+\beta\mu \tag{4.31}$$

and

$$\left(\sigma_{t+1,m}^{dyna}\right)^2 = (1-\beta)^2\left(\frac{\left(\sigma_{t,m}^{dyna}\right)^2\left(\sigma_{t,m}^{in}\right)^2}{\left(\sigma_{t,m}^{dyna}\right)^2+\left(\sigma_{t,m}^{in}\right)^2}\right)+\beta^2\zeta. \tag{4.32}$$

4.4.3.2 The Azimuth Support $\mathbf{a}_t$ Passing Through Time

The azimuth support prior information passing to the next time $\upsilon_{a_{t,n_x} \to k_{t+1,n_x}}(a_{t,n_x})$ can be given as

$$\upsilon_{a_{t,n_x} \to k_{t+1,n_x}}(a_{t,n_x}) = \rho_{t+1,n_x}^{(A)dyna} \delta\left(a_{t,n_x} - 1\right) + \left(1 - \rho_{t+1,n_x}^{(A)dyna}\right) \delta\left(a_{t,n_x}\right), \tag{4.33}$$

with

$$\rho_{t+1,n_x}^{(A)dyna} = \frac{\gamma_{t,n_x}^{(A)f} \gamma_{t,n_x}^{(A)b} \rho_{t,n_x}^{(A)in}}{\gamma_{t,n_x}^{(A)f} \gamma_{t,n_x}^{(A)b} \rho_{t,n_x}^{(A)in} + (1 - \gamma_{t,n_x}^{(A)f})(1 - \gamma_{t,n_x}^{(A)b})(1 - \rho_{t,n_x}^{(A)in})}. \tag{4.34}$$

4.4.3.3 The Elevation Support $\boldsymbol{e}_t$ Passing Through Time

Similarly, the elevation support prior information passing to the next time can be written as

$$\upsilon_{e_{t,n_y} \to r_{t+1,n_y}}(e_{t,n_y}) = \rho_{t+1,n_y}^{(E)dyna} \delta\left(e_{t,n_y} - 1\right) + \left(1 - \rho_{t+1,n_y}^{(E)dyna}\right) \delta\left(e_{t,n_y}\right), \tag{4.35}$$

with

$$\rho_{t+1,n_y}^{(E)dyna} = \frac{\gamma_{t,n_y}^{(E)f} \gamma_{t,n_y}^{(E)b} \rho_{t,n_y}^{(E)in}}{\gamma_{t,n_y}^{(E)f} \gamma_{t,n_y}^{(E)b} \rho_{t,n_y}^{(E)in} + (1 - \gamma_{t,n_y}^{(E)f})(1 - \gamma_{t,n_y}^{(E)b})(1 - \rho_{t,n_y}^{(E)in})}. \tag{4.36}$$

Finally, the overall proposed 3D-DTAMP is summarized in **Algorithm 4.2**.

4.5 Numerical Results

In this section, the complexity and the recovery performance of the proposed algorithm are evaluated. For space-air communication system, the sparsity ratio for both azimuth and elevation directions is $\lambda = \frac{1}{N_x} = \frac{1}{N_y}$ which indicates only one dom-

```
Algorithm 4.2: 3D-DTAMP Dynamic Channel Tracking
Input: $\boldsymbol{y}_1, ..., \boldsymbol{y}_T$, sensing matrix $\boldsymbol{\Phi}$.
Output: $\hat{\boldsymbol{g}}_1, ..., \hat{\boldsymbol{g}}_T$
for t=1,...,T do
    Initialization:$\boldsymbol{g}_{\mathcal{A},t}^{pri} = \mathbf{0}$, $v_{\mathcal{A},t}^{pri} = \sigma^2$
    while not converge do
    %Module $\mathcal{A}$ LMMSE estimator:
        Update $\boldsymbol{g}_{\mathcal{A},t}^{post}$ and $v_{\mathcal{A},t}^{post}$ in (4.16) and (4.17), respectively.
        Calculate $\boldsymbol{g}_{\mathcal{B},t}^{pri} = \boldsymbol{g}_{\mathcal{A},t}^{ext}$, $v_{\mathcal{B},t}^{pri} = v_{\mathcal{A},t}^{ext}$ using (4.19) and (4.20).
    %Module $\mathcal{B}$ 3D-2D-MM MMSE denoiser:
        Message passing over Gaussian hidden value vector $\boldsymbol{\vartheta}_t$ using (B.1).
        Message passing over $g_{t,m} \to \pi_{t,m} \to b_{t,m} \to u_{t,m} \to a_{t,n_x}$ using (B.2)-(B.6)
        AoD azimuth support $\boldsymbol{a}_t$ and elevation support $\boldsymbol{e}_t$ estimation in Algorithm 4.1.
        Message passing over elevation support $e_{t,n_y} \to u_{t,m} \to b_{t,m} \to \pi_{t,m} \to g_{t,m}$ using (B.23)-(B.26)
        Calculate the approximate posterior distributions $p\left(g_{t,m}|\boldsymbol{g}_{\mathcal{B},t}^{pri}\right)$ using (4.21).
        Update $\boldsymbol{g}_{\mathcal{B},t,m}^{post}$ and $v_{\mathcal{B},t}^{post}$ using (4.22) and (4.23).
        Update $\boldsymbol{g}_{\mathcal{A},t}^{pri} = \boldsymbol{g}_{\mathcal{B},t}^{ext}$ and $v_{\mathcal{A},t}^{pri} = v_{\mathcal{B},t}^{ext}$ using (4.24) and (4.25).
        Repeat previous steps until convergence.
    end while
    Update dynamic messages to the next time slot $(t+1)$: $v_{a_{t,n_x} \to k_{t+1,n_x}}$, $v_{e_{t,n_y} \to r_{t+1,n_y}}$ and $v_{f_{t+1,m} \to \vartheta_{t+1,m}}$ using (4.33), (4.35) and (4.30).
end for
```

inant LoS ray. The parameters are set as follows[5]: the azimuth spatial correlation parameters $\{q_{0,1}^{(AS)}, q_{1,0}^{(AS)}\} = \{0.05, 0.05\}$, the azimuth temporal correlation parameters $\{q_{0,1}^{(AT)}, q_{1,0}^{(AT)}\} = \{0.05, 0.05\}$, the azimuth joint parameters $\{q_{1,1,1}^{(A)}, q_{0,0,1}^{(A)}, q_{0,1,1}^{(A)}, q_{1,0,1}^{(A)}\} = \{0.9958, 0.0013, 0.3276, 0.3936\}$, the elevation spatial correlation parameters $\{q_{0,1}^{(ES)}, q_{1,0}^{(ES)}\} = \{0.025, 0.025\}$, the elevation temporal correlation parameters $\{q_{0,1}^{(ET)}, q_{1,0}^{(ET)}\} = \{0.025, 0.175\}$, the elevation joint parameters $\{q_{1,1,1}^{(E)}, q_{0,0,1}^{(E)}, q_{0,1,1}^{(E)}, q_{1,0,1}^{(E)}\} = \{0.99546, 0.0007, 0.5, 0.1078\}$, the hidden value vector parameters $\{\mu, \sigma^2, \beta, \zeta\} = \{0, \frac{1}{3}, 0.5, 1\}$, $\mathrm{Th} = 1 - 10^{-2}$, $\gamma_{\boldsymbol{b}_t} = 1$ and $\epsilon = 10^{-7}$. The benchmark algorithms which can be employed in our 3D channel model are SBL and D-OMP proposed in [42] and [108], respectively.

[5] The expectation maximization algorithm is proposed for training parameters in [53]. Since our paper is focused on the channel tracking based on known probabilities, we recommend readers to refer to [53] for details of the parameters trained algorithm. The parameters in this paper are referred 2D settings in [112].

4.5.1 Complexity Analysis

The complexity of the LMMSE estimator in module $\mathcal{A}$ is $\mathcal{O}\left(MN_s\right)$ which is mainly determined by the multiplication of the matrix. The complexity of the proposed MMSE denoiser in module $\mathcal{B}$ is the simple sum-product message passing rule with the $\mathcal{O}\left(N_x + N_y\right)$ complexity. To sum up, the total complexity of the proposed algorithm over time T is $\mathcal{O}\left(T\left(MN_s + N_x + N_y\right)\right)$ for each iteration.

As shown in Fig. 4.7, the SBL takes longer computation time, especially for larger antenna elements. This is caused by the required higher computational complexity of the multiplication and inversion of the matrix. Compared with SBL, the simulation time of both the proposed algorithm and D-OMP drops dramatically. To be noticed, the complexity of D-OMP is $\mathcal{O}\left(T\left(N_s\left(M+1+1\right)\right)\right)$ over total time period T. With the number of antennas increase, the proposed algorithm shows comparable computation time.

4.5.2 Simulation Results

To evaluate the recovery performance of the algorithms, we refer the performance metric which is called the time-averaged normalized mean square error (TNMSE) as follows [53]

$$\text{TNMSE} = \frac{1}{T}\sum_{t}^{T}\frac{\left\|\boldsymbol{g}_t - \hat{\boldsymbol{g}}_t\right\|^2}{\left\|\boldsymbol{g}_t\right\|^2}, \tag{4.37}$$

where $\hat{\boldsymbol{g}}_t$ is the estimated result of $\boldsymbol{g}_t$ in time t.

In Fig.4.8, we further compare the TNMSE performance versus SNR when the total number of antenna array elements is $M = 64$ and 256, respectively. It can be observed that under the different numbers of antennas, the proposed algorithm achieves sufficient performance gain over the benchmark algorithms SBL and D-OMP. This proves the advantage of effectively exploiting the detail dynamic azimuth and elevation spatial sparsity structure of the channel in the proposed algorithm.

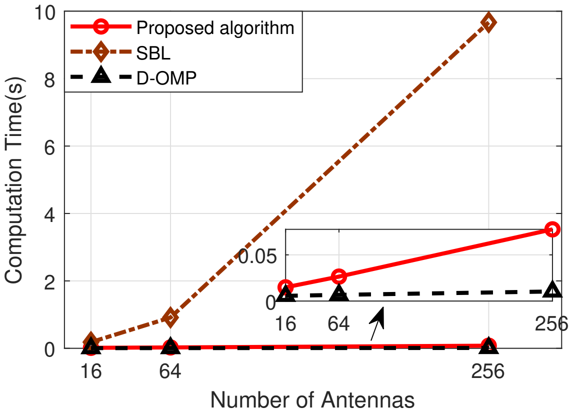


Figure 4.7: Computation time of various algorithms versus the number of antennas when $\text{SNR} = 15\text{dB}, N_s = 10, T = 50, M = 16, 64, 256$ and $N_x = N_y = 4, 8, 16$.

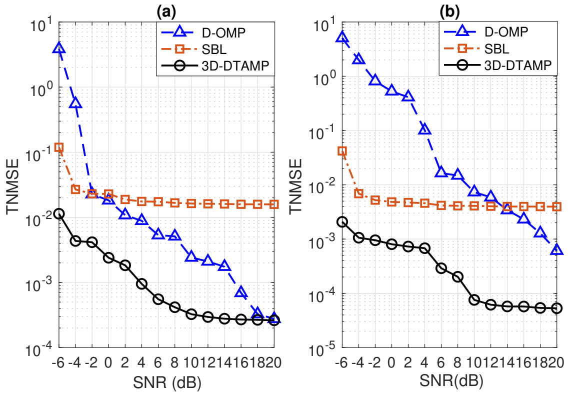


Figure 4.8: TNMSE versus SNR when $T = 35, N_s = 20$. (a) $M = 64, N_x = N_y = 8$. (b) $M = 256, N_x = N_y = 16$.

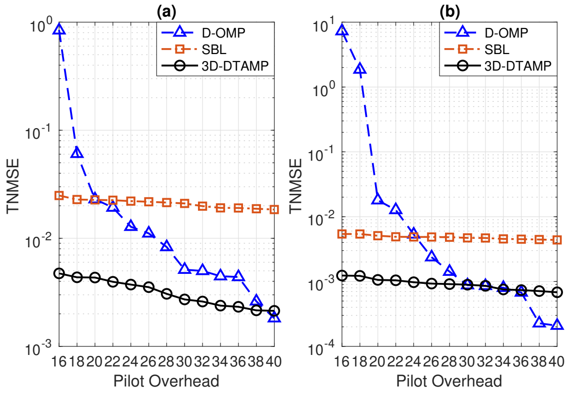


Figure 4.9: TNMSE versus pilot overheads when $T = 35, \text{SNR} = 0dB$. (a) $M = 64, N_x = N_y = 8$. (b) $M = 256, N_x = N_y = 16$.

The impact of the pilot number is further compared in Fig.4.9. It is obvious that all algorithms have better recovery performance with the pilot number increase, the proposed algorithm always remains the superior performance gain. Moreover, the results also demonstrate that the proposed algorithm can achieve a solid performance with a smaller size of the pilot overheads. The reason is that conventionally, increasing pilot number can improve the recovery performance based on multiple estimations through time, while our proposed algorithm can efficiently achieve good accuracy by exploring the dynamic channel structure.

The impact of the antenna size is further compared in Fig.4.10. It is obvious that with antenna size increases (also the virtual grid increases) the proposed algorithm shows superior performance gain. The reason is because that with denser virtual grid the estimated 3D support can be much closer to the true angle.

4.6 Summary

In this chapter, we investigated the 3D channel modelling and tracking for Ka-band space-air communications with the orbit movement of an LEO satellite and 3D UAV trajectory in SAGIN. The spatial correlation on the continuous relative displacement between the UAV and satellite, and the temporal correlation of the dynamic channel were explored. We proposed a statistical dynamic channel model called 3D-2D-MM in which the insight probability relationship of both hidden value vector and joint hidden support vector has been developed. We further designed a novel 3D-DTAMP algorithm to recursively track the dynamic channel with additional 3D-2D-MM prior information. The benefits of the proposed algorithm can be summarized in three aspects. Firstly, the message passing rule has taken the more realistic 3D channel environment into account. Secondly, the consideration of the sparsity structure of the dynamic channel from both spatial and temporal domains have enhanced the recovery performance. Finally, it was more comparable in tracking a longer time series dynamic channel with lower pilot overheads. Analytical and numerical results have shown that the proposed algorithm

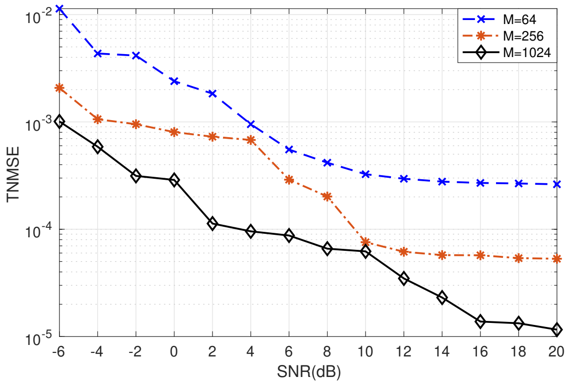


Figure 4.10: TNMSE versus SNR when $T = 35, N_S = 20$.

achieved superior recovery performance against two other benchmark algorithms as well as required fewer pilot overheads and time complexity.

Chapter 5

3D on-grid and off-grid Channel Tracking for MU Space-Air Communications

The work in Chapter 4 takes SU on-grid space-air communications into consideration. However, in real-world, one satellite is normally communicate with multiple UAVs and base stations to provide wider coverage. Therefore, it is more realistic to further consider the relationships in MU space-air links. Moreover, apart from on-grid azimuth and elevation estimation, off-grid offset estimation can further increase the accuracy of the CSI tracking. Hence, in this chapter, channel tracking for 3D on-grid and off-grid MU space-air communications system is studied. This chapter is organized as follows. The related work and main contributions are introduced in Section 5.1. The downlink MU space-air communications system model is introduced in Section 5.2. In Section 5.3, the 3D on-grid and off-grid MD-MM channel model are elaborated. The problem of channel tracking is formulated in Section 5.4. The proposed algorithm MD-DTAMP and the 3D gradient off-grid estimation scheme are presented in Section 5.5. Numerical results and conclusions of this chapter are then provided in Section 5.6 and Section 5.7, respectively.

5.1 Introduction

5.1.1 Related Work

Jointly considering both spatial correlation and temporal correlation can improve the channel tracking performance [110], [112], [122], [123]. [122] proposed the channel tracking scheme based on the prior-aided iterative angle and Doppler shift estimation algorithm and with the help of the previously estimated channel parameters. [110] decoupled the channel information into AoAs/AoDs information and gain knowledge. A modified unscented Kalman filter tracks the AoAs/ AoDs, and the gain is estimated from the proposed beam training. [112] and [123] formulated the more realistic dynamic models of the AoDs support over time through hidden Markov model in flat and 3D space, respectively.

Although our work [123] in the last chapter considered a more realistic 3D space-air communications with only one UAV, the multiple-UAVs system can provide wider coverage in practice. Compared to one UAV in the system, for multiple UAVs, there will have more than one structured sparse points on the quantized grids to represent the MU spatial correlation with sharing azimuth or elevation supports. Hence, a more complicated scenario with multiple UAVs should be further explored. The algorithms proposed in all reviewed related works assumed that the true angles lie on the quantized grids. However, in the real world, the true angle is continuously distributed [124]. To solve the off-grid problem, [124] and [125] all proposed algorithms based on a two-dimensional (2D) flat environment. Therefore, none of the mentioned work considered the 3D off-grid case of the channel model.

5.1.2 Motivation and Contribution

Motivated by the above challenges, we consider the MU space-air links with multiple UAVs in the system to recursively track the 3D off-grid dynamic channel with smaller training overheads, comparable complexity, and improved estimation performance. The main contributions are summarized as follows:

- We develop a statistical 3D on-grid and off-grid channel model called MD-MM for space-air communication links with 3D navigation multiple UAVs and orbit LEO satellite in the system. The MD-MM captures the structured sparsity of the MU azimuth and elevation spatial domains and the probabilistic relationship between MU and SU support.

- Based on the proposed MD-MM channel model, we propose a novel MD-DTAMP algorithm which can recursively track the 3D on-grid dynamic channel over time.

- We further develop a gradient azimuth and elevation 3D off-grid estimation scheme for a more realistic channel model, which iteratively update the optimum 3D offset through gradient calculation.

- The proposed MD-DTAMP algorithm with and without an off-grid scheme all show superior estimation performance compared with benchmark algorithms. Moreover, the proposed algorithm can achieve solid performance with smaller pilot overheads and comparable complexity.

5.2 System model

In this paper, we consider the narrow-band flat fading transmission model through space-air downlink communication from LEO satellite to multiple UAVs. Since Ka-band is weather dependent, to mitigate the high path loss, the LEO satellite is installed with UPA with $M = N_x \times N_y$ antenna elements to compensate the transmission loss [24], [116]. The $k^{th} \in \{1, ..., K\}$ UAV is equipped with a high gain omni-directional single antenna[1][126], [127]. The illustration of the system model can be found in Fig. 5.1. The important notations of the system and channel model are shown in Table 5-A. In training procedure, as symbols $\boldsymbol{S}_k \in \mathbb{C}^{M \times N_S}$ are transmitted from satellite to UAV, the

[1] For simplicity of only exploring the temporal and spatial correlation of the azimuth and elevation AoD of the satellite, the high gain omni-directional single antenna is assumed to be carried by the k^{th} UAV. However, the proposed system model can be further extended with UAVs also installed with UPA. The main difference is that the channel model should further take the receiver array response vectors with azimuth and elevation directions of AoAs into consideration.

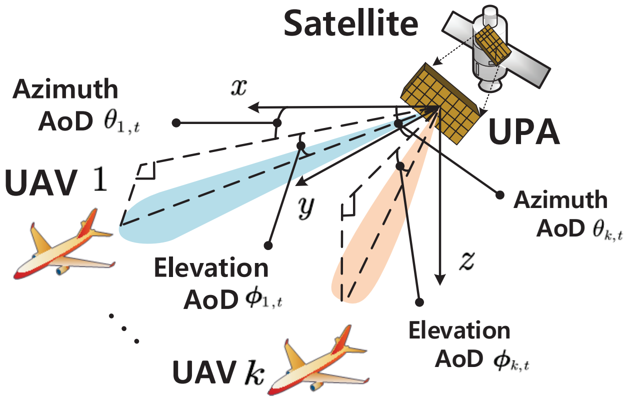

Satellite
UPA
Azimuth AoD $\theta_{1,t}$
x
UAV 1
y
Elevation AoD $\phi_{1,t}$
Azimuth AoD $\theta_{k,t}$
z
Elevation AoD $\phi_{k,t}$
UAV k

Figure 5.1: The illustration of space-air communications system model at time slot t.

received signal at k^{th} UAV at time slot t can be expressed as

$$\boldsymbol{y}_{k,t}^{\mathrm{RX}} = \boldsymbol{H}_{k,t}\boldsymbol{S}_k + \boldsymbol{n}^{\mathrm{RX}}, \tag{5.1}$$

with $\boldsymbol{H}_{k,t} \in \mathbb{C}^{1\times M}$ as the frequency domain channel matrix, and $\boldsymbol{n}^{\mathrm{RX}} \sim \mathcal{CN}(0, \sigma_n^2 \boldsymbol{I})$ is the additive white Gaussian noise.

5.2.1 Channel Model

The LoS ray is the dominant path in space-air communication links [105]. Hence, the $\boldsymbol{H}_{k,t}$ denotes the frequency domain channel vector with one dominant LoS path, which is written as,

$$\boldsymbol{H}_{k,t} = \tilde{g}_{k,t}\boldsymbol{A}^H\left(\theta_{k,t}, \phi_{k,t}\right), \tag{5.2}$$

where $\tilde{g}_{k,t}$ is the complex channel gain, $\boldsymbol{A}\left(\theta_{k,t}, \phi_{k,t}\right) \in \mathbb{C}^{(N_x\times N_y)\times 1}$ represents the transmit array response vectors, $\theta_{k,t}$ and $\phi_{k,t}$ as azimuth and elevation directions of AoD for k^{th} UAV at time slot t. As for UPA installed LEO satellite, the steering vector can be written as

$$\begin{aligned}\boldsymbol{A}\left(\theta_{k,t}, \phi_{k,t}\right) = \frac{1}{\sqrt{N_x \times N_y}}[1, ..., e^{j\frac{2\pi}{\lambda}d\sin\theta_{k,t}(n_x\sin\phi_{k,t}+n_y\cos\phi_{k,t})} \\ , ..., e^{j\frac{2\pi}{\lambda}d\sin\theta_{k,t}(N_x\sin\phi_{k,t}+N_y\cos\phi_{k,t})}]^T,\end{aligned} \tag{5.3}$$

Table 5-A: Table of Important Notations

Notations	Explanations
n_x, N_x	Index and total number of x-axis antenna/sparsity size
n_y, N_y	Index and total number of y-axis antenna/sparsity size
N_S	Total pilot number
m, M	Index and total number of the antenna and the sparse grid
t, T	Index and total number of the time slot
$\theta_{k,t},\ \phi_{k,t}$	k^{th} UAV azimuth and elevation directions of AoD at t
$\tilde{\boldsymbol{\theta}}_{k,t},\ \tilde{\boldsymbol{\phi}}_{k,t}$	k^{th} UAV on-grid azimuth and elevation directions of AoD at t
$\mathit{\Delta}\boldsymbol{\theta}_{k,t},\ \mathit{\Delta}\boldsymbol{\phi}_{k,t}$	k^{th} UAV azimuth and elevation offset directions of AoD at t
θ_{range}	The range of azimuth directions of AoD
ϕ_{range}	The range of elevation directions of AoD
$\boldsymbol{A}\left(\theta_{k,t}, \phi_{k,t}\right)$	Transmit array response vector at t
$\boldsymbol{H}_{k,t}$	k^{th} UAV frequency domain channel model at t
$\boldsymbol{S}_k$	k^{th} UAV transmit pilot symbols
$\boldsymbol{y}^{\mathrm{RX}}_{k,t}, \boldsymbol{y}_t$	k^{th} UAV received signal at t
$\boldsymbol{y}^{(t)}_k$	k^{th} UAV total received signal for total period t
$\boldsymbol{n}^{\mathrm{RX}}_{k,t}, \boldsymbol{n}_{k,t}$	k^{th} UAV additive white Gaussian noise at t
$\tilde{g}_{k,t}$	k^{th} UAV complex channel gain at t
$g_{k,t,m}$	k^{th} UAV channel gain element for the m^{th} AoD at t
$\boldsymbol{g}_{k,t}$	k^{th} UAV sparse matrix complex channel gain at t
$\boldsymbol{g}^{(t)}_k, \boldsymbol{g}^{(T)}_k$	k^{th} UAV total sparse matrix complex channel gain for period t or T
$b_{k,t,m}$	k^{th} UAV hidden support element for the m^{th} AoD at t
$\boldsymbol{b}_{k,t}$	k^{th} UAV hidden support vector at t
$\boldsymbol{b}^{(t)}_k, \boldsymbol{b}^{(T)}_k$	k^{th} UAV hidden support vector for total period t or T
$\tilde{\boldsymbol{b}}_t$	MU Hidden support vector at t
$a_{k,t,m}$	k^{th} UAV azimuth support element for the m^{th} AoD at t
$e_{k,t,m}$	k^{th} UAV elevation support element for the m^{th} AoD at t
$\boldsymbol{a}_{k,t}, \boldsymbol{e}_{k,t}$	k^{th} UAV azimuth and elevation support vector at t
$\boldsymbol{a}^{(t)}_k, \boldsymbol{a}^{(T)}_k$	k^{th} UAV azimuth support vector for total period t or T
$\boldsymbol{e}^{(t)}_k, \boldsymbol{e}^{(T)}_k$	k^{th} UAV elevation support vector for total period t or T
$\tilde{a}_{t,m},\ \tilde{e}_{t,m}$	MU azimuth and elevation support element for the m^{th} AoD at t
$\tilde{\boldsymbol{a}}_t,\tilde{\boldsymbol{e}}_t$	MU azimuth and elevation support vector at t
$\tilde{\boldsymbol{a}}^{(t)}, \tilde{\boldsymbol{a}}^{(T)}$	MU azimuth support vector for total period t or T
$\tilde{\boldsymbol{e}}^{(t)}, \tilde{\boldsymbol{e}}^{(T)}$	MU elevation support vector for total period t or T
$\vartheta_{k,t,m}$	k^{th} UAV hidden value element for the m^{th} AoD at t
$\boldsymbol{\vartheta}_{k,t}$	k^{th} UAV hidden value vector at t
$\boldsymbol{\vartheta}^{(t)}_k, \boldsymbol{\vartheta}^{(T)}_k$	k^{th} UAV hidden value vector for total period t or T

where λ is the signal wavelength, $0 \leqslant n_x < N_x$ and $0 \leqslant n_y < N_y$ are the indices of an antenna element with N_x and N_y as antenna array sizes.

5.2.2 On-grid and Off-grid Virtual Channel Model

Conventionally, to explore the sparse property of the channel, AoDs are assumed to lie on the quantized grids for simplicity [33]. With discretized AoDs, the channel model is called on-grid channel. However, in practice, the true angle is continuously distributed. The channel with this kind of AoDs is named as off-grid channel [125]. Hence in this subsection, we present both on-gird and off-grid cases.

For on-grid case, the channel model can be further written into a sparse representation as

$$\boldsymbol{H}_{k,t} = \boldsymbol{g}_{k,t} \tilde{\boldsymbol{A}}^{H} \left(\tilde{\boldsymbol{\theta}}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t} \right), \tag{5.4}$$

where $\boldsymbol{g}_{k,t} \in \mathbb{C}^{1 \times M}$ is a sparse vector with one non-zero elements (which means only one dominant path for the k^{th} UAV), M is the total number of spatial grids, the quantized AoD pairs are considered, i.e., $\tilde{\boldsymbol{\theta}}_{k,t} = [0, \theta^{\text{range}}/N_x, ..., \theta^{\text{range}}(N_x - 1)/N_x]$ and $\tilde{\boldsymbol{\phi}}_{k,t} = [0, \phi^{\text{range}}/N_y, ..., \phi^{\text{range}}(N_y - 1)/N_y]$, which means the virtual k^{th} UAV's AoD pair can consist of a M uniform grid, and $\tilde{\boldsymbol{A}} \left(\tilde{\boldsymbol{\theta}}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t} \right) \in \mathbb{C}^{M \times M}$ is the dictionary matrix. For off-grid case, we define $\varDelta\theta_{k,t,n_x}$ and $\varDelta\phi_{k,t,n_y}$ to represent the offset angles for both azimuth and elevation directions, respectively. Hence, the true AoD azimuth and elevation angle should be the combination of the nearest on-grid angle and the relative offset angle. As an example, we demonstrate the k^{th} UAV's AoD azimuth direction. The illustration can be found in Fig. 5.2. We assume that the true azimuth angle $\theta_{k,t}$ has the nearest on-grid angle $\tilde{\theta}_{k,t,n_x}$ and the offset angle is $\varDelta\theta_{k,t,n_x} = \theta_{k,t} - \tilde{\theta}_{k,t,n_x}$. Similarly, the AoD elevation direction offset angle can be written as $\varDelta\phi_{k,t,n_x} = \phi_{k,t} - \tilde{\phi}_{k,t,n_y}$. Hence, the off-grid channel can be written as

$$\boldsymbol{H}_{k,t} = \boldsymbol{g}_{k,t} \tilde{\boldsymbol{A}}^{H} \left(\tilde{\boldsymbol{\theta}}_{k,t} + \boldsymbol{\varDelta\theta}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t} + \boldsymbol{\varDelta\phi}_{k,t} \right), \tag{5.5}$$

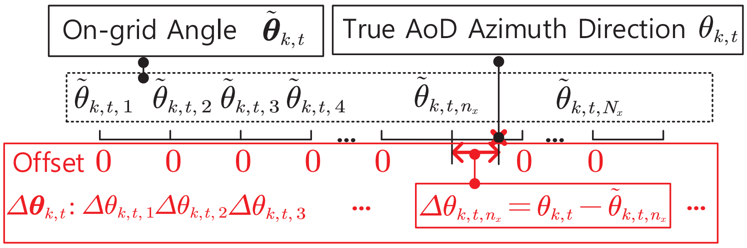

On-grid Angle $\tilde{\boldsymbol{\theta}}_{k,t}$
True AoD Azimuth Direction $\theta_{k,t}$
$\tilde{\theta}_{k,t,1}$ $\tilde{\theta}_{k,t,2}$ $\tilde{\theta}_{k,t,3}$ $\tilde{\theta}_{k,t,4}$ $\tilde{\theta}_{k,t,n_x}$ $\tilde{\theta}_{k,t,N_x}$
...
Offset 0 0 0 0 0 0 0
$\Delta\boldsymbol{\theta}_{k,t}$: $\Delta\theta_{k,t,1}$ $\Delta\theta_{k,t,2}$ $\Delta\theta_{k,t,3}$ ...
$\Delta\theta_{k,t,n_x} = \theta_{k,t} - \tilde{\theta}_{k,t,n_x}$ ...

Figure 5.2: Illustration of the off-grid AoD azimuth direction.

where

$$\tilde{\boldsymbol{A}}\left(\tilde{\boldsymbol{\theta}}_{k,t}+\Delta\boldsymbol{\theta}_{k,t},\tilde{\boldsymbol{\phi}}_{k,t}+\Delta\boldsymbol{\phi}_{k,t}\right)=\Big[\boldsymbol{A}\left(\tilde{\theta}_{k,t,1}+\Delta\theta_{k,t,1},\tilde{\phi}_{k,t,1}+\Delta\phi_{k,t,1}\right),...,$$
$$\boldsymbol{A}\left(\tilde{\theta}_{k,t,n_x}+\Delta\theta_{k,t,n_x},\tilde{\phi}_{k,t,n_y}+\Delta\phi_{k,t,n_y}\right),...,\boldsymbol{A}\left(\tilde{\theta}_{k,t,N_x}+\Delta\theta_{k,t,N_x},\tilde{\phi}_{k,t,N_y}+\Delta\phi_{k,t,N_y}\right)\Big] \tag{5.6}$$

is the off-grid array response vector dictionary matrix. To be noticed, the AoD elevation offset vector $\Delta\boldsymbol{\theta}_{k,t} = [\Delta\theta_{k,t,1}..., \Delta\theta_{k,t,N_x}]^T$ with

$$\Delta\theta_{k,t,n_x} = \begin{cases} \theta_{k,t} - \tilde{\theta}_{k,t,n_x}, & \text{Active path,} \\ 0, & \text{Otherwise.} \end{cases} \tag{5.7}$$

The otherwise case represents the non-active virtual on-grid offsets which can be observed in Fig. 5.2 with the notation from $\Delta\theta_{k,t,1}$ to $\Delta\theta_{k,t,N}$ without the active path offset $\Delta\theta_{k,t,n_x}$. Similarly, the AoD azimuth offset vector $\Delta\boldsymbol{\phi}_{k,t} = \left[\Delta\phi_{k,t,1}..., \Delta\phi_{k,t,N_y}\right]^T$ with

$$\Delta\phi_{k,t,n_y} = \begin{cases} \phi_{k,t} - \tilde{\phi}_{k,t,n_y}, & \text{Active path,} \\ 0, & \text{Otherwise.} \end{cases} \tag{5.8}$$

When $\Delta\boldsymbol{\theta}_{k,t} = \mathbf{0}$ and $\Delta\boldsymbol{\phi}_{k,t} = \mathbf{0}$, the off-grid channel can be treated as the on-grid channel in (5.4).

5.3 Multi-Dimensional Markov Multiuser Channel Model

In our model, communication links connected between LEO satellite and multiple UAVs are considered. Because of the LEO satellite height, a certain portion of the earth's surface is covered at any given time slot. This means a constellation of LEO satellites is necessary for full coverage of the earth [128]. We consider a condition that one LEO satellite communicates with the same group of multiple UAVs simultaneously before any communication link is handed over to the adjacent LEO satellite. This means the LEO satellite would orbit in a certain distance range, limiting the azimuth and elevation angle in fixed ranges. With assumptions above, we transfer the complex continuous relative displacement relationships between LEO satellite and UAVs into time-varying angular domain variance with transition probability similar to [110], [112], [123] (this is further elaborated in Section 5.3.3).

In the MU scenario with multiple UAVs, we specially explore the relationship between the MU and SU hidden support vector. The MU azimuth $\tilde{\boldsymbol{a}}^{(T)}$ and elevation $\tilde{\boldsymbol{e}}^{(T)}$ support vector are both modeled as two dimensional Markov model (2D-MM) similarly in [123]. $\tilde{\boldsymbol{a}}^{(T)}$ and $\tilde{\boldsymbol{e}}^{(T)}$ represent the MU azimuth and elevation support vector over total time period T. During time slot t, $\tilde{\boldsymbol{a}}_{\boldsymbol{t}} = [\tilde{a}_{t,1}, ..., \tilde{a}_{t,N_x}] \in \{0,1\}^{N_x}$ with $\tilde{a}_{t,n_x} = a_{1,t,m} \vee a_{2,t,m} \vee ... \vee a_{K,t,m}$, which means MU azimuth support is the union of the SU azimuth support of total K UAVs at time slot t. Similarly, for MU elevation support $\tilde{\boldsymbol{e}}_{\boldsymbol{t}} = [\tilde{e}_{t,1}, ..., \tilde{e}_{t,N_x}] \in \{0,1\}^{N_x}$ with $\tilde{e}_{t,n_x} = e_{1,t,m} \vee e_{2,t,m} \vee ... \vee e_{K,t,m}$. Furthermore, the SU joint hidden support vector $\boldsymbol{b}_{k,t}$ is modeled as three dimensional hidden Markov model (3D-HMM) which jointly combines the SU azimuth and elevation support.

The details of the complete MD-MM channel model are introduced in four parts: probability model for SU channel, 2D-MM for azimuth and elevation MU hidden support vector, 3D-HMM for azimuth and elevation SU joint hidden support vector, and Gaussian Markov model for SU hidden value vector.

5.3.1 Probability Model for SU Channel

The dynamic angular domain channel for the k^{th} UAV $\boldsymbol{g}_k^{(T)} = \{\boldsymbol{g}_{k,1}, ..., \boldsymbol{g}_{k,T}\}$ with $\boldsymbol{g}_{k,t} = \{g_{k,t,1}, ..., g_{k,t,M}\}$ can be modeled as a probabilistic signal model with two hidden random processes, SU joint hidden support vector $\boldsymbol{b}_k^{(T)} = \{\boldsymbol{b}_{k,1}, ..., \boldsymbol{b}_{k,T}\}$ and SU hidden value vector $\boldsymbol{\vartheta}_k^{(T)} = \{\boldsymbol{\vartheta}_{k,1}, ..., \boldsymbol{\vartheta}_{k,T}\}$. Here, $\boldsymbol{b}_{k,t} = \left[b_{k,t,1,1}, ..., b_{k,t,N_x,N_y}\right] = [b_{k,t,1}, ..., b_{k,t,M}] \in \{0,1\}^M$ represents the SU joint hidden support vector at time t, which indicates the k^{th} UAV channel sparsity with $b_{k,t,m} \in \{1, 0\}$. The SU hidden value vector $\boldsymbol{\vartheta}_{k,t} = [\vartheta_{k,t,1}, ..., \vartheta_{k,t,M}] \in \mathbb{C}^M$ represents the temporal correlation of the k^{th} UAV channel coefficients. The dynamic channel element for the m^{th} AoD direction in the time t of the k^{th} UAV can be written as

$$g_{k,t,m} = b_{k,t,m}\vartheta_{k,t,m}, \tag{5.9}$$

where $b_{k,t,m}$ denotes whether there is an active path or not (i.e., $b_{k,t,m} = 1$ means the m^{th} AoD path is activated) and $\vartheta_{k,t,m}$ represents the path gain[2].

To formulate the more realistic dynamic channel (i.e., the inactive AoD path at the current time t may be activated at time $t+1$), a probabilistic channel model of k^{th} UAV with channel prior distribution can be formed as

$$p\left(\boldsymbol{g}_k^{(T)}, \boldsymbol{b}_k^{(T)}, \boldsymbol{\vartheta}_k^{(T)}\right) = p\left(\boldsymbol{g}_k^{(T)} \mid \boldsymbol{b}_k^{(T)}, \boldsymbol{\vartheta}_k^{(T)}\right) p\left(\boldsymbol{b}_k^{(T)}\right) p\left(\boldsymbol{\vartheta}_k^{(T)}\right), \tag{5.10}$$

where the k^{th} UAV channel vector conditional prior is

$$p\left(\boldsymbol{g}_k^{(T)} \mid \boldsymbol{b}_k^{(T)}, \boldsymbol{\vartheta}_k^{(T)}\right) = \prod_{t=1}^{T}\prod_{m=1}^{M} p\left(g_{k,t,m} \mid b_{k,t,m}, \vartheta_{k,t,m}\right) = \prod_{t=1}^{T}\prod_{m=1}^{M} \delta\left(g_{k,t,m} - b_{k,t,m}\vartheta_{k,t,m}\right), \tag{5.11}$$

with $\delta(\cdot)$ is the Dirac delta function. The factor graph of the combined k^{th} UAV channel can be found in Fig. 5.3, with $\pi_{k,t,m}$ denotes the conditional prior $p\left(g_{k,t,m} \mid b_{k,t,m}, \vartheta_{k,t,m}\right)$. This factor graph shows that the final k^{th} UAV channel gain $\mathbf{g}_{k,t}$ is the combination of

[2] For paring the azimuth and elevation directions, $m = (n_x - 1) \times N_x + n_y$.

the SU joint hidden support vector $\mathbf{b}_{k,t}$ (on the left hand side of Fig. 5.3) and the SU hidden value vector $\vartheta_{k,t}$ (on the right hand side of Fig. 5.3).

5.3.2 Gaussian Markov Model for SU Hidden Value Vector

The illustration of the factor graph for the k^{th} UAV SU hidden value vector $\boldsymbol{\vartheta}_{k,t}$ and the Gaussian-Markov model can be found in Fig. 5.3 (on the right hand side). Due to the path gains change smoothly over time [108], the hidden value vector can be formulated as the Gauss-Markov processes [53]

$$\vartheta_{k,t,m} = (1-\beta_k)\left(\vartheta_{k,t-1,m} - \mu_k\right) + \beta_k \omega_{k,t,m} + \mu_k, \tag{5.12}$$

where $\beta_k \in [0,1]$, $\omega_{k,t,m} \sim \mathcal{CN}(0,\zeta_k)$, $\mu_k \in \mathbb{C}$ is the mean of the process. When $\beta_k = 0$, then $\vartheta_{k,t,m} = \vartheta_{k,t-1,m}$, which indicates that $\boldsymbol{\vartheta}_{k,t}$ remain unchanged over time. When $\beta_k = 1$, then $\vartheta_{k,t,m} = \beta_k \omega_{k,t,m} + \mu_k$, which means that the SU hidden value vector shows the i.i.d Gaussian distribution with mean μ_k over time. If $0 < \beta_k < 1$, the conditional probability can be written as

$$p\left(\vartheta_{k,t,m} \mid \vartheta_{k,t-1,m}\right) \sim \mathcal{CN}\left(\vartheta_{k,t,m}; (1-\beta_k)\,\vartheta_{k,t-1,m} + \beta_k \mu_k, \beta_k^2 \zeta_k\right). \tag{5.13}$$

The distribution of the steady state of the process is $\vartheta_{k,t,m} \sim \mathcal{CN}(\mu_k, \sigma_k^2 = \frac{\beta_k \zeta_k}{2-\beta_k})$. And the joint distribution of $\boldsymbol{\vartheta}_{\boldsymbol{k}}^{(T)}$ can be formulated as

$$p\left(\boldsymbol{\vartheta}_k^{(T)}\right) = \prod_{m=1}^{M} p\left(\vartheta_{k,1,m}\right) \prod_{t=2}^{T} p\left(\vartheta_{k,t,m} \mid \vartheta_{k,t-1,m}\right), \tag{5.14}$$

where $p\left(\vartheta_{k,1,m}\right)$ and $p\left(\vartheta_{k,t,m} \mid \vartheta_{k,t-1,m}\right)$ are the transition probabilities that equal to the factor nodes $f_{k,1,m}$ and $f_{k,t,m}$ in Fig. 5.3.

5.3.3 2D-MM for Azimuth and Elevation MU Hidden Support Vector

The details of the azimuth and elevation MU hidden support vector are introduced as follows: To explain the 2D-MM for MU hidden support vector of azimuth and elevation

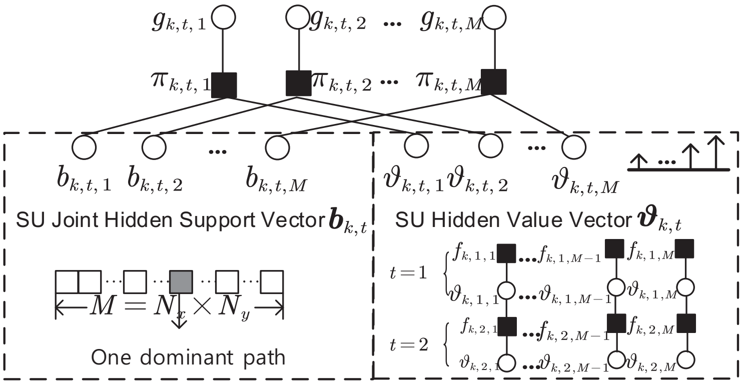


Figure 5.3: Factor graph of the SU channel.

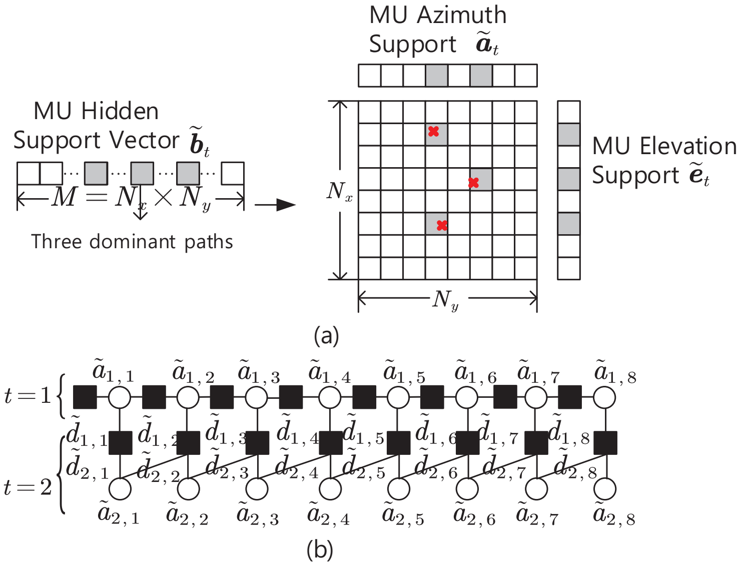


Figure 5.4: (a)MU hidden support matrix illustration at time slot t with $N_x = 8$ $N_y = 8$ when three UAV users are considered with three activated paths. (b) Factor graph of 2D-MM of the MU azimuth support vectors when $T = 2$ and $N_x = 8$.

directions, we take three UAV users as an example in Fig. 5.4(a). The relationship of the MU azimuth support $\tilde{\boldsymbol{a}}_t$ and the MU elevation support $\tilde{\boldsymbol{e}}_t$ with three activated paths are shown. To be noticed, in this example, there are two users share the same MU azimuth support. Hence, the MU azimuth support $\tilde{\boldsymbol{a}}_t$ has only two activated points at time t. For MU elevation support $\tilde{\boldsymbol{e}}_t$ has three activated points at time t.

Because of the navigation of the UAVs [115], together with the known orbital voyage of the LEO satellite, the relative displacement of the UAV and satellite is continous.

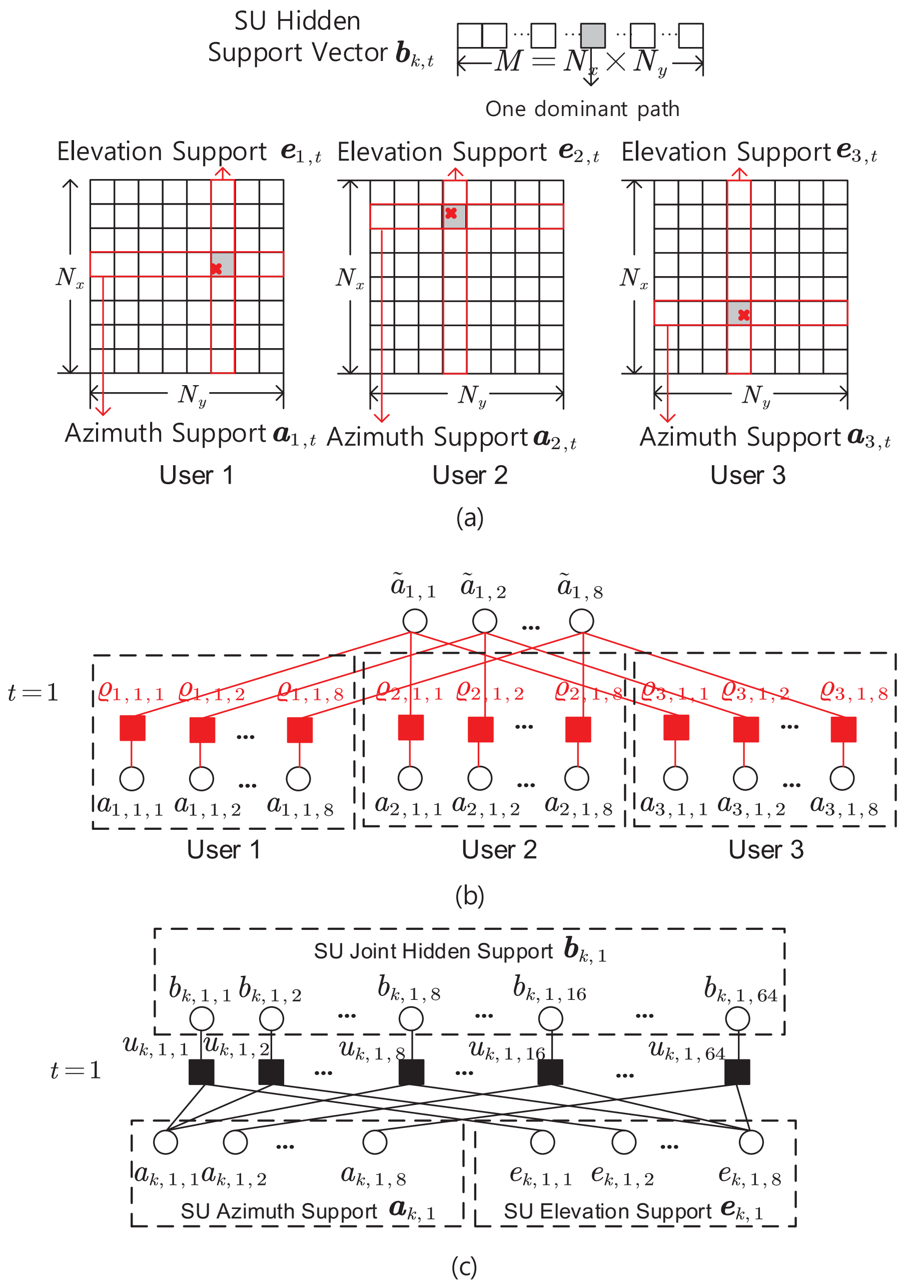

SU Hidden Support Vector $\boldsymbol{b}_{k,t}$
$M = N_x \times N_y$
One dominant path
Elevation Support $\boldsymbol{e}_{1,t}$
Elevation Support $\boldsymbol{e}_{2,t}$
Elevation Support $\boldsymbol{e}_{3,t}$
N_x
N_y
Azimuth Support $\boldsymbol{a}_{1,t}$
Azimuth Support $\boldsymbol{a}_{2,t}$
Azimuth Support $\boldsymbol{a}_{3,t}$
User 1
User 2
User 3
(a)
$\tilde{a}_{1,1}$ $\tilde{a}_{1,2}$ $\tilde{a}_{1,8}$
$t=1$
$\varrho_{1,1,1}$ $\varrho_{1,1,2}$ $\varrho_{1,1,8}$
$\varrho_{2,1,1}$ $\varrho_{2,1,2}$ $\varrho_{2,1,8}$
$\varrho_{3,1,1}$ $\varrho_{3,1,2}$ $\varrho_{3,1,8}$
$a_{1,1,1}$ $a_{1,1,2}$ $a_{1,1,8}$
$a_{2,1,1}$ $a_{2,1,2}$ $a_{2,1,8}$
$a_{3,1,1}$ $a_{3,1,2}$ $a_{3,1,8}$
User 1
User 2
User 3
(b)
SU Joint Hidden Support $\boldsymbol{b}_{k,1}$
$b_{k,1,1}$ $b_{k,1,2}$ $b_{k,1,8}$ $b_{k,1,16}$ $b_{k,1,64}$
$t=1$
$u_{k,1,1}$ $u_{k,1,2}$ $u_{k,1,8}$ $u_{k,1,16}$ $u_{k,1,64}$
$a_{k,1,1}$ $a_{k,1,2}$ $a_{k,1,8}$
$e_{k,1,1}$ $e_{k,1,2}$ $e_{k,1,8}$
SU Azimuth Support $\boldsymbol{a}_{k,1}$
SU Elevation Support $\boldsymbol{e}_{k,1}$
(c)

Figure 5.5: (a) Illustration of SU joint hidden support matrix based on MU hidden support matrix in Fig.5.4. (b) HMM of SU azimuth support a_{k,t,n_x} with three users. (c) 3D-HMM Factor graph of the SU joint hidden support vector $\boldsymbol{b}_{k,t}$ when $T = 1$, $N_x = 8$, $N_y = 8$.

Hence, the spatial sparsity pattern will act highly related with the prior adjacent pattern, i.e., $\tilde{a}_{t+1,n_x}$ has higher probability be influenced by the adjacent $\tilde{a}_{t,n_x-1}$ in previous time slot [36]. [108] has also verified that the channel support change slowly over time, which indicates $\tilde{a}_{t+1,n_x}$ has higher probability to be influenced by itself in previous time slot $\tilde{a}_{t,n_x}$. This relationship which considers both spatial correlation and temporal correlation is named as dynamic 2D sparsity. The illustration of this 2D-MM can be found in the factor graph in Fig. 5.4(b). The model over total time period T can be expressed as:

$$\begin{aligned} p\left(\tilde{\boldsymbol{a}}^{(T)}\right) &= \underbrace{p\left(\tilde{a}_{1,1}\right)}_{q_{\tilde{a}_{1,1}}} \prod_{n_x=2}^{N_x} \underbrace{p\left(\tilde{a}_{1,n_x} \mid \tilde{a}_{1,n_x-1}\right)}_{q^{(AS)}_{\tilde{a}_{1,n_x},\tilde{a}_{1,n_x-1}}} \\ &\times \prod_{t=2}^{T} \left(\underbrace{p\left(\tilde{a}_{t,1} \mid \tilde{a}_{t-1,1}\right)}_{q^{(AT)}_{\tilde{a}_{t,1},\tilde{a}_{t-1,1}}} \prod_{n_x=1}^{N_x} \underbrace{p(\tilde{a}_{t,n_x} \mid \tilde{a}_{t-1,n_x-1}, \tilde{a}_{t-1,n_x})}_{q^{(A)}_{\tilde{a}_{t,n_x},\tilde{a}_{t-1,n_x-1},\tilde{a}_{t-1,n_x}}} \right), \end{aligned} \tag{5.15}$$

where $q_{\tilde{a}_{1,1}}$, $q^{(AS)}_{\tilde{a}_{1,n_x},\tilde{a}_{1,n_x-1}}$, $q^{(AT)}_{\tilde{a}_{t,1},\tilde{a}_{t-1,1}}$ and $q^{(A)}_{\tilde{a}_{t,n_x},\tilde{a}_{t-1,n_x-1},\tilde{a}_{t-1,n_x}}$ are the transition probabilities that equal to the factor nodes $\tilde{d}_{1,1}$, $\tilde{d}_{1,n_x}$, $\tilde{d}_{t,1}$ and $\tilde{d}_{t,n_x}$ in Fig. 5.4(b) of the azimuth directions. To be noticed, the MU azimuth support $\tilde{a}_{t,n_x} \in \{0,1\}$. The support of the MU elevation support of AoD $p\left(\tilde{\boldsymbol{e}}^{(T)}\right)$ can be modeled similarly.

5.3.4 3D-HMM for Azimuth and Elevation SU Joint Hidden Support Vector

To explore the relationship between MU and SU, we modeled the joint prior distribution of MU and SU azimuth or elevation support as HMM with conditional probability. Take k^{th} UAV SU azimuth support as an example:

$$p\left(\tilde{\boldsymbol{a}}^{(T)}, \boldsymbol{a}_k^{(T)}\right) = \prod_{t=1}^{T}\prod_{k=1}^{K} p\left(\tilde{\boldsymbol{a}}_t\right) p\left(\boldsymbol{a}_{k,t} | \tilde{\boldsymbol{a}}_t\right) = \prod_{t=1}^{T}\prod_{k=1}^{K}\prod_{n_x=1}^{N_x} p\left(\tilde{\boldsymbol{a}}_t\right) \underbrace{p\left(a_{k,t,n_x} | \tilde{a}_{t,n_x}\right)}_{\varsigma^{(A)}_{k,\tilde{a}_{t,n_x},a_{k,t,n_x}}}. \tag{5.16}$$

where $\varsigma^{(A)}_{k,\tilde{a}_{t,n_x},a_{k,t,n_x}}$ is the transition probability equal to the factor node ϱ_{k,t,n_x} in Fig.5.5(b). Furthermore, $\tilde{\boldsymbol{a}}^{(T)}$ and $\boldsymbol{a}_k^{(T)}$ are the MU azimuth support vector and k^{th} UAV SU azimuth support vector over time period T, respectively. Although in practice, multiple UAVs may share partially MU azimuth or elevation support, i.e., in Fig.5.5, user 2 and user

3 both have the 4^{th} grid activated which means the same azimuth support is shared, our HMM model can still apply. To be noticed, each element in $\boldsymbol{a}_k^{(T)}$ is generated by the MU azimuth support vector in the HMM with certain conditional probability, i.e., $p(a_{k,t,n_x} = 1|\tilde{a}_{t,n_x} = 0) = \varsigma_{k,t,0,1}^{(A)} = 0$ and $p(a_{k,t,n_x} = 1|\tilde{a}_{t,n_x} = 0) = \varsigma_{k,t,1,1}^{(A)}$. In another word, when the MU azimuth support element $\tilde{a}_{t,n_x} = 0$, there is no way that the k^{th} UAV SU azimuth support element $a_{k,t,n_x} = 1$. However, when the MU azimuth support element $\tilde{a}_{t,n_x} = 1$, there is a certain probability $\varsigma_{k,t,1,1}^{(A)}$ that the n_x^{th} element a_{k,t,n_x} in the k^{th} UAV SU azimuth support vector is 1. The k^{th} UAV SU elevation support can be modeled similarly.

Hence, the 3D-HMM of the SU joint hidden support $\boldsymbol{b}_k^{(T)}$ over total period time T which combines both azimuth and elevation direction supports can be found in Fig. 5.5(c). The joint conditional prior of the channel support probability of the k^{th} UAV is given by

$$p\left(\boldsymbol{b}_k^{(T)}|\boldsymbol{a}_k^{(T)}, \boldsymbol{e}_k^{(T)}\right) = \prod_{t=1}^{T} \prod_{n_x=1}^{N_x} \prod_{n_y=1}^{N_y} p\left(b_{k,t,n_x,n_y}|a_{k,t,n_x}, e_{k,t,n_y}\right). \tag{5.17}$$

5.4 3D Off-grid Dynamic Sparse Channel Tracking Problem Formulation

By taking the off-grid channel (5.5) into consideration, the received signal at k^{th} UAV can be written as

$$\boldsymbol{y}_{k,t}^{\mathrm{RX}} = \boldsymbol{g}_{k,t}\tilde{\boldsymbol{A}}^H\left(\tilde{\boldsymbol{\theta}}_{k,t} + \Delta\boldsymbol{\theta}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t} + \Delta\boldsymbol{\phi}_{k,t}\right)\boldsymbol{S}_k + \boldsymbol{n}_{k,t}^{\mathrm{RX}}. \tag{5.18}$$

with $\boldsymbol{y}_{k,t}^{\mathrm{RX}} \in \mathbb{C}^{1\times N_S}$. To convert the received signal into the standard CS model, we reshape $\boldsymbol{g}_{k,t}$ from $1 \times M$ to $M \times 1$. Hence the received signal can be rewritten as

$$\boldsymbol{y}_{k,t} = \boldsymbol{\Phi}_{k,t}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right)\boldsymbol{g}_{k,t} + \boldsymbol{n}_{k,t}, \tag{5.19}$$

with the sensing matrix $\boldsymbol{\varPhi}_{k,t}\left(\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)=\boldsymbol{S}_k^T\left(\tilde{\boldsymbol{A}}^H\left(\tilde{\boldsymbol{\theta}}_{k,t}+\varDelta\boldsymbol{\theta}_{k,t},\tilde{\boldsymbol{\phi}}_{k,t}+\varDelta\boldsymbol{\phi}_{k,t}\right)\right)^T\in\mathbb{C}^{N_S\times M}$, $\boldsymbol{y}_{k,t}\in\mathbb{C}^{N_S\times 1}$. For the training phase, the transmitted pilot $\boldsymbol{S}_k$ is designed based on the partial discrete Fourier transform (DFT) random permutation measurement matrix $\boldsymbol{\varPhi}_{k,t}\left(\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)$ [37], [118]. Details are shown in **Appendix A**. Given the received signals of total t time period $\boldsymbol{y}_k^{(t)}$, the purpose is to track the dynamic channel vector $\boldsymbol{g}_{k,t}$ at time t. The channel vector can be estimated as $\hat{g}_{k,t,m}=E(g_{k,t,m}|\boldsymbol{y}_k^{(t)},\varDelta\boldsymbol{\theta}_{k,t}$, $\varDelta\boldsymbol{\phi}_{k,t})$, when the expectation is over the marginal posterior

$$\begin{aligned}&p\left(g_{k,t,m}|\boldsymbol{y}_k^{(t)},\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)\propto\\&\sum_{\boldsymbol{b}_k^{(t)}}\int_{\boldsymbol{g}_{-(k,t,m)}^{(t)},\boldsymbol{\vartheta}_k^{(t)}}p\left(\boldsymbol{g}_k^{(t)},\tilde{\boldsymbol{a}}^{(t)},\tilde{\boldsymbol{e}}^{(t)},\boldsymbol{b}_k^{(t)},\boldsymbol{\vartheta}_k^{(t)},\boldsymbol{y}_k^{(t)}|\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)\end{aligned}\tag{5.20}$$

with $\boldsymbol{g}_{-(k,t,m)}^{(t)}$ denotes the vector $\boldsymbol{g}_k^{(t)}$ without the element $g_{k,t,m}$. Our goal is to obtain the MMSE of the estimated $\boldsymbol{g}_k$. At the same time, the optimal off-grid azimuth and elevation offset $\varDelta\boldsymbol{\theta}_{k,t}{}^{(opt)}$ and $\varDelta\boldsymbol{\phi}_{k,t}^{(opt)}$ are obtained by maximum likelihood as follows,

$$\begin{aligned}\varDelta\boldsymbol{\theta}_{k,t}^{(opt)},\varDelta\boldsymbol{\phi}_{k,t}^{(opt)}&=\underset{\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}}{\operatorname{argmax}}\ln p\left(\boldsymbol{y}_k^{(t)},\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)\\&=\underset{\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}}{\operatorname{argmax}}\int_{\boldsymbol{g}_k^{(t)}}p\left(\boldsymbol{g}_k^{(t)},\boldsymbol{y}_k^{(t)},\varDelta\boldsymbol{\theta}_{k,t},\varDelta\boldsymbol{\phi}_{k,t}\right)d\boldsymbol{g}_k^{(t)}.\end{aligned}\tag{5.21}$$

5.5 Multi-dimensional Dynamic Turbo Approximate Message Passing Algorithm with 3D Off-grid Estimation

AMP based algorithms have been widely studied for user detection [129], [130] and sparse channel estimation [53], [131], [132] because of its good reconstruction accuracy and low complexity. Traditionally, AMP is applied based on a static prior information. However, the prior distribution knowledge is time-varying in our structured dynamic channel model. As can be seen from the factor graph Fig. 5.6, the complex structure makes it more challenging to gain the exact marginal posterior in (5.20). We further employ a turbo iterative framework [121] to improve the reconstruction performance.

Table 5-B: Table of Factors, Distributions and Functional Forms Associated with Our MD-MM

Factor	Distribution	Function Form
$c_{k,t,m}\left(g_{k,t,m}, g^{pri}_{\mathcal{B},k,t,m}\right)$	$p(g^{pri}_{\mathcal{B},k,t,m} \mid g_{k,t,m})$	$\mathcal{CN}\left(\boldsymbol{g}_{k,t}; \boldsymbol{g}^{pri}_{\mathcal{B},k,t}, v^{pri}_{\mathcal{B},k,t}\boldsymbol{I}\right)$
$\pi_{k,t,m}\left(g_{k,t,m}, b_{k,t,m}, \vartheta_{k,t,m}\right)$	$p\left(g_{k,t,m} \mid b_{k,t,m}, \vartheta_{k,t,m}\right)$	$\delta\left(g_{k,t,m} - b_{k,t,m}\vartheta_{k,t,m}\right)$
$f_{k,1,m}\left(\vartheta_{k,1,m}\right)$	$p\left(\vartheta_{k,1,m}\right)$	$\mathcal{CN}\left(\vartheta_{k,1,m}; \mu, \sigma^2\right)$
$f_{k,t,m}\left(\vartheta_{k,t,m}, \vartheta_{k,t-1,m}\right)$	$p\left(\vartheta_{k,t,m} \mid \vartheta_{k,t-1,m}\right)$	$\mathcal{CN}\left(\vartheta_{k,t,m}; \left(1-\beta\right)\vartheta_{k,t-1,m} + \beta\mu, \beta^2\zeta\right)$
$\tilde{d}_{1,1}(\tilde{a}_{1,1})$	$p(\tilde{a}_{1,1})$	$(1-\lambda)^{1-\tilde{a}_{1,1}}\lambda^{\tilde{a}_{1,1}}$
$\tilde{d}_{1,m}(\tilde{a}_{1,m}, \tilde{a}_{1,m-1})$	$p(\tilde{a}_{1,m} \mid \tilde{a}_{1,m-1})$	$\begin{cases} \left(q^{(AS)}_{0,1}\right)^{\tilde{a}_{1,m}}\left(1-q^{(AS)}_{0,1}\right)^{(1-\tilde{a}_{1,m})}, \tilde{a}_{1,m-1}=0 \\ \left(1-q^{(AS)}_{1,0}\right)^{\tilde{a}_{1,m}}\left(q^{(AS)}_{1,0}\right)^{(1-\tilde{a}_{1,m})}, \tilde{a}_{1,m-1}=1 \end{cases}$
$\tilde{d}_{t,1}(\tilde{a}_{t,1}, \tilde{a}_{t-1,1})$	$p(\tilde{a}_{t,1} \mid \tilde{a}_{t-1,1})$	$\begin{cases} \left(q^{(AT)}_{0,1}\right)^{\tilde{a}_{t,1}}\left(1-q^{(AT)}_{0,1}\right)^{(1-\tilde{a}_{t,1})}, \tilde{a}_{t-1,1}=0 \\ \left(1-q^{(AT)}_{1,0}\right)^{\tilde{a}_{t,1}}\left(q^{(AT)}_{1,0}\right)^{(1-\tilde{a}_{t,1})}, \tilde{a}_{t-1,1}=1 \end{cases}$
$\tilde{d}_{t,m}(\tilde{a}_{t,m}, \tilde{a}_{t,m-1}, \tilde{a}_{t-1,m})$	$p(\tilde{a}_{t,m} \mid \tilde{a}_{t-1,m-1}, \tilde{a}_{t-1,m})$	$\begin{cases} \left(q^{(A)}_{\tilde{a}_{t-1,m-1},\tilde{a}_{t-1,m},1}\right)^{\tilde{a}_{t,m}}\left(1-q^{(A)}_{\tilde{a}_{t-1,m-1},\tilde{a}_{t-1,m},1}\right)^{(1-\tilde{a}_{t,m})} \\ \tilde{a}_{t-1,m-1}, \tilde{a}_{t-1,m} \in \{0,1\} \end{cases}$
$\tilde{r}_{1,1}(\tilde{e}_{1,1})$	$p(\tilde{e}_{1,1})$	$(1-\lambda)^{1-\tilde{e}_{1,1}}\lambda^{\tilde{e}_{1,1}}$
$\tilde{r}_{1,m}(\tilde{e}_{1,m}, \tilde{e}_{1,m-1})$	$p(\tilde{e}_{1,m} \mid \tilde{e}_{1,m-1})$	$\begin{cases} \left(q^{(ES)}_{0,1}\right)^{\tilde{e}_{1,m}}\left(1-q^{(ES)}_{0,1}\right)^{(1-\tilde{e}_{1,m})}, \tilde{e}_{1,m-1}=0 \\ \left(1-q^{(ES)}_{1,0}\right)^{\tilde{e}_{1,m}}\left(q^{(ES)}_{1,0}\right)^{(1-\tilde{e}_{1,m})}, \tilde{e}_{1,m-1}=1 \end{cases}$
$\tilde{r}_{t,1}(\tilde{e}_{t,1}, \tilde{e}_{t-1,1})$	$p(\tilde{e}_{t,1} \mid \tilde{e}_{t-1,1})$	$\begin{cases} \left(q^{(ET)}_{0,1}\right)^{\tilde{e}_{t,1}}\left(1-q^{(ET)}_{0,1}\right)^{(1-\tilde{e}_{t,1})}, \tilde{e}_{t-1,1}=0 \\ \left(1-q^{(ET)}_{1,0}\right)^{\tilde{e}_{t,1}}\left(q^{(ET)}_{1,0}\right)^{(1-\tilde{e}_{t,1})}, \tilde{e}_{t-1,1}=1 \end{cases}$
$\tilde{r}_{t,m}(\tilde{e}_{t,m}, \tilde{e}_{t-1,m-1}, \tilde{e}_{t-1,m})$	$p(\tilde{e}_{t,m} \mid \tilde{e}_{t-1,m-1}, \tilde{e}_{t-1,m})$	$\begin{cases} \left(q^{(E)}_{\tilde{e}_{t-1,m-1},\tilde{e}_{t-1,m},1}\right)^{\tilde{e}_{t,m}}\left(1-q^{(E)}_{\tilde{e}_{t-1,m-1},\tilde{e}_{t-1,m},1}\right)^{(1-\tilde{e}_{t,m})} \\ \tilde{e}_{t-1,m-1}, \tilde{e}_{t-1,m} \in \{0,1\} \end{cases}$
$\varrho_{k,t,m}(\tilde{a}_{t,n_x}, a_{k,t,n_x})$	$p(a_{k,t,n_x} \mid \tilde{a}_{t,n_x})$	$p(a_{k,t,n_x}=1 \mid \tilde{a}_{t,n_x}=0)=0$, $p(a_{k,t,n_x}=1 \mid \tilde{a}_{t,n_x}=1)=\varsigma^{(A)}_{k,t,1,1}$
$\varpi_{k,t,m}(\tilde{e}_{t,n_x}, e_{k,t,n_x})$	$p(e_{k,t,n_x} \mid \tilde{e}_{t,n_x})$	$p(e_{k,t,n_x}=1 \mid \tilde{e}_{t,n_x}=0)=0$, $p(e_{k,t,n_x}=1 \mid \tilde{e}_{t,n_x}=1)=\varsigma^{(E)}_{k,t,1,1}$
$u_{k,t,m}(b_{k,t,m}, a_{k,t,n_x}, e_{k,t,n_y})$	$p(b_{k,1,m} \mid a_{k,t,n_x}, e_{k,t,n_y})$	$a_{k,t,n_x}e_{k,t,n_y}(\gamma_{\boldsymbol{b}_{k,t}})^{b_{k,t,m}}(1-\gamma_{\boldsymbol{b}_{k,t}})^{(1-b_{k,t,m})} + (1-a_{k,t,n_x}e_{k,t,n_y})\delta(b_{k,t,m})$

The details of the proposed algorithm is elaborated in this section. The factor graph of the total MD-MM distribution $p(\boldsymbol{g}^{pri}_{\mathcal{B},t}, \tilde{\boldsymbol{a}}^{(t)}, \tilde{\boldsymbol{e}}^{(t)}, \boldsymbol{b}^{(t)}_{k}, \boldsymbol{\vartheta}^{(t)}_{k}, \boldsymbol{y}^{(t)}_{k})$ is shown in Fig. 5.6. This factor graph contains three sub-graphs: the MU azimuth and elevation vector $\tilde{\boldsymbol{a}}_t$ and $\tilde{\boldsymbol{e}}_t$ (on the left-hand side of Fig. 5.6), the SU hidden support vector $\boldsymbol{b}_{k,t}$ (in the middle of Fig. 5.6), and the k^{th} UAV SU hidden value vector $\boldsymbol{\vartheta}_{k,t}$ (on the right-hand side of Fig. 5.6) all through time T. The MU azimuth and elevation vector $\tilde{\boldsymbol{a}}_t$ and $\tilde{\boldsymbol{e}}_t$ are both structured as 2D-MM. The k^{th} UAV SU hidden support vector $\boldsymbol{b}_{k,t}$ is structured as 3D-HMM which contains both SU azimuth and elevation vector $\boldsymbol{a}_{k,t}$ and $\boldsymbol{e}_{k,t}$. The details of all the necessary nodes in the factor graph are already explained in Section 5.3. The expression of the factor nodes, distributions and functional forms are shown in Table 5-B.

Due to the complex structure of the total factor graph of the MD-MM, it is challenging to achieve the exact marginal posterior in (5.20). In this section, to explore the dynamic

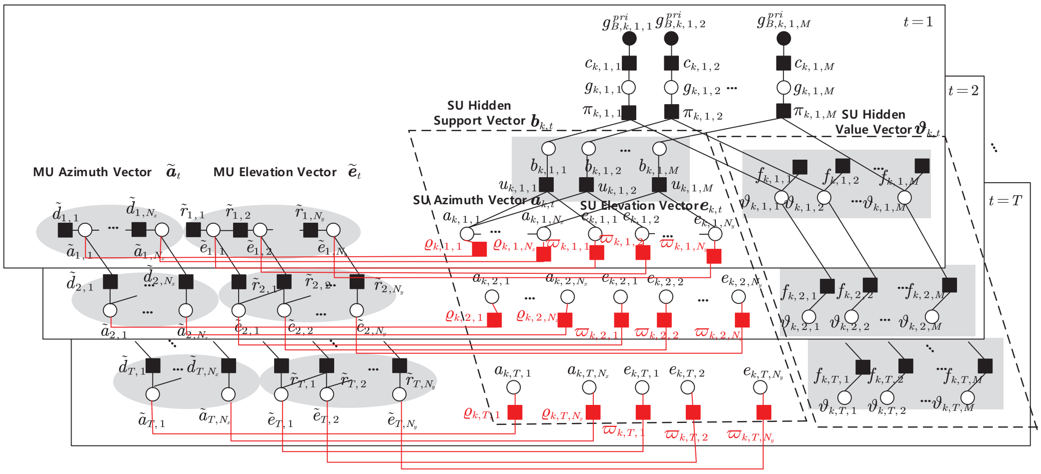


Figure 5.6: Factor graph representation of the MD-MM.

channel structure, we explain the structural details of the proposed algorithm based on the turbo framework with the proposed MD-MM MMSE denoiser. In [121], conventional turbo compressed sensing is achieved based on a two-step iteration framework: LMMSE estimator and i.i.d. prior-based MMSE estimator. Our proposed algorithm will be introduced in four parts: Module $\mathcal{A}$ LMMSE Estimator, Module $\mathcal{B}$ MD-MM MMSE denoiser, message passing through time, and 3D off-grid estimation.

5.5.1 Module $\mathcal{A}$: LMMSE Estimator

In Module $\mathcal{A}$, the angular channel vector $\boldsymbol{g}_{k,t}$ at time t for k^{th} UAV is estimated based on the the received signal $\boldsymbol{y}_{k,t}$ with the prior distribution $\mathcal{CN}\left(\boldsymbol{g}_{k,t};\boldsymbol{g}^{pri}_{\mathcal{A},k,t},v^{pri}_{\mathcal{A},k,t}\boldsymbol{I}\right)$. The extrinsic mean and variance are defined as [121]

$$\boldsymbol{g}^{post}_{\mathcal{A},k,t}=\boldsymbol{g}^{pri}_{\mathcal{A},k,t}+\frac{v^{pri}_{\mathcal{A},k,t}}{v^{pri}_{\mathcal{A},k,t}+\sigma^2_e}\boldsymbol{\Phi}^H_{k,t}\left(\boldsymbol{y}_{k,t}-\boldsymbol{\Phi}_{k,t}\boldsymbol{g}^{pri}_{\mathcal{A},k,t}\right) \tag{5.22}$$

and

$$v^{post}_{\mathcal{A},k,t}=v^{pri}_{\mathcal{A},k,t}-\frac{N_S}{M}\frac{\left(v^{pri}_{\mathcal{A},k,t}\right)^2}{v^{pri}_{\mathcal{A},k,t}+\sigma^2_e}, \tag{5.23}$$

Algorithm 5.1: MD-MM k^{th} SU Azimuth and Elevation Support Estimation

```
Input: $\tilde{\rho}_{t,n_x}^{(A)in}$, $\tilde{\rho}_{t,n_x}^{(A)dyna}$, $\tilde{\rho}_{t,n_y}^{(E)in}$, $\tilde{\rho}_{t,n_y}^{(E)dyna}$, $\rho_{k,t,n_x}^{(A)in}$, $\rho_{k,t,n_y}^{(E)in}$
Output: $\rho_{k,t,n_x}^{(A)out}$, $\rho_{k,t,n_y}^{(E)out}$
 1: Calculate $\upsilon_{u_{k,t,m} \to a_{k,t,n_x}}$ as in (C.8) and $\upsilon_{\varrho_{k,t,n_x} \to \tilde{a}_{t,n_x}}$ as in (C.17)
    # HMM $k^{th}$ SU Azimuth Support Estimation Based on 2D-MM MU Azimuth Support:
 2: if $t = 1$ then
 3:   Initialization: $\tilde{\gamma}_{t,1}^{(A)f} = \lambda$, $\tilde{\gamma}_{t,N_x}^{(A)b} = \frac{1}{2}$
 4:   for $n_x = 2, ..., N_x$ do
 5:     Use (C.11) to calculate $\tilde{\gamma}_{t,n_x}^{(A)f}$.
 6:   end for
 7:   for $n_x = 1, ..., N_x - 1$ do
 8:     Use (C.12) to calculate $\tilde{\gamma}_{t,n_x}^{(A)b}$.
 9:   end for
10: else if $t > 1$ then
11:   Initialization: $\tilde{\gamma}_{t,1}^{f} = q_{0,1}^{(T)}(1 - \tilde{\rho}_{t,1}^{dyna}) + \left(1 - q_{1,0}^{(T)}\right) \tilde{\rho}_{t,1}^{dyna}$, $\tilde{\gamma}_{t,N_x}^{b} = \frac{1}{2}$.
12:   for $n_x = 2, ..., N_x$ do
13:     Calculate the $\tilde{\gamma}_{t,n_x}^{(A)f}$ as in (C.13)
14:   end for
15:   for $n_x = 1, ..., N_x - 1$ do
16:     Calculate $\tilde{\gamma}_{t,n_x}^{(A)b} = \frac{1}{1+\tilde{\gamma}^{(A)}}$ use (C.14).
17:   end for
18: end if
19: Calculate the $k^{th}$ SU AoD azimuth support vector $\rho_{k,t,n_x}^{(A)out} = \tilde{\rho}_{\tilde{\boldsymbol{a}}_{t},k,n_x}^{out} \varsigma_{k,t,1,1}^{(A)}$.
    #Message Passing Over $k^{th}$ SU Azimuth to Elevation Support:
20: Calculate individual message $\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}$ using (C.19).
21: Calculate message $\upsilon_{a_{k,t,n_x} \to u_{k,t,m}}(a_{k,t,n_x})$ and $\upsilon_{u_{k,t,m} \to e_{k,t,n_y}}(e_{k,t,n_y})$ as in (C.18) and
    (C.20), respectively.
22: Calculate $\upsilon_{u_{k,t,m} \to e_{k,t,n_y}}$ as in (C.22) and $\upsilon_{\varpi_{k,t,n_y} \to \tilde{e}_{t,n_y}}$ as in (C.31).
    # HMM $k^{th}$ SU Elevation Support Estimation Based on 2D-MM MU Elevation:
23: Similar to the procedures from step 2 to step 18 by using elevation related
    parameters. Details can be found from (C.25) to (C.28).
24: Calculate the $k^{th}$ SU AoD elevation support vector $\rho_{k,t,n_y}^{(E)out} = \tilde{\rho}_{\tilde{\boldsymbol{e}}_{t},k,n_y}^{out} \varsigma_{k,t,1,1}^{(E)}$.
```

respectively. The extrinsic distribution of $\boldsymbol{g}_{k,t}$ is given by

$$\mathcal{CN}\left(\boldsymbol{g}_{k,t}; \boldsymbol{g}_{\mathcal{A},k,t}^{post}, v_{\mathcal{A},k,t}^{post}\boldsymbol{I}\right) \propto \mathcal{CN}\left(\boldsymbol{g}_{k,t}; \boldsymbol{g}_{\mathcal{A},k,t}^{pri}, v_{\mathcal{A},k,t}^{pri}\boldsymbol{I}\right) \mathcal{CN}\left(\boldsymbol{g}_{k,t}; \boldsymbol{g}_{\mathcal{A},k,t}^{ext}, v_{\mathcal{A},k,t}^{ext}\boldsymbol{I}\right), \quad (5.24)$$

where the extrinsic mean and variance are respectively given by

$$\boldsymbol{g}_{\mathcal{A},k,t}^{ext} = v_{\mathcal{A},k,t}^{ext} \left(\frac{\boldsymbol{g}_{\mathcal{A},k,t}^{post}}{v_{\mathcal{A},k,t}^{post}} - \frac{\boldsymbol{g}_{\mathcal{A},k,t}^{pri}}{v_{\mathcal{A},k,t}^{pri}} \right) \tag{5.25}$$

and

$$v_{\mathcal{A},k,t}^{ext} = \left(\frac{1}{v_{\mathcal{A},k,t}^{post}} - \frac{1}{v_{\mathcal{A},k,t}^{pri}} \right)^{-1}. \tag{5.26}$$

5.5.2 Module $\mathcal{B}$: Message Passing MMSE Denoiser

Following LMMSE estimator in module $\mathcal{A}$, the main task for module $\mathcal{B}$ is to exploit the k^{th} UAV spatial sparsity details of the proposed MD-MM in the multi-UAVs system. Hence, the extrinsic mean and variance pass through the graph, which means $\boldsymbol{g}_{\mathcal{B},k,t}^{pri} = \boldsymbol{g}_{\mathcal{A},k,t}^{ext}$ and $v_{\mathcal{B},k,t}^{pri} = v_{\mathcal{A},k,t}^{ext}$ [121]. In module $\mathcal{B}$, the purpose is to calculate the approximate posterior distributions of $p(g_{k,t,m}|\boldsymbol{g}_{\mathcal{B},k}^{pri})$ instead of (5.20) using the sum-product message passing structure shown in the factor graph. For message passing algorithm, the basic assumption is that $\boldsymbol{g}_{\mathcal{B},k,t}^{pri} = \boldsymbol{g}_{k,t} + \boldsymbol{z}_{k,t}$ where $\boldsymbol{z}_{k,t} \sim \mathcal{CN}(0, v_{\mathcal{B},k,t}^{pri}\boldsymbol{I})$ is independent of $\boldsymbol{g}_{k,t}$ [121].

The formulation details are elaborated in **Appendix C** and the message passing procedure can be summarized as follows:

- Firstly, in module $\mathcal{A}$, based on the prior information passed from the last time slot t, the message passing over the k^{th} UAV Gaussian hidden value vector $\boldsymbol{\vartheta}_{k,t}$ are given by (C.1) with μ^{dyna} and $(\sigma^{dyna})^2$ as the input to update the message $\upsilon_{\vartheta_{k,t,m} \to \pi_{k,t,m}}$.

- Secondly, in module $\mathcal{B}$, $\boldsymbol{g}_{\mathcal{B},k,t}^{pri}$ and $v_{\mathcal{B},k,t}^{pri}$ from module $\mathcal{A}$ are passed over the path $g_{k,t,m} \to \pi_{t,m} \to b_{k,t,m} \to u_{k,t,m} \to a_{k,t,n_x}$ using (C.2) to (C.5).

- Thirdly, the forward-backward message passing is executed over the proposed MD-MM MU azimuth branch with three paths: $a_{k,t,n_x} \to \varrho_{k,t,n_x} \to \tilde{a}_{t,n_x}$, the 2D-MM of the MU azimuth support $\tilde{\boldsymbol{a}}_t$, and $\tilde{a}_{t,n_x} \rightarrow \varrho_{k,t,n_x} \rightarrow a_{k,t,n_x}$ using (C.8) to (C.17).

- Then the message pass through path $a_{t,n_x} \rightarrow u_{t,m} \rightarrow e_{t,n_y}$ (using (C.18) to (C.20)) to another MD-MM elevation branch with similar three paths: $e_{k,t,n_y} \rightarrow \varpi_{k,t,n_y} \rightarrow \tilde{e}_{t,n_y}$, the 2D-MM of the MU elevation support $\tilde{\boldsymbol{e}}_t$, and $\tilde{e}_{t,n_y} \rightarrow \varpi_{k,t,n_y} \rightarrow e_{k,t,n_y}$ using (C.20) to (C.31). The overall details of the MD-MM azimuth and elevation SU joint hidden support estimation can be found in **Algorithm 5.1**.
- The message is finally passed back over the path $e_{k,t,n_y} \rightarrow u_{k,t,m} \rightarrow b_{k,t,m} \rightarrow \pi_{k,t,m} \rightarrow g_{k,t,m}$ using (C.32) to (C.35).

Based on the calculated updated messages $\upsilon_{\pi_{k,t,m} \rightarrow g_{k,t,m}}$, the posterior distributions can be written as

$$p\left(g_{k,t,m}|\boldsymbol{g}_{\mathcal{B},k,t}^{pri}\right) \propto \upsilon_{\pi_{k,t,m} \rightarrow g_{k,t,m}}\left(g_{k,t,m}\right) \upsilon_{c_{k,t,m} \rightarrow g_{k,t,m}}\left(g_{k,t,m}\right), \tag{5.27}$$

where $\upsilon_{c_{k,t,m} \rightarrow g_{k,t,m}}\left(g_{k,t,m}\right) = \mathcal{CN}\left(g_{k,t,m}; g_{\mathcal{B},k,t,m}^{pri}, v_{\mathcal{B},k,t}^{pri}\right)$. Then the posterior mean and variance of each element of $\boldsymbol{g}_{k,t}$ can be updated as

$$g_{\mathcal{B},k,t,m}^{post} = \int_{g_{k,t,m}} g_{k,t,m} p\left(g_{k,t,m}|\boldsymbol{g}_{\mathcal{B},k,t}^{pri}\right) \tag{5.28}$$

and

$$v_{\mathcal{B},k,t}^{post} = \frac{1}{M} \sum_{m=1}^{M} \mathrm{Var}\left(g_{k,t,m}|\boldsymbol{g}_{\mathcal{B},k,t}^{pri}\right), \tag{5.29}$$

where $\mathrm{Var}\left(g_{k,t,m}|\boldsymbol{g}_{\mathcal{B},k,t}^{pri}\right)$ denotes the conditional variance of $g_{k,t,m}$ given $\boldsymbol{g}_{\mathcal{B},k,t}^{pri}$. Then similar to (5.25) and (5.26), the extrinsic update mean and covariance can be written as

$$\boldsymbol{g}_{\mathcal{A},k,t}^{pri} = \boldsymbol{g}_{\mathcal{B},k,t}^{ext} = v_{\mathcal{A},k,t}^{pri}\left(\frac{\boldsymbol{g}_{\mathcal{B},k,t}^{post}}{v_{\mathcal{B},k,t}^{post}} - \frac{\boldsymbol{g}_{\mathcal{B},k,t}^{pri}}{v_{\mathcal{B},k,t}^{pri}}\right) \tag{5.30}$$

and

$$v_{\mathcal{A},k,t}^{pri} = v_{\mathcal{B},k,t}^{ext} = \left(\frac{1}{v_{\mathcal{B},k,t}^{post}} - \frac{1}{v_{\mathcal{B},k,t}^{pri}} \right)^{-1}. \tag{5.31}$$

5.5.3 Message Passing Through Time

For dynamic scenario, prior information which is also treated as the temporal correlation can be considered for better reconstruction of the time-varying channel model. Hence, we further consider the message passing through time for channel tracking. There are mainly three branches passing through time in Fig. 5.6 the factor graph of the MD-MM: the path of the k^{th} UAV Gaussian hidden value vector $\boldsymbol{\vartheta}_{k,t}$, the path of MU azimuth support $\tilde{\boldsymbol{a}}_t$ and the path of MU elevation support $\tilde{\boldsymbol{e}}_t$.

5.5.3.1 The k^{th} UAV Gaussian Hidden Value Vector $\boldsymbol{\vartheta}_{k,t}$ Passing Through Time

The message passing from factor node $\pi_{k,t,m}$ to variable node $\vartheta_{k,t,m}$ should be

$$v_{\pi_{k,t,m} \to \vartheta_{k,t,m}}^{true} \left(\vartheta_{k,t,m} \right) = \rho_{k,t,m}^{out} \mathcal{CN} \left(\vartheta_{k,t,m}; \boldsymbol{g}_{\mathcal{B},k,t}^{pri}, v_{\mathcal{B},k,t}^{pri} \right) + \left(1 - \rho_{k,t,m}^{out} \right) \mathcal{CN} \left(0; \boldsymbol{g}_{\mathcal{B},k,t}^{pri}, v_{\mathcal{B},k,t}^{pri} \right). \tag{5.32}$$

To be noticed, if $b_{k,t,m} = 0$, $g_{k,t,m}$ would be 0 in (5.9), which makes $\vartheta_{k,t,m}$ unobservable. Hence, similarly in [53], a threshold which is slightly less than 1 is introduced. The modified message passing is given by

$$v_{\pi_{k,t,m} \to \vartheta_{k,t,m}} \left(\vartheta_{k,t,m} \right) = \mathcal{CN} \left(\vartheta_{k,t,m}; \mu_{k,t,m}^{in}, (\sigma_{k,t,m}^{in})^2 \right), \tag{5.33}$$

with

$$\left(\mu_{k,t,m}^{in}, (\sigma_{k,t,m}^{in})^2 \right) = \begin{cases} \left(\frac{1}{\epsilon} \boldsymbol{g}_{\mathcal{B},k,t}^{pri}, \frac{1}{\epsilon^2} v_{\mathcal{B},k,t}^{pri} \right) & , \rho_{k,t,m}^{out} \leqslant \text{Th}, \\ \left(\boldsymbol{g}_{\mathcal{B},k,t}^{pri}, v_{\mathcal{B},k,t}^{pri} \right) & , \rho_{k,t,m}^{out} > \text{Th}, \end{cases} \tag{5.34}$$

where ϵ is set as a small positive value close to 0. Then the message passing from factor node $f_{k,t+1,m}$ to variable node $\vartheta_{k,t+1,m}$ is formed as

$$\upsilon_{f_{k,t+1,m}\to\vartheta_{k,t+1,m}}\left(\vartheta_{k,t+1,m}\right)=\mathcal{CN}\left(\vartheta_{k,t+1,m};\mu_{k,t+1,m}^{dyna},\left(\sigma_{k,t+1,m}^{dyna}\right)^{2}\right), \tag{5.35}$$

where

$$\mu_{k,t+1,m}^{dyna}=(1-\beta)\left(\frac{\left(\sigma_{k,t,m}^{dyna}\right)^{2}\left(\sigma_{k,t,m}^{in}\right)^{2}}{\left(\sigma_{k,t,m}^{dyna}\right)^{2}+\left(\sigma_{k,t,m}^{in}\right)^{2}}\right)\times\left(\frac{\mu_{k,t,m}^{dyna}}{\left(\sigma_{k,t,m}^{dyna}\right)^{2}}+\frac{\mu_{k,t,m}^{in}}{\left(\sigma_{k,t,m}^{in}\right)^{2}}\right)+\beta\mu \tag{5.36}$$

and

$$\left(\sigma_{k,t+1,m}^{dyna}\right)^{2}=(1-\beta)^{2}\left(\frac{\left(\sigma_{k,t,m}^{dyna}\right)^{2}\left(\sigma_{k,t,m}^{in}\right)^{2}}{\left(\sigma_{k,t,m}^{dyna}\right)^{2}+\left(\sigma_{k,t,m}^{in}\right)^{2}}\right)+\beta^{2}\zeta. \tag{5.37}$$

5.5.3.2 The MU Azimuth Support $\tilde{a}_t$ Passing Through Time

The azimuth support prior information passing to the next time $\upsilon_{\tilde{a}_{t,n_x}\to\tilde{d}_{t+1,n_x}}\left(\tilde{a}_{t,n_x}\right)$ can be given as

$$\upsilon_{\tilde{a}_{t,n_x}\to\tilde{d}_{t+1,n_x}}\left(\tilde{a}_{t,n_x}\right)=\tilde{\rho}_{t+1,n_x}^{(A)dyna}\delta\left(\tilde{a}_{t,n_x}-1\right)+\left(1-\tilde{\rho}_{t+1,n_x}^{(A)dyna}\right)\delta\left(\tilde{a}_{t,n_x}\right), \tag{5.38}$$

with

$$\tilde{\rho}_{t+1,n_x}^{(A)dyna}=\frac{\gamma_{t,n_x}^{(A)f}\gamma_{t,n_x}^{(A)b}\tilde{\rho}_{t,n_x}^{(A)in}}{\gamma_{t,n_x}^{(A)f}\gamma_{t,n_x}^{(A)b}\tilde{\rho}_{t,n_x}^{(A)in}+(1-\gamma_{t,n_x}^{(A)f})(1-\gamma_{t,n_x}^{(A)b})(1-\tilde{\rho}_{t,n_x}^{(A)in})}. \tag{5.39}$$

5.5.3.3 The MU Elevation Support $\tilde{e}_t$ Passing Through Time

Similarly, the elevation support prior information passing to the next time can be written as

$$\upsilon_{\tilde{e}_{t,n_y}\to\tilde{r}_{t+1,n_y}}\left(\tilde{e}_{t,n_y}\right)=\tilde{\rho}_{t+1,n_y}^{(E)dyna}\delta\left(\tilde{e}_{t,n_y}-1\right)+\left(1-\tilde{\rho}_{t+1,n_y}^{(E)dyna}\right)\delta\left(\tilde{e}_{t,n_y}\right), \tag{5.40}$$

with

$$\tilde{\rho}^{(E)dyna}_{t+1,n_y} = \frac{\gamma^{(E)f}_{t,n_y}\gamma^{(E)b}_{t,n_y}\tilde{\rho}^{(E)in}_{t,n_y}}{\gamma^{(E)f}_{t,n_y}\gamma^{(E)b}_{t,n_y}\tilde{\rho}^{(E)in}_{t,n_y} + (1-\gamma^{(E)f}_{t,n_y})(1-\gamma^{(E)b}_{t,n_y})(1-\tilde{\rho}^{(E)in}_{t,n_y})}. \tag{5.41}$$

5.5.4 3D Off-grid Estimation

As there is no closed-form expression due to the complex MD-MM structure of $\boldsymbol{g}_k^{(t)}$, it's hard to directly maximize the log-likelihood function $\ln p\left(\boldsymbol{y}_k^{(t)}, \Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right)$ in (5.21). We extend the proposed in-exact majorization-minimization (MM) algorithm in [42] to help us solve the off-grid problem. Firstly, we iteratively construct the lower bound for $\ln p\left(\boldsymbol{y}_k^{(t)}, \Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right)$ which can be named as surrogate function, i.e., for any fixed azimuth and elevation offset pairs $\left(\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right)$, the surrogate function can be defined as [42]

$$\begin{aligned}&\mathcal{L}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}|\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right) = \\ &\int p\left(\boldsymbol{g}_{k,t}|\boldsymbol{y}_k^{(t)}, \Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right)\ln\frac{p\left(\boldsymbol{g}_{k,t}\boldsymbol{y}_k^{(t)}, \Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right)}{p\left(\boldsymbol{g}_{k,t}|\boldsymbol{y}_k^{(t)}, \Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right)}d\boldsymbol{g}_{k,t},\end{aligned} \tag{5.42}$$

which satisfies the following properties:

$$\mathcal{L}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}|\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right) \leqslant \ln p\left(\boldsymbol{y}_k^{(t)}, \Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right), \forall\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}, \tag{5.43}$$

$$\mathcal{L}\left(\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}|\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right) = \ln p\left(\boldsymbol{y}_k^{(t)}, \Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right), \tag{5.44}$$

$$\left.\frac{\partial\mathcal{L}(\Delta\boldsymbol{\theta}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}|\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t})}{\partial\Delta\boldsymbol{\theta}_{k,t}}\right|_{\Delta\boldsymbol{\theta}_{k,t}=\Delta\dot{\boldsymbol{\theta}}_{k,t}} = \left.\frac{\partial\ln p\left(\boldsymbol{y}_k^{(t)}, \Delta\boldsymbol{\theta}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t}\right)}{\partial\Delta\boldsymbol{\theta}_{k,t}}\right|_{\Delta\boldsymbol{\theta}_{k,t}=\Delta\dot{\boldsymbol{\theta}}_{k,t}}, \tag{5.45}$$

Algorithm 5.2: 3D Off-grid Estimation

```
Input: $\boldsymbol{g}^{post}_{\mathcal{B},k,t,m}$, $v^{post}_{\mathcal{B},k,t}$, , $\Delta\boldsymbol{\theta}^{(0)}_{k,t} = \mathbf{0}$, $\Delta\boldsymbol{\phi}^{(0)}_{k,t} = \mathbf{0}$, $I$.
Output: $\Delta\boldsymbol{\theta}^{(opt)}_{k,t}$, $\Delta\boldsymbol{\phi}^{(opt)}_{k,t}$
for $i = 1 : I$ do
    Azimuth offset $\Delta\boldsymbol{\theta}^{(i+1)}_{k,t}$ can be updated use gradient update (5.49) and (D.2)
    with known $i^{th}$ iteration elevation offset $\Delta\boldsymbol{\phi}^{(i)}_{k,t}$.
    Elevation offset $\Delta\boldsymbol{\phi}^{(i+1)}_{k,t}$ can be updated similarly with known $(i+1)^{th}$ iteration
    azimuth offset $\Delta\boldsymbol{\theta}^{(i+1)}_{k,t}$.
end for
$\Delta\boldsymbol{\theta}^{(opt)}_{k,t} = \Delta\boldsymbol{\theta}^{(I)}_{k,t}, \Delta\boldsymbol{\phi}^{(opt)}_{k,t} = \Delta\boldsymbol{\phi}^{(I)}_{k,t}$
```

$$\left.\frac{\partial\mathcal{L}(\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}|\Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\dot{\boldsymbol{\phi}}_{k,t})}{\partial\Delta\boldsymbol{\phi}_{k,t}}\right|_{\Delta\boldsymbol{\phi}_{k,t}=\Delta\dot{\boldsymbol{\phi}}_{k,t}} = \left.\frac{\partial\ln p\left(\boldsymbol{y}^{(t)}_{k}, \Delta\dot{\boldsymbol{\theta}}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}\right)}{\partial\Delta\boldsymbol{\phi}_{k,t}}\right|_{\Delta\boldsymbol{\phi}_{k,t}=\Delta\dot{\boldsymbol{\phi}}_{k,t}}. \tag{5.46}$$

Then $\Delta\boldsymbol{\theta}_{k,t}$ and $\Delta\boldsymbol{\phi}_{k,t}$ can be updated at each iteration as

$$\Delta\boldsymbol{\theta}^{(i+1)}_{k,t} = \underset{\boldsymbol{\theta}_{k,t}}{\operatorname{argmax}}\,\mathcal{L}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}^{(i)}_{k,t}|\Delta\boldsymbol{\theta}^{(i)}_{k,t}, \Delta\boldsymbol{\phi}^{(i)}_{k,t}\right), \tag{5.47}$$

$$\Delta\boldsymbol{\phi}^{(i+1)}_{k,t} = \underset{\boldsymbol{\phi}_{k,t}}{\operatorname{argmax}}\,\mathcal{L}\left(\Delta\boldsymbol{\theta}^{(i+1)}_{k,t}, \Delta\boldsymbol{\phi}_{k,t}|\Delta\boldsymbol{\theta}^{(i+1)}_{k,t}, \Delta\boldsymbol{\phi}^{(i)}_{k,t}\right), \tag{5.48}$$

with $(\cdot)^{(i)}$ represents for the $i^{(th)}$ iteration. To be noticed, for the $(i+1)^{th}$ iteration elevation update, we use the updated $(i+1)^{th}$ azimuth offset as fixed known knowledge in (5.48). However, the problems in (5.47) and (5.48) are non-convex. Hence the gradient update can be applied as follows:

$$\Delta\boldsymbol{\theta}^{(i+1)}_{k,t} = \Delta\boldsymbol{\theta}^{(i)}_{k,t} + \Delta^{(i)}_{\theta}\left.\frac{\partial\mathcal{L}\left(\Delta\boldsymbol{\theta}_{k,t}, \Delta\boldsymbol{\phi}^{(i)}_{k,t}|\Delta\boldsymbol{\theta}^{(i)}_{k,t}, \Delta\boldsymbol{\phi}^{(i)}_{k,t}\right)}{\partial\Delta\boldsymbol{\theta}_{k,t}}\right|_{\Delta\boldsymbol{\theta}_{k,t}=\Delta\boldsymbol{\theta}^{(i)}_{k,t}} \tag{5.49}$$

and

$$\varDelta\boldsymbol{\phi}_{k,t}^{(i+1)} = \varDelta\boldsymbol{\phi}_{k,t}^{(i)} + \varDelta_{\phi}^{(i)} \left.\frac{\partial\mathcal{L}\left(\varDelta\boldsymbol{\theta}_{k,t}^{(i+1)}, \varDelta\boldsymbol{\phi}_{k,t}|\varDelta\boldsymbol{\theta}_{k,t}^{(i+1)}, \varDelta\boldsymbol{\phi}_{k,t}^{(i)}\right)}{\partial\varDelta\boldsymbol{\phi}_{k,t}}\right|_{\varDelta\boldsymbol{\phi}_{k,t}=\varDelta\boldsymbol{\phi}_{k,t}^{(i)}}, \tag{5.50}$$

where $\varDelta_{\theta}^{(i)}$ and $\varDelta_{\phi}^{(i)}$ are the stepsizes. The off-grid estimation algorithm is summarized in **Algorithm 5.2** and the detailed equation can be found in **Appendix D**. Finally, the overall proposed MD-DTAMP can be summarized in **Algorithm 5.3**.

5.6 Numerical Results

In this section, the complexity and the estimation performance of the proposed algorithm MD-DTAMP with and without 3D off-grid estimation are evaluated. We assume that the proposed algorithm is performed in the distributed UAVs. For the space-air link of each single UAV, the sparsity ratio for both azimuth and elevation directions is $\lambda = \frac{1}{N_x} = \frac{1}{N_y}$ which indicates only one dominant LoS ray for each space-air link. The parameters are set as follows[3]: We assume that there are total 3 UAVs in the system. The MU azimuth and elevation spatial correlation parameters $\{q_{0,1}^{(AS)}, q_{1,0}^{(AS)}, q_{0,1}^{(ES)}, q_{1,0}^{(ES)}\} = \{0.05, 0.05, 0.025, 0.025\}$, the MU azimuth and elevation temporal correlation parameters $\{q_{0,1}^{(AT)}, q_{1,0}^{(AT)}, q_{0,1}^{(ET)}, q_{1,0}^{(ET)}\} = \{0.05, 0.05, 0.025, 0.175\}$, the MU azimuth and elevation joint parameters $\{q_{1,1,1}^{(A)}, q_{0,0,1}^{(A)}, q_{0,1,1}^{(A)}, q_{1,0,1}^{(A)}, q_{1,1,1}^{(E)}, q_{0,0,1}^{(E)}, q_{0,1,1}^{(E)}, q_{1,0,1}^{(E)}\} = \{0.9958, 0.0013, 0.3276, 0.3936, 0.99546, 0.0007, 0.5, 0.1078\}$, the hidden value vector parameters for k^{th} UAV $\{\mu_k, \sigma_k^2, \beta_k, \zeta_k, \varsigma_{k,t,1,1}^{(A)}, \varsigma_{k,t,1,1}^{(E)}\} = \{0, \frac{1}{3}, 0.5, 1, 0.9375, 0.9375\}$, $\mathrm{Th} = 1 - 10^{-2}$, $\gamma_{\boldsymbol{b}_{k,t}} = 1$, the azimuth range $\theta^{\text{range}} = \pi$ and the elevation range $\phi^{\text{range}} = \pi/4$, 3D Off-grid azimuth and elevation update step sizes are $\varDelta_{\theta}^{(i)} = \pi/(100 \times M)$ and $\varDelta_{\phi}^{(i)} = \pi/(4 \times 100 \times M)$.

To be noticed, in our simulation, our proposed algorithm MD-DTAMP with 3D off-grid estimation is presented in **Algorithm 5.3**. MD-DTAMP on grid which is **Algo-**

[3] The parameters can be trained through expectation maximization algorithm which can refer to [53] for details. The MU azimuth and elevation related parameters for mutiple UAVs system in this paper are referred 2D settings in [112].

Algorithm 5.3: MD-DTAMP Dynamic Channel Tracking

```
Input: $\boldsymbol{y}_{k,1}, ..., \boldsymbol{y}_{k,T}$, sensing matrix $\boldsymbol{\Phi}(\mathbf{0}, \mathbf{0})$
Output: $\hat{\boldsymbol{g}}_{k,1}, ..., \hat{\boldsymbol{g}}_{k,T}$
1:  for t=1,...,T do
2:    while not converge do
3:      Initialization: $\boldsymbol{g}^{pri}_{\mathcal{A},k,t} = \mathbf{0}$, $v^{pri}_{\mathcal{A},k,t} = \sigma^2$.
    #Module $\mathcal{A}$ LMMSE estimator:
4:      Update $\boldsymbol{g}^{post}_{\mathcal{A},k,t}$ and $v^{post}_{\mathcal{A},k,t}$ in (5.22) and (5.23), respectively.
5:      Calculate $\boldsymbol{g}^{pri}_{\mathcal{B},k,t} = \boldsymbol{g}^{ext}_{\mathcal{A},k,t}$, $v^{pri}_{\mathcal{B},k,t} = v^{ext}_{\mathcal{A},k,t}$ using (5.25) and (5.26).
    #Module $\mathcal{B}$ MD-MM MMSE denoiser:
6:      Message passing over Gaussian hidden value vector $\boldsymbol{\vartheta}_{k,t}$ using (C.1).
7:      Message Passing Over $g_{k,t,m} \rightarrow \pi_{k,t,m} \rightarrow b_{k,t,m} \rightarrow u_{k,t,m} \rightarrow a_{k,t,n_x}$ using
    (C.2)-(C.7)
8:      AoD SU azimuth support $\boldsymbol{a}_{k,t}$ and SU elevation support $\boldsymbol{e}_{k,t}$ estimation in
    Algorithm 5.1.
9:      Message Passing Over the Path $e_{k,t,n_y} \rightarrow u_{k,t,m} \rightarrow b_{k,t,m} \rightarrow \pi_{k,t,m} \rightarrow g_{k,t,m}$
    using (C.32)-(C.35)
10:     Calculate the approximate posterior distributions $p\left(g_{k,t,m}|\boldsymbol{g}^{pri}_{\mathcal{B},k,t}\right)$ using (5.27).
11:     Update $\boldsymbol{g}^{post}_{\mathcal{B},k,t,m}$ and $v^{post}_{\mathcal{B},k,t}$ using (5.28) and (5.29).
12:     Update $\boldsymbol{g}^{pri}_{\mathcal{A},k,t} = \boldsymbol{g}^{ext}_{\mathcal{B},k,t}$ and $v^{pri}_{\mathcal{A},k,t} = v^{ext}_{\mathcal{B},k,t}$ using (5.30) and (5.31).
13:     Repeat Module $\mathcal{A}$ and Module $\mathcal{B}$ until convergence.
14:     Hence $\hat{p}\left(\boldsymbol{g}_{k,t}|\boldsymbol{y}^{(t)}_k, \Delta\boldsymbol{\theta}^{(opt)}_{k,t}, \Delta\boldsymbol{\phi}^{(opt)}_{k,t}\right) = \mathcal{CN}\left(\boldsymbol{g}^{post}_{\mathcal{B},k,t}, v^{post}_{\mathcal{B},k,t}\boldsymbol{I}\right)$.
    #3D Off-grid estimation:
15:     Update azimuth offset $\Delta\boldsymbol{\theta}^{(opt)}_{k,t}$ and elevation offset $\Delta\boldsymbol{\phi}^{(opt)}_{k,t}$ use Algorithm 5.2.
16:   end while
17:   Update dynamic messages to the next time slot $(t+1)$: $\upsilon_{a_{k,t,n_x} \rightarrow d_{k,t+1,n_x}}$,
    $\upsilon_{e_{k,t,n_y} \rightarrow r_{k,t+1,n_y}}$ and $\upsilon_{f_{k,t+1,m} \rightarrow \vartheta_{k,t+1,m}}$ using (5.38), (5.40) and (5.35).
18: end for
```

rithm 5.3 without step 15. The benchmark algorithms which can be employed in our MU 3D channel model are SBL and D-OMP proposed in [42] and [108], respectively.

5.6.1 Complexity Analysis

The computation complexity of the LMMSE estimator in module $\mathcal{A}$ and MD-MM MMSE denoiser in module $\mathcal{B}$ are $\mathcal{O}\left(MN_S\right)$ and $\mathcal{O}\left(N_x + N_y\right)$, respectively. Hence, the total complexity of the proposed MD-DTAMP without 3D off-grid estimation over time T is $\mathcal{O}\left(T\left(MN_S + N_x + N_y\right)\right)$ for each iteration. For off-grid estimation, with defined stepsize $\Delta^{(i)}_{\theta}$, the complexity for update azimuth offset $\Delta\boldsymbol{\theta}^{(i+1)}_{k,t}$ is $\mathcal{O}\left(N_S M^2\right)$. The reason why

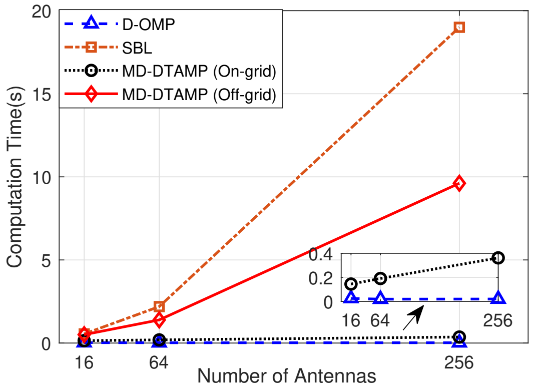


Figure 5.7: Computation time of various algorithms versus the number of antennas when SNR $= 10$dB, $N_s = 10, T = 50, M = 16, 64, 256$ and $N_x = N_y = 4, 8, 16$.

the complexity is not $\mathcal{O}\left(N_S N_x^2\right)$ is that the original dimension has to be extended for paring the azimuth and elevation angles. Similarly, the complexity for update elevation offset $\varDelta\boldsymbol{\phi}_{k,t}^{(i+1)}$ is also $\mathcal{O}\left(N_S M^2\right)$.

Fig. 5.7 shows the computation time of various algorithms versus the number of antennas. It can be observed that the proposed on-gird MD-DTAMP shows comparable complexity compared with D-OMP. Furthermore, compared with SBL algorithm which requires matrix multiplication and inversion with high computation complexity, the proposed MD-DTAMP with off-grid estimation consumes less computation time for different number of antennas.

5.6.2 Simulation Results

We choose TNMSE as the performance metric as [53]

$$\text{TNMSE} = \frac{1}{T}\sum_{t}^{T}\frac{\|\boldsymbol{g}_t - \hat{\boldsymbol{g}}_t\|^2}{\|\boldsymbol{g}_t\|^2}, \tag{5.51}$$

where $\hat{\boldsymbol{g}}_t$ is the estimated result of $\boldsymbol{g}_t$ in time t. This performance metric is to evaluate the recovery performance of various algorithms.

In Fig. 5.8, we further compare the TNMSE performance metric versus SNR among various algorithms with antenna array elements set as $M = 64, 256$. With SNR increases,

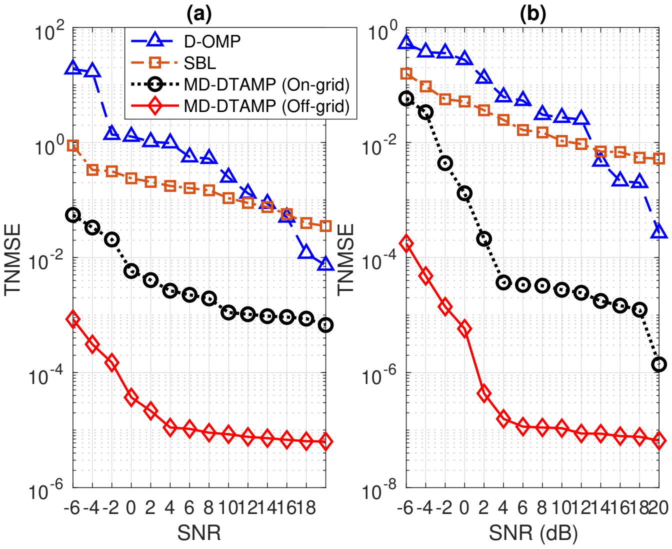


Figure 5.8: TNMSE versus SNR when $T = 35$. (a) $M = 64, N_x = N_y = 8, N_s = 20$. (b) $M = 256, N_x = N_y = 16, N_s = 26$.

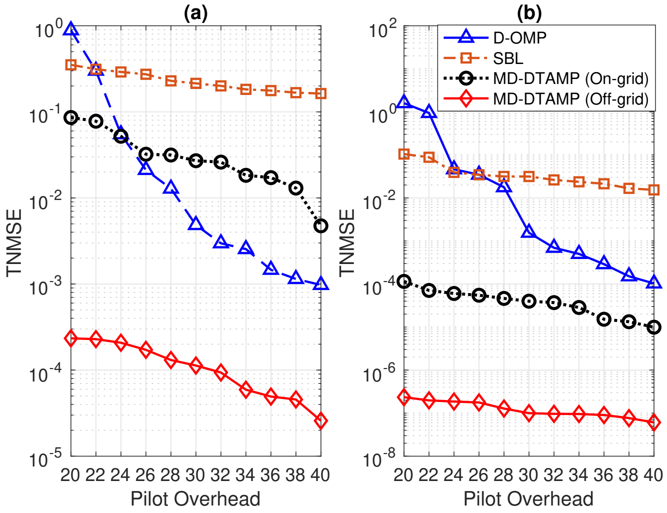


Figure 5.9: TNMSE versus total pilot overheads when $T = 35$. (a) $M = 64, N_x = N_y = 8, \mathrm{SNR} = 0\mathrm{dB}$. (b) $M = 256, N_x = N_y = 16, \mathrm{SNR} = 5\mathrm{dB}$.

all four algorithms show better recovery performance. To be noticed, our proposed MD-DTAMP on-grid and off-grid tracking algorithms always show superior performance compared to benchmark algorithms. Moreover, the off-grid can further improve the performance through gradient update of the 3D offset.

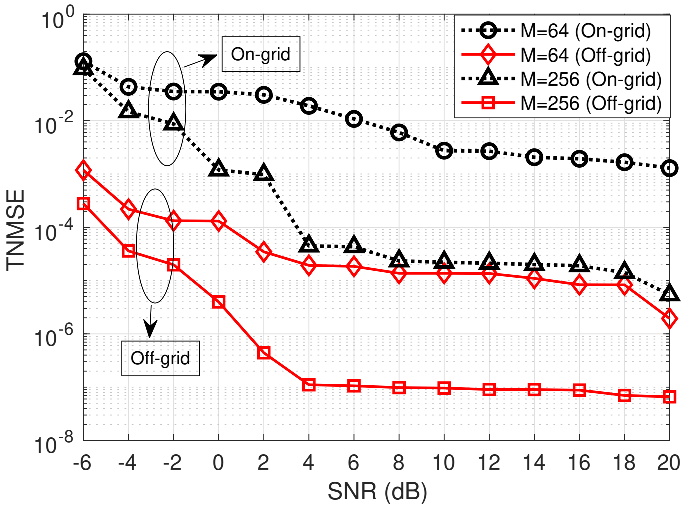


Figure 5.10: TNMSE versus SNR when $T = 35, N_s = 22$.

In Fig. 5.9, we further consider the relationship between the tracking performance and the training pilot overheads. It is easy to be noticed that our proposed algorithm with or without off-grid estimation can maintain a good performance with smaller pilot overheads. This proves that our proposed algorithms can achieve a solid performance by taking the dynamic channel spatial and temporal correlation into consideration.

The proposed on-grid and off-grid algorithms are compared in Fig. 5.10. The off-grid estimation achieves sufficient performance gain over the on-grid algorithm. Moreover, with M increases from 64 to 256, the tracking performance also improves. This is because with a larger antenna array and denser virtual gird, and the on-grid 3D tracking angle can be estimated with a narrower grid and closer to the true angle.

5.7 Summary

This chapter exploited the on-grid and off-grid 3D channel modelling and tracking for space-air communications with multiple UAVs and an LEO satellite in the system. The relative displacement between the orbit movement of the LEO satellite and the 3D trajectory of multiple UAVs was modelled based on the statistical dynamic channel model called MD-MM. The proposed MD-MM was composed of the insight probability relationship of the spatial and temporal correlation of MU hidden support vector, SU joint hidden support vector, and SU hidden value vector. We further developed a novel MD-DTAMP algorithm to track the dynamic channel recursively. Furthermore, the gradient update for the 3D off-grid estimation scheme was proposed, which can iteratively find the azimuth and elevation offset. In addition, we compared the computation complexity between benchmark algorithms and the proposed algorithm with and without the off-grid scheme. Simulations have verified that the proposed algorithm derived from the MD-MM dynamic channel model can better exploit the more realistic space-air 3D dynamic channel to achieve more accurate recovery performance over other benchmark algorithms with smaller pilot overheads.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The space-air-ground integrated network (SAGIN) is a structure based on the modern communication networks and interconnecting space, air, and ground segments to form a huge and complicated network topology. This multidimensional network can provide seamless connectivity to rural, ocean, and mountain areas for many practical fields, such as earth mapping, intelligent transportation systems, military missions, disaster rescue, and so on. Due to the enormous smart devices brought online, the heavily utilized microwave spectrum bands below 6GHz become insufficient to attain remarkable data rate enhancement for emerging applications. To resolve the spectrum crunch crisis and meet the demands of beyond the fifth generation (B5G) communication, millimetre wave (mmWave) frequencies have been drawn hugely attention to help support dramatically increasing mobile data. Moreover, mmWave frequencies such as Ka-band has been already used in low earth orbit (LEO) and medium earth orbit (MEO) satellite based communications. Therefore, the future SAGIN with multiple mmWave frequencies operated across three segments is the inevitable trend. To mitigate the atmospheric attenuation and the severe penetration shadowing of mmWave transmission, massive multiple input multiple output (MIMO) and intelligent reflecting surfaces (IRSs) are two advanced technologies that can generate or reconfigure the directional beams to improve communication performance in SAGIN. Since good knowledge of dynamic channel state information (CSI) is the cornerstone of building a stable connectivity, in this thesis, we presented several 3D channel tracking algorithms based on special characteristics in different cross-segment links in SAGIN specifically.

In Chapter 3, we firstly proposed a three dimensional (3D) geometry-based IRS-assisted air-ground mmWave channel model with both UAV navigation and mobile user movement. We further developed a deep learning (DL) based channel tracking algorithm which consists of two modules: deep neural network (DNN) for channel pre-estimation denoising and stacked bi-directional long short term memory (Stacked Bi-LSTM) for channel tracking. The Stacked Bi-LSTM is a framework that can trace back historical time sequence together with bidirectional structure over multiple stacked layers. The proposed algorithm achieves superior prediction accuracy with small pilot overheads.

In Chapter 4, we proposed a statistical dynamic mmWave channel model for space-air downlink with a single user (SU) unmanned ariel vehicle (UAV) in the system. The model named 3D two-dimensional Markov model (3D-2D-MM) explores the probabilistic insight relationship of both hidden value vector and joint hidden support vector of the dynamic model. The realistic 3D on-grid azimuth and elevation directions are considered in our model. The spatial sparsity and the temporal correlation are studied for each direction. Moreover, we further derive a novel 3D dynamic turbo approximate message passing (3D-DTAMP) algorithm to track the dynamic channel based on the 3D-2D-MM priors. The numerical results prove that our proposed algorithm can achieve better channel tracking performance compared to benchmark algorithms with smaller pilot overheads.

Lastly, in Chapter 5, we further proposed a more realistic statistical dynamic mmWave channel model, namely multi-dimensional Markov model (MD-MM) for multiple users (MU) UAVs in the space-air downlinks with 3D off-grid angle offsets. The spatial and temporal relationships of MU hidden support vector, SU joint hidden support vector, and SU hidden value vector are explored. More importantly, this model defines specific transition probabilities that connect the SU and MU hidden support vector for both azimuth and elevation directions. Based on the proposed MD-MM, a novel multi-dimensional dynamic turbo approximate message passing (MD-DTAMP) algorithm is derived. Furthermore, a gradient update scheme to recursively find the off-grid directions is developed. Numerical results verify the superiority of our proposed algorithm.

To sum up, in this thesis, various 3D channel tracking algorithms are proposed by considering different characteristics of the cross-segment links in SAGIN. For IRS-assisted air-ground links, a DL-based channel tracking algorithm is proposed. For space-air links with both SU and MU cases, the channel tracking algorithms are derived based on specific sparsity structures of the channel that include both spatial and temporal correlations. The proposed channel tracking algorithms for different links were proved to be superior to the benchmark algorithms.

6.2 Future Work

There are a few research opportunities will be addressed in the future work.

6.2.1 Distributed Learning for MU Channel Tracking in SAGIN

Conventionally, channel estimation via machine learning (ML) requires model training on a dataset, which usually includes the received pilot signals as input and channel data as output. In these traditional ways, model training is mostly done in a centralized manner, where the whole training dataset is collected from the users at the base station (BS). Due to the necessity of privacy protection in wireless communication and the outstanding privacy protection characteristics of distributed learning, the combination of distributed learning and wireless communication has gradually emerged. Unlike traditional machine learning, which processes data centrally, the physical framework of distributed learning includes several clients and a central server. And the data for training is distributed to each client which means managed and processed locally. In SAGIN, data can be transmitted and collected through multiple segments, which impose the possibility of utilizing distributed learning for MU channel tracking.

6.2.2 Unsupervised DL-based Online Training Design for Channel Tracking

Our proposed supervised DL-based channel tracking algorithm is designed based on the supervised manner, which means during the training phase, the true channel information is assumed to be known in advance as labelled data. However, it is not practical in the real world. Hence an online training scheme with certain designed loss functions for training without the knowledge of the true channel information should be studied. To be specific, in future work, the DL-based channel estimator can be trained online in the mobile devices based on real-time received pilot measurements from the BS without knowledge of the true channels, and can simultaneously generate channel tracking in real-time.

6.2.3 Federated Learning-based IRSs-assisted Channel Tracking

In Federated Learning, acting as one of the distributed learning algorithms, instead of sending the whole dataset, only the model updates, i.e., gradients of the model parameters, are transmitted. For the IRSs-assisted mmWave communication, the controller of IRSs can act as the central server, and the user equipment participating in communication as the client. The dataset held by the client consists of a series of sampled data points/mobile users within its range of activity, each of which consists of the CSI corresponding to its location and the optimal configuration of IRS. By utilizing federated Learning, an optimal global channel model can be trained in the specific communication area. Apart from IRSs act as central servers, an access point connected to multiple IRSs can also be regarded as the central server, where each IRS as a client and the dataset held by each IRS is composed of a series of optimal IRS configurations and the sampled mobile devices location information pairs. Through the federated learning of all IRSs, the optimal model can be trained to acquire overall CSI.

Appendix A

Training Pilot Matrix Design

It has been proved that a partial DFT sensing matrix outperforms the independent identically distributed sensing matrix in solving the CS problem with AMP algorithm [37], [118]. Hence, in our proposed model, the sensing matrix $\boldsymbol{\varPhi}_{k,t}\left(\varDelta\boldsymbol{\theta}_{k,t}, \varDelta\boldsymbol{\phi}_{k,t}\right)$ is referred to be a partial DFT random permutation matrix as $\boldsymbol{\varPhi}_{k,t}\left(\varDelta\boldsymbol{\theta}_{k,t}, \varDelta\boldsymbol{\phi}_{k,t}\right) \approx \boldsymbol{S}_{\mathrm{SEL}}\boldsymbol{S}_{\mathrm{DFT}}\boldsymbol{S}_{\mathrm{RAND}}$, where $\boldsymbol{S}_{\mathrm{SEL}}$ is a selection matrix consisting of randomly selected and reordered rows of $N_S \times M$ identity matrix, $\boldsymbol{S}_{\mathrm{DFT}}$ is the $M \times M$ DFT matrix, and $\boldsymbol{S}_{\mathrm{RAND}}$ is random permutation matrix generated by a randomly reordered $M \times M$ identity matrix. For large M antenna elements array, $\boldsymbol{S}_{\mathrm{SEL}}\boldsymbol{S}_{\mathrm{DFT}}\boldsymbol{S}_{\mathrm{RAND}} = \boldsymbol{S}_k^T\left(\tilde{\boldsymbol{A}}^H\left(\tilde{\boldsymbol{\theta}}_{k,t} + \varDelta\boldsymbol{\theta}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t} + \varDelta\boldsymbol{\phi}_{k,t}\right)\right)^T$. Hence, for simplicity with fixed on-grid and approximate partial orthogonal, the pilot matrix $\boldsymbol{S}_k^T = \boldsymbol{S}_{\mathrm{SEL}}\boldsymbol{S}_{\mathrm{DFT}}\boldsymbol{S}_{\mathrm{RAND}}\left(\left(\tilde{\boldsymbol{A}}^H\left(\tilde{\boldsymbol{\theta}}_{k,t}, \tilde{\boldsymbol{\phi}}_{k,t}\right)\right)^T\right)^{-1}$.

Appendix B

3D-2D-MM Message Passing in Module $\mathcal{B}$

The details of the message passing sequence paths are shown in Fig. B1.

1. **Message Passing Over Gaussian Hidden Value Vector $\boldsymbol{\vartheta}_t$:** The message $\upsilon_{f_{t,m}\to\vartheta_{t,m}}$ passed from the last time slot can be formed as $\mathcal{CN}(\vartheta_{t,m}; \mu_{t,m}^{dyna}, (\sigma_{t,m}^{dyna})^2)$. The message passed from variable node $\vartheta_{t,m}$ to factor node $\pi_{t,m}$ can be formed as

$$\upsilon_{\vartheta_{t,m}\to\pi_{t,m}} = \mathcal{CN}(\vartheta_{t,m}; \mu_{t,m}^{out}, (\sigma_{t,m}^{out})^2), \tag{B.1}$$

with $(\mu_{t,m}^{out}, (\sigma_{t,m}^{out})^2) = (\mu_{t,m}^{dyna}, (\sigma_{t,m}^{dyna})^2)$. When $t = 1$, we set $(\mu_{1,m}^{dyna}, (\sigma_{1,m}^{dyna})^2) = (\mu, \sigma^2)$

2. **Message Passing Over $g_{t,m} \to \pi_{t,m} \to b_{t,m} \to u_{t,m} \to a_{t,n_x}$:** The message from variable node $g_{t,m}$ to factor node $\pi_{t,m}$ is

$$\upsilon_{g_{t,m}\to\pi_{t,m}}(g_{t,m}) = \mathcal{CN}\left(g_{t,m}; g_{\mathcal{B},t,m}^{pri}, v_{\mathcal{B},t,m}^{pri}\right). \tag{B.2}$$

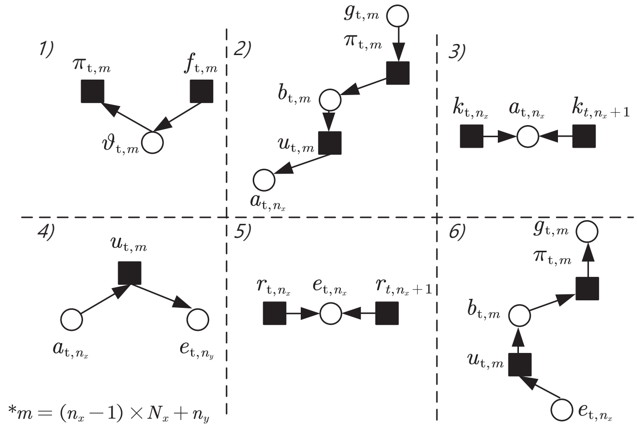


Figure B1: The message passing sequence paths in Module $\mathcal{B}$.

The message from factor node $\pi_{t,m}$ to variable node $b_{t,m}$ is

$$\upsilon_{\pi_{t,m}\to b_{t,m}}\left(b_{t,m}\right)=\rho^{in}_{b,t,m}\delta\left(b_{t,m}-1\right)+\left(1-\rho^{in}_{b,t,m}\right)\delta\left(b_{t,m}\right), \tag{B.3}$$

where $\rho^{in}_{\boldsymbol{b}_t,t,m}=\left(1+\frac{\mathcal{CN}(0;g^{pri}_{\mathcal{B},t,m},\upsilon^{pri}_{\mathcal{B},t})}{\mathcal{CN}(0;g^{pri}_{\mathcal{B},t,m}-\mu^{out}_{t,m},\upsilon^{pri}_{\mathcal{B},t}+(\sigma^{out}_{t,m})^2)}\right)^{-1}$.The message $\upsilon_{b_{t,m}\to u_{t,m}}$ from variable node $b_{t,m}$ to factor node $u_{t,m}$ is the same as $\upsilon_{\pi_{t,m}\to b_{t,m}}\left(b_{t,m}\right)$. The message from factor node $u_{t,m}$ to variable node a_{t,n_x} is

$$\upsilon_{u_{t,m}\to a_{t,n_x}}\left(a_{t,n_x}\right)=\rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y}\delta\left(a_{t,n_x}-1\right)+\left(1-\rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y}\right)\delta\left(a_{t,n_x}\right), \tag{B.4}$$

where

$$\rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y}=\frac{\rho_{\boldsymbol{a}_t}}{1+\rho_{\boldsymbol{a}_t}-\rho^{in}_{\boldsymbol{b}_t,t,m}}, \tag{B.5}$$

with

$$\begin{aligned}\rho_{\boldsymbol{a}_t}=\gamma_{\boldsymbol{b}_t}\rho^{in}_{\boldsymbol{b}_t,t,m}\rho^{out}_{\boldsymbol{e}_t,t,n_x,n_y}+(1-\gamma_{\boldsymbol{b}_t})(1-\rho^{in}_{\boldsymbol{b}_t,t,m})\rho^{out}_{\boldsymbol{e}_t,t,n_x,n_y}+\\(1-\rho^{in}_{\boldsymbol{b}_t,t,m})(1-\rho^{out}_{\boldsymbol{e}_t,t,n_x,n_y})\end{aligned} \tag{B.6}$$

and $\rho^{out}_{\boldsymbol{e}_t,t,n_x,n_y}=\upsilon_{e_{t,n_y}\to u_{t,m}}(e_{t,n_y}=1)$.

3. **Message Passing Over the 2D-MM of the Azimuth Support $\boldsymbol{a}_t$:** We firstly introduce the details of the azimuth support estimation. The condition can be divided into two section, one is when $t = 1$, the other is when $t > 1$. To be notice, the message from factor node $u_{t,m}$ to variable node a_{t,n_x} is

$$\upsilon_{u_{t,m}\to a_{t,n_x}} = \rho^{(A)in}_{t,n_x} = \frac{\prod_{n_y} \rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y}}{\prod_{n_y} \rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y} + \prod_{n_y}\left(1-\rho^{in}_{\boldsymbol{a}_t,t,n_x,n_y}\right)}, \tag{B.7}$$

which is deriviated based on the the matched dimension M from the total gain, n_x from the azimuth support and n_y from the elevation support.

When $t = 1$, the azimuth forward parameter is

$$\gamma^{(A)f}_{t,n_x} = \frac{q^{(AS)}_{0,1}\left(1-\rho^{(A)in}_{t,n_x-1}\right)\left(1-\gamma^{(A)f}_{t,n_x-1}\right) + q^{(AS)}_{1,1}\rho^{(A)in}_{t,n_x-1}\gamma^{(A)f}_{t,n_x-1}}{\left(1-\rho^{(A)in}_{t,n_x-1}\right)\left(1-\gamma^{(A)f}_{t,n_x-1}\right) + \rho^{(A)in}_{t,n_x-1}\gamma^{(A)f}_{t,n_x-1}}. \tag{B.8}$$

The azimuth backward parameter $\gamma^{(A)b}_{t,n_x}$ can be written as

$$\gamma^{(A)b}_{t,n_x} = \frac{q^{(AS)}_{1,0}\left(1-\rho^{(A)in}_{t,n_x+1}\right)\left(1-\gamma^{(A)b}_{t,n_x+1}\right) + \left(1-q^{(S)}_{1,0}\right)\rho^{(A)in}_{t,n_x+1}\gamma^{(A)b}_{t,n_x+1}}{\left(q^{(AS)}_{0,0}+q^{(S)}_{1,0}\right)\left(1-\rho^{(A)in}_{t,n_x+1}\right)\left(1-\gamma^{(A)b}_{t,n_x+1}\right) + \left(q^{(AS)}_{1,1}+q^{(AS)}_{0,1}\right)\rho^{(A)in}_{t,n_x+1}\gamma^{(A)b}_{t,n_x+1}}. \tag{B.9}$$

When $t > 1$, the azimuth forward parameter $\gamma^{(A)f}_{t,n_x}$ is

$$\begin{aligned}\gamma^{(A)f}_{t,n_x} = &\frac{q^{(A)}_{1,1,1}\rho^{dyna}_{t,n_x} + q^{(A)}_{1,0,1}\left(1-\rho^{(A)dyna}_{t,n_x}\right)}{1+\left(\left(\rho^{(A)in}_{t,n_x-1}\right)^{-1}-1\right)\left(\left(\gamma^{(A)f}_{t,n_x-1}\right)^{-1}-1\right)} + \\ &\frac{q^{(A)}_{0,1,1}\rho^{(A)dyna}_{t,n_x} + q^{(A)}_{0,0,1}\left(1-\rho^{(A)dyna}_{t,n_x}\right)}{\left(\left(\rho^{(A)in}_{t,n_x-1}\right)^{-1}-1\right)^{-1}\left(\left(\left(\gamma^{(A)f}_{t,n_x-1}\right)^{-1}-1\right)\right)^{-1}+1}.\end{aligned} \tag{B.10}$$

The azimuth backward parameter $\gamma_{t,n_x}^{(A)b} = \frac{1}{1+\gamma^{(A)}}$ is given as

$$\gamma^{(A)} = \frac{\left(1-\rho_{t,n_x+1}^{(A)in}-\gamma_{t,n_x+1}^{(A)b}\right)\left[q_{0,1,0}^{(A)}\rho_{t,n_x+1}^{(A)dyna}+q_{0,0,0}^{(A)}\left(1-\rho_{t,n_x+1}^{(A)dyna}\right)\right]+\rho_{t,n_x+1}^{(A)in}\gamma_{t,n_x+1}^{(A)b}}{\left(1-\rho_{t,n_x+1}^{(A)in}-\gamma_{t,n_x+1}^{(A)b}\right)\left[q_{1,1,0}^{(A)}\rho_{t,n_x+1}^{(A)dyna}+q_{1,0,0}^{(A)}\left(1-\rho_{t,n_x+1}^{(A)dyna}\right)\right]+\rho_{t,n_x+1}^{(A)in}\gamma_{t,n_x+1}^{(A)b}}. \tag{B.11}$$

The final azimuth message which represents the nonzero probability can be written as

$$\upsilon_{a_{t,n_x}\rightarrow u_{t,m}} = \rho_{t,n_x}^{(A)out} = \frac{\gamma_{t,n_x}^{(A)f}\gamma_{t,n_x}^{(A)b}}{\gamma_{t,n_x}^{(A)f}\gamma_{t,n_x}^{(A)b}+\left(1-\gamma_{t,n_x}^{(A)f}\right)\left(1-\gamma_{t,n_x}^{(A)b}\right)}. \tag{B.12}$$

4. **Message Passing Over the Path $a_{t,n_x} \rightarrow u_{t,m} \rightarrow e_{t,n_y}$:** The message from factor node a_{t,n_x} to variable node $u_{t,m}$ is

$$\upsilon_{a_{t,n_x}\rightarrow u_{t,m}}(a_{t,n_x}) = \rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out}\delta(a_{t,n_x}-1)+(1-\rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out})\delta(a_{t,n_x}), \tag{B.13}$$

where

$$\rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out} = \frac{\rho_{t,n_x}^{(A)out}}{\rho_{t,n_x}^{(A)out}+\left(1-\rho_{t,n_x}^{(A)out}\right)\prod_{n_y'\neq n_y}\left(\left(\rho_{\boldsymbol{a}_t,t,n_x,n_y'}^{in}\right)^{-1}-1\right)}. \tag{B.14}$$

To be noticed, $\rho_{t,n_x}^{(A)out}$ indicates the message passing over the n_x^{th} azimuth support element and $\rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out}$ indicates the matching relationship between n_x^{th} azimuth support element and n_y^{th} elevation support element. This condition is similar for elevation support message passing equations. The message from factor node $u_{t,m}$ to variable node e_{t,n_y} is

$$\upsilon_{u_{t,m}\rightarrow e_{t,n_y}}(e_{t,n_y}) = \rho_{\boldsymbol{e}_t,t,n_x,n_y}^{in}\delta\left(e_{t,n_y}-1\right)+\left(1-\rho_{\boldsymbol{e},t,n_x,n_y}^{in}\right)\delta\left(e_{t,n_y}\right), \tag{B.15}$$

where $\rho^{in}_{\boldsymbol{e}_t,t,n_x,n_y}$ can be written as

$$\rho^{in}_{\boldsymbol{e}_t,t,n_x,n_y} = \frac{\gamma_{\boldsymbol{b}_t}\rho^{in}_{\boldsymbol{b}_t,t,m}\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y}+(1-\gamma_{\boldsymbol{b}_t})(1-\rho^{in}_{\boldsymbol{b}_t,t,m})\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y}+(1-\rho^{in}_{\boldsymbol{b}_t,t,m})(1-\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y})}{\gamma_{\boldsymbol{b}_t}\rho^{in}_{\boldsymbol{b}_t,t,m}\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y}+(1-\gamma_{\boldsymbol{b}_t})(1-\rho^{in}_{\boldsymbol{b}_t,t,m})\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y}+(1-\rho^{in}_{\boldsymbol{b}_t,t,m})(1-\rho^{out}_{\boldsymbol{a}_t,t,n_x,n_y})+1-\rho^{in}_{\boldsymbol{b}_t,t,m}}. \tag{B.16}$$

5. **Message Passing Over the 2D-MM of the Elevation Support $\boldsymbol{e}_t$:** The details can be derived similarly as the message passing process of the azimuth support $\mathbf{a}_t$ estimation. The message passing from factor node $u_{t,m}$ to variable node e_{t,n_y} is

$$\upsilon_{u_{t,m}\rightarrow e_{t,n_y}} = \rho^{(E)in}_{t,n_y} = \frac{\prod_{n_x}\rho^{in}_{\boldsymbol{e}_t,t,n_x,n_y}}{\prod_{n_x}\rho^{in}_{\boldsymbol{e}_t,t,n_x,n_y}+\prod_{n_x}\left(1-\rho^{in}_{\boldsymbol{e}_t,t,n_x,n_y}\right)}. \tag{B.17}$$

When $t=1$, the elevation forward parameter is

$$\gamma^{(E)f}_{t,n_y} = \frac{q^{(ES)}_{0,1}\left(1-\rho^{(E)in}_{t,n_y-1}\right)\left(1-\gamma^{(E)f}_{t,n_y-1}\right)+q^{(ES)}_{1,1}\rho^{(E)in}_{t,n_y-1}\gamma^{(E)f}_{t,n_y-1}}{\left(1-\rho^{(E)in}_{t,n_y-1}\right)\left(1-\gamma^{(E)f}_{t,n_y-1}\right)+\rho^{(E)in}_{t,n_y-1}\gamma^{(E)f}_{t,n_y-1}}. \tag{B.18}$$

The elevation backward parameter $\gamma^{(E)b}_{t,n_y}$ can be found in (B.19). When $t>1$, the

$$\gamma^{(E)b}_{t,n_y} = \frac{q^{(ES)}_{1,0}\left(1-\rho^{(E)in}_{t,n_y+1}\right)\left(1-\gamma^{(E)b}_{t,n_y+1}\right)+\left(1-q^{(ES)}_{1,0}\right)\rho^{(E)in}_{t,n_y+1}\gamma^{(E)b}_{t,n_y+1}}{\left(q^{(ES)}_{0,0}+q^{(ES)}_{1,0}\right)\left(1-\rho^{(E)in}_{t,n_y+1}\right)\left(1-\gamma^{(E)b}_{t,n_y+1}\right)+\left(q^{(ES)}_{1,1}+q^{(ES)}_{0,1}\right)\rho^{(E)in}_{t,n_y+1}\gamma^{(E)b}_{t,n_y+1}} \tag{B.19}$$

elevation forward parameter $\gamma^{(E)f}_{t,n_y}$ is given in (B.20). And the elevation backward

$$\begin{aligned}\gamma^{(E)f}_{t,n_y} = &\frac{q^{(E)}_{1,1,1}\rho^{(E)dyna}_{t,n_y}+q^{(E)}_{1,0,1}\left(1-\rho^{(E)dyna}_{t,n_x}\right)}{1+\left(\left(\rho^{(E)in}_{t,n_y-1}\right)^{-1}-1\right)\left(\left(\gamma^{(E)f}_{t,n_y-1}\right)^{-1}-1\right)}+\\ &\frac{q^{(E)}_{0,1,1}\rho^{(E)dyna}_{t,n_y}+q^{(E)}_{0,0,1}\left(1-\rho^{(E)dyna}_{t,n_y}\right)}{\left(\left(\rho^{(E)in}_{t,n_y-1}\right)^{-1}-1\right)^{-1}\left(\left(\left(\gamma^{(E)f}_{t,n_y-1}\right)^{-1}-1\right)\right)^{-1}+1}\end{aligned} \tag{B.20}$$

parameter $\gamma_{t,n_y}^{(E)b} = \frac{1}{1+\gamma^{(E)}}$ with $\gamma^{(E)}$ given as

$$\gamma^{(E)} = \frac{\left(1-\rho_{t,n_y+1}^{(E)in}-\gamma_{t,n_y+1}^{(E)b}\right)\left[q_{0,1,0}^{(E)}\rho_{t,n_y+1}^{(E)dyna}+q_{0,0,0}^{(E)}\left(1-\rho_{t,n_y+1}^{(E)dyna}\right)\right]+\rho_{t,n_y+1}^{(E)in}\gamma_{t,n_y+1}^{(E)b}}{\left(1-\rho_{t,n_y+1}^{(E)in}-\gamma_{t,n_y+1}^{(E)b}\right)\left[q_{1,1,0}^{(E)}\rho_{t,n_y+1}^{(E)dyna}+q_{1,0,0}^{(E)}\left(1-\rho_{t,n_y+1}^{(E)dyna}\right)\right]+\rho_{t,n_y+1}^{(E)in}\gamma_{t,n_y+1}^{(E)b}}. \tag{B.21}$$

The final elevation nonzero probability output can be written as

$$\upsilon_{e_{t,n_x}\rightarrow u_{t,m}} = \rho_{t,n_y}^{(E)out} = \frac{\gamma_{t,n_y}^{(E)f}\gamma_{t,n_y}^{(E)b}}{\gamma_{t,n_y}^{(E)f}\gamma_{t,n_y}^{(E)b}+\left(1-\gamma_{t,n_y}^{(E)f}\right)\left(1-\gamma_{t,n_y}^{(E)b}\right)}. \tag{B.22}$$

6. **Message Passing Over the Path** $e_{t,n_y} \rightarrow u_{t,m} \rightarrow b_{t,m} \rightarrow \pi_{t,m} \rightarrow g_{t,m}$**:** The message from factor node e_{t,n_y} to variable node $u_{t,m}$ is

$$\upsilon_{e_{t,n_y}\rightarrow u_{t,m}}(e_{t,n_y}) = \rho_{\boldsymbol{e}_t,t,n_x,n_y}^{out}\delta\left(e_{t,n_y}-1\right)+\left(1-\rho_{\boldsymbol{e}_t,t,n_x,n_y}^{out}\right)\delta\left(e_{t,n_y}\right), \tag{B.23}$$

where

$$\rho_{\boldsymbol{e}_t,t,n_x,n_y}^{out} = \frac{\rho_{t,n_y}^{(E)out}}{\rho_{t,n_y}^{(E)out}+\left(1-\rho_{t,n_y}^{(E)out}\right)\prod_{n_x'\neq n_x}\left(\left(\rho_{\boldsymbol{e}_t,t,n_x',n_y}^{in}\right)^{-1}-1\right)}. \tag{B.24}$$

The message from factor node $u_{t,m}$ to variable node $b_{t,m}$ is

$$\upsilon_{u_{t,m}\rightarrow b_{t,m}}\left(b_{t,m}\right) = \rho_{\boldsymbol{b}_t,t,m}^{out}\delta(b_{t,m}-1)+\left(1-\rho_{\boldsymbol{b}_t,t,m}^{out}\right)\delta\left(b_{t,m}\right), \tag{B.25}$$

where $\rho_{\boldsymbol{b}_t,t,m}^{out} = \rho_{\boldsymbol{a}_t,t,n_x,n_y}^{out}\rho_{\boldsymbol{e}_t,t,n_x,n_y}^{out}\gamma_{\boldsymbol{b}_t}$. The message $\upsilon_{b_{t,m}\rightarrow\pi_{t,m}}$ from variable node $b_{t,m}$ to factor node $\pi_{t,m}$ is the same as $\upsilon_{u_{t,m}\rightarrow b_{t,m}}(b_{t,m})$. The message from factor node $\pi_{t,m}$ back to variable node $g_{t,m}$ is

$$\upsilon_{\pi_{t,m}\rightarrow g_{t,m}}\left(g_{t,m}\right) = \rho_{\boldsymbol{b}_t,t,m}^{out}\mathcal{CN}\left(g_{t,m};\mu_{t,m}^{out},(\sigma_{t,m}^{out})^2\right)+\left(1-\rho_{\boldsymbol{b}_t,t,m}^{out}\right)\delta\left(g_{t,m}\right). \tag{B.26}$$

Appendix C

MD-MM Message Passing in Module $\mathcal{B}$

1. **Message Passing Over Gaussian Hidden Value Vector $\boldsymbol{\vartheta}_{k,t}$** The message $\upsilon_{f_{k,t,m}\to\vartheta_{k,t,m}}$ passed from the last time slot can be formed as $\mathcal{CN}(\vartheta_{k,t,m}; \mu^{dyna}_{k,t,m}, (\sigma^{dyna}_{k,t,m})^2)$. The message passed from variable node $\vartheta_{k,t,m}$ to factor node $\pi_{k,t,m}$ can be formed as

$$\upsilon_{\vartheta_{k,t,m}\to\pi_{k,t,m}} = \mathcal{CN}(\vartheta_{k,t,m}; \mu^{out}_{k,t,m}, (\sigma^{out}_{k,t,m})^2), \tag{C.1}$$

with $(\mu^{out}_{k,t,m}, (\sigma^{out}_{k,t,m})^2) = (\mu^{dyna}_{k,t,m}, (\sigma^{dyna}_{k,t,m})^2)$. When $t = 1$, we set $(\mu^{dyna}_{1,m}, (\sigma^{dyna}_{1,m})^2) = (\mu, \sigma^2)$

2. **Message Passing Over** $g_{k,t,m} \to \pi_{k,t,m} \to b_{k,t,m} \to u_{k,t,m} \to a_{k,t,n_x}$ The message from variable node $g_{k,t,m}$ to factor node $\pi_{k,t,m}$ is

$$\upsilon_{g_{k,t,m}\to\pi_{k,t,m}}(g_{k,t,m}) = \mathcal{CN}\left(g_{k,t,m}; g^{pri}_{\mathcal{B},k,t,m}, v^{pri}_{\mathcal{B},k,t,m}\right). \tag{C.2}$$

The message from factor node $\pi_{k,t,m}$ to variable node $b_{k,t,m}$ is

$$\upsilon_{\pi_{k,t,m}\to b_{k,t,m}}\left(b_{k,t,m}\right)=\rho^{in}_{b,k,t,m}\delta\left(b_{k,t,m}-1\right)+\left(1-\rho^{in}_{b,k,t,m}\right)\delta\left(b_{k,t,m}\right), \tag{C.3}$$

where

$$\rho^{in}_{\boldsymbol{b}_{k,t},k,t,m}=\left(1+\frac{\mathcal{CN}(0;g^{pri}_{\mathcal{B},k,t,m},v^{pri}_{\mathcal{B},k,t})}{\mathcal{CN}(0;g^{pri}_{\mathcal{B},k,t,m}-\mu^{out}_{k,t,m},v^{pri}_{\mathcal{B},k,t}+(\sigma^{out}_{k,t,m})^2)}\right)^{-1}. \tag{C.4}$$

The message $\upsilon_{b_{k,t,m}\to u_{k,t,m}}$ from variable node $b_{k,t,m}$ to factor node $u_{k,t,m}$ is the same as $\upsilon_{\pi_{k,t,m}\to b_{k,t,m}}\left(b_{k,t,m}\right)$. The message from factor node $u_{k,t,m}$ to variable node a_{k,t,n_x} is

$$\begin{aligned}&\upsilon_{u_{k,t,m}\to a_{k,t,n_x}}\left(a_{k,t,n_x}\right)=\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}\delta\left(a_{k,t,n_x}-1\right)+\\&\left(1-\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}\right)\delta\left(a_{k,t,n_x}\right),\end{aligned} \tag{C.5}$$

where

$$\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}=\frac{\rho_{\boldsymbol{a}_{k,t}}}{1+\rho_{\boldsymbol{a}_{k,t}}-\rho^{in}_{\boldsymbol{b}_{k,t},k,t,m}}, \tag{C.6}$$

with

$$\begin{aligned}&\rho_{\boldsymbol{a}_{k,t}}=\gamma_{\boldsymbol{b}_{k,t}}\rho^{in}_{\boldsymbol{b}_{k,t},k,t,m}\rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}+\left(1-\gamma_{\boldsymbol{b}_{k,t}}\right)(1-\\&\rho^{in}_{\boldsymbol{b}_{k,t},k,t,m})\rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}+\left(1-\rho^{in}_{\boldsymbol{b}_{k,t},k,t,m}\right)\left(1-\rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}\right)\end{aligned} \tag{C.7}$$

and $\rho^{out}_{\boldsymbol{e}_t,k,t,n_x,n_y}=\upsilon_{e_{k,t,n_y}\to u_{k,t,m}}(e_{k,t,n_y}=1)$.

3. **Message Passing Over $a_{k,t,n_x}\to\varrho_{k,t,n_x}\to\tilde{a}_{t,n_x}$, the 2D-MM of the MU Azimuth Support $\tilde{\boldsymbol{a}}_{t,n_x}$, and $\tilde{a}_{t,n_x}\to\varrho_{k,t,n_x}\to a_{k,t,n_x}$**

The forward-backward message passing is performed over the HMM $\boldsymbol{a}_{k,t}$ for given input messages $\upsilon_{u_{k,t,n_x,n_y}\to a_{k,t,n_x}}$.

We firstly introduce the details of the MU azimuth support estimation. The condition can be divided into two section, one is when $t = 1$, the other is when $t > 1$. To be notice, the message from factor node $u_{t,m}$ to variable node a_{t,n_x} is

$$\begin{aligned}&\upsilon_{u_{k,t,m}\to a_{k,t,n_x}} = \rho^{(A)in}_{k,t,n_x} = \\ &\frac{\prod_{n_y}\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}}{\prod_{n_y}\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}+\prod_{n_y}\left(1-\rho^{in}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}\right)},\end{aligned} \tag{C.8}$$

which is deriviated based on the the matched dimension M from the total gain, n_x from the azimuth support and n_y from the elevation support.

The message from factor node ϱ_{k,t,n_x} to variable node $\tilde{a}_{t,n_x}$ is

$$\begin{aligned}&\upsilon_{\varrho_{k,t,n_x}\to\tilde{a}_{t,n_x}} \\ &\propto \sum_{a_{k,t,n_x}}\prod_{n_x}\upsilon_{u_{k,t,m}\to a_{k,t,n_x}}(a_{k,t,n_x})\varsigma^{(A)}_{\tilde{a}_{t,n_x},a_{k,t,n_x}} \\ &= \rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x}\delta(\tilde{a}_{t,n_x}-1)+(1-\rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x})\tilde{a}_{t,n_x},\end{aligned} \tag{C.9}$$

where

$$\begin{aligned}&\rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x} = \\ &\frac{\varsigma^{(A)}_{k,t,1,1}\rho^{(A)in}_{k,t,n_x}+(1-\varsigma^{(A)}_{k,t,1,1})(1-\rho^{(A)in}_{k,t,n_x})}{\varsigma^{(A)}_{k,t,1,1}\rho^{(A)in}_{k,t,n_x}+(1-\varsigma^{(A)}_{k,t,1,1})(1-\rho^{(A)in}_{k,t,n_x})+(1-\rho^{(A)in}_{k,t,n_x})}.\end{aligned} \tag{C.10}$$

When $t = 1$, the MU azimuth forward parameter is

$$\tilde{\gamma}^{(A)f}_{t,n_x} = \frac{q^{(AS)}_{0,1}\left(1-\tilde{\rho}^{(A)in}_{t,n_x-1}\right)\left(1-\tilde{\gamma}^{(A)f}_{t,n_x-1}\right)+q^{(AS)}_{1,1}\tilde{\rho}^{(A)in}_{t,n_x-1}\tilde{\gamma}^{(A)f}_{t,n_x-1}}{\left(1-\tilde{\rho}^{(A)in}_{t,n_x-1}\right)\left(1-\tilde{\gamma}^{(A)f}_{t,n_x-1}\right)+\tilde{\rho}^{(A)in}_{t,n_x-1}\tilde{\gamma}^{(A)f}_{t,n_x-1}}, \tag{C.11}$$

where $\tilde{\rho}^{(A)in}_{t,n_x} = \frac{\prod_k\rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x}}{\prod_k\rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x}+\prod_k(1-\rho^{in}_{\tilde{\boldsymbol{a}}_t,k,n_x})}$. The azimuth backward parameter $\tilde{\gamma}^{(A)b}_{t,n_x}$

can be found as

$$\tilde{\gamma}_{t,n_x}^{(A)b} = \frac{q_{1,0}^{(AS)}\left(1-\tilde{\rho}_{t,n_x+1}^{(A)in}\right)\left(1-\tilde{\gamma}_{t,n_x+1}^{(A)b}\right)+\left(1-q_{1,0}^{(S)}\right)\tilde{\rho}_{t,n_x+1}^{(A)in}\tilde{\gamma}_{t,n_x+1}^{(A)b}}{\left(q_{0,0}^{(AS)}+q_{1,0}^{(S)}\right)\left(1-\tilde{\rho}_{t,n_x+1}^{(A)in}\right)\left(1-\tilde{\gamma}_{t,n_x+1}^{(A)b}\right)+\left(q_{1,1}^{(AS)}+q_{0,1}^{(AS)}\right)\tilde{\rho}_{t,n_x+1}^{(A)in}\tilde{\gamma}_{t,n_x+1}^{(A)b}}. \tag{C.12}$$

When $t>1$, the azimuth forward parameter $\tilde{\gamma}_{t,n_x}^{(A)f}$ can be found as

$$\begin{aligned}\tilde{\gamma}_{t,n_x}^{(A)f} = & \frac{q_{1,1,1}^{(A)}\tilde{\rho}_{t,n_x}^{dyna}+q_{1,0,1}^{(A)}\left(1-\tilde{\rho}_{t,n_x}^{(A)dyna}\right)}{1+\left(\left(\tilde{\rho}_{t,n_x-1}^{(A)in}\right)^{-1}-1\right)\left(\left(\tilde{\gamma}_{t,n_x-1}^{(A)f}\right)^{-1}-1\right)}+ \\ & \frac{q_{0,1,1}^{(A)}\tilde{\rho}_{t,n_x}^{(A)dyna}+q_{0,0,1}^{(A)}\left(1-\tilde{\rho}_{t,n_x}^{(A)dyna}\right)}{\left(\left(\tilde{\rho}_{t,n_x-1}^{(A)in}\right)^{-1}-1\right)^{-1}\left(\left(\left(\tilde{\gamma}_{t,n_x-1}^{(A)f}\right)^{-1}-1\right)\right)^{-1}+1}.\end{aligned} \tag{C.13}$$

The azimuth backward parameter $\tilde{\gamma}_{t,n_x}^{(A)b}=\frac{1}{1+\tilde{\gamma}^{(A)}}$ is given as

$$\tilde{\gamma}^{(A)} = \frac{\left(1-\tilde{\rho}_{t,n_x+1}^{(A)in}-\tilde{\gamma}_{t,n_x+1}^{(A)b}\right)\left[q_{0,1,0}^{(A)}\tilde{\rho}_{t,n_x+1}^{(A)dyna}+q_{0,0,0}^{(A)}\left(1-\tilde{\rho}_{t,n_x+1}^{(A)dyna}\right)\right]+\tilde{\rho}_{t,n_x+1}^{(A)in}\tilde{\gamma}_{t,n_x+1}^{(A)b}}{\left(1-\tilde{\rho}_{t,n_x+1}^{(A)in}-\tilde{\gamma}_{t,n_x+1}^{(A)b}\right)\left[q_{1,1,0}^{(A)}\tilde{\rho}_{t,n_x+1}^{(A)dyna}+q_{1,0,0}^{(A)}\left(1-\tilde{\rho}_{t,n_x+1}^{(A)dyna}\right)\right]+\tilde{\rho}_{t,n_x+1}^{(A)in}\tilde{\gamma}_{t,n_x+1}^{(A)b}}. \tag{C.14}$$

The final MU azimuth message which represents the non-zero probability can be written as

$$\upsilon_{\tilde{a}_{t,n_x}\to\varrho_{k,t,n_x}} = \tilde{\rho}_{\tilde{\mathbf{a}}_{t,k,n_x}}^{out}\delta(\tilde{a}_{t,n_x}-1)+(1-\tilde{\rho}_{\tilde{\mathbf{a}}_{t,k,n_x}}^{out})\delta(\tilde{a}_{t,n_x}), \tag{C.15}$$

where $\tilde{\rho}_{\tilde{\mathbf{a}}_{t,k,n_x}}^{out}$ can be written as

$$\tilde{\rho}_{\tilde{\mathbf{a}}_{t,k,n_x}}^{out} = \frac{\tilde{\gamma}_{t,n_x}^{(A)f}\tilde{\gamma}_{t,n_x}^{(A)b}}{\tilde{\gamma}_{t,n_x}^{(A)f}\tilde{\gamma}_{t,n_x}^{(A)b}+\prod_{k'\neq k}\left((\rho_{\tilde{\mathbf{a}}_{t,k',n_x}}^{in})^{-1}-1\right)\left(1-\tilde{\gamma}_{t,n_x}^{(A)f}\right)\left(1-\tilde{\gamma}_{t,n_x}^{(A)b}\right)}. \tag{C.16}$$

The message from factor node ϱ_{k,t,n_x} to variable node a_{k,t,n_x} is

$$\upsilon_{\varrho_{k,t,n_x}\rightarrow a_{k,t,n_x}}(a_{k,t,n_x}) = \rho_{k,t,n_x}^{(A)out}\delta(a_{k,t,n_x}-1)+(1-\rho_{k,t,n_x}^{(A)out})\delta(a_{k,t,n_x}), \tag{C.17}$$

where $\rho_{k,t,n_x}^{(A)out} = \tilde{\rho}_{\tilde{\boldsymbol{a}}_t,k,n_x}^{out}\varsigma_{k,t,1,1}^{(A)}$.

4. **Message Passing Over the Path** $a_{k,t,n_x} \rightarrow u_{k,t,m} \rightarrow e_{k,t,n_y}$ The message from factor node a_{k,t,n_x} to variable node $u_{k,t,m}$ is

$$\upsilon_{a_{k,t,n_x}\rightarrow u_{k,t,m}}(a_{k,t,n_x}) = \rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}\delta(a_{k,t,n_x}-1)+(1-\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out})\delta(a_{k,t,n_x}), \tag{C.18}$$

where $\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}$ can be formulated as

$$\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out} = \frac{\rho_{k,t,n_x}^{(A)out}}{\rho_{k,t,n_x}^{(A)out}+\left(1-\rho_{k,t,n_x}^{(A)out}\right)\prod_{n_y'\neq n_y}\left(\left(\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y'}^{in}\right)^{-1}-1\right)}. \tag{C.19}$$

The message from factor node $u_{k,t,m}$ to variable node e_{k,t,n_y} is

$$\upsilon_{u_{k,t,m}\rightarrow e_{k,t,n_y}}(e_{k,t,n_y}) = \rho_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}^{in}\delta\left(e_{k,t,n_y}-1\right)+\left(1-\rho_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}^{in}\right)\delta\left(e_{k,t,n_y}\right), \tag{C.20}$$

where $\rho_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}^{in}$ can be written as

$$\rho_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}^{in} = \frac{\genfrac{}{}{0pt}{}{\gamma_{\boldsymbol{b}_{k,t}}\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in}\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}+(1-\gamma_{\boldsymbol{b}_{k,t}})}{(1-\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in})\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}+(1-\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in})(1-\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out})}}{\genfrac{}{}{0pt}{}{\gamma_{\boldsymbol{b}_{k,t}}\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in}\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}+(1-\gamma_{\boldsymbol{b}_{k,t}})(1-\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in})}{\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out}+(1-\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in})(1-\rho_{\boldsymbol{a}_{k,t},k,t,n_x,n_y}^{out})+1-\rho_{\boldsymbol{b}_{k,t},k,t,m}^{in}}}. \tag{C.21}$$

5. **Message Passing Over** $e_{k,t,n_y} \rightarrow \varpi_{k,t,n_y} \rightarrow \tilde{e}_{t,n_y}$ **, the 2D-MM of the MU Elevation Support** $\tilde{\boldsymbol{e}}_{t,n_y}$**, and** $\tilde{e}_{t,n_y} \rightarrow \varpi_{k,t,n_y} \rightarrow e_{k,t,n_y}$ The details can be derived similarly as the message passing process of the MU azimuth support $\tilde{\boldsymbol{a}}_t$ estimation.

The message from factor node $u_{k,t,m}$ to variable node e_{k,t,n_y} is

$$\upsilon_{u_{k,t,m}\to e_{k,t,n_y}} = \rho^{(E)in}_{k,t,n_y} = \frac{\prod_{n_x} \rho^{in}_{\mathbf{e}_{k,t},k,t,n_x,n_y}}{\prod_{n_x} \rho^{in}_{\mathbf{e}_{k,t},k,t,n_x,n_y} + \prod_{n_x}\left(1-\rho^{in}_{\mathbf{e}_{k,t},k,t,n_x,n_y}\right)}, \tag{C.22}$$

which is deriviated based on the the matched dimension M from the total gain, n_x from the azimuth support and n_y from the elevation support.

The message from factor node ϖ_{k,t,n_y} to variable node $\tilde{e}_{t,n_y}$ is

$$\begin{aligned} \upsilon_{\varpi_{k,t,n_y}\to\tilde{e}_{t,n_y}} &\propto \sum_{e_{k,t,n_y}} \prod_{n_y} \upsilon_{u_{k,t,m}\to e_{k,t,n_y}}(e_{k,t,n_y})\varsigma^{(E)}_{\tilde{e}_{t,n_y},e_{k,t,n_y}} \\ &= \rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y}\delta(\tilde{e}_{t,n_y}-1) + (1-\rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y})\tilde{e}_{t,n_y}, \end{aligned} \tag{C.23}$$

where

$$\rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y} = \frac{\varsigma^{(E)}_{k,t,1,1}\rho^{(E)in}_{k,t,n_y} + (1-\varsigma^{(E)}_{k,t,1,1})(1-\rho^{(E)in}_{k,t,n_y})}{\varsigma^{(E)}_{k,t,1,1}\rho^{(E)in}_{k,t,n_y} + (1-\varsigma^{(E)}_{k,t,1,1})(1-\rho^{(E)in}_{k,t,n_y}) + (1-\rho^{(E)in}_{k,t,n_y})}. \tag{C.24}$$

When $t=1$, the MU elevation forward parameter is

$$\tilde{\gamma}^{(E)f}_{t,n_y} = \frac{q^{(ES)}_{0,1}\left(1-\tilde{\rho}^{(E)in}_{t,n_y-1}\right)\left(1-\tilde{\gamma}^{(E)f}_{t,n_y-1}\right) + q^{(ES)}_{1,1}\tilde{\rho}^{(E)in}_{t,n_y-1}\tilde{\gamma}^{(E)f}_{t,n_y-1}}{\left(1-\tilde{\rho}^{(E)in}_{t,n_y-1}\right)\left(1-\tilde{\gamma}^{(E)f}_{t,n_y-1}\right) + \tilde{\rho}^{(E)in}_{t,n_y-1}\tilde{\gamma}^{(E)f}_{t,n_y-1}}, \tag{C.25}$$

where $\tilde{\rho}^{(E)in}_{t,n_y} = \frac{\prod_k \rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y}}{\prod_k \rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y} + \prod_k (1-\rho^{in}_{\tilde{\mathbf{e}}_t,k,n_y})}$. The MU elevation backward parameter $\tilde{\gamma}^{(E)b}_{t,n_y}$ can be found as

$$\tilde{\gamma}^{(E)b}_{t,n_y} = \frac{q^{(ES)}_{1,0}\left(1-\tilde{\rho}^{(E)in}_{t,n_y+1}\right)\left(1-\tilde{\gamma}^{(E)b}_{t,n_y+1}\right) + \left(1-q^{(S)}_{1,0}\right)\tilde{\rho}^{(E)in}_{t,n_y+1}\tilde{\gamma}^{(E)b}_{t,n_y+1}}{\left(q^{(ES)}_{0,0}+q^{(S)}_{1,0}\right)\left(1-\tilde{\rho}^{(E)in}_{t,n_y+1}\right)\left(1-\tilde{\gamma}^{(E)b}_{t,n_y+1}\right) + \left(q^{(ES)}_{1,1}+q^{(ES)}_{0,1}\right)\tilde{\rho}^{(E)in}_{t,n_y+1}\tilde{\gamma}^{(E)b}_{t,n_y+1}}. \tag{C.26}$$

When $t > 1$, the azimuth forward parameter $\tilde{\gamma}^{(E)f}_{t,n_x}$ can be found as

$$\begin{aligned}\tilde{\gamma}^{(E)f}_{t,n_y} = & \frac{q^{(E)}_{1,1,1}\tilde{\rho}^{dyna}_{t,n_y} + q^{(E)}_{1,0,1}\left(1 - \tilde{\rho}^{(E)dyna}_{t,n_y}\right)}{1 + \left(\left(\tilde{\rho}^{(E)in}_{t,n_y-1}\right)^{-1} - 1\right)\left(\left(\tilde{\gamma}^{(E)f}_{t,n_y-1}\right)^{-1} - 1\right)} + \\ & \frac{q^{(E)}_{0,1,1}\tilde{\rho}^{(E)dyna}_{t,n_y} + q^{(E)}_{0,0,1}\left(1 - \tilde{\rho}^{(E)dyna}_{t,n_y}\right)}{\left(\left(\tilde{\rho}^{(E)in}_{t,n_y-1}\right)^{-1} - 1\right)^{-1}\left(\left(\left(\tilde{\gamma}^{(E)f}_{t,n_y-1}\right)^{-1} - 1\right)\right)^{-1} + 1}.\end{aligned} \tag{C.27}$$

The azimuth backward parameter $\tilde{\gamma}^{(E)b}_{t,n_y} = \frac{1}{1+\tilde{\gamma}^{(E)}}$ is given as

$$\tilde{\gamma}^{(E)} = \frac{\left(1-\tilde{\rho}^{(E)in}_{t,n_y+1}-\tilde{\gamma}^{(E)b}_{t,n_y+1}\right)\left[q^{(E)}_{0,1,0}\tilde{\rho}^{(E)dyna}_{t,n_y+1}+q^{(E)}_{0,0,0}\left(1-\tilde{\rho}^{(E)dyna}_{t,n_y+1}\right)\right]+\tilde{\rho}^{(E)in}_{t,n_y+1}\tilde{\gamma}^{(E)b}_{t,n_y+1}}{\left(1-\tilde{\rho}^{(E)in}_{t,n_y+1}-\tilde{\gamma}^{(E)b}_{t,n_y+1}\right)\left[q^{(E)}_{1,1,0}\tilde{\rho}^{(E)dyna}_{t,n_y+1}+q^{(E)}_{1,0,0}\left(1-\tilde{\rho}^{(E)dyna}_{t,n_y+1}\right)\right]+\tilde{\rho}^{(E)in}_{t,n_y+1}\tilde{\gamma}^{(E)b}_{t,n_y+1}}. \tag{C.28}$$

The final MU elevation message which represents the non-zero probability can be written as

$$\upsilon_{\tilde{e}_{t,n_y}\to\varpi_{k,t,n_y}} = \tilde{\rho}^{out}_{\tilde{\mathbf{e}}_t,k,n_y}\delta(\tilde{e}_{t,n_y} - 1) + (1 - \tilde{\rho}^{out}_{\tilde{\mathbf{e}}_t,k,n_y})\delta(\tilde{e}_{t,n_y}), \tag{C.29}$$

where $\tilde{\rho}^{out}_{\tilde{\mathbf{e}}_t,k,n_y}$ can be seen in (C.30).

$$\tilde{\rho}^{out}_{\tilde{\mathbf{e}}_t,k,n_y} = \frac{\tilde{\gamma}^{(E)f}_{t,n_y}\tilde{\gamma}^{(E)b}_{t,n_y}}{\tilde{\gamma}^{(E)f}_{t,n_y}\tilde{\gamma}^{(E)b}_{t,n_y} + \prod_{k'\neq k}\left((\rho^{in}_{\tilde{\mathbf{e}}_t,k',n_y})^{-1} - 1\right)\left(1 - \tilde{\gamma}^{(E)f}_{t,n_y}\right)\left(1 - \tilde{\gamma}^{(E)b}_{t,n_y}\right)}. \tag{C.30}$$

The message from factor node ϖ_{k,t,n_y} to variable node e_{k,t,n_y} is

$$\upsilon_{\varpi_{k,t,n_y}\to e_{k,t,n_y}}(e_{k,t,n_y}) = \rho^{(E)out}_{k,t,n_y}\delta(e_{k,t,n_y} - 1) + (1 - \rho^{(E)out}_{k,t,n_y})\delta(e_{k,t,n_y}), \tag{C.31}$$

where $\rho^{(E)out}_{k,t,n_y} = \tilde{\rho}^{out}_{\tilde{\mathbf{e}}_t,k,n_y}\varsigma^{(E)}_{k,t,1,1}$.

6. **Message Passing Over the Path** $e_{k,t,n_y} \to u_{k,t,m} \to b_{k,t,m} \to \pi_{k,t,m} \to g_{k,t,m}$

   The message from factor node e_{k,t,n_y} to variable node $u_{k,t,m}$ is

$$\upsilon_{e_{k,t,n_y} \to u_{k,t,m}}(e_{k,t,n_y}) = \rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y} \delta\left(e_{k,t,n_y} - 1\right) + \left(1 - \rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}\right) \delta\left(e_{k,t,n_y}\right), \tag{C.32}$$

   where $\rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y}$ can be written as

$$\rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y} = \frac{\rho^{(E)out}_{k,t,n_y}}{\rho^{(E)out}_{k,t,n_y} + \left(1 - \rho^{(E)out}_{k,t,n_y}\right) \prod_{n'_x \neq n_x} \left(\left(\rho^{in}_{\boldsymbol{e}_{k,t},k,t,n'_x,n_y}\right)^{-1} - 1\right)}. \tag{C.33}$$

   The message from factor node $u_{k,t,m}$ to variable node $b_{k,t,m}$ is

$$\upsilon_{u_{k,t,m} \to b_{k,t,m}}\left(b_{k,t,m}\right) = \rho^{out}_{\boldsymbol{b}_{k,t},k,t,m} \delta(b_{k,t,m} - 1) + \left(1 - \rho^{out}_{\boldsymbol{b}_{k,t},k,t,m}\right) \delta\left(b_{k,t,m}\right), \tag{C.34}$$

   where $\rho^{out}_{\boldsymbol{b}_{k,t},k,t,m} = \rho^{out}_{\boldsymbol{a}_{k,t},k,t,n_x,n_y} \rho^{out}_{\boldsymbol{e}_{k,t},k,t,n_x,n_y} \gamma_{\boldsymbol{b}_{k,t}}$. The message $\upsilon_{b_{k,t,m} \to \pi_{k,t,m}}$ from variable node $b_{k,t,m}$ to factor node $\pi_{k,t,m}$ is the same as $\upsilon_{u_{k,t,m} \to b_{k,t,m}}(b_{k,t,m})$. The message from factor node $\pi_{k,t,m}$ back to variable node $g_{k,t,m}$ is

$$\begin{aligned} \upsilon_{\pi_{k,t,m} \to g_{k,t,m}}\left(g_{k,t,m}\right) =& \rho^{out}_{\boldsymbol{b}_{k,t},k,t,m} \mathcal{CN}\left(g_{k,t,m}; \mu^{out}_{k,t,m}, (\sigma^{out}_{k,t,m})^2\right) + \\ & \left(1 - \rho^{out}_{\boldsymbol{b}_{k,t},k,t,m}\right) \delta\left(g_{k,t,m}\right). \end{aligned} \tag{C.35}$$

Appendix D

Gradient Update for 3D Off-grid Estimation

As the 3D off-grid estimation step is following the step 14 in **Algorithm 5.3** which represents the posterior estimation at the i^{th} iteration with $\hat{p}\left(\boldsymbol{g}_{k,t}|\boldsymbol{y}_k^{(t)}, \boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i)}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}\right) = \mathcal{CN}\left(\boldsymbol{g}_{\mathcal{B},k,t}^{post}, v_{\mathcal{B},k,t}^{post}\boldsymbol{I}\right)$. The surrogate function can be calculated as

$$\begin{aligned}
&\hat{\mathcal{L}}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}|\boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i)}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}\right) \propto E_{\hat{p}\left(\boldsymbol{g}_{k,t}|\boldsymbol{y}_k^{(t)}, \boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i)}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}\right)} \\
&\left[-\frac{1}{\sigma_e^2}\|\,\boldsymbol{y}_{k,t} - \boldsymbol{\mathit{\Phi}}_{k,t}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}\right)\boldsymbol{g}_{k,t}\|^2\right] \propto \\
&-\frac{1}{\sigma_e^2}\left(\|\,\boldsymbol{y}_{k,t} - \boldsymbol{\mathit{\Phi}}_{k,t}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}\right)\boldsymbol{g}_{\mathcal{B},k,t}^{post}\|^2 + \right. \\
&\left. v_{\mathcal{B},k,t}^{post}\mathrm{tr}\left(\boldsymbol{\mathit{\Phi}}_{k,t}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}\right)\boldsymbol{\mathit{\Phi}}_{k,t}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}\right)^H\right)\right).
\end{aligned} \tag{D.1}$$

1. **Azimuth off-grid estimation** The azimuth offset $\boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i+1)}$ can be updated use (5.49)[1]. The derivative $\left.\frac{\partial\mathcal{L}\left(\boldsymbol{\mathit{\Delta\theta}}_{k,t}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}|\boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i)}, \boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}\right)}{\partial\boldsymbol{\mathit{\Delta\theta}}_{k,t}}\right|_{\boldsymbol{\mathit{\Delta\theta}}_{k,t}=\boldsymbol{\mathit{\Delta\theta}}_{k,t}^{(i)}}$ with previous iteration updated elevation offset $\boldsymbol{\mathit{\Delta\phi}}_{k,t}^{(i)}$ can be written as $\boldsymbol{\mathit{\Lambda}}_{\boldsymbol{\mathit{\Delta\theta}}_{k,t}}^{(i)} = [\mathit{\Lambda}^{(i)}(\mathit{\Delta\theta}_{k,t,1}), ...,$

[1] To be noticed, in order to have azimuth and elevation one-one correspondence, we extend the original N_x dimension of offset into M dimension with azimuth and elevation paring.

$\varLambda^{(i)}(\varDelta\theta_{k,t,M})]^T$, with $\varLambda^{(i)}(\varDelta\theta_{k,t,m})$ as in (D.2),

$$\begin{aligned}\varLambda^{(i)}(\varDelta\theta_{k,t,m}) = 2\mathrm{Re}\Bigg(\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi^{(i)}_{k,t,m}\right)^H \boldsymbol{S}_k\boldsymbol{S}_k^H\\ \boldsymbol{A}\left(\tilde{\theta}_{k,t,m}+\varDelta\theta_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi^{(i)}_{k,t,m}\right)\varTheta_1^{(i)} + \\ 2\mathrm{Re}\left(\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi^{(i)}_{k,t,m}\right)^H \boldsymbol{S}_k\boldsymbol{\varTheta}_2^{(i)}\right),\end{aligned} \tag{D.2}$$

where $\varTheta_1^{(i)} = -\frac{1}{\sigma_e^2}\left(g^{post}_{\mathcal{B},k,t,m}+v^{post}_{\mathcal{B},k,t}\right)$, $\boldsymbol{\varTheta}_2^{(i)} = \frac{1}{\sigma_e^2}\left(g^{post}_{\mathcal{B},k,t,m}\right) * \boldsymbol{y}_{k,t,-m}$, $\boldsymbol{y}_{k,t,-m} = \boldsymbol{y}_{k,t} - \boldsymbol{S}_k^H\sum_{j\neq m}\boldsymbol{A}\left(\tilde{\theta}_{k,t,j}+\varDelta\theta_{k,t,j},\tilde{\phi}_{k,t,j}+\varDelta\phi^{(i)}_{k,t,j}\right)g^{post}_{\mathcal{B},k,t,j}$, and

$$\begin{aligned}\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi^{(i)}_{k,t,m}\right) = \\ d\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi^{(i)}_{k,t,m}\right)/d\varDelta\theta_{k,t,m}.\end{aligned} \tag{D.3}$$

2. **Elevation off-grid estimation** Similarly, the elevation offset $\boldsymbol{\varDelta\phi}^{(i+1)}_{k,t}$ can be updated use (5.50). The derivative $\left.\frac{\partial\mathcal{L}\left(\boldsymbol{\varDelta\theta}^{(i+1)}_{k,t},\boldsymbol{\varDelta\phi}_{k,t}|\boldsymbol{\varDelta\theta}^{(i+1)}_{k,t},\boldsymbol{\varDelta\phi}^{(i)}_{k,t}\right)}{\partial\boldsymbol{\varDelta\phi}_{k,t}}\right|_{\boldsymbol{\varDelta\phi}_{k,t}=\boldsymbol{\varDelta\phi}^{(i)}_{k,t}}$ with updated azimuth offset $\boldsymbol{\varDelta\theta}^{(i+1)}_{k,t}$ can be written as $\boldsymbol{\varLambda}^{(i)}_{\boldsymbol{\varDelta\phi}_{k,t}} = [\varLambda^{(i)}(\varDelta\phi_{k,t,1}),...,$ $\varLambda^{(i)}(\varDelta\phi_{k,t,M})]^T$, with $\varLambda^{(i)}(\varDelta\phi_{k,t,m})$ as

$$\begin{aligned}\varLambda^{(i)}(\varDelta\phi_{k,t,m}) = 2\mathrm{Re}\Bigg(\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta^{(i+1)}_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi_{k,t,m}\right)^H\\ \boldsymbol{S}_k\boldsymbol{S}_k^H\boldsymbol{A}\left(\tilde{\theta}_{k,t,m}+\varDelta\theta^{(i+1)}_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi_{k,t,m}\right)\varTheta_3^{(i)} + \\ 2\mathrm{Re}\left(\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta^{(i+1)}_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi_{k,t,m}\right)^H \boldsymbol{S}_k\boldsymbol{\varTheta}_4^{(i)}\right),\end{aligned} \tag{D.4}$$

where $\varTheta_3^{(i)} = -\frac{1}{\sigma_e^2}\left(g^{post}_{\mathcal{B},k,t,m}+v^{post}_{\mathcal{B},k,t}\right)$, $\boldsymbol{\varTheta}_4^{(i)} = \frac{1}{\sigma_e^2}\left(g^{post}_{\mathcal{B},k,t,m}\right) * \boldsymbol{y}_{k,t,-m}$, $\boldsymbol{y}_{k,t,-m} = \boldsymbol{y}_{k,t} - \boldsymbol{S}_k^H\sum_{j\neq m}\boldsymbol{A}\left(\tilde{\theta}_{k,t,j}+\varDelta\theta^{(i+1)}_{k,t,j},\tilde{\phi}_{k,t,j}+\varDelta\phi_{k,t,j}\right)g^{post}_{\mathcal{B},k,t,j}$, and

$$\begin{aligned}\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta^{(i+1)}_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi_{k,t,m}\right) = \\ d\boldsymbol{A}'\left(\tilde{\theta}_{k,t,m}+\varDelta\theta^{(i+1)}_{k,t,m},\tilde{\phi}_{k,t,m}+\varDelta\phi_{k,t,m}\right)/d\varDelta\phi_{k,t,m}.\end{aligned} \tag{D.5}$$

Bibliography

[1] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2714–2741, 2018. DOI: 10.1109/COMST.2018.2841996.

[2] A. N. Uwaechia and N. M. Mahyuddin, "A comprehensive survey on millimeter wave communications for fifth-generation wireless networks: Feasibility and challenges," *IEEE Access*, vol. 8, pp. 62 367–62 414, 2020. DOI: 10.1109/ACCESS.2020.2984204.

[3] S. M. R. Jones, R. E. Sheriff, and P. A. Watson, "Ka-band mobile-satellite service propagation environment," in *IEE Colloquium on Future of Ka Band for Satellite Communications*, 1993, pp. 1/1–1/7.

[4] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: it will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013. DOI: 10.1109/ACCESS.2013.2260813.

[5] H. Zhao, R. Mayzus, S. Sun, M. Samimi, J. K. Schulz, Y. Azar, K. Wang, G. N. Wong, F. Gutierrez, and T. S. Rappaport, "28 GHz millimeter wave cellular communication measurements for reflection and penetration loss in and around buildings in New York city," in *2013 IEEE International Conference on Communications (ICC)*, 2013, pp. 5163–5167. DOI: 10.1109/ICC.2013.6655403.

[6] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, 2014. DOI: 10.1109/MCOM.2014.6736761.

[7] M. Di Renzo, K. Ntontin, J. Song, F. H. Danufane, X. Qian, F. Lazarakis, J. De Rosny, D.-T. Phan-Huy, O. Simeone, R. Zhang, M. Debbah, G. Lerosey, M. Fink, S. Tretyakov, and S. Shamai, "Reconfigurable intelligent surfaces vs. relaying: Differences, similarities, and performance comparison," *IEEE Open J. of the Commun. Soc.*, vol. 1, pp. 798–807, 2020. DOI: 10.1109/OJCOMS.2020.3002955.

[8] Z. Li, Y. Wang, M. Liu, R. Sun, Y. Chen, J. Yuan, and J. Li, "Energy efficient resource allocation for UAV-assisted space-air-ground internet of remote things networks," *IEEE Access*, vol. 7, pp. 145 348–145 362, 2019. DOI: 10.1109/ACCESS.2019.2945478.

[9] J. Radtke, C. Kebschull, and E. Stoll, "Interactions of the space debris environment with mega constellations—using the example of the oneweb constellation," *Acta Astronautica*, vol. 131, pp. 55–68, 2017.

[10] T. Pultarova, "Telecommunications-space tycoons go head to head over mega satellite network [news briefing]," *Eng. Technol.*, vol. 10, no. 2, pp. 20–20, 2015.

[11] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, 2017, ISSN: 1558-1896. DOI: 10.1109/MCOM.2017.1601156.

[12] M. Conti and S. Giordano, "Mobile ad hoc networking: Milestones, challenges, and new research directions," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 85–96, 2014. DOI: 10.1109/MCOM.2014.6710069.

[13] S.-Y. Lien, S.-L. Shieh, Y. Huang, B. Su, Y.-L. Hsu, and H.-Y. Wei, "5G new radio: waveform, frame structure, multiple access, and initial access," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 64–71, 2017. DOI: 10.1109/MCOM.2017.1601107.

[14] M. Shafi, A. F. Molisch, P. J. Smith, T. Haustein, P. Zhu, P. De Silva, F. Tufvesson, A. Benjebbour, and G. Wunder, “5G: a tutorial overview of standards, trials, challenges, deployment, and practice,” *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1201–1221, 2017. DOI: 10.1109/JSAC.2017.2692307.

[15] M. Szydelko and M. Dryjanski, “3GPP spectrum access evolution towards 5G,” *EAI Endorsed Transactions on Cognitive Communications*, vol. 3, no. 10, 2017.

[16] A. Ghosh, A. Maeder, M. Baker, and D. Chandramouli, “5G evolution: a view on 5G cellular technology beyond 3GPP release 15,” *IEEE Access*, vol. 7, pp. 127 639–127 651, 2019. DOI: 10.1109/ACCESS.2019.2939938.

[17] C. Chen, O. Kedem, C. R. C. M. d. Silva, and C. Cordeiro, “Millimeter-wave fixed wireless access using IEEE 802.11ay,” *IEEE Commun. Mag.*, vol. 57, no. 12, pp. 98–104, 2019. DOI: 10.1109/MCOM.001.1900076.

[18] S. Gong, D. Wei, X. Xue, and M. Y. Chen, “Study on the channel model and BER performance of single-polarization satellite-earth MIMO communication systems at Ka band,” *IEEE Trans. Antennas Propag.*, vol. 62, no. 10, pp. 5282–5297, 2014, ISSN: 0018-926X. DOI: 10.1109/TAP.2014.2342754.

[19] T. Hong, W. Zhao, R. Liu, and M. Kadoch, “Space-air-ground IoT network and related key technologies,” *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 96–104, 2020. DOI: 10.1109/MWC.001.1900186.

[20] L. Zhang, H. Zhao, S. Hou, Z. Zhao, H. Xu, X. Wu, Q. Wu, and R. Zhang, “A survey on 5G millimeter wave communications for UAV-assisted wireless networks,” *IEEE Access*, vol. 7, pp. 117 460–117 504, 2019. DOI: 10.1109/ACCESS.2019.2929241.

[21] Y. Banday, G. Mohammad Rather, and G. R. Begh, “Effect of atmospheric absorption on millimetre wave frequencies for 5G cellular networks,” *IET Commun.*, vol. 13, no. 3, pp. 265–270, 2019.

[22] D. Nandi and A. Maitra, “Study of rain attenuation effects for 5G Mm-wave cellular communication in tropical location,” *IET Microw., Antennas & Propag.*, vol. 12, no. 9, pp. 1504–1507, 2018.

[23] I. A. Hemadeh, K. Satyanarayana, M. El-Hajjar, and L. Hanzo, “Millimeter-wave communications: Physical channel models, design considerations, antenna constructions, and link-budget,” *IEEE Comm. Surv. & Tutorials.*, vol. 20, no. 2, pp. 870–913, 2018. DOI: `10.1109/COMST.2017.2783541`.

[24] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, “Scaling up MIMO: opportunities and challenges with very large arrays,” *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, 2013. DOI: `10.1109/MSP.2011.2178495`.

[25] Q. Xue, P. Zhou, X. Fang, and M. Xiao, “Performance analysis of interference and eavesdropping immunity in narrow beam mmwave networks,” *IEEE Access*, vol. 6, pp. 67 611–67 624, 2018. DOI: `10.1109/ACCESS.2018.2878741`.

[26] Y. Yang, B. Zheng, S. Zhang, and R. Zhang, “Intelligent reflecting surface meets OFDM: protocol design and rate maximization,” *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 4522–4535, 2020. DOI: `10.1109/TCOMM.2020.2981458`.

[27] M. Di Renzo, A. Zappone, M. Debbah, M.-S. Alouini, C. Yuen, J. de Rosny, and S. Tretyakov, “Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and the road ahead,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, 2020. DOI: `10.1109/JSAC.2020.3007211`.

[28] Y. Liu, X. Liu, X. Mu, T. Hou, J. Xu, Z. Qin, M. Di Renzo, and N. Al-Dhahir, “Reconfigurable intelligent surfaces: Principles and opportunities,” *arXiv preprint arXiv:2007.03435*, 2020.

[29] L. You, K.-X. Li, J. Wang, X. Gao, X.-G. Xia, and B. Ottersten, “Massive MIMO transmission for LEO satellite communications,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1851–1865, 2020. DOI: `10.1109/JSAC.2020.3000803`.

[30] P.-D. Arapoglou, P. Burzigotti, M. Bertinelli, A. Bolea Alamanac, and R. De Gaudenzi, “To MIMO or not to MIMO in mobile satellite broadcasting systems,” *IEEE Trans. Wireless Commun.*, vol. 10, no. 9, pp. 2807–2811, 2011. DOI: `10.1109/TWC.2011.071411.101599`.

[31] Z. Ma, B. Ai, R. He, G. Wang, Y. Niu, M. Yang, J. Wang, Y. Li, and Z. Zhong, "Impact of UAV Rotation on MIMO channel characterization for air-to-ground communication systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12 418–12 431, 2020. DOI: 10.1109/TVT.2020.3028301.

[32] H. Jiang, Z. Zhang, L. Wu, and J. Dang, "Three-dimensional geometry-based UAV-MIMO channel modeling for A2G communication environments," *IEEE Commun. Lett.*, vol. 22, no. 7, pp. 1438–1441, 2018. DOI: 10.1109/LCOMM.2018.2828110.

[33] A. Alkhateeb, O. El Ayach, G. Leus, and R. W. Heath, "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, 2014. DOI: 10.1109/JSTSP.2014.2334278.

[34] R. W. Heath, N. González-Prelcic, S. Rangan, W. Roh, and A. M. Sayeed, "An overview of signal processing techniques for millimeter wave MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 436–453, 2016. DOI: 10.1109/JSTSP.2016.2523924.

[35] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: From theory to applications," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4053–4085, 2011. DOI: 10.1109/TSP.2011.2161982.

[36] A. Liu, V. K. N. Lau, and W. Dai, "Exploiting burst-sparsity in massive MIMO with partial channel support information," *IEEE Trans. on Wireless Commun.*, vol. 15, no. 11, pp. 7820–7830, 2016, ISSN: 1536-1276. DOI: 10.1109/TWC.2016.2608342.

[37] L. Chen, A. Liu, and X. Yuan, "Structured turbo compressed sensing for massive MIMO channel estimation using a Markov prior," *IEEE Trans. on Veh. Technol.*, vol. 67, no. 5, pp. 4635–4639, 2018. DOI: 10.1109/TVT.2017.2787708.

[38] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, 2007. DOI: 10.1109/TIT.2007.909108.

[39] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and computational harmonic analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[40] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2230–2249, 2009. DOI: 10.1109/TIT.2009.2016006.

[41] R. Prasad, C. R. Murthy, and B. D. Rao, "Joint approximately sparse channel estimation and data detection in ofdm systems using sparse bayesian learning," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3591–3603, 2014. DOI: 10.1109/TSP.2014.2329272.

[42] J. Dai, A. Liu, and V. K. N. Lau, "FDD massive MIMO channel estimation with arbitrary 2D-array geometry," *IEEE Trans. Signal Process.*, vol. 66, no. 10, pp. 2584–2599, 2018. DOI: 10.1109/TSP.2018.2807390.

[43] D. Wipf and B. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, 2004. DOI: 10.1109/TSP.2004.831016.

[44] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing: I. motivation and construction," in *Proc. Inf. Theory Workshop (ITW)*, 2010, pp. 1–5. DOI: 10.1109/ITWKSPS.2010.5503193.

[45] X. Wei, C. Hu, and L. Dai, "Deep learning for beamspace channel estimation in millimeter-wave massive mimo systems," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 182–193, 2021. DOI: 10.1109/TCOMM.2020.3027027.

[46] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.

[47] H. Huang, J. Yang, H. Huang, Y. Song, and G. Gui, "Deep learning for super-resolution channel estimation and doa estimation based massive mimo system," *IEEE Trans. on Veh. Technol.*, vol. 67, no. 9, pp. 8549–8560, 2018. DOI: 10.1109/TVT.2018.2851783.

[48] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, 2018. DOI: 10.1109/LWC.2017.2757490.

[49] X. Rao and V. K. N. Lau, "Compressive sensing with prior support quality information and application to massive MIMO channel estimation with temporal correlation," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4914–4924, 2015. DOI: 10.1109/TSP.2015.2446444.

[50] X. Zhu, L. Dai, W. Dai, Z. Wang, and M. Moonen, "Tracking a dynamic sparse channel via differential orthogonal matching pursuit," in *IEEE Mil.Commun. Conf.*, 2015, pp. 792–797. DOI: 10.1109/MILCOM.2015.7357541.

[51] L. Dai and X. Gao, "Priori-aided channel tracking for millimeter-Wave beamspace massive MIMO systems," in *2016 URSI Asia-Pacific Radio Science Conference (URSI AP-RASC)*, 2016, pp. 1493–1496. DOI: 10.1109/URSIAP-RASC.2016.7601171.

[52] X. Liu, Y. Shi, J. Zhang, and K. B. Letaief, "Massive csi acquisition for dense cloud-rans with spatial-temporal dynamics," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2557–2570, 2018. DOI: 10.1109/TWC.2018.2797969.

[53] J. Ziniel and P. Schniter, "Dynamic compressive sensing of time-varying signals via approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5270–5284, 2013. DOI: 10.1109/TSP.2013.2273196.

[54] J. Connor, R. Martin, and L. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 240–254, 1994. DOI: 10.1109/72.279188.

[55] W. Jiang and H. D. Schotten, "Neural network-based fading channel prediction: A comprehensive overview," *IEEE Access*, vol. 7, pp. 118 112–118 124, 2019. DOI: 10.1109/ACCESS.2019.2937588.

[56] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation,"

IEEE J. Sel. Areas Commun, vol. 32, no. 6, pp. 1164–1179, 2014. DOI: 10.1109/JSAC.2014.2328154.

[57] J. Zhang, W. Xu, H. Gao, M. Pan, Z. Han, and P. Zhang, "Codebook-based beam tracking for conformal array-enabled UAV mmWave networks," *IEEE Internet Things J.*, vol. 8, no. 1, pp. 244–261, 2021. DOI: 10.1109/JIOT.2020.3005394.

[58] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, 2016, ISSN: 0163-6804. DOI: 10.1109/MCOM.2016.7470933.

[59] B. Li, Z. Fei, and Y. Zhang, "UAV communications for 5G and beyond: recent advances and future trends," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2241–2263, 2019. DOI: 10.1109/JIOT.2018.2887086.

[60] C. Zhang, W. Zhang, W. Wang, L. Yang, and W. Zhang, "Research challenges and opportunities of UAV millimeter-wave communications," *IEEE Wireless Communi.*, vol. 26, no. 1, pp. 58–62, 2019. DOI: 10.1109/MWC.2018.1800214.

[61] S. Li, B. Duo, X. Yuan, Y.-C. Liang, and M. Di Renzo, "Reconfigurable intelligent surface assisted UAV communication: joint trajectory design and passive beamforming," *IEEE Wireless Commun. Lett.*, vol. 9, no. 5, pp. 716–720, 2020. DOI: 10.1109/LWC.2020.2966705.

[62] Z. Wei, Y. Cai, Z. Sun, D. W. K. Ng, J. Yuan, M. Zhou, and L. Sun, "Sum-rate maximization for IRS-assisted UAV OFDMA communication systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2530–2550, 2021. DOI: 10.1109/TWC.2020.3042977.

[63] Z. Mohamed and S. Aïssa, "Leveraging UAVs with intelligent reflecting surfaces for energy-efficient communications with cell-edge users," in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2020, pp. 1–6. DOI: 10.1109/ICCWorkshops49005.2020.9145273.

[64] L. Ge, P. Dong, H. Zhang, J.-B. Wang, and X. You, "Joint beamforming and trajectory optimization for intelligent reflecting surfaces-assisted UAV communi-

cations," *IEEE Access*, vol. 8, pp. 78 702–78 712, 2020. DOI: 10.1109/ACCESS.2020.2990166.

[65] M. A. ElMossallamy, H. Zhang, L. Song, K. G. Seddik, Z. Han, and G. Y. Li, "Reconfigurable intelligent surfaces for wireless communications: Principles, challenges, and opportunities," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 3, pp. 990–1002, 2020. DOI: 10.1109/TCCN.2020.2992604.

[66] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, "Intelligent reflecting surface aided wireless communications: A tutorial," *IEEE Trans. Commun.*, pp. 1–1, 2021. DOI: 10.1109/TCOMM.2021.3051897.

[67] H. Yin, D. Gesbert, M. Filippou, and Y. Liu, "A coordinated approach to channel estimation in large-scale multiple-antenna systems," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 264–273, 2013. DOI: 10.1109/JSAC.2013.130214.

[68] C. R. Berger, Z. Wang, J. Huang, and S. Zhou, "Application of compressive sensing to sparse channel estimation," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 164–174, 2010. DOI: 10.1109/MCOM.2010.5621984.

[69] T. Jiang, Y. Shi, J. Zhang, and K. B. Letaief, "Joint activity detection and channel estimation for IoT networks: phase transition and computation-estimation tradeoff," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6212–6225, 2019. DOI: 10.1109/JIOT.2018.2881486.

[70] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile internet of things: Can uavs provide an energy-efficient mobile architecture?" In *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–6. DOI: 10.1109/GLOCOM.2016.7841993.

[71] X. Cheng and Y. Li, "A 3-D geometry-based stochastic model for UAV-MIMO wideband nonstationary channels," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1654–1662, 2019. DOI: 10.1109/JIOT.2018.2874816.

[72] H. Jiang, Z. Zhang, and G. Gui, "Three-dimensional non-stationary wideband geometry-based UAV channel model for A2G communication environments," *IEEE Access*, vol. 7, pp. 26 116–26 122, 2019. DOI: 10.1109/ACCESS.2019.2897431.

[73] H. Jiang, Z. Zhang, C. X. Wang, J. Zhang, J. Dang, L. Wu, and H. Zhang, "A novel 3D UAV channel model for A2G communication environments using AoD and AoA estimation algorithms," *IEEE Trans. Commun.*, vol. 68, no. 11, pp. 7232–7246, 2020. DOI: 10.1109/TCOMM.2020.3011716.

[74] E. Basar, "Reconfigurable intelligent surface-based index modulation: A new beyond MIMO paradigm for 6G," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 3187–3196, 2020.

[75] Z. Wang, Y. Shi, Y. Zhou, H. Zhou, and N. Zhang, "Wireless-powered over-the-air computation in intelligent reflecting surface-aided IoT networks," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1585–1598, 2021. DOI: 10.1109/JIOT.2020.3015489.

[76] C. Huang, R. Mo, and C. Yuen, "Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning," *IEEE J. Sel. Areas in Commun.*, vol. 38, no. 8, pp. 1839–1850, 2020.

[77] G. Yu, X. Chen, C. Zhong, D. W. Kwan Ng, and Z. Zhang, "Design, analysis, and optimization of a large intelligent reflecting surface-aided B5G cellular internet of things," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8902–8916, 2020. DOI: 10.1109/JIOT.2020.2996984.

[78] Z. Wang, L. Liu, and S. Cui, "Channel estimation for intelligent reflecting surface assisted multiuser communications: Framework, algorithms, and analysis," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6607–6620, 2020. DOI: 10.1109/TWC.2020.3004330.

[79] B. Zheng, C. You, and R. Zhang, "Intelligent reflecting surface assisted multi-user OFDMA: channel estimation and training design," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8315–8329, 2020. DOI: 10.1109/TWC.2020.3021434.

[80] Y. Wang, H. Lu, and H. Sun, "Channel estimation in IRS-enhanced mmWave system with super-resolution network," *IEEE Commun. Lett.*, vol. 25, no. 8, pp. 2599–2603, 2021. DOI: 10.1109/LCOMM.2021.3079322.

[81] X. Hu, R. Zhang, and C. Zhong, "Semi-passive elements assisted channel estimation for intelligent reflecting surface-aided communications," *IEEE Trans. Wireless Commun.*, pp. 1–1, 2021. DOI: 10.1109/TWC.2021.3102446.

[82] C. You, B. Zheng, and R. Zhang, "Channel estimation and passive beamforming for intelligent reflecting surface: Discrete phase shift and progressive refinement," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2604–2620, 2020. DOI: 10.1109/JSAC.2020.3007056.

[83] X. Ma, Z. Chen, W. Chen, Z. Li, Y. Chi, C. Han, and S. Li, "Joint channel estimation and data rate maximization for intelligent reflecting surface assisted Terahertz MIMO communication systems," *IEEE Access*, vol. 8, pp. 99 565–99 581, 2020. DOI: 10.1109/ACCESS.2020.2994100.

[84] B. Zheng, C. You, and R. Zhang, "Efficient channel estimation for double-IRS aided multi-user MIMO system," *IEEE Trans. Commun.*, pp. 1–1, 2021. DOI: 10.1109/TCOMM.2021.3064947.

[85] C. Liu, X. Liu, D. W. K. Ng, and J. Yuan, "Deep residual learning for channel estimation in intelligent reflecting surface-assisted multi-user communications," *arXiv preprint arXiv:2009.01423*, 2020.

[86] Y. Zhang, Y. Mu, Y. Liu, T. Zhang, and Y. Qian, "Deep learning-based beamspace channel Estimation in mmWave massive MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 12, pp. 2212–2215, 2020. DOI: 10.1109/LWC.2020.3019321.

[87] S. Liu, Z. Gao, J. Zhang, M. D. Renzo, and M. S. Alouini, "Deep denoising neural network assisted compressive channel estimation for mmWave intelligent reflecting surfaces," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9223–9228, 2020. DOI: 10.1109/TVT.2020.3005402.

[88] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Enabling large intelligent surfaces with compressive sensing and deep learning," *IEEE Access*, pp. 1–1, 2021. DOI: 10.1109/ACCESS.2021.3064073.

[89] A. M. Elbir, A. Papazafeiropoulos, P. Kourtessis, and S. Chatzinotas, "Deep channel learning for large intelligent surfaces aided mm-Wave massive MIMO

systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1447–1451, 2020. DOI: 10.1109/LWC.2020.2993699.

[90] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 106–112, 2020. DOI: 10.1109/MCOM.001.1900107.

[91] X. Liu, Y. Liu, and Y. Chen, "Machine learning empowered trajectory and passive beamforming design in UAV-RIS wireless networks," *arXiv preprint arXiv:2010.02749*, 2020.

[92] H. Iimori, G. T. F. de Abreu, O. Taghizadeh, R.-A. Stoica, T. Hara, and K. Ishibashi, "Stochastic learning robust beamforming for millimeter-wave systems with path blockage," *IEEE Wireless Commun. Lett.*, vol. 9, no. 9, pp. 1557–1561, 2020. DOI: 10.1109/LWC.2020.2997366.

[93] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Deep learning based channel estimation schemes for IEEE 802.11p standard," *IEEE Access*, vol. 8, pp. 113 751–113 765, 2020. DOI: 10.1109/ACCESS.2020.3003286.

[94] Y. Yang, F. Gao, X. Ma, and S. Zhang, "Deep learning-based channel estimation for doubly selective fading channels," *IEEE Access*, vol. 7, pp. 36 579–36 589, 2019. DOI: 10.1109/ACCESS.2019.2901066.

[95] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[96] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[97] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: a search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017. DOI: 10.1109/TNNLS.2016.2582924.

[98] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, 1997. DOI: `10.1109/78.650093`.

[99] Y. Bengio, *Learning deep architectures for AI.* Now Publishers Inc, 2009.

[100] T. L. Jensen and E. De Carvalho, "An optimal channel estimation scheme for intelligent reflecting surfaces based on a minimum variance unbiased estimator," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 5000–5004. DOI: `10.1109/ICASSP40776.2020.9053695`.

[101] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv preprint arXiv:1402.1128*, 2014.

[102] N. Hosseini and D. W. Matolak, "Software defined radios as cognitive relays for satellite ground stations incurring terrestrial interference," in *IEEE CCAA*, 2017, pp. 1–4. DOI: `10.1109/CCAAW.2017.8001874`.

[103] N. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "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, ISSN: 1558-0008. DOI: `10.1109/JSAC.2019.2906789`.

[104] A. Varasteh, S. Hofmann, N. Deric, A. Blenk, D. Schupke, W. Kellerer, and C. M. Machuca, "Toward optimal mobility-aware vm placement and routing in space-air-ground integrated networks," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 1–2. DOI: `10.1109/INFCOMW.2019.8845123`.

[105] J. Zhao, F. Gao, Q. Wu, S. Jin, Y. Wu, and W. Jia, "Beam tracking for UAV mounted SatCom on-the-move with massive antenna array," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 2, pp. 363–375, 2018, ISSN: 0733-8716. DOI: `10.1109/JSAC.2018.2804239`.

[106] X. Rao and V. K. N. Lau, "Distributed compressive CSIT estimation and feedback for FDD multi-user massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3261–3271, 2014, ISSN: 1941-0476. DOI: 10.1109/TSP.2014.2324991.

[107] M. Masood, L. H. Afify, and T. Y. Al-Naffouri, "Efficient coordinated recovery of sparse channels in massive MIMO," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 104–118, 2015, ISSN: 1941-0476. DOI: 10.1109/TSP.2014.2369005.

[108] X. Zhu, L. Dai, G. Gui, W. Dai, Z. Wang, and F. Adachi, "Structured matching pursuit for reconstruction of dynamic sparse channels," in *IEEE Global Commun. Conf. (GLOBECOM)*, 2015, pp. 1–5. DOI: 10.1109/GLOCOM.2015.7416980.

[109] X. Gao, L. Dai, Y. Zhang, T. Xie, X. Dai, and Z. Wang, "Fast channel tracking for terahertz beamspace massive mimo systems," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 5689–5696, 2017, ISSN: 1939-9359. DOI: 10.1109/TVT.2016.2614994.

[110] J. Zhao, F. Gao, W. Jia, S. Zhang, S. Jin, and H. Lin, "Angle domain hybrid precoding and channel tracking for millimeter wave massive mimo systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6868–6880, 2017, ISSN: 1558-2248. DOI: 10.1109/TWC.2017.2732405.

[111] Q. Qin, L. Gui, P. Cheng, and B. Gong, "Time-varying channel estimation for millimeter wave multiuser MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9435–9448, 2018. DOI: 10.1109/TVT.2018.2854735.

[112] L. Lian, A. Liu, and V. K. N. Lau, "Exploiting dynamic sparsity for downlink FDD-massive MIMO channel tracking," *IEEE Trans. Signal Process.*, vol. 67, no. 8, pp. 2007–2021, 2019, ISSN: 1053-587X. DOI: 10.1109/TSP.2019.2896179.

[113] R. He, C. Schneider, B. Ai, G. Wang, D. Dupleich, R. Thomae, M. Boban, J. Luo, z. Zhong, and Y. Zhang, "Propagation channels of 5G millimeter wave vehicle-to-vehicle communications: recent advances and future challenges," *IEEE Veh. Technol. Mag.*, pp. 1–1, 2019, ISSN: 1556-6080. DOI: 10.1109/MVT.2019.2928898.

[114] R. He, B. Ai, G. L. Stuber, G. Wang, and Z. Zhong, "Geometrical-based modeling for millimeter-wave MIMO mobile-to-mobile channels," *IEEE Trans. Veh.*

Technol., vol. 67, no. 4, pp. 2848–2863, 2018, ISSN: 1939-9359. DOI: `10.1109/TVT.2017.2774808`.

[115] W. Shi, J. Li, N. Cheng, F. Lyu, S. Zhang, H. Zhou, *et al.*, "Multi-drone 3D trajectory planning and scheduling in drone assisted radio access networks," *arXiv preprint arXiv:1906.00777*, 2019.

[116] H. Fenech, S. Amos, and T. Waterfield, "The role of array antennas in commercial telecommunication satellites," in *2016 10th European Conference on Antennas and Propagation (EuCAP)*, 2016, pp. 1–4. DOI: `10.1109/EuCAP.2016.7481857`.

[117] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Wireless communication using unmanned aerial vehicles (UAVs): optimal transport theory for hover time optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 8052–8066, 2017, ISSN: 1558-2248. DOI: `10.1109/TWC.2017.2756644`.

[118] J. Ma, X. Yuan, and L. Ping, "On the performance of turbo signal recovery with partial DFT sensing matrices," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1580–1584, 2015. DOI: `10.1109/LSP.2015.2414951`.

[119] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[120] P. Schniter, "Turbo reconstruction of structured sparse signals," in *Proc. Conf. Inf. Sci. Syst. (CISS)*, 2010, pp. 1–6.

[121] Z. Xue, J. Ma, and X. Yuan, "Denoising-based turbo compressed sensing," *IEEE Access*, vol. 5, pp. 7193–7204, 2017. DOI: `10.1109/ACCESS.2017.2697978`.

[122] A. Liao, Z. Gao, D. Wang, H. Wang, H. Yin, D. W. K. Ng, and M.-S. Alouini, "Terahertz ultra-massive mimo-based aeronautical communications in space-air-ground integrated networks," *IEEE J. Sel. Areas Commun.*, pp. 1–1, 2021. DOI: `10.1109/JSAC.2021.3071834`.

[123] J. Yu, X. Liu, Y. Gao, and X. Shen, "3D channel tracking for UAV-satellite communications in space-air-ground integrated networks," *IEEE J. Sel. Areas Commun.*, pp. 1–1, 2020.

[124] B. Qi, W. Wang, and B. Wang, "Off-grid compressive channel estimation for mm-Wave massive MIMO with hybrid precoding," *IEEE Commun. Lett.*, vol. 23, no. 1, pp. 108–111, 2019.

[125] C. K. Anjinappa, A. C. Gürbüz, Y. Yapıcı, and Güvenç, "Off-grid aware channel and covariance estimation in mmwave networks," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3908–3921, 2020.

[126] Y. Xu and C. Ruan, "A novel design of circularly polarized omni-directional antenna for Ka-Band," in *2008 Global Symposium on Millimeter Waves*, 2008, pp. 378–379. DOI: 10.1109/GSMM.2008.4534652.

[127] Güngör and A. Ünal, "Design of a vertically polarized omni-directional antenna at Ka-Band," in *2016 IEEE International Symposium on Antennas and Propagation (APSURSI)*, 2016, pp. 129–130. DOI: 10.1109/APS.2016.7695773.

[128] Y. Wu, G. Hu, F. Jin, and J. Zu, "A satellite handover strategy based on the potential game in LEO satellite networks," *IEEE Access*, vol. 7, pp. 133 641–133 652, 2019.

[129] M. Ke, Z. Gao, Y. Wu, X. Gao, and K.-K. Wong, "Massive access in cell-free massive MIMO-based internet of things: cloud computing and edge computing paradigms," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 3, pp. 756–772, 2021. DOI: 10.1109/JSAC.2020.3018807.

[130] M. Ke, Z. Gao, Y. Wu, X. Gao, and R. Schober, "Compressive sensing-based adaptive active user detection and channel estimation: massive access meets massive MIMO," *IEEE Trans. Signal Process.*, vol. 68, pp. 764–779, 2020. DOI: 10.1109/TSP.2020.2967175.

[131] S. Wu, H. Yao, C. Jiang, X. Chen, L. Kuang, and L. Hanzo, "Downlink channel estimation for massive MIMO systems relying on vector approximate message passing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 5145–5148, 2019.

[132] N. Dandanov, K. Tonchev, V. Poulkov, and P. Koleva, "Approximate message passing for downlink sparse channel estimation in FDD massive MIMO," in

2019 42nd International Conference on Telecommunications and Signal Processing (TSP), 2019, pp. 487–491.