Large Multimodal Model-Based Environment-Aware Beam Management

Seungnyun Kim, *Member, IEEE*, Subham Saha, *Student Member, IEEE*, Seokhyun Jeong, *Student Member, IEEE*, Byonghyo Shim, *Fellow, IEEE*, and Moe Z. Win, *Fellow, IEEE*

***Abstract*—Beam management is an essential operation of next-generation (xG) wireless networks to compensate severe signal attenuation and ensure reliable communications over millimeter wave (mmWave) and terahertz (THz) bands. The primary objective of beam management is to determine beam directions that are properly aligned with the signal propagation paths. Conventional beam management techniques typically rely on geometric channel parameters such as angles, delays, and path gains. However, due to their lack of contextual awareness of the surrounding environment, these techniques fall short in handling the piecewise continuous changes in the geometric channel parameters caused by sudden obstructions or variations in scatterers. In this paper, we propose a novel beam management framework that utilizes environmental information extracted from sensor data (e.g., images) and pilot measurements to optimize beam directions. The main idea of the proposed scheme is to utilize visual channel parameters, including the positions of user equipment (UE), reflection points, and scatterers, which provide a direct visualization of the propagation environment. By tracking the visual channel parameters, dynamic changes in scatterers and propagation paths can be monitored over time. To analyze multimodal data and track these parameters, we exploit large multimodal model (LMM), a generative artificial intelligence (AI) model specialized in extracting correlated features across multimodal data and generating subsequent data. Simulation results show that the proposed scheme can accurately track the visual channel parameters and enhance data rate.**

***Index Terms*—Large multimodal model (LMM), integrated sensing and communications, environment-awareness, beam management.**

I. Introduction

NEXT-GENERATION (xG) wireless networks are expected to support immersive mobile services, including digital twin, telepresence, and high-fidelity holographic display, paving the way for intelligent society [1], [2], [3]. To meet the tremendous data rate demands of emerging services, the utilization of higher frequency bands (e.g., millimeter wave (mmWave) and terahertz (THz) bands) is imperative. However, despite the abundant spectrum resources in mmWave/THz bands, their primary limitation is the severe signal attenuation caused by propagation, reflection, penetration, and atmospheric absorption. To compensate for this signal attenuation, beamforming techniques that concentrate signal power from multiple antennas in specific directions are widely used [4], [5], [6], [7], [8], [9], [10], [11], [12], [13]. Since beamforming gain varies significantly with beam directions, the base station (BS) needs to select the optimal beams that align precisely with the signal propagation paths.

The process of establishing, optimizing, and maintaining beam direction is collectively called *beam management*. In 5G New Radio (NR), a two-stage beam management scheme is employed. In the first stage, the BS sequentially transmits multiple training beams carrying reference signals (e.g., synchronization signal block (SSB) and channel state information reference signal (CSI-RS)) toward different directions (i.e., beam sweeping). In the second stage, the user equipment (UE) reports the index of the beam maximizing the reference signal received power (RSRP) back to the BS (i.e., beam reporting) [14]. Also, hierarchical beam training algorithms and codebook designs have been proposed in [15] and [16]. While these beam training schemes are effective to some extent, the power consumption and signaling latency associated with the beam sweeping operation are considerable since the beam direction is determined only after measuring the RSRP of all training beams. For example, in 5G NR Frequency Range 2 (FR2) band (i.e., 24-71 GHz), up to 64 SSB beams are sequentially transmitted within the SSB burst set. Furthermore, in mmWave/THz systems where massive antenna arrays are deployed, the number of training beams required to sweep a specific angular area increases significantly due to the narrower beamwidth, thereby exacerbating the beam training overhead.

To reduce beam training overhead and improve beamforming gain, beam tracking techniques that predict future beam direction by leveraging the sequence of previous beam directions and channel measurements have attracted much attention in recent years [17]. Since the complicated handshaking between the BS and the UE is unnecessary, beam tracking can reduce the signaling latency in beam management. Over the years, various beam tracking techniques that track the changes in geometric channel parameters (GCPs) (i.e., angles, delays,

Received 14 February 2025; revised 8 July 2025; accepted 5 September 2025. Date of publication 21 January 2026; date of current version 4 February 2026. The fundamental research described in this paper was supported, in part, by the National Research Foundation of Korea under Grant RS-2023-00252789 and Grant RS-2024-00409492, by the Institute of Information and Communications Technology Planning and Evaluation of Korea under Grant RS-2024-00398157 and Grant IITP-2025-RS-2024-00418784, and by the Robert R. Taylor Professorship. *(Corresponding author: Moe Z. Win.)*

Seungnyun Kim and Subham Saha are with the Wireless Information and Network Sciences Laboratory (WINS Lab), Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: snkim94@mit.edu; subhams@mit.edu).

Seokhyun Jeong and Byonghyo Shim are with the Department of Electrical and Computer Engineering and the Institute of New Media and Communications (INMC), Seoul National University, Seoul 08826, Republic of Korea (e-mail: shjeong@islab.snu.ac.kr; bshim@snu.ac.kr).

Moe Z. Win is with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: moewin@mit.edu).

Digital Object Identifier 10.1109/JSAC.2025.3611419

and path gains) have been proposed [4]. Among these, the most widely used method is the Kalman filter, which recursively estimates the states of a system using linear models. For example, in [18], [19], and [20], extended Kalman filter (EKF)-based beam tracking techniques for mmWave systems have been proposed. Furthermore, to handle the nonlinearity in GCPs, beam tracking techniques employing unscented Kalman filter (UKF) algorithm have been introduced in [21] and [22]. Moreover, there have been significant efforts to leverage artificial intelligence (AI)/machine learning (ML) capabilities for beam tracking [23]. In [24] and [25], ML-based approaches have been investigated. In [26] and [27], beam tracking techniques using recurrent neural network (RNN) architectures (e.g., long short-term memory (LSTM)) have been proposed. Also, deep reinforcement learning (DRL) has been adopted for beam management in [28].

The major drawback of the conventional beam tracking techniques is their deficiency in handling abrupt transitions in GCPs caused by blockages and mobilities. Specifically, while these parameters vary continuously within the same set of scatterers, they shift discontinuously when the scatterers change (i.e., piecewise continuity). Since GCPs do not exhibit contextual information about the surrounding environments (e.g., positions and sizes of scatterers), they fail to capture scatterer dynamics, causing significant performance degradation in the conventional beam tracking techniques. This issue becomes even more pronounced in high-mobility scenarios such as vehicle-to-everything (V2X) communications where scatterers change more frequently. To address this issue, approaches that leverage sensing information (e.g., images and radar signals) collected from various sensors (e.g., cameras, radar, and LiDAR) have been introduced recently [29]. In [30], [31], and [32], beam tracking techniques based on the integrated sensing and communications (ISAC) framework have been proposed. These techniques utilize radar signals for sensing the surrounding environment and transmitting data. Also, vision-aided beam tracking schemes which exploit visual information extracted from images have been proposed in [33] and [34]. However, a key limitation of conventional sensing-aided beam tracking techniques is that they exploit environmental information exclusively from the line-of-sight (LOS) path while neglecting scatterers in the non-line-of-sight (NLOS) paths. Due to this reason, they may not perform well when the LOS path is blocked or when both LOS and NLOS paths coexist.

Recently, sensing-aided beam tracking techniques that also tracks the NLOS components of channel have been proposed [35], [36]. In [35], an ISAC-assisted beam tracking scheme that tracks the NLOS components by analyzing the radar echo signals using the EKF technique has been proposed. Also, in [36], a joint beam training and target sensing framework is introduced for reconfigurable intelligent surfaces (RIS)-assisted ISAC systems, where NLOS paths are converted into virtual LOS links by dynamically adjusting RIS phase shifts to direct and collect reflected echo signals. While these techniques are effective to some extent, they primarily focus on estimating the numerical values of NLOS components and lack a contextual understanding of signal propagation and the surrounding environment. As a result, these techniques fall short in real-time adaptation to dynamically varying environments with transient obstacles.

The fundamental questions related to the environment-aware beam management are: 1) how to identify comprehensive propagation environments including UE and scatterers; and 2) how to effectively track this environmental information over time. The answers to these questions will facilitate *environment-awareness*, the ability to comprehend the surrounding physical environment and proactively respond to its changes, thereby achieving the data rate maximization.

The aim of this paper is to propose an environment-aware multimodal beam management framework based on sensing and large multimodal model (LMM). The key idea of the proposed scheme, henceforth referred to as LMM-based beam management (LMM-BM), is to track visual channel parameters (VCPs), i.e., positions of UE, reflection points, and scatterers, using both pilot measurements (e.g., sounding reference signal (SRS)) and sensing measurements (e.g., images). Unlike GCPs which lack environmental information, VCPs directly visualize the scattering geometry, offering contextual understanding of signal propagation. By tracking scatterers and reflection points in VCPs, LMM-BM can effectively predict discontinuous signal reflections and determine beam directions. Furthermore, LMM-BM enables environment-aware, proactive beam management by leveraging semantic features (e.g., boundaries of scatterers) in VCPs to interpret the interactions between signals and the surrounding environment.

As a main engine for the task at hand, we use LMM, a generative AI model specialized in capturing the underlying probability distribution of sequential data and generating subsequent elements [37]. Specifically, LMM extracts VCPs from the sequential pilot and sensing measurements and then predicts future VCPs using previously extracted parameters and measurements. The core component of LMM for processing sequential multimodal data is the Transformer architecture, which utilizes the attention mechanism to quantify correlations across all input elements [38]. By capturing both intra-modal correlations within time-sequential measurements and cross-modal correlations between pilot and sensing measurements, LMM learns how visual features (e.g., positions of scatterers) in sensing measurements correspond to numerical features (e.g., angles, delays, and path gains) in pilot measurements, as well as visual features in previous sensing measurements, thereby facilitating effective tracking of VCPs. Also, leveraging its vast network size and extensive pre-training, LMM can perform a broad range of tasks either without examples (i.e., zero-shot prompting) or with a smaller number of examples (i.e., few-shot prompting). Moreover, due to its advanced natural language processing (NLP) capabilities, LMM can interpret detailed task instructions to further enhance task performance. This prompt engineering is particularly crucial for communication tasks since they depend heavily on network structure and physical models (e.g., the laws of reflection).

The main contributions of this paper are as follows:

- we propose an environment-aware multimodal beamforming framework that utilizes VCPs, i.e., positions of UE, scatterers, and reflection point;

- we develop a reflection tracking technique that employs LMM to tract VCPs using sequential pilot and sensing measurements; and
- we demonstrate the beam management performance of LMM-BM through extensive simulations conducted on the NVIDIA Sionna™ platform.

The rest of this article is organized as follows: Section II explains sensing-aided mmWave/THz systems. Sections III proposes LMM-BM. Section IV presents the numerical results. Section V concludes the paper.

Notations: Random variables are displayed in sans serif, upright fonts; their realizations in serif, italic fonts. Vectors and matrices are denoted by bold lowercase and uppercase letters, respectively. For example, a random variable and its realization are denoted by x and x for scalars, $\mathbf{x}$ and $\boldsymbol{x}$ for vectors, and $\mathbf{X}$ and $\boldsymbol{X}$ for matrices. Sets and random sets are denoted by upright sans serif and calligraphic font, respectively. For example, a random set and its realization are denoted by X and $\mathcal{X}$, respectively. The m-by-n matrix of zeros is denoted by $\mathbf{0}_{m\times n}$; when $n=1$, the m-dimensional vector of zeros is simply denoted by $\mathbf{0}_m$. The m-by-m identity matrix is denoted by $\boldsymbol{I}_m$. The operators $\mathrm{tr}(\boldsymbol{X})$, $\|\boldsymbol{x}\|_2$, and $\|\boldsymbol{X}\|_{\mathrm{F}}$ denote the trace, the Euclidean norm, and the Frobenius norm, respectively. The transpose, conjugate, and conjugate transpose of $\boldsymbol{X}$ are denoted by $(\cdot)^{\mathrm{T}}$, $(\cdot)^*$, and $(\cdot)^{\mathrm{H}}$, respectively. The operations $\otimes$ and $*$ denote the Kronecker product and the column-wise Khatri-Rao product, respectively. The vectorization of $\boldsymbol{X}$ is denoted by $\mathrm{vec}(\boldsymbol{X})$. The notation $\mathrm{diag}(\cdot)$ represents a diagonal matrix with the arguments being its diagonal elements. The nth element of $\boldsymbol{x}$ is denoted by $[\boldsymbol{x}]_n$. The notation $\boldsymbol{X}_{\mathcal{S}}$ represents a matrix whose ith column is the $\mathcal{S}[i]$th column of $\boldsymbol{X}$.

II. Sensing-Aided mmWave/THz Systems

In this section, we introduce sensing-aided mmWave/THz system and geometry-based channel model. We then provide an overview of conventional beam management schemes.

A. Sensing-Aided mmWave/THz System Model

We consider a mmWave/THz multiple-input multiple-output (MIMO) system where a BS equipped with $N_{\mathrm{bs}}=N_{\mathrm{bs,h}}\times N_{\mathrm{bs,v}}$ uniform rectangular array (URA) antennas serves a UE equipped with $N_{\mathrm{ue}}=N_{\mathrm{ue,h}}\times N_{\mathrm{ue,v}}$ URA antennas. To identify the wireless environments, a circular array of N_{cam} red-green-blue-depth (RGB-D) cameras, each covering a specific angular sector, is mounted on the BS.[1] Also, the BS is equipped with an LMM agent that processes both sensing and pilot measurements to extract environmental features and determine beamforming vectors. The BS and UE employ analog beamforming architectures comprising N_{bs} and N_{ue} analog phase shifters, respectively. We adopt a global coordinate system (GCS) where the position vectors of the BS and its nth antenna are $\boldsymbol{p}_{\mathrm{bs}}\in\mathbb{R}^3$ and $\boldsymbol{p}_{\mathrm{bs},n}\in\mathbb{R}^3$, respectively.[2] Also, the position vectors of the UE and its mth antenna at the tth time slot (i.e., channel coherence block) are $\boldsymbol{p}_{\mathrm{ue}}^{(t)}\in\mathbb{R}^3$ and $\boldsymbol{p}_{\mathrm{ue},m}^{(t)}\in\mathbb{R}^3$, respectively. The BS antenna orientation is defined by three angles: the bearing angle α_{bs}, the downtilt angle β_{bs}, and the slant angle γ_{bs}, which correspond to rotations about the z-, y-, and x-axes, respectively [39]. Similarly, the UE antenna orientation is specified as $(\alpha_{\mathrm{ue}},\beta_{\mathrm{ue}},\gamma_{\mathrm{ue}})$. We consider an orthogonal frequency-division multiplexing (OFDM) system with S subcarriers operating at a carrier frequency of f_{c} and a system bandwidth of B where the frequency of the sth subcarrier is $f_s=f_{\mathrm{c}}+\left(s-\frac{S}{2}\right)\Delta f$ for $s=0,1,\ldots,S-1$ where $\Delta f\triangleq\frac{B}{S}$ is the subcarrier spacing.

We assume a time-division duplexing (TDD) system where each channel coherence block is divided into two phases: 1) uplink training phase; and 2) downlink data transmission phase. In the uplink training phase, the UE transmits uplink pilot signals (i.e., SRS), and the BS determines beam directions based on the historical pilot and sensing measurements. The BS also predicts beam directions for the subsequent uplink training. In the downlink data transmission phase, by exploiting the channel reciprocity between uplink and downlink channels, the BS and UE transmit and receive data symbols using the determined beam directions.[3]

1) Uplink Training: The uplink pilot transmission is scheduled over P OFDM symbols and Q pilot subcarriers in each channel coherence block. The sets of OFDM symbol and pilot subcarrier indices defined as $\mathcal{K}\triangleq\{k_1,k_2,\ldots,k_P\}$ and $\mathcal{S}\triangleq\{s_1,s_2,\ldots,s_Q\}$, respectively. Specifically, for each kth symbol of the sth subcarrier, the UE transmits the uplink pilot symbol $x_{\mathrm{ue},k}[s]=\sqrt{P_{\mathrm{ue}}}$ using the beamforming vector $\boldsymbol{f}_{\mathrm{ue},k}^{(t)}\in\mathbb{C}^{N_{\mathrm{ue}}}$ where P_{ue} is the per subcarrier transmit power of UE. The received pilot signal $\mathbf{r}_{\mathrm{bs},k}^{(t)}[s]\in\mathbb{C}^{N_{\mathrm{bs}}}$ of the BS at the sth subcarrier and the kth symbol is expressed as

$$\mathbf{r}_{\mathrm{bs},k}^{(t)}[s]=\sqrt{P_{\mathrm{ue}}}\,\boldsymbol{H}^{(t)}[s]\,\boldsymbol{f}_{\mathrm{ue},k}^{(t)}+\mathbf{n}_{\mathrm{bs},k}^{(t)}[s] \tag{1}$$

where $\boldsymbol{H}^{(t)}[s]\in\mathbb{C}^{N_{\mathrm{bs}}\times N_{\mathrm{ue}}}$ is the uplink channel matrix and $\mathbf{n}_{\mathrm{bs},k}^{(t)}[s]\sim\mathcal{CN}(\mathbf{0}_{N_{\mathrm{bs}}},\sigma_{\mathrm{n}}^2\boldsymbol{I}_{N_{\mathrm{bs}}})$ is the additive Gaussian noise at the tth channel coherence block. Then, the BS combines $\mathbf{r}_{\mathrm{bs},k}^{(t)}[s]$ using the BS combining vector $\boldsymbol{w}_{\mathrm{bs},k}^{(t)}\in\mathbb{C}^{N_{\mathrm{bs}}}$ to obtain the post-processed signal $\mathsf{y}_{\mathrm{bs},k}^{(t)}[s]\triangleq(\boldsymbol{w}_{\mathrm{bs},k}^{(t)})^{\mathrm{H}}\mathbf{r}_{\mathrm{bs},k}^{(t)}[s]\in\mathbb{C}$:

$$\mathsf{y}_{\mathrm{bs},k}^{(t)}[s]=\sqrt{P_{\mathrm{ue}}}\left(\boldsymbol{w}_{\mathrm{bs},k}^{(t)}\right)^{\mathrm{H}}\boldsymbol{H}^{(t)}[s]\,\boldsymbol{f}_{\mathrm{ue},k}^{(t)}+\mathsf{z}_{\mathrm{bs},k}^{(t)}[s] \tag{2}$$

$$\overset{(a)}{=}\sqrt{P_{\mathrm{ue}}}\left(\left(\boldsymbol{f}_{\mathrm{ue},k}^{(t)}\right)^*\otimes\boldsymbol{w}_{\mathrm{bs},k}^{(t)}\right)^{\mathrm{H}}\mathrm{vec}\left(\boldsymbol{H}^{(t)}[s]\right)+\mathsf{z}_{\mathrm{bs},k}^{(t)}[s] \tag{3}$$

[1]For example, the Intel RealSense L515 RGB-D camera provides the horizontal and vertical field of views (FoVs) of 70° and 45°, respectively. Since angular spreads of azimuth and elevation angles in mmWave/THz bands are approximately 10–15° and 7–11°, respectively, each RGB-D camera can effectively capture the multipath propagation of the pilot signals [39].

[2]Note that the BS antenna located at the xth row and yth column of the antenna array is referred to as the nth BS antenna where $n=(x-1)\,N_{\mathrm{bs,v}}+y$. Also, the UE antenna indices are determined similarly.

[3]Note that the proposed scheme can be readily extended to frequency-division duplexing (FDD) system. Although full channel reciprocity does not hold in FDD systems, partial channel reciprocity (also referred to as angular reciprocity) ensures that the propagation paths of uplink and downlink channels remain fairly similar [40]. Based on this partial channel reciprocity, the BS can acquire downlink channel state information (CSI) by exchanging pilot symbols (i.e., CSI-RS) and CSI report over the determined beam directions.

where $\mathsf{z}_{\mathrm{bs},k}^{(t)}[s] \triangleq \big(\boldsymbol{w}_{\mathrm{bs},k}^{(t)}\big)^{\mathrm{H}}\mathbf{n}_{\mathrm{bs},k}^{(t)}[s] \in \mathbb{C}$ and (a) is from the property that $\boldsymbol{a}^{\mathrm{H}}\boldsymbol{X}\boldsymbol{b} = (\boldsymbol{b}^{*} \otimes \boldsymbol{a})^{\mathrm{H}}\mathrm{vec}(\boldsymbol{X})$.[1] Also, the post-processed signal vector $\mathbf{y}_{\mathrm{bs}}^{(t)}[s] \in \mathbb{C}^{P}$ is expressed as

$$\mathbf{y}_{\mathrm{bs}}^{(t)}[s] \triangleq \big[\, \mathsf{y}_{\mathrm{bs},k_1}^{(t)}[s]\ \mathsf{y}_{\mathrm{bs},k_2}^{(t)}[s] \cdots \mathsf{y}_{\mathrm{bs},k_P}^{(t)}[s] \,\big]^{\mathrm{T}} \tag{4}$$

$$= \sqrt{P_{\mathrm{ue}}}\left(\big(\boldsymbol{F}_{\mathrm{ue}}^{(t)}\big)^{*} * \boldsymbol{W}_{\mathrm{bs}}^{(t)}\right)^{\mathrm{H}}\mathrm{vec}\big(\boldsymbol{H}^{(t)}[s]\big) + \mathbf{z}_{\mathrm{bs}}^{(t)}[s] \tag{5}$$

where $\boldsymbol{F}_{\mathrm{ue}}^{(t)} \triangleq \big[\, \boldsymbol{f}_{\mathrm{ue},k_1}^{(t)}\, \boldsymbol{f}_{\mathrm{ue},k_2}^{(t)} \cdots \boldsymbol{f}_{\mathrm{ue},k_P}^{(t)} \,\big] \in \mathbb{C}^{N_{\mathrm{ue}} \times P}$ is the UE beamforming matrix, $\boldsymbol{W}_{\mathrm{bs}}^{(t)} \triangleq \big[\, \boldsymbol{w}_{\mathrm{bs},k_1}^{(t)}\, \boldsymbol{w}_{\mathrm{bs},k_2}^{(t)} \cdots \boldsymbol{w}_{\mathrm{bs},k_P}^{(t)} \,\big] \in \mathbb{C}^{N_{\mathrm{bs}} \times P}$ is the BS combining matrix, and $\mathbf{z}_{\mathrm{bs}}^{(t)}[s] \triangleq \big[\, \mathsf{z}_{\mathrm{bs},k_1}^{(t)}[s]\, \mathsf{z}_{\mathrm{bs},k_2}^{(t)}[s] \cdots \mathsf{z}_{\mathrm{bs},k_P}^{(t)}[s] \,\big]^{\mathrm{T}} \in \mathbb{C}^{P}$ is the noise vector. Finally, by combining $\mathbf{y}_{\mathrm{bs}}^{(t)}[s]$ for all $s \in \mathcal{S}$, we have the pilot measurement matrix $\sum^{R} \boldsymbol{\Upsilon}^{(t)} \in \mathbb{C}^{Q \times P}$ as

$$\boldsymbol{\Upsilon}^{(t)} \triangleq \big[\, \mathbf{y}_{\mathrm{bs}}^{(t)}[s_1]\ \mathbf{y}_{\mathrm{bs}}^{(t)}[s_2] \cdots \mathbf{y}_{\mathrm{bs}}^{(t)}[s_Q] \,\big] \tag{6}$$

$$= \sqrt{P_{\mathrm{ue}}}\left(\big(\boldsymbol{F}_{\mathrm{ue}}^{(t)}\big)^{*} * \boldsymbol{W}_{\mathrm{bs}}^{(t)}\right)^{\mathrm{H}}\big[\, \boldsymbol{H}^{(t)} \,\big]_{\mathcal{S}} + \mathbf{Z}^{(t)} \tag{7}$$

where $\boldsymbol{H}^{(t)} \in \mathbb{C}^{N_{\mathrm{bs}}N_{\mathrm{ue}} \times S}$ is the combined channel matrix defined as

$$\boldsymbol{H}^{(t)} \triangleq \big[\, \mathrm{vec}(\boldsymbol{H}^{(t)}[1])\ \mathrm{vec}(\boldsymbol{H}^{(t)}[2]) \cdots \mathrm{vec}(\boldsymbol{H}^{(t)}[S]) \,\big]. \tag{8}$$

Also, $\mathbf{Z}^{(t)} \triangleq \big[\, \mathbf{z}_{\mathrm{bs}}^{(t)}[s_1]\ \mathbf{z}_{\mathrm{bs}}^{(t)}[s_2] \cdots \mathbf{z}_{\mathrm{bs}}^{(t)}[s_Q] \,\big] \in \mathbb{C}^{Q \times P}$.

Along with the pilot measurements, the BS exploits sensing measurements collected from the RGB-D camera to gather contextual information about the surrounding environments. The RGB-D image capture is synchronized with the uplink training phase. One major issue in acquiring sensing measurements is determining the appropriate sensing direction (or picking the correct camera from the array) to accurately capture the signal propagation. As a remedy, we exploit the SSB beam index, which provides a coarse estimate of the UE direction. Specifically, when the RGB-D image capture is triggered, the BS activates the camera covering the angular region designated by the reported SSB beam index. Specifically, the $W \times H$ pixels RGB-D image captured at the tth channel coherence block is represented as a three-dimensional tensor $\mathcal{D}^{(t)} \in \mathbb{R}^{H \times W \times 4}$ comprising the RGB data $\mathcal{D}_{\mathrm{rgb}}^{(t)} \in \mathbb{R}^{H \times W \times 3}$ and the depth data $\mathcal{D}_{\mathrm{d}}^{(t)} \in \mathbb{R}^{H \times W \times 1}$:

$$\mathcal{D}^{(t)} = \big[\, \mathcal{D}_{\mathrm{rgb}}^{(t)};\ \mathcal{D}_{\mathrm{d}}^{(t)} \,\big] \tag{9}$$

where $[\cdot\,;\,\cdot]$ denotes the concatenation along the third dimension.

2) Downlink Data Transmission: Once the beam management process is completed, the BS transmits downlink data symbols using the determined beamforming vector. Then, the UE receives the transmitted symbols by applying the corresponding combining vector. Specifically, the post-processed signal $\mathsf{y}_{\mathrm{ue},k}^{(t)}[s] \in \mathbb{C}$ at the kth symbol and sth subcarrier at time slot t is expressed as

$$\mathsf{y}_{\mathrm{ue},k}^{(t)}[s] = \sqrt{P_{\mathrm{bs}}}\,\big(\boldsymbol{w}_{\mathrm{ue}}^{(t)}\big)^{\mathrm{H}}\big(\boldsymbol{H}^{(t)}[s]\big)^{\mathrm{H}}\boldsymbol{f}_{\mathrm{bs}}^{(t)}x_{\mathrm{bs},k}^{(t)}[s] + \mathsf{z}_{\mathrm{ue},k}^{(t)}[s] \tag{10}$$

where P_{bs} is the BS transmit power per subcarrier, $\boldsymbol{f}_{\mathrm{bs}}^{(t)} \in \mathbb{C}^{N_{\mathrm{bs}}}$ and $\boldsymbol{w}_{\mathrm{ue}}^{(t)} \in \mathbb{C}^{N_{\mathrm{ue}}}$ are the BS beamforming and UE combining vectors, respectively, $x_{\mathrm{bs},k}^{(t)}[s] \in \mathbb{C}$ is the data symbol, and $\mathsf{z}_{\mathrm{ue},k}^{(t)}[s] \sim \mathcal{CN}(0, \sigma_{\mathrm{n}}^2)$ is the additive Gaussian noise. Then, the average downlink data rate $R^{(t)}$ of the UE at time slot t is given by

$$R^{(t)} = \frac{1}{S}\sum_{s=1}^{S}\log_2\left(1 + \frac{P_{\mathrm{bs}}}{\sigma_{\mathrm{n}}^2}\left|\big(\boldsymbol{w}_{\mathrm{ue}}^{(t)}\big)^{\mathrm{H}}\big(\boldsymbol{H}^{(t)}[s]\big)^{\mathrm{H}}\boldsymbol{f}_{\mathrm{bs}}^{(t)}\right|^2\right). \tag{11}$$

B. Geometry-Based MIMO Channel Model

In mmWave/THz bands, signal power is focused mostly onto the LOS path and a few NLOS paths due to the high directivity and severe path losses associated with propagation, reflection, and refraction. For instance, the numbers of effective paths in FR2 and sub-THz bands are less than 6 [41], [42]. The scattering geometry is primarily determined by dominant scatterers (e.g., building) in the vicinity of the BS. Based on this sparse scattering property, we adopt the geometric block-fading frequency-selective MIMO channel model consisting of L discrete rays, where each ray is characterized by its zenith angle of arrival (ZOA) and azimuth angle of arrival (AOA) (θ, ϕ), zenith angle of departure (ZOD) and azimuth angle of departure (AOD) (ϑ, φ), delay τ, and path gain β [15], [39].

Specifically, the uplink channel frequency response (CFR) $h_{m,n}^{(t)}[s] \in \mathbb{C}$ from the mth UE antenna to the nth BS antenna for the sth subcarrier at time slot t is expressed as a sum of L multipath components as

$$h_{m,n}^{(t)}[s] = \sum_{l=1}^{L}\beta_l^{(t)}[s]\, e^{-j2\pi s\Delta f\tau_l^{(t)}}e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\mathrm{T}}(\vartheta_l^{(t)},\varphi_l^{(t)})\,\boldsymbol{p}_{\mathrm{ue},m}^{(t)}} \times e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\mathrm{T}}(\theta_l^{(t)},\phi_l^{(t)})\,\boldsymbol{p}_{\mathrm{bs},n}} \tag{12}$$

where $\lambda = \frac{c}{f_{\mathrm{c}}}$ is the signal wavelength with c being the speed of light, $\big(\theta_l^{(t)}, \phi_l^{(t)}\big)$ are the ZOA and AOA, $\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big)$ is the ZOD and AOD, $\tau_l^{(t)} \in [0, \tau_{\max}]$ is the delay with $\tau_{\max}$ being the delay spread, and $\beta_l^{(t)}[s]$ is the path gain of the lth path for the sth subcarrier at time slot t. Also, $\hat{\boldsymbol{r}}(\theta, \phi) \in \mathbb{R}^3$ is the spherical unit vector defined as

$$\hat{\boldsymbol{r}}(\theta, \phi) \triangleq [\, \sin\theta\cos\phi \quad \sin\theta\sin\phi \quad \cos\theta \,]^{\mathrm{T}}. \tag{13}$$

The path gain $\beta_l^{(t)}[s]$, which consists of four components, i.e., free-space path loss, antenna radiation pattern, reflection loss, and molecular absorption, is expressed as [43]

$$\beta_l^{(t)}[s] = \frac{1}{4\pi f_s\tau_l^{(t)}}F_{\mathrm{bs}}\big(\theta_l^{(t)}, \phi_l^{(t)}\big)\,F_{\mathrm{ue}}\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big) \times \big|\Gamma_l^{(t)}(f_s)\big|e^{-\frac{1}{2}k_{\mathrm{abs}}(f_s)\,c\tau_l^{(t)}} \tag{14}$$

where $F_{\mathrm{bs}}\big(\theta_l^{(t)}, \phi_l^{(t)}\big)$ and $F_{\mathrm{ue}}\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big)$ are the BS and UE antenna radiation patterns [39]. Also, $k_{\mathrm{abs}}(f_s)$ is the molecular absorption coefficient, and $\Gamma_l^{(t)}(f_s)$ is the reflection coefficient given by [44]

$$\Gamma_l^{(t)}(f_s) \triangleq \frac{\cos\psi_l^{(t)} - \sqrt{\big(\eta_l^{(t)}\big)^2 - \sin^2\psi_l^{(t)}}}{\cos\psi_l^{(t)} + \sqrt{\big(\eta_l^{(t)}\big)^2 - \sin^2\psi_l^{(t)}}}e^{-\frac{\left(4\pi f_s\sigma_l^{(t)}\cos\psi_l^{(t)}\right)^2}{2c^2}} \tag{15}$$

where ψ_l is the incidence angle, η_l is the refractive index of the surface material, and σ_l is the standard deviation of the surface roughness [45]. If the lth path correspond to the LOS path, then $\Gamma_l(f) = 1$. One can see that the path gain decreases significantly with the signal frequency. As a result, diffused and diffracted rays undergo considerable attenuation in mmWave/THz bands. For this reason, we focus on single-bounce specularly reflected rays for modeling the NLOS paths.

Finally, by combining the CFR for all BS and UE antennas, we obtain the uplink channel matrix $\boldsymbol{H}^{(t)}[s] \in \mathbb{C}^{N_{\rm bs}\times N_{\rm ue}}$ as

$$\boldsymbol{H}^{(t)}[s] = \sum_{l=1}^{L} \beta_l^{(t)}[s]\, e^{-j2\pi s \Delta f \tau_l^{(t)}} \boldsymbol{a}_{\rm bs}\big(\theta_l^{(t)}, \phi_l^{(t)}\big) \times \big(\boldsymbol{a}_{\rm ue}^{(t)}\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big)\big)^{\rm T} \tag{16}$$

where $\boldsymbol{a}_{\rm bs}\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big) \in \mathbb{C}^{N_{\rm bs}}$ and $\boldsymbol{a}_{\rm ue}^{(t)}\big(\theta_l^{(t)}, \phi_l^{(t)}\big) \in \mathbb{C}^{N_{\rm ue}}$ are the BS and UE array response vectors defined as

$$\boldsymbol{a}_{\rm bs}\big(\theta_l^{(t)}, \phi_l^{(t)}\big) \triangleq \Big[\, e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\theta_l^{(t)}, \phi_l^{(t)})\boldsymbol{p}_{{\rm bs},1}} \; e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\theta_l^{(t)}, \phi_l^{(t)})\boldsymbol{p}_{{\rm bs},2}} \cdots e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\theta_l^{(t)}, \phi_l^{(t)})\boldsymbol{p}_{{\rm bs},N}} \,\Big]^{\rm T} \tag{17}$$

$$\boldsymbol{a}_{\rm ue}^{(t)}\big(\vartheta_l^{(t)}, \varphi_l^{(t)}\big) \triangleq \Big[\, e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\vartheta_l^{(t)}, \varphi_l^{(t)})\boldsymbol{p}_{{\rm ue},1}^{(t)}} \; e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\vartheta_l^{(t)}, \varphi_l^{(t)})\boldsymbol{p}_{{\rm ue},2}^{(t)}} \cdots e^{j\frac{2\pi}{\lambda}\hat{\boldsymbol{r}}^{\rm T}(\vartheta_l^{(t)}, \varphi_l^{(t)})\boldsymbol{p}_{{\rm ue},M}^{(t)}} \,\Big]^{\rm T}. \tag{18}$$

Note that $\boldsymbol{H}^{(t)}$ is a function of $\boldsymbol{\theta}^{(t)} \triangleq \big[\theta_1^{(t)}\, \theta_2^{(t)} \cdots \theta_L^{(t)}\big]^{\rm T}$, $\boldsymbol{\phi}^{(t)} \triangleq \big[\phi_1^{(t)}\, \phi_2^{(t)} \cdots \phi_L^{(t)}\big]^{\rm T}$, $\boldsymbol{\vartheta}^{(t)} \triangleq \big[\vartheta_1^{(t)}\, \vartheta_2^{(t)} \cdots \vartheta_L^{(t)}\big]^{\rm T}$, $\boldsymbol{\varphi}^{(t)} \triangleq \big[\varphi_1^{(t)}\, \varphi_2^{(t)} \cdots \varphi_L^{(t)}\big]^{\rm T}$, $\boldsymbol{\tau}^{(t)} \triangleq \big[\tau_1^{(t)}\, \tau_2^{(t)} \cdots \tau_L^{(t)}\big]^{\rm T}$, and $\boldsymbol{B}^{(t)} \triangleq \big[\boldsymbol{\beta}^{(t)}[1]\; \boldsymbol{\beta}^{(t)}[2] \cdots \boldsymbol{\beta}^{(t)}[S]\big]$ where $\boldsymbol{\beta}^{(t)}[s] \triangleq \big[\beta_1^{(t)}[s]\, \beta_2^{(t)}[s] \cdots \beta_L^{(t)}[s]\big]^{\rm T}$. These parameters are collectively referred to as the GCPs $\mathcal{G}^{(t)} \subseteq \mathbb{R}^{(S+5)L}$ as

$$\mathcal{G}^{(t)} \triangleq \big\{\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)}, \boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}, \boldsymbol{\tau}^{(t)}, \boldsymbol{B}^{(t)}\big\}. \tag{19}$$

Lemma 1: The channel matrix $\boldsymbol{H}^{(t)}$ is expressed as a function of the GCPs $\mathcal{G}^{(t)}$ via the geometric map $f_{\rm geo}^{(t)} : \mathbb{R}^{(S+5)L} \to \mathbb{C}^{N_{\rm bs}N_{\rm ue}\times S}$:

$$\boldsymbol{H}^{(t)} = f_{\rm geo}^{(t)}\big(\mathcal{G}^{(t)}\big) \tag{20}$$

$$= \Big(\boldsymbol{A}_{\rm ue}^{(t)}\big(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}\big) * \boldsymbol{A}_{\rm bs}\big(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)}\big)\Big) \times \Big(\boldsymbol{B}^{(t)} \odot \boldsymbol{E}\big(\boldsymbol{\tau}^{(t)}\big)\Big) \tag{21}$$

where $\boldsymbol{A}_{\rm bs}\big(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)}\big) \in \mathbb{C}^{N_{\rm bs}\times L}$ and $\boldsymbol{A}_{\rm ue}^{(t)}\big(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}\big) \in \mathbb{C}^{N_{\rm ue}\times L}$ are the BS and UE array response matrices, respectively, defined as

$$\boldsymbol{A}_{\rm bs}\big(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)}\big) \triangleq \Big[\, \boldsymbol{a}_{\rm bs}\big(\theta_1^{(t)}, \phi_1^{(t)}\big) \;\; \boldsymbol{a}_{\rm bs}\big(\theta_2^{(t)}, \phi_2^{(t)}\big) \cdots \boldsymbol{a}_{\rm bs}\big(\theta_L^{(t)}, \phi_L^{(t)}\big) \,\Big] \tag{22}$$

$$\boldsymbol{A}_{\rm ue}^{(t)}\big(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}\big) \triangleq \Big[\, \boldsymbol{a}_{\rm ue}^{(t)}\big(\vartheta_1^{(t)}, \varphi_1^{(t)}\big) \; \boldsymbol{a}_{\rm ue}^{(t)}\big(\vartheta_2^{(t)}, \varphi_2^{(t)}\big) \cdots \boldsymbol{a}_{\rm ue}^{(t)}\big(\vartheta_L^{(t)}, \varphi_L^{(t)}\big) \,\Big]. \tag{23}$$

Also, $\boldsymbol{E}\big(\boldsymbol{\tau}^{(t)}\big)$ is a $L \times S$-dimensional matrix whose (l,s)th element is $[\,\boldsymbol{E}\big(\boldsymbol{\tau}^{(t)}\big)\,]_{l,s} = e^{-j2\pi(s-1)\,\Delta f \tau_l^{(t)}}$. □

Proof: See Appendix A. ⊠

C. Conventional GCP-Based Beam Management

As shown in Lemma 1, the beam directions aligned with the signal propagation paths can be derived from GCPs. Based on this observation, conventional beam tracking techniques determine the beam directions by recursively updating GCPs using Kalman filter or deep learning (DL) techniques [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28]. For example, in the EKF-based approaches, GCPs are modeled as random variables, i.e., $\mathsf{G}^{(t)} = \big\{\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)}, \boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}, \boldsymbol{\tau}^{(t)}, \mathbf{B}^{(t)}\big\}$, and then the state vector $\mathbf{s}^{(t)} \in \mathbb{R}^{(S+5)L}$ representing $\mathsf{G}^{(t)}$ is defined as [18]

$$\mathbf{s}^{(t)} \triangleq \mathrm{vec}\big(\big[\,\boldsymbol{\theta}^{(t)}\; \boldsymbol{\phi}^{(t)}\; \boldsymbol{\vartheta}^{(t)}\; \boldsymbol{\varphi}^{(t)}\; \boldsymbol{\tau}^{(t)}\; \mathbf{B}^{(t)}\,\big]\big). \tag{24}$$

Then, the discrete state evolution model is formulated as

$$\mathbf{s}^{(t)} = \boldsymbol{F}\mathbf{s}^{(t-1)} + \mathbf{u}^{(t)} \tag{25}$$

where $\boldsymbol{F} \in \mathbb{C}^{(S+5)L\times(S+5)L}$ is the state transition matrix and $\mathbf{u}^{(t)} \sim \mathcal{CN}(\mathbf{0}_{(S+5)L}, \boldsymbol{\Sigma}_{\mathbf{u}})$ is the process noise vector. At each time slot t, the estimated state vector $\hat{\mathbf{s}}^{(t\,|\,t-1)} \in \mathbb{R}^{(S+5)L}$ and its error covariance matrix $\boldsymbol{P}^{(t\,|\,t-1)} \in \mathbb{R}^{(S+5)L\times(S+5)L}$ are first predicted using the state evolution model (25) as

$$\hat{\mathbf{s}}^{(t\,|\,t-1)} = \boldsymbol{F}\hat{\mathbf{s}}^{(t-1\,|\,t-1)} + \mathbf{u}^{(t)} \tag{26}$$

$$\boldsymbol{P}^{(t\,|\,t-1)} = \boldsymbol{F}\boldsymbol{P}^{(t-1\,|\,t-1)}\boldsymbol{F}^{\rm H} + \boldsymbol{\Sigma}_{\mathbf{u}}. \tag{27}$$

In the DL-based approaches, RNN architectures such as LSTM are employed for the prediction [26]:

$$\big(\hat{\mathbf{s}}^{(t\,|\,t-1)}, \boldsymbol{P}^{(t\,|\,t-1)}\big) = f_{\rm lstm}\Big(\big\{\hat{\mathbf{s}}^{(t-t'\,|\,t-t')}\big\}_{t'=1}^{T}; \boldsymbol{\eta}\Big) \tag{28}$$

where T is the sequence length and $\boldsymbol{\eta}$ is the network parameters. Then, the predicted values are updated using the pilot measurements $\mathbf{y}^{(t)} \triangleq \big[\,\Re^{\rm T}\big\{\mathrm{vec}\big(\mathbf{Y}^{(t)}\big)\big\}\; \Im^{\rm T}\big\{\mathrm{vec}\big(\mathbf{Y}^{(t)}\big)\big\}\,\big]^{\rm T}$ as

$$\hat{\mathbf{s}}^{(t\,|\,t)} = \hat{\mathbf{s}}^{(t\,|\,t-1)} + \boldsymbol{K}^{(t)}\big(\mathbf{y}^{(t)} - \tilde{\mathbf{y}}\big(\hat{\mathbf{s}}^{(t\,|\,t-1)}\big)\big) \tag{29}$$

$$\boldsymbol{P}^{(t\,|\,t)} = \big(\boldsymbol{I}_{(S+5)L} - \boldsymbol{K}^{(t)}\boldsymbol{J}^{(t)}\big)\,\boldsymbol{P}^{(t\,|\,t-1)} \tag{30}$$

where $\boldsymbol{K}^{(t)} = \boldsymbol{P}^{(t\,|\,t-1)}\big(\boldsymbol{J}^{(t)}\big)^{\rm H}\big(\boldsymbol{J}^{(t)}\boldsymbol{P}^{(t\,|\,t-1)}\big(\boldsymbol{J}^{(t)}\big)^{\rm H} + \boldsymbol{\Sigma}_{\mathbf{Y}}\big)^{-1}$ is the Kalman gain, $\tilde{\mathbf{y}}(\mathbf{s}^{(t)})$ is the measurement function, $\boldsymbol{J}^{(t)} \triangleq \nabla_{\mathbf{s}}\tilde{\mathbf{y}}\,|_{\mathbf{s}=\hat{\mathbf{s}}^{(t\,|\,t-1)}}$, and $\boldsymbol{\Sigma}_{\mathbf{y}}$ is the measurement noise covariance matrix.

One major limitation of the conventional GCP-based beam tracking techniques is, since they depend on continuous state evolution models, they cannot effectively handle abrupt changes in GCP. In practice, due to the scattered and irregular positioning of scatterers, combined with dynamically varying wireless environments, GCPs exhibit discontinuous variations over time. To address the piecewise continuity of GCPs, leveraging contextual information about surrounding environments is essential. In this context, approaches utilizing sensing information (e.g., images and radar signals) have been proposed recently [30], [31], [32], [33], [34]. By exploiting the environmental information associated with the LOS component, the conventional sensing-aided beam tracking techniques can effectively track UE trajectory and predict blockage. However, they overlook environmental information related to scatterers in the NLOS paths. As a result, they fall short in simultaneously tracking multipath components, leading to significant performance degradation when the LOS path is unavailable.

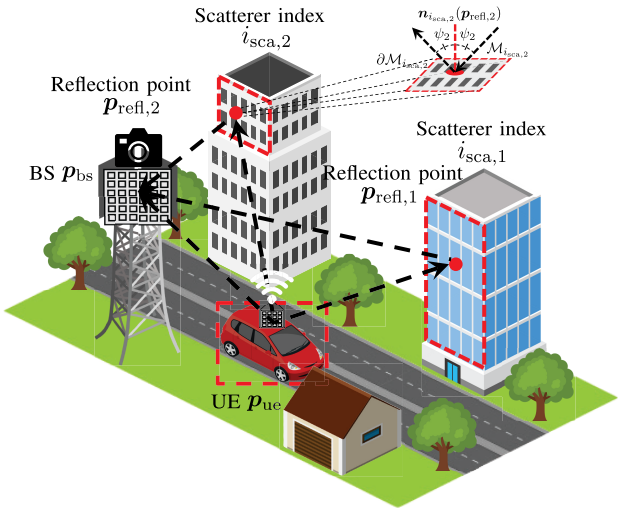


Fig. 1. Visualization of signal propagation using VCPs.

III. Large Multimodal Model-Based Environment-Aware Beam Management

The primary objectives of the proposed LMM-BM are to identify signal propagation environments and determine the proper beam directions. To achieve these goals, we exploit VCPs, i.e., the reflection points and scatterer indices, extracted from pilot and sensing measurements (see Fig. 1). In contrast to GCPs which analytically model the wireless channel, VCPs provide a visual representation of signal propagation as well as scatterer dynamics. By utilizing VCPs, we can track the wireless channel and determine accurate beam directions. Furthermore, by integrating VCPs with environmental information (e.g., positions and sizes of scatterers) obtained from sensing measurements (e.g., RGB-D images), we can analyze the physical interactions between the signal and the surrounding environments, thereby facilitating the prediction of abrupt changes in propagation environments (e.g., changes of scatterers and transient blockages).

A key ingredient of LMM-BM is the *reflection tracking* technique that predicts and refines VCPs from pilot and sensing measurements using LMM. LMM is a generative AI model specialized in extracting correlated features across multimodal data and generating subsequent data. By leveraging cross-modal correlations between pilot and sensing measurements, the LMM identifies the intrinsic relationship between GCPs in pilot measurements and VCPs in sensing measurements, thereby facilitating the extraction of VCPs and environmental information. Also, by analyzing intra-modal correlations within time-sequential measurements, the LMM learns the underlying probability distribution of VCPs and utilizes it to predict future parameters.

In a nutshell, the proposed LMM-BM framework consists of two major stages (see Fig. 2): 1) a prediction stage that predicts the future VCPs from the historical parameters; and 2) an update stage that refines the predicted VCPs by incorporating environmental information from real-time pilot and sensing measurements.

A. Basics of Large Multimodal Model

In this subsection, we outline the general architecture of Transformer-based LMMs, which includes a multimodal encoder, a backbone, and a language modelling (LM) head.

1) Multimodal Encoder: The multimodal encoder transforms diverse input modalities (e.g., texts and images) into unified latent representations. Let I_{mod} be the number of input modalities. For each ith modality, the input is first segmented into a sequence of tokens $\{v_{i,n}\}_{n=1}^{N_i} \subseteq \mathcal{V}$ where $\mathcal{V} = \{1, 2, \ldots, V\}$ is the token vocabulary and N_i is the number of tokens in the ith modality. Then, these tokens are encoded to embedding vectors $\{\boldsymbol{e}_{i,n}\}_{n=1}^{N_i} \subseteq \mathbb{R}^{e_i}$ using a modality-specific encoder $f_{\text{enc},i} : \mathcal{V} \to \mathbb{R}^{e_i}$ as

$$\boldsymbol{e}_{i,n} = f_{\text{enc},i}(v_{i,n})\,. \tag{31}$$

For example, textual data is first tokenized using a subword algorithm (e.g., byte pair encoding (BPE)) and then fed into the Transformer-based text encoder such as bidirectional encoder representations from transformers (BERT). Similarly, visual data like RGB-D images is segmented into smaller, fixed-size patches (e.g., 16×16 pixels) and then processed by encoders such as convolutional neural network (CNN) or vision transformer (ViT).

After the token embedding process, the embedding vectors are projected onto a unified latent space $\mathbb{R}^d$ as

$$\boldsymbol{y}_{i,n} = \boldsymbol{W}_{\text{proj},i}\boldsymbol{e}_{i,n} + \boldsymbol{b}_{\text{proj},i} \tag{32}$$

where $\boldsymbol{y}_{i,n} \in \mathbb{R}^d$ is the latent vector, $\boldsymbol{W}_{\text{proj},i} \in \mathbb{R}^{d\times e_i}$ is the projection matrix, and $\boldsymbol{b}_{\text{proj},i} \in \mathbb{R}^d$ is the bias vector. While the projection is a form of dimensionality reduction, it is a learned, meaningful compression rather than a destructive process. Because these linear layers are trained end-to-end with the entire model, they are optimized to preserve salient, task-relevant features identified by the encoders, while discarding irrelevant noise. Finally, the latent matrices $\boldsymbol{Y}_i = [\,\boldsymbol{y}_{i,1}\,\boldsymbol{y}_{i,2}\,\cdots\,\boldsymbol{y}_{i,N_i}\,]^{\mathrm{T}} \in \mathbb{R}^{N_i\times d}$ of all modalities are concatenated to form the latent representation matrix $\boldsymbol{Y} \in \mathbb{R}^{N_{\text{tok}}\times d}$ where $N_{\text{tok}} \triangleq \sum_{i=1}^{I_{\text{mod}}} N_i$ is the total number of tokens:

$$\boldsymbol{Y} = \left[\,\boldsymbol{Y}_1^{\mathrm{T}}\,\boldsymbol{Y}_2^{\mathrm{T}}\,\cdots\,\boldsymbol{Y}_{I_{\text{mod}}}^{\mathrm{T}}\,\right]^{\mathrm{T}}. \tag{33}$$

2) Backbone: The backbone processes the latent representation matrix $\boldsymbol{Y}$ through sequential transformer blocks to extract both intra- and cross-modal features. Each transformer block consists of multi-head attention layer, layer normalization layer, feed-forward layer, and another layer normalization layer [38], [46]. In the multi-head attention layer, $\boldsymbol{Y}$ is attended by H parallel attention heads which assign specific weights to its elements based on their pairwise correlations. To do so, each hth attention head constructs three different embedding matrices from $\boldsymbol{Y}$: 1) query $\boldsymbol{Q}_h = \boldsymbol{Y}\boldsymbol{W}_{\text{q},h} \in \mathbb{R}^{N_{\text{tok}}\times d_h}$; 2) key $\boldsymbol{K}_h = \boldsymbol{Y}\boldsymbol{W}_{\text{k},h} \in \mathbb{R}^{N_{\text{tok}}\times d_h}$; and 3) value $\boldsymbol{V}_h = \boldsymbol{Y}\boldsymbol{W}_{\text{v},h} \in \mathbb{R}^{N_{\text{tok}}\times d_h}$. Here, $\boldsymbol{W}_{\text{q},h}, \boldsymbol{W}_{\text{k},h}, \boldsymbol{W}_{\text{v},h} \in \mathbb{R}^{d\times d_h}$ are the weight matrices and $d_h = \frac{d}{H}$. The output $\boldsymbol{Y}_{\text{att},h} \in \mathbb{R}^{N_{\text{tok}}\times d_h}$ of the hth attention head is given by

$$\boldsymbol{Y}_{\text{att},h} = f_{\text{smax}}\left(\frac{\boldsymbol{Q}_h\boldsymbol{K}_h^{\mathrm{T}}}{\sqrt{d_h}} + \boldsymbol{M}\right)\boldsymbol{V}_h\,. \tag{34}$$

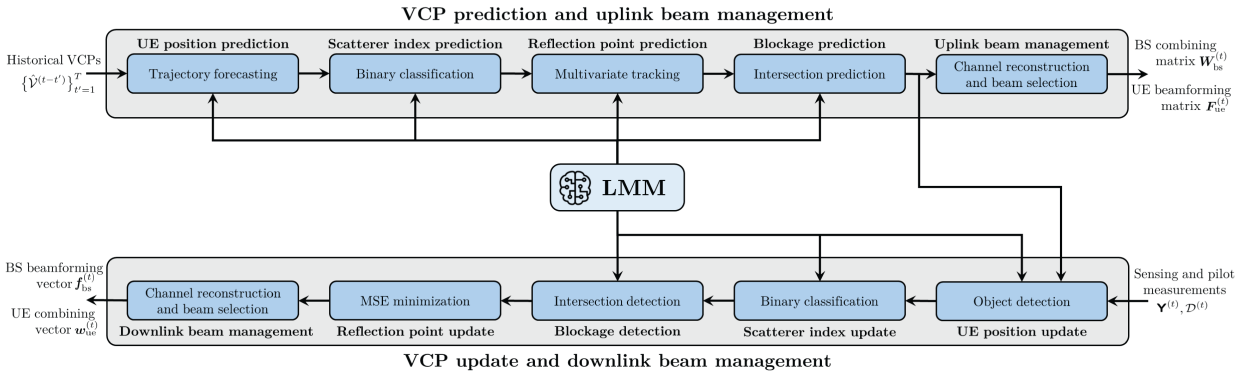


Fig. 2. Block diagram of the proposed LMM-BM framework.

Note that in (34), a scaled dot-product between $\boldsymbol{Q}_h$ and $\boldsymbol{K}_h$ is used to measure the pairwise correlations of the elements of $\boldsymbol{Y}$. Also, $\boldsymbol{M} \in \mathbb{R}^{N_{\rm tok} \times N_{\rm tok}}$ is a mask matrix such that $[\boldsymbol{M}]_{i,j} = 0$ if $i \leqslant j$ and $[\boldsymbol{M}]_{i,j} = -\infty$ otherwise. The mask matrix ensures that only present and past information is used when predicting future tokens. Then, the outputs of all attention heads are combined into $\boldsymbol{Y}_{\rm att} \in \mathbb{R}^{N_{\rm tok} \times d}$ as

$$\boldsymbol{Y}_{\rm att} = [\, \boldsymbol{Y}_{{\rm att},1}\, \boldsymbol{Y}_{{\rm att},2} \cdots \boldsymbol{Y}_{{\rm att},H}\,]\, \boldsymbol{W}_{\rm o} \tag{35}$$

where $\boldsymbol{W}_{\rm o} \in \mathbb{R}^{d \times d}$ is the output weight matrix.

After the multi-head attention layer, the input $\boldsymbol{Y}$ is added to the attention output $\boldsymbol{Y}_{\rm att}$ via a residual connection. Subsequently, each row vector of $\boldsymbol{Y}_{\rm att}+\boldsymbol{Y}$ is normalized by applying layer normalization, which scales and shifts the elements to ensure zero mean and unit variance [38]:

$$\bar{\boldsymbol{Y}} = f_{\rm ln}(\boldsymbol{Y}_{\rm att} + \boldsymbol{Y})\,. \tag{36}$$

Then, $\bar{\boldsymbol{Y}}$ is passed through the feed-forward network (FFN) as

$$\tilde{\boldsymbol{Y}} = f_{\rm ffn}(\bar{\boldsymbol{Y}})\,. \tag{37}$$

Finally, another residual connection and layer normalization is applied to $\tilde{\boldsymbol{Y}}$ to produce the output matrix $\boldsymbol{Z}_1 \in \mathbb{R}^{N_{\rm tok} \times d}$ of the first transformer block:

$$\boldsymbol{Z}_1 = f_{\rm ln}(\bar{\boldsymbol{Y}} + \tilde{\boldsymbol{Y}})\,. \tag{38}$$

The feature matrix $\boldsymbol{Z} \in \mathbb{R}^{N_{\rm tok} \times d}$ is obtained from the output of the last (i.e., Lth) transformer block as $\boldsymbol{Z} = \boldsymbol{Z}_L \in \mathbb{R}^{N_{\rm tok} \times d}$.

3) LM Head: The core operation of the LM head is *next-token generation*, which estimates the probability distribution over the token vocabulary based on the extracted feature $\boldsymbol{Z}$ and then sequentially generates output tokens. This mechanism is crucial for enabling the generative capabilities of LMM. Initially, $\boldsymbol{Z}$ is decomposed into feature vectors as $\boldsymbol{Z} = [\, \boldsymbol{z}_1\, \boldsymbol{z}_2 \cdots \boldsymbol{z}_{N_{\rm tok}}\,]^{\rm T}$. Then, each $\boldsymbol{z}_n$ associated with the nth token v_n is transformed into a logit vector $\boldsymbol{l}_n \in \mathbb{R}^V$, representing unnormalized log-probabilities over the token vocabulary $\mathcal{V}$, conditioned on the preceding tokens $\{v_{n'}\}_{n'=1}^{n-1}$:

$$\boldsymbol{l}_n = \boldsymbol{W}_{\rm lm} \boldsymbol{z}_n + \boldsymbol{b}_{\rm lm} \tag{39}$$

where $\boldsymbol{W}_{\rm lm} \in \mathbb{R}^{V \times d}$ is the weight matrix and $\boldsymbol{b}_{\rm lm} \in \mathbb{R}^V$ is the bias vector. Then, the token probability vector $\boldsymbol{p}_n \in \mathbb{R}^V$ is obtained by applying the softmax operation to $\boldsymbol{l}_n$ as

$$\boldsymbol{p}_n = f_{\rm smax}(\boldsymbol{l}_n)\,. \tag{40}$$

Finally, the output token x_n is sequentially generated as

$$x_n = \underset{v \in \mathcal{V}}{\rm argmax}\ [\, \boldsymbol{p}_n\,]_v\,. \tag{41}$$

B. VCP-Based Channel Representation

The primary factors affecting the signal propagation are the scatterers (e.g., buildings and walls) around the BS. For instance, the path angles can be derived from the relative positions of the reflection points with respect to the BS and UE. Also, the delays and path gains can be modeled as functions of the angles of incidence, the propagation distance, and the material properties of the scatterers. Thus, we define the VCPs $\mathcal{V}^{(t)} \subseteq \mathcal{I}_{\rm obj}^L \times \mathbb{R}^{3(L+1)}$ at time slot t:

$$\mathcal{V}^{(t)} \triangleq \{\boldsymbol{p}_{\rm ue}^{(t)}, \mathcal{I}_{\rm sca}^{(t)}, \mathcal{P}_{\rm refl}^{(t)}\} \tag{42}$$

where $\boldsymbol{p}_{\rm ue}^{(t)} \in \mathbb{R}^3$ is the UE position, $\mathcal{I}_{\rm sca}^{(t)} \subseteq \mathcal{I}_{\rm obj}$ are the object indices of the scatterers, and $\mathcal{P}_{\rm refl}^{(t)} \subseteq \mathbb{R}^3$ are the reflection points, defined as

$$\mathcal{P}_{\rm refl}^{(t)} \triangleq \{\boldsymbol{p}_{{\rm refl},1}^{(t)}, \boldsymbol{p}_{{\rm refl},2}^{(t)}, \ldots, \boldsymbol{p}_{{\rm refl},L}^{(t)}\} \tag{43}$$

$$\mathcal{I}_{\rm sca}^{(t)} \triangleq \{\, i_{{\rm sca},1}^{(t)},\ i_{{\rm sca},2}^{(t)},\ \ldots,\ i_{{\rm sca},L}^{(t)}\,\}\,. \tag{44}$$

Here, $i_{{\rm sca},l}^{(t)} \in \mathcal{I}_{\rm obj}$ and $\boldsymbol{p}_{{\rm refl},l}^{(t)} \in \mathbb{R}^3$ are the object index and the reflection point of the lth scatterer, respectively.[4] The set $\mathcal{I}_{\rm obj} \triangleq \{1, 2, \ldots, N_{\rm obj}\}$ represents all objects (i.e., potential scatterers) within the environment where each object i is characterized by its boundary $\partial\mathcal{M}_i \subseteq \mathbb{R}^3$ and class c_i. We then define the environmental information $\mathcal{E} \triangleq \{\mathcal{E}_i\}_{i=1}^{N_{\rm obj}}$ as

$$\mathcal{E}_i \triangleq \{\partial\mathcal{M}_i, c_i\}\,. \tag{45}$$

Using $\mathcal{V}^{(t)}$, we can re-express the AOAs and ZOAs as

$$\theta_l^{(t)} = \arccos\left(\frac{[\, \boldsymbol{R}_{\rm bs}^{-1}(\boldsymbol{p}_{\rm bs} - \boldsymbol{p}_{{\rm refl},l}^{(t)})\,]_3}{\|\boldsymbol{R}_{\rm bs}^{-1}(\boldsymbol{p}_{\rm bs} - \boldsymbol{p}_{{\rm refl},l}^{(t)})\|_2}\right) \tag{46}$$

$$\phi_l^{(t)} = {\rm atan2}\Big([\, \boldsymbol{R}_{\rm bs}^{-1}(\boldsymbol{p}_{\rm bs} - \boldsymbol{p}_{{\rm refl},l}^{(t)})\,]_2, [\, \boldsymbol{R}_{\rm bs}^{-1}(\boldsymbol{p}_{\rm bs} - \boldsymbol{p}_{{\rm refl},l}^{(t)})\,]_1\Big) \tag{47}$$

where $\boldsymbol{R}_{\rm bs} \triangleq \boldsymbol{R}_x(\alpha_{\rm bs})\, \boldsymbol{R}_y(\beta_{\rm bs})\, \boldsymbol{R}_z(\gamma_{\rm bs}) \in \mathbb{R}^{3\times 3}$ is the rotation matrix for the BS antenna orientation [39]. Similarly, the AODs $\boldsymbol{\vartheta}^{(t)}$ and ZODs $\boldsymbol{\varphi}^{(t)}$ can be derived by replacing $\boldsymbol{p}_{\rm bs}$

[4] If the lth path is the LOS path, then $i_{{\rm sca},l}^{(t)} = i_{\rm ue}$ and $\boldsymbol{p}_{\rm refl}^{(t)} = \boldsymbol{p}_{\rm ue}^{(t)}$.

with $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$. Also, the delays $\boldsymbol{\tau}^{(t)}$ can be obtained by calculating the total propagation distance as

$$\tau_l = \frac{\left\|\boldsymbol{p}_{\mathrm{bs}} - \boldsymbol{p}_{\mathrm{refl},l}^{(t)}\right\|_2 + \left\|\boldsymbol{p}_{\mathrm{refl},l}^{(t)} - \boldsymbol{p}_{\mathrm{ue}}^{(t)}\right\|_2}{c}. \tag{48}$$

For the path gain, recall that $\beta_l^{(t)}[s]$ is a function of the path angles $\left(\theta_l^{(t)}, \phi_l^{(t)}, \vartheta_l^{(t)}, \varphi_l^{(t)}\right)$, the delay $\tau_l^{(t)}$, and the angle of incidence $\psi_l^{(t)}$ as well as the refractive index $\eta_l^{(t)}$ and the surface roughness $\sigma_l^{(t)}$ (see (14) and (15)). Among these, $\psi_l^{(t)}$ can be obtained from $\boldsymbol{p}_{\mathrm{bs}}$, $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ and $\boldsymbol{p}_{\mathrm{refl},l}^{(t)}$ using the law of cosine as $\psi_l^{(t)} = \frac{1}{2}\arccos\left(\frac{\|\boldsymbol{p}_{\mathrm{bs}}-\boldsymbol{p}_{\mathrm{refl},l}^{(t)}\|_2^2+\|\boldsymbol{p}_{\mathrm{refl},l}^{(t)}-\boldsymbol{p}_{\mathrm{ue}}^{(t)}\|_2^2-\|\boldsymbol{p}_{\mathrm{bs}}-\boldsymbol{p}_{\mathrm{ue}}^{(t)}\|_2^2}{2\|\boldsymbol{p}_{\mathrm{bs}}-\boldsymbol{p}_{\mathrm{refl},l}^{(t)}\|_2\|\boldsymbol{p}_{\mathrm{refl},l}^{(t)}-\boldsymbol{p}_{\mathrm{ue}}^{(t)}\|_2}\right)$. Also, the material properties $\eta_l^{(t)}$ and $\sigma_l^{(t)}$ of the scatterer are directly associated with the scatterer class $c_{i_{\mathrm{sca},l}}$. Thus, we can express $\beta_l^{(t)}[s]$ as a function of $\left(\boldsymbol{p}_{\mathrm{bs}}, \boldsymbol{p}_{\mathrm{ue}}^{(t)}, i_{\mathrm{sca},l}^{(t)}, f_s\right)$, parametrized by $\mathcal{E}$ as

$$\beta_l^{(t)}[s] = \beta\left(\boldsymbol{p}_{\mathrm{bs}}, \boldsymbol{p}_{\mathrm{ue}}^{(t)}, i_{\mathrm{sca},l}^{(t)}, f_s; \mathcal{E}\right). \tag{49}$$

By combining (46)–(49), we obtain the conversion map between the GCPs $\mathcal{G}^{(t)}$ and the VCPs $\mathcal{V}^{(t)}$.

Remark 1: The GCPs $\mathcal{G}^{(t)}$ are expressed as functions of the VCPs $\mathcal{V}^{(t)}$ by using the conversion map $f_{\mathrm{conv}} : \mathcal{I}_{\mathrm{obj}}^{L} \times \mathbb{R}^{3(L+1)} \to \mathbb{R}^{(S+5)L}$ as

$$\mathcal{G}^{(t)} = f_{\mathrm{conv}}\left(\mathcal{V}^{(t)}; \mathcal{E}\right). \tag{50}$$

The detailed equations defining f_{conv} are provided in (46)–(49). Note that f_{conv} is parametrized by the environmental information $\mathcal{E}$ defined in (45). □

Finally, by plugging (50) to (21), we can rewrite the channel matrix $\boldsymbol{H}^{(t)}$ as a function of $\mathcal{V}^{(t)}$.

Remark 2: The channel matrix $\boldsymbol{H}^{(t)}$ is expressed as a function of the VCP $\mathcal{V}^{(t)}$ as

$$\boldsymbol{H}^{(t)} = f_{\mathrm{geo}}^{(t)}\left(f_{\mathrm{conv}}(\mathcal{V}^{(t)}; \mathcal{E})\right). \tag{51}$$

where $f_{\mathrm{geo}}^{(t)}$ and f_{conv} are the geometric map and the conversion map defined in (21) and (50), respectively. □

C. VCP Prediction and Uplink Beam Management

In the prediction stage, the future VCPs $\mathcal{V}^{(t)}$ are estimated based on the historical parameters $\{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}$ where T denotes the number of sequential measurements. The predicted VCPs $\hat{\mathcal{V}}^{(t)} = \{\hat{\boldsymbol{p}}_{\mathrm{ue}}^{(t)}, \hat{\mathcal{I}}_{\mathrm{sca}}^{(t)}, \hat{\mathcal{P}}_{\mathrm{refl}}^{(t)}\}$ are then used to determine the beamformers and combiners for uplink training. For this task, we employ the LMM to model the conditional probability $f\left(\mathcal{V}^{(t)} \mid \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right)$, parameterized by the environmental information $\mathcal{E}$ in (45).

Remark 3: The conditional probability of VCPs $f\left(\mathcal{V}^{(t)} \mid \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right)$ can be factorized as shown in (52), as shown at the bottom of the page. It reveals that the VCP tracking comprises three interconnected tasks: UE position tracking, scatterer index classification, and reflection point tracking. □

Note that VCPs are determined independently for each propagation path. Hence, by introducing a binary variable $S_i \in \{0, 1\}$ that indicates whether the object i acts as a scatterer or not, we can decompose the VCP prediction problem into independent subproblems for each object. Specifically, the scatterer indicator S_i for the object i is defined as

$$S_i = \begin{cases} 1 & \text{if the object } i \text{ acts as a scatterer} \\ 0 & \text{otherwise}. \end{cases} \tag{53}$$

Remark 4: The conditional probabilities of the reflection points, scatterer indices, and UE positions are decomposed as shown in (54)–(56), as shown at the bottom of the page. In particular, (54) shows that, due to the piecewise continuous variation of reflection points, the reflection point $\boldsymbol{p}_{\mathrm{refl},i}^{(t)}$ on the object i is determined by the historical reflection points $\{\boldsymbol{p}_{\mathrm{refl},i}^{(t-t')}\}_{t'=1}^{T-1}$ on the same object i and the historical UE positions $\{\boldsymbol{p}_{\mathrm{ue}}^{(t-t')}\}_{t'=1}^{T}$. In contrast, (55) implies that, due to the discontinuous variation of scatterer indices, the scatterer indicator S_i depends sorely on the current UE position $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$. □

Building on these observations, the prediction stage is structured into five sequential steps, i.e., UE position tracking, scatterer index classification, reflection point tracking, blockage prediction, and uplink beam management.

1) UE Position Prediction: Initially, the LMM predicts the UE position $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ based on the historical UE positions $\{\boldsymbol{p}_{\mathrm{ue}}^{(t-t')}\}_{t'=1}^{T-1}$ by learning the conditional probability $f\left(\boldsymbol{p}_{\mathrm{ue}}^{(t)} \mid \{\boldsymbol{p}_{\mathrm{ue}}^{(t-t')}\}_{t'=1}^{T-1}\right)$ in (56). For this task, various trajectory forecasting techniques that utilize Kalman filter or RNN models can be used [47]. Also, LMM-based techniques that incorporate lane geometry and pedestrian behavior dynamics can be employed for context-aware UE trajectory prediction [48].

2) Scatterer Index Prediction: After obtaining the predicted UE position $\hat{\boldsymbol{p}}_{\mathrm{ue}}$, the scatterer indicator $S_i^{(t)}$ is determined independently for each object $i \in \mathcal{I}_{\mathrm{obj}}$ based on $\hat{\boldsymbol{p}}_{\mathrm{ue}}$. As shown in (55), $S_i^{(t)}$ exhibits discrete variation with respect $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ and is therefore, determined sorely by $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$. To capture this intrinsic relationship, the LMM learns the conditional probability $f\left(S_i^{(t)} \mid \boldsymbol{p}_{\mathrm{ue}}^{(t)}; \mathcal{E}_i\right)$ using a labeled dataset $\mathcal{X}_{\mathrm{sca},i} = \left\{\left(\boldsymbol{p}_{\mathrm{ue}}^{(t)}[n], S_i^{(t)}[n]\right)\right\}_{n=1}^{N_{\mathrm{data}}}$, which consist of pairs of UE positions and scatterer indicators.

Since $S_i^{(t)}$ is a binary variable, identifying $f\left(S_i^{(t)} \mid \boldsymbol{p}_{\mathrm{ue}}^{(t)}; \mathcal{E}_i\right)$ is a binary classification problem. In conventional DL model-based approaches, a fixed classification function (e.g., FFN) is trained using $\mathcal{X}_{\mathrm{sca},i}$. However, a major limitation of these models is that their classification accuracy degrades significantly when the sample size is insufficient. Also, the training overhead is substantial as the network parameters must be retrained for every new dataset. To address this issue, the LMM exploits

$$f\left(\mathcal{V}^{(t)} \mid \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right) = f\left(\mathcal{P}_{\mathrm{refl}}^{(t)} \mid \mathcal{I}_{\mathrm{sca}}^{(t)}, \boldsymbol{p}_{\mathrm{ue}}^{(t)}, \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right) f\left(\mathcal{I}_{\mathrm{sca}}^{(t)} \mid \boldsymbol{p}_{\mathrm{ue}}^{(t)}, \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right) f\left(\boldsymbol{p}_{\mathrm{ue}}^{(t)} \mid \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\right). \tag{52}$$

the in-context learning, which learns an algorithm f_{sel} that selects the most appropriate function $\hat{f}_{\text{sca},i} \in \mathcal{F}$ from a family of pretrained binary classifiers $\mathcal{F}$ [49]

$$\hat{f}_{\text{sca},i} = f_{\text{sel}}\big(\mathcal{X}_{\text{sca},i}; \mathcal{R}_{\text{sca},i}\big) \tag{57}$$

where $\mathcal{Q}_{\text{sca},i}$ is the instruction prompt. To facilitate this function selection process, $\mathcal{Q}_{\text{sca},i}$ must contain detailed descriptions of the target task. Specifically, it consists of three components for each sample, i.e., input description prompt, scenario specification prompt, and task instruction prompt:

- **Input-output description prompt**: "*The input is a three-dimensional UE position and the output is a scatterer indicator of the object i which takes the value 1 if the reflection point on the object i exists and 0 otherwise.*"
- **Scenario specification prompt**: "*The BS is located at $\boldsymbol{p}_{\text{bs}}$ and the environmental information of the object i, including its boundary and class, is $\mathcal{E}_i = \{\partial\mathcal{M}_i, c_i\}$.*"
- **Task instruction prompt**: "*Perform binary classification to determine the scatterer indicator for a given UE position $\boldsymbol{p}_{\text{ue}}^{(t)}[n]$ using $\mathcal{E}_i$.*"

At the inference stage, the LMM generates the response prompt $\mathcal{A}_{\text{sca},i}$ as "*The scatterer indicator is $S_i^{(t)}[n]$.*". Using $\{\hat{S}_i^{(t)}\}_{i\in\mathcal{I}_{\text{obj}}}$, the scatterer indices $\mathcal{I}_{\text{sca}}^{(t)}$ are obtained as

$$\hat{\mathcal{I}}_{\text{sca}}^{(t)} = \big\{i \in \mathcal{I}_{\text{obj}} \,|\, \hat{S}_i^{(t)} = 1\big\}. \tag{58}$$

3) Reflection Point Prediction: Once the scatterers are identified, the LMM predicts the reflection point $\boldsymbol{p}_{\text{refl},i}^{(t)}$ for each object i identified as scatterer (i.e., $\hat{S}_i^{(t)} = 1$), by learning the conditional probability $f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, S_i^{(t)}, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, S_i^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}_i\big)$ in (54). Let T_i be the longest sequence of consecutive time slots such that $S_i^{(t)} = S_i^{(t-1)} = \cdots = S_i^{(t-T_i)} = 1$. Then, the conditional probability of $\boldsymbol{p}_{\text{refl},i}^{(t)}$ can be rewritten as

$$\begin{aligned} &f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, S_i^{(t)}, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, S_i^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}_i\big) \\ &\quad = f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T_i}; \mathcal{E}_i\big). \end{aligned} \tag{59}$$

Determining $f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T_i}; \mathcal{E}_i\big)$ constitutes a multivariate tracking problem. Unlike conventional DL models that directly learn input-output mapping, LMMs infer the outputs by leveraging the contextual information embedded in the instruction prompt with their extensive pre-trained knowledge. Hence, to maximize the tracking performance of LMM, it is essential to structure its reasoning process through chain-of-throught (CoT) prompting, a step-by-step strategy that breaks down a complex problem into simpler subproblems [50]. To apply CoT prompting to reflection point tracking, in the following lemma, we show that the relationship between the UE position and the reflection point is fully characterized by the normal vector and offset of the surface.

Lemma 2: For each object i such that $S_i^{(t)} = 1$, the tangent space $\mathcal{T}_{\boldsymbol{p}_{\text{refl},i}^{(t)}}\mathcal{M}_i$ at the reflection point $\boldsymbol{p}_{\text{refl},i}^{(t)}$ of the reflecting surface $\mathcal{M}_i$ is given by

$$\mathcal{T}_{\boldsymbol{p}_{\text{refl},i}^{(t)}}\mathcal{M} = \big\{\boldsymbol{p} \in \mathbb{R}^3 \,|\, \boldsymbol{n}_i^{\text{T}}\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)\,\boldsymbol{p} = o_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)\big\} \tag{60}$$

where $\boldsymbol{n}_i : \mathcal{M}_i \to \mathbb{R}^3$ is the Gauss map that assigns a unit normal vector and $o_i : \mathcal{M}_i \to \mathbb{R}$ is the offset map given by

$$\boldsymbol{n}_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big) = \frac{\frac{\boldsymbol{p}_{\text{bs}}-\boldsymbol{p}_{\text{refl},i}^{(t)}}{\|\boldsymbol{p}_{\text{bs}}-\boldsymbol{p}_{\text{refl},i}^{(t)}\|_2} + \frac{\boldsymbol{p}_{\text{ue}}^{(t)}-\boldsymbol{p}_{\text{refl},i}^{(t)}}{\|\boldsymbol{p}_{\text{ue}}^{(t)}-\boldsymbol{p}_{\text{refl},i}^{(t)}\|_2}}{\left\|\frac{\boldsymbol{p}_{\text{bs}}-\boldsymbol{p}_{\text{refl},i}^{(t)}}{\|\boldsymbol{p}_{\text{bs}}-\boldsymbol{p}_{\text{refl},i}^{(t)}\|_2} + \frac{\boldsymbol{p}_{\text{ue}}^{(t)}-\boldsymbol{p}_{\text{refl},i}^{(t)}}{\|\boldsymbol{p}_{\text{ue}}^{(t)}-\boldsymbol{p}_{\text{refl},i}^{(t)}\|_2}\right\|_2} \tag{61}$$

$$o_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big) = \boldsymbol{n}_i^{\text{T}}\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)\,\boldsymbol{p}_{\text{refl},i}^{(t)}. \tag{62}$$

Moreover, for a given $\boldsymbol{n}_i^{(t)} \triangleq \boldsymbol{n}_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)$ and $o_i^{(t)} \triangleq o_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)$, $\boldsymbol{p}_{\text{refl},i}^{(t)}$ can be expressed a rational function of $\boldsymbol{p}_{\text{ue}}^{(t)}$, as shown in (63), as shown at the bottom of this page. □

Proof: See Appendix B. ⊠

Lemma 2 shows that the reflection point $\boldsymbol{p}_{\text{refl},i}^{(t)}$ can be derived from the corresponding normal vector $\boldsymbol{n}_i^{(t)} \triangleq \boldsymbol{n}_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)$ and the offset $o_i^{(t)} \triangleq o_i\big(\boldsymbol{p}_{\text{refl},i}^{(t)}\big)$, and vice versa. This means the reflection point prediction can be equivalently converted to the normal vector and offset prediction. Thus, the conditional probability of $\boldsymbol{p}_{\text{refl},i}^{(t)}$ in (59) is rewritten as

$$\begin{aligned} &f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T_i}; \mathcal{E}_i\big) \\ &= f\big(\boldsymbol{n}_i^{(t)}, o_i^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{n}_i^{(t-t')}, o_i^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T_i}; \mathcal{E}_i\big). \end{aligned} \tag{64}$$

Using (64), the LMM predicts the future normal vector $\boldsymbol{n}_i^{(t)}$ and offset $o_i^{(t)}$ from previous normal vectors, offsets,

$$f\big(\mathcal{P}_{\text{refl}}^{(t)} \,|\, \mathcal{I}_{\text{sca}}^{(t)}, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\big) = \prod_{i\in\mathcal{I}_{\text{obj}}} f\big(\boldsymbol{p}_{\text{refl},i}^{(t)} \,|\, S_i^{(t)}, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\boldsymbol{p}_{\text{refl},i}^{(t-t')}, S_i^{(t-t')}, \boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}_i\big) \tag{54}$$

$$f\big(\mathcal{I}_{\text{sca}}^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}, \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}; \mathcal{E}\big) = \prod_{i\in\mathcal{I}_{\text{sca}}} f\big(S_i^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}; \mathcal{E}_i\big) \prod_{j\in\mathcal{I}_{\text{obj}}\setminus\mathcal{I}_{\text{sca}}} \Big(1 - f\big(S_j^{(t)} \,|\, \boldsymbol{p}_{\text{ue}}^{(t)}; \mathcal{E}_j\big)\Big) \tag{55}$$

$$f\big(\boldsymbol{p}_{\text{ue}}^{(t)} \,|\, \{\mathcal{V}^{(t-t')}\}_{t'=1}^{T-1}\big) = f\big(\boldsymbol{p}_{\text{ue}}^{(t)} \,|\, \{\boldsymbol{p}_{\text{ue}}^{(t-t')}\}_{t'=1}^{T-1}\big). \tag{56}$$

$$\boldsymbol{p}_{\text{refl},i}^{(t)} = \frac{\big(\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{ue}}^{(t)} - o_i^{(t)}\big)\,\boldsymbol{p}_{\text{bs}} + \big(\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{bs}} - o_i^{(t)}\big)\,\boldsymbol{p}_{\text{ue}}^{(t)} - 2\,\big(\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{ue}}^{(t)} - o_i^{(t)}\big)\,\big(\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{bs}} - o_i^{(t)}\big)\,\boldsymbol{n}_i^{(t)}}{\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\big(\boldsymbol{p}_{\text{bs}} + \boldsymbol{p}_{\text{ue}}^{(t)}\big) - 2o_i^{(t)}}. \tag{63}$$

and UE positions.[5] For this task, we use the dataset $\mathcal{X}_{\text{refl},i} = \left\{\{(\boldsymbol{p}_{\text{ue}}^{(t)}[n], \boldsymbol{p}_{\text{refl},i}^{(t)}[n])\}_{t=1}^{T_i[n]}\right\}_{n=1}^{N_{\text{data}}}$ comprising sequential pairs of UE positions and reflection points. Then, using (61) and (62), we compute the normal vectors and offsets to construct $\mathcal{X}_{\text{nor},i} = \left\{\{(\boldsymbol{p}_{\text{ue}}^{(t)}[n], (\boldsymbol{n}_i^{(t)}[n], o_i^{(t)}[n]))\}_{t=1}^{T_i[n]}\right\}_{n=1}^{N_{\text{data}}}$. The instruction prompt $\mathcal{Q}_{\text{refl},i}$ for each sample is given by

- **Input-output description prompt**: "*The inputs are a sequence of $T_i[n]$ three-dimensional UE positions, and sequences of $T_i[n]-1$ normal vectors and offsets of the object i at the reflection points for each UE position. The outputs are the $T_i[n]$th normal vector and offset.*"
- **Scenario specification prompt**: "*The BS is located at $\boldsymbol{p}_{\text{bs}}$ and the environmental information of the object i, including its boundary and class, is $\mathcal{E}_i = \{\partial\mathcal{M}_i, c_i\}$. The scatterer indicator of the object i is $\hat{S}_i^{(t)} = 1$.*"
- **Task instruction prompt**: "*Perform multivariate tracking to predict the $T_i[n]$th normal vector and offset corresponding to the $T_i[n]$th UE position $\boldsymbol{p}_{\text{ue}}^{(T_i[n])}$ based on the previous $T_i[n]-1$ pairs of UE positions and normal vector/offset $\{\boldsymbol{p}_{\text{ue}}^{(t)}[n], (\boldsymbol{n}_i^{(t)}[n], o_i^{(t)}[n])\}_{t=1}^{T_i[n]-1}$, while incorporating $\mathcal{E}_i$. If $T_i[n] = 1$, then perform multivariate regression to directly estimate the $T_i[n]$th normal vector and offset from the $T_i[n]$th UE position $\boldsymbol{p}_{\text{ue}}^{(T_i[n])}$ and $\mathcal{E}_i$.*"

At the inference stage, the LMM generates the response prompt $\mathcal{A}_{\text{refl},i}$ as "*The Tth normal vector and offset are $\left(\boldsymbol{n}_i^{(T)}[n], o_i^{(T)}[n]\right)$.*". Once $\left(\boldsymbol{n}_i^{(t)}, o_i^{(t)}\right)$ are obtained, $\boldsymbol{p}_{\text{refl},i}^{(t)}$ is computed using (63). Lastly, $\hat{\mathcal{P}}_{\text{refl}}^{(t)}$ is determined as

$$\hat{\mathcal{P}}_{\text{refl}}^{(t)} = \left\{\hat{\boldsymbol{p}}_{\text{refl},i}^{(t)} \,|\, \hat{S}_i^{(t)} = 1, i \in \mathcal{I}_{\text{obj}}\right\}. \tag{65}$$

4) Blockage Prediction: Using the predicted VCPs $\hat{\mathcal{V}}^{(t)}$, we assess how the transmitted signal interacts with nearby objects such as blockages. Specifically, we determine whether the two-hop path (from the BS to a reflection point and from that reflection point to the UE) will be obstructed by dynamic objects.[6] The blockage prediction procedure consists of two main steps: obstacle trajectory prediction and occlusion detection. In the first step, we estimate future 2D bounding boxes $\mathcal{B}^{(t)} \subseteq \mathbb{R}^2$ (i.e., the smallest rectangles that enclose the object) in RGB images and the corresponding depth map $\left[\mathcal{D}_{\text{d}}^{(t)}\right]_{\mathcal{B}^{(t)}}$ for each potential obstacle nearby the UE and reflection points using their historical data $\left\{\mathcal{B}^{(t-t')}, \left[\mathcal{D}_{\text{d}}^{(t-t')}\right]_{\mathcal{B}^{(t-t')}}\right\}_{t'=1}^{T-1}$. Note that these obstacles are identified beforehand during the update stages of earlier time slots from the RGB-D images $\{\mathcal{D}^{(t-t')}\}_{t'=1}^{T-1}$ (see Section III-D.3 for detailed operations). For this task, trajectory forecasting techniques analogous to those used for UE position prediction can be employed [47], [48].

In the second step, we check if either line segment from the BS to the reflection point $\hat{\boldsymbol{p}}_{\text{refl},i}^{(t)}$, denoted as $\hat{\mathcal{L}}_{\text{bs}-\text{refl},i}^{(t)}$, or from that reflection point to the UE, denoted as $\mathcal{L}_{ue-\text{refl}}^{(t)}$, intersects any obstacles. These line segments are expressed as

$$\hat{\mathcal{L}}_{\text{bs}-\text{refl},i}^{(t)} = \left\{\boldsymbol{p}_{\text{bs}} + t\left(\hat{\boldsymbol{p}}_{\text{refl},i}^{(t)} - \boldsymbol{p}_{\text{bs}}\right) \,|\, t \in [0,1]\right\} \tag{66}$$

$$\hat{\mathcal{L}}_{\text{ue}-\text{refl},i}^{(t)} = \left\{\hat{\boldsymbol{p}}_{\text{ue}}^{(t)} + t\left(\hat{\boldsymbol{p}}_{\text{refl},i}^{(t)} - \hat{\boldsymbol{p}}_{\text{ue}}^{(t)}\right) \,|\, t \in [0,1]\right\}. \tag{67}$$

This requires verifying two intersection conditions. First, we identify the intersections within the image plane between the obstacle and the line segments. For example, in case of UE-reflection point link, we check:

$$\mathcal{B}_{\text{int},i}^{(t)} \triangleq \left(\mathcal{B}^{(t)} \cap \text{proj}_{\text{img}}\left(\hat{\mathcal{L}}_{\text{ue}-\text{refl},i}^{(t)}\right)\right) \neq \emptyset \tag{68}$$

where proj_{img} is the projection operator onto the image plane. Second, if the intersection $\mathcal{B}_{\text{int},i}^{(t)}$ exists, then we compare the depth values of the intersection points with those of the line segment. Since obstacles are 3D objects, their bounding boxes on the image may correspond to multiple depth values, and only the closest surfaces are recorded at the depth map. To address this issue, we verify that there exists a 2D intersection point $\boldsymbol{p}_{\text{int}-2\text{D}}^{(t)} \in \mathcal{B}_{\text{int},i}^{(t)}$ and the corresponding 3D point $\boldsymbol{p}_{\text{int}-3\text{D}}^{(t)} \in \hat{\mathcal{L}}_{\text{ue}-\text{refl},i}^{(t)}$ (i.e., $\text{proj}_{\text{img}}\left(\boldsymbol{p}_{\text{int}-3\text{D}}^{(t)}\right) = \boldsymbol{p}_{\text{int}-2\text{D}}^{(t)}$) such that the depth value of $\boldsymbol{p}_{\text{int}-3\text{D}}^{(t)}$ falls between that of $\boldsymbol{p}_{\text{int}-2\text{D}}^{(t)}$ and the maximum depth values within the bounding box:

$$\left[\mathcal{D}_{\text{d}}^{(t)}\right]_{\boldsymbol{p}_{\text{int}-2\text{D}}^{(t)}} \leqslant \left\|\boldsymbol{p}_{\text{int}-2\text{D}}^{(t)} - \boldsymbol{p}_{\text{cam}}\right\|_2 \leqslant \max_{\boldsymbol{p}\in\mathcal{B}^{(t)}}\left[\mathcal{D}_{\text{d}}^{(t)}\right]_{\boldsymbol{p}} \tag{69}$$

where $\boldsymbol{p}_{\text{cam}} \in \mathbb{R}^3$ is the RGB-D camera position. The blockage of the BS-reflection point link is predicted similarly. Once the blockages are predicted, the VCPs $\hat{\mathcal{V}}^{(t)} = \left\{\hat{\boldsymbol{p}}_{\text{ue}}^{(t)}, \hat{\mathcal{I}}_{\text{sca}}^{(t)}, \hat{\mathcal{P}}_{\text{refl}}^{(t)}\right\}$ are updated accordingly.

5) Uplink Beam Management: After the VCP prediction, the BS determines its combining matrix $\boldsymbol{W}_{\text{bs}}^{(t)}$ and the UE beamforming matrix $\boldsymbol{F}_{\text{ue}}^{(t)}$ using $\hat{\mathcal{V}}^{(t)}$ for uplink training. The goal of uplink training is to maximize the UE detection probability while ensuring received signal power. To do so, the BS samples P candidate positions $\left\{\hat{\boldsymbol{p}}_{\text{ue},p}^{(t)}\right\}_{p=1}^{P}$ near $\hat{\boldsymbol{p}}_{\text{ue}}^{(t)}$ and generates the corresponding VCPs and channels $\left\{\hat{\mathcal{V}}_p^{(t)}, \hat{\boldsymbol{H}}_p^{(t)}\right\}_{p=1}^{P}$:

$$\hat{\boldsymbol{H}}_p^{(t)} = f_{\text{geo}}^{(t)}\left(f_{\text{conv}}(\hat{\mathcal{V}}_p^{(t)}; \mathcal{E})\right) \qquad p = 1, 2, \ldots, P. \tag{70}$$

Then, for each k_pth symbol, $\boldsymbol{w}_{\text{bs},k_p}^{(t)}$ and $\boldsymbol{f}_{\text{ue},k_p}^{(t)}$ are chosen from the BS and UE beam codebooks $\mathcal{F}_{\text{bs}}$ and $\mathcal{F}_{\text{ue}}$:

$$\left(\boldsymbol{w}_{\text{bs},k_p}^{(t)}, \boldsymbol{f}_{\text{ue},k_p}^{(t)}\right) = \underset{\boldsymbol{w}\in\mathcal{F}_{\text{bs}}, \boldsymbol{f}\in\mathcal{F}_{\text{ue}}}{\text{argmax}} \sum_{s\in\mathcal{S}} \log_2\left(1 + \frac{P_{\text{ue}}}{\sigma_{\text{n}}^2}\left|\boldsymbol{w}^{\text{H}}\left(\hat{\boldsymbol{H}}_p^{(t)}[s]\right)^{\text{H}}\boldsymbol{f}\right|^2\right). \tag{71}$$

Finally, we obtain $\boldsymbol{W}_{\text{bs}}^{(t)} \triangleq \left[\boldsymbol{w}_{\text{bs},k_1}^{(t)}\ \boldsymbol{w}_{\text{bs},k_2}^{(t)}\ \cdots\ \boldsymbol{w}_{\text{bs},k_P}^{(t)}\right]$ and $\boldsymbol{F}_{\text{ue}}^{(t)} \triangleq \left[\boldsymbol{f}_{\text{ue},k_1}^{(t)}\ \boldsymbol{f}_{\text{ue},k_2}^{(t)}\ \cdots\ \boldsymbol{f}_{\text{ue},k_P}^{(t)}\right]$.

D. VCP Update and Downlink Beam Management

In the update stage, by incorporating the environmental features obtained from pilot and sensing measurements $(\mathbf{Y}^{(t)}, \mathcal{D}^{(t)})$, the predicted VCPs $\hat{\mathcal{V}}^{(t)}$ are refined to $\mathcal{V}^{(t)}$ and used for the downlink beam management. The update stage consists of five steps, i.e., UE position update, scatterer index update, blockage detection, reflection point update, and downlink beam management.

[5]Tracking the normal vector and offset offers additional benefits compared to directly tracking the reflection point, as their variations with the UE movements are much smaller. For example, in the case of a building-type scatterer with a planar surface, the reflection point shifts rapidly as the UE moves, whereas the normal vector and offset of the surface remain constant.

[6]We focus solely on obstructions caused by dynamic objects, as blockages from static structures (e.g., buildings) have already been considered during the scatterer index prediction step.

1) UE Position Update: The LMM first updates the UE position using the sensing measurement $\mathcal{D}^{(t)}$. Specifically, the LMM detects the UE within the RGB-D image and extracts its position $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ through object detection.[7] Object detection is a computer vision (CV) technique that identifies the bounding box and the class of target object [51]. Compared to traditional DL-based object detection, the LMM-based approach provides superior detection accuracy and contextual understanding owing to its advanced vision-language reasoning capabilities. The corresponding instruction prompt $\mathcal{Q}_{\mathrm{od}}$ is

- **Input-output description prompt**: "*The input are an RGB-D image and the predicted three-dimensional UE position. The output is the UE position extracted from the detected bounding box within the image.*"
- **Scenario specification prompt**: "*The RGB-D image is captured by the BS and the class of the UE is* $c_{i_{\mathrm{ue}}}$.
- **Task instruction prompt**: "*Perform object detection to identify the UE within a local region around the predicted UE position* $\hat{\boldsymbol{p}}_{\mathrm{ue}}^{(t)}$ *in the RGB-D image* $\mathcal{D}^{(t)}$. *Extract its three-dimensional position from the bounding box.*"

If the UE is not detected, then the BS can acquire $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ using the global navigation satellite system (GNSS) combined with xG positioning techniques based on time and angle measurements [52], [53], [54], [55], [56], [57], [58], [59].

2) Scatterer Index Update: Once the UE position is refined, the LMM updates the scatterer indicators based on $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$. For this task, we use the same instruction prompt $\mathcal{Q}_{\mathrm{sca},i}$ with that employed during the scatterer index prediction step (see Section III-C2). Also, the refined scatterer indices $\mathcal{I}_{\mathrm{sca}}$ are obtained from $\{S_i^{(t)}\}_{i\in\mathcal{I}_{\mathrm{obj}}}$ using (58).

3) Blockage Detection: In this step, the LMM identifies potential obstacles near the UE position $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$ and scatterers $\mathcal{I}_{\mathrm{sca}}^{(t)}$ and detects signal path blockages by analyzing the sensing measurement $\mathcal{D}^{(t)}$. First, to identify the potential obstacles and obtain their bounding boxes $\{\mathcal{B}_i^{(t)}\}_{i=1}^{N_{\mathrm{obs}}}$ and classes $\{b_i\}_{i=1}^{N_{\mathrm{obs}}}$, the LMM performs object detection on $\mathcal{D}^{(t)}$. The instruction prompt for this task can be obtained by slightly modifying the object detection prompt $\mathcal{Q}_{\mathrm{od}}$ used in the UE position update step (see Section III-D.1). Alternatively, obstacles can also be detected by identifying pixel-wise differences between $\mathcal{D}^{(t)}$ and a reference background image without dynamic objects.

Second, using the identified bounding boxes $\{\mathcal{B}_i^{(t)}\}_{i=1}^{N_{\mathrm{obs}}}$ along with the depth map $\{[\mathcal{D}_{\mathrm{d}}^{(t)}]_{\mathcal{B}_i^{(t)}}\}_{i=1}^{N_{\mathrm{obs}}}$ of the detected obstacles, the LMM determines whether each signal path is blocked. Following the approach used in the blockage prediction step in Section III-C.4, the LMM first constructs two line segments, i.e., $\mathcal{L}_{\mathrm{bs-rel},i}^{(t)}$ and $\mathcal{L}_{\mathrm{ue-rel},i}^{(t)}$, for the BS-reflection point and reflection point-UE links. Note that $\mathcal{L}_{\mathrm{bs-rel},i}^{(t)}$ and $\mathcal{L}_{\mathrm{ue-rel},i}^{(t)}$ slightly differ from those in (66) and (67) since they are constructed using the refined UE position $\boldsymbol{p}_{\mathrm{ue}}^{(t)}$, whereas those in (66) and (67) are based on the predicted UE position $\hat{\boldsymbol{p}}_{\mathrm{ue}}^{(t)}$. The LMM then verifies the occurrence of blockage by checking the two intersection conditions in (68) and (69).

4) Reflection Point Update: In this step, the BS recalculates the reflection points $\hat{\mathcal{P}}_{\mathrm{refl}}^{(t)}$ based on the updated UE position $\boldsymbol{p}_{\mathrm{ue}}$ and scatterer indices $\mathcal{I}_{\mathrm{sca}}^{(t)}$. Then, the BS refines the reflection points using the pilot measurements $\mathbf{Y}^{(t)}$. Specifically, the BS searches for $\mathcal{P}_{\mathrm{refl}}^{(t)}$ in the vicinity of $\hat{\mathcal{P}}_{\mathrm{refl}}^{(t)}$ that minimizes the mean squared error (MSE) between $\mathbf{Y}^{(t)}$ and that reconstructed from $\mathcal{P}_{\mathrm{refl}}^{(t)}$. Furthermore, by utilizing the boundary information $\partial\mathcal{M}_{i_{\mathrm{sca},l}}$ of each lth scatterer, the search space for $\boldsymbol{p}_{\mathrm{refl},l}^{(t)}$ can be restricted to the interior region $\mathrm{int}(\partial\mathcal{M}_{i_{\mathrm{sca},l}})$ of $\partial\mathcal{M}_{i_{\mathrm{sca},l}}$. The reflection point refinement problem $\mathscr{P}$ is formulated as

$$\mathscr{P}: \quad \underset{\mathcal{P}_{\mathrm{refl}}^{(t)}}{\text{minimize}} \quad L\big(\mathcal{P}_{\mathrm{refl}}^{(t)}\big) \tag{72a}$$

$$\text{subject to} \quad \boldsymbol{p}_{\mathrm{refl},l}^{(t)} \in \mathrm{int}(\partial\mathcal{M}_{i_{\mathrm{sca},l}}) \qquad \forall l \in \mathcal{L} \tag{72b}$$

where $\mathcal{L} \triangleq \{1,2,\ldots,L\}$ and $L\big(\mathcal{P}_{\mathrm{refl}}^{(t)}\big)$ is defined as

$$L\big(\mathcal{P}_{\mathrm{refl}}^{(t)}\big) \triangleq \Big\|\mathbf{Y}^{(t)} - \sqrt{P_{\mathrm{ue}}}\,\big(\big(\boldsymbol{F}_{\mathrm{ue}}^{(t)}\big)^{*} * \boldsymbol{W}_{\mathrm{bs}}^{(t)}\big)^{\mathrm{H}} \times \big[f_{\mathrm{geo}}^{(t)}\big(f_{\mathrm{conv}}(\mathcal{V}^{(t)};\mathcal{E})\big)\big]_{\mathcal{S}}\Big\|_2^2. \tag{73}$$

To solve $\mathscr{P}$, we employ an alternating approach that updates one reflection point at a time while keeping the others fixed. Also, each reflection point can be effectively optimized using grid search or the quasi-Newton method.

5) Downlink Beam Management: Once the refined VCPs $\mathcal{V}^{(t)}$ are obtained, the BS reconstructs the channel matrix $\mathcal{V}^{(t)}$ as $\boldsymbol{H}^{(t)} = f_{\mathrm{geo}}^{(t)}\big(f_{\mathrm{conv}}(\mathcal{V}^{(t)};\mathcal{E})\big)$. Then, the BS determines the downlink BS beamforming vector $\boldsymbol{f}_{\mathrm{bs}}^{(t)}$ and the UE combining vector $\boldsymbol{w}_{\mathrm{ue}}^{(t)}$ maximizing the data rate R (see (11)). Specifically, $\boldsymbol{f}_{\mathrm{bs}}^{(t)}$ and $\boldsymbol{w}_{\mathrm{ue}}^{(t)}$ and selected from $\mathcal{F}_{\mathrm{bs}}$ and $\mathcal{F}_{\mathrm{ue}}$ as

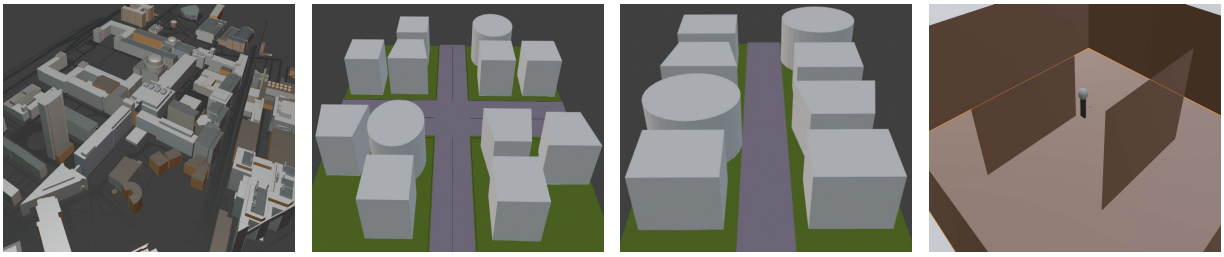
$$\big(\boldsymbol{f}_{\mathrm{bs}}^{(t)}, \boldsymbol{w}_{\mathrm{ue}}^{(t)}\big) = \underset{\boldsymbol{f}\in\mathcal{F}_{\mathrm{bs}},\boldsymbol{w}\in\mathcal{F}_{\mathrm{ue}}}{\mathrm{argmax}} \sum_{s=1}^{S} \log_2\left(1 + \frac{P_{\mathrm{bs}}}{\sigma_{\mathrm{n}}^2}\left|\boldsymbol{w}^{\mathrm{H}}\big(\boldsymbol{H}^{(t)}[s]\big)^{\mathrm{H}}\boldsymbol{f}\right|^2\right). \tag{74}$$

IV. Numerical Results

A. Simulation Setup

We consider a mmWave MIMO-OFDM system where a BS equipped with a $N_{\mathrm{bs}} = 8\times 8$ URA antenna serves a UE equipped with $N_{\mathrm{ue}} = 2\times 2$ URA antennas. The BS is located at $\boldsymbol{p}_{\mathrm{bs}} = [\,0\ \mathrm{m}\ \ 0\,\mathrm{m}\ \ 15\,\mathrm{m}\,]$ and the BS and UE antenna orientations are $[\,\alpha_{\mathrm{bs}}\ \beta_{\mathrm{bs}}\ \gamma_{\mathrm{bs}}\,] = [\,0^\circ\ -10^\circ\ 0^\circ\,]$ and $[\,\alpha_{\mathrm{ue}}\ \beta_{\mathrm{ue}}\ \gamma_{\mathrm{ue}}\,] = [\,0^\circ\ 0^\circ\ 0^\circ\,]$, respectively. The position vector of the BS antenna located at the xth row and yth column is $\boldsymbol{p}_{\mathrm{bs},(x-1)N_{\mathrm{bs,v}}+y} = \boldsymbol{p}_{\mathrm{bs}} + \boldsymbol{R}_{\mathrm{bs}}\,[\,(x-1)d_{\mathrm{bs}}\ (y-1)d_{\mathrm{bs}}\ 0\,]^{\mathrm{T}}$ where $\boldsymbol{R}_{\mathrm{bs}}$ is the rotation matrix defined in (46) and d_{bs} is the BS antenna spacing. The position vectors of the UE antennas are defined similarly. We employ the geometric mmWave channel model with carrier frequency $f_{\mathrm{c}} = 28\,\mathrm{GHz}$ and bandwidth $B = 400\,\mathrm{MHz}$ [39]. We set the signal-to-noise ratio (SNR) to 20 dB. The uplink pilot transmission and downlink

[7] In LMM-BM, the VCP tracking is conducted independently for each UE. However, since the BS uses sensing measurements to acquire UE positions, distinguishing individual UEs in multi-user scenarios can be challenging. For UE identification, one can exploit a two-stage approach that first identifies the UEs via small-range beam sweeping and then uses their visual features as unique identifiers. Initially, the BS constructs a beam codebook in which each codeword is steered toward a position extracted by the object detector, and performs beam sweeping using this codebook. Then, each UE feeds back the index of the best beam codeword through the scheduled physical uplink shared channel (PUSCH). After the initial UE identification, the BS extracts their visual features and then uses them as unique identifiers.



(a) UMi (MIT campus) (b) UC (intersection) (c) UC (straight road) (d) InH

Fig. 3. Examples images of various wireless environments.

data transmission span 4 and 10 OFDM symbols across 273 resource blocks (RBs). For the BS and UE beam codebooks, we use $\mathcal{F}_{\rm bs} = \boldsymbol{F}_4 \otimes \boldsymbol{F}_2$ and $\mathcal{F}_{\rm ue} = \boldsymbol{F}_2$, respectively, where $\boldsymbol{F}_N \in \mathbb{C}^{N\times N}$ is the N-point discrete Fourier transform (DFT) matrix. The number of prior measurements is $T = 5$. We use LLaVA-NeXT-Interleave 7B as the LMM model [60].

As a performance metric, we use the channel normalized mean square error (NMSE) and defined as $\text{NMSE} = \mathbb{E}\big[\|\boldsymbol{H}_{\rm true}^{(t)} - \boldsymbol{H}_{\rm est}^{(t)}\|_{\rm F}^2 / \|\boldsymbol{H}_{\rm true}^{(t)}\|_{\rm F}^2\big]$ and the downlink data rate R in (11). We employ four benchmark schemes: 1) LMM-based GCP tracking scheme; 2) LSTM-based GCP tracking scheme [26]; 3) Kalman filter-based GCP tracking scheme [18]; and 4) ISAC-based beam tracking scheme [31].

B. Multimodal Dataset Generation and LMM Fine-Tuning

Initially, we construct four wireless environments using Blender: 1) urban micro (UMi) environment based on the Massachusetts Institute of Technology (MIT) campus; 2) urban canyon (UC) environment with an intersection; 3) UC environment on straight loads; and 4) indoor hotspot (InH) environment (see Fig. 3). Unless stated otherwise, the UC environment on straight roads is used as the default setting. The UC environments include up to 8 buildings and 6 lanes, with the UE moving along the road while randomly switching lanes. For structural diversity, we use 4 rectangular and 4 cylindrical buildings. In particular, we consider a dynamic multi-user scenario in the UC environment with an intersection, featuring 2 UEs and a bus serving as a dynamic obstacle, with turning trajectories. The InH environment consists of four side walls and two obstacles. We then perform ray tracing on the generated environments using the NVIDIA Sionna™ platform to determine VCPs. These parameters are then used to generate pilot measurements. Also, by using the rendering function of Sionna, we capture images with an RGB-D camera with a pixel resolution of 1024×768 and a FoV of $120°$. Finally, we generate $N_{\rm data} = 8{,}000$, $N_{\rm val} = 1{,}000$, and $N_{\rm test} = 1{,}000$ samples for training, validation, and test, respectively.

For the fine-tuning of the proposed LMM-based reflection tracking scheme, we first collect a dataset comprising sequences of VCPs $\mathcal{X} = \{\{(\boldsymbol{p}_{\rm ue}^{(t)}[n], \mathcal{I}_{\rm sca}^{(t)}[n], \mathcal{P}_{\rm refl}^{(t)}[n])\}_{t=1}^{T}\}_{n=1}^{N_{\rm data}}$. Based on $\mathcal{X}$, we construct task-specific datasets $\mathcal{X}_{{\rm sca},i}$ and $\mathcal{X}_{{\rm refl},i}$ for each object i. We then generate pairs of instruction and response prompts for each task. As a loss function, we utilize the cross-entropy between the probability distributions of predicted and ground truth tokens. Note that the LMM module used in each step of the prediction and update stages is fine-tuned separately in a supervised manner. One major issue of the LMM fine-tuning is that, due to the large network size, updating the entire network parameters is computationally prohibitive. To address this challenge, we adopt the low-rank adaptation (LoRA) technique which updates only a small subset of task-specific parameters [61]. These parameters are augmented with the product of two learnable low-rank matrices. In our work, we fine-tune the attention layer of the LMM. The dataset generation is performed on a Dell PowerEdge R750 server equipped with an NVIDIA RTX 6000 Ada Generation GPU, and the LMM fine-tuning is accomplished using a Supermicro Grace Hopper MGX system equipped with NVIDIA GH200 superchips.

TABLE I
PERFORMANCE OF LMM-BM FOR DIFFERENT PARAMETER CONFIGURATIONS

Number of LMM parameters	Inference time	Channel NMSE	Data rate
13B	300 ms	−9.69 dB	5.72 bps/Hz
7B	192 ms	−9.23 dB	5.68 bps/Hz

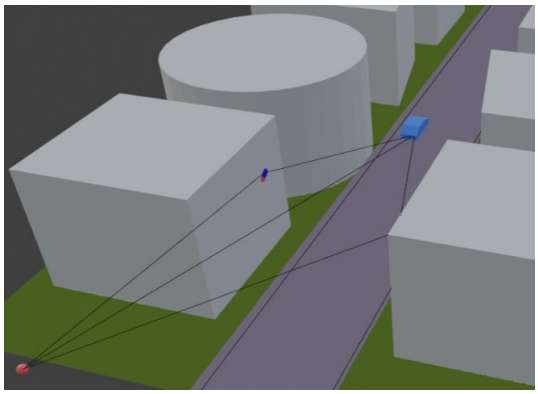


Fig. 4. Visual comparison of predicted (red dot) and ground truth reflection point (blue dot).

C. Simulation Results

In Table I, we evaluate the inference time, channel NMSE, and data rate of LMM-BM. We observe that the inference time of LLaVA-NeXT-Interleave 7B model is around 0.19 s. To assess the feasibility of LMM-BM, we compare this inference

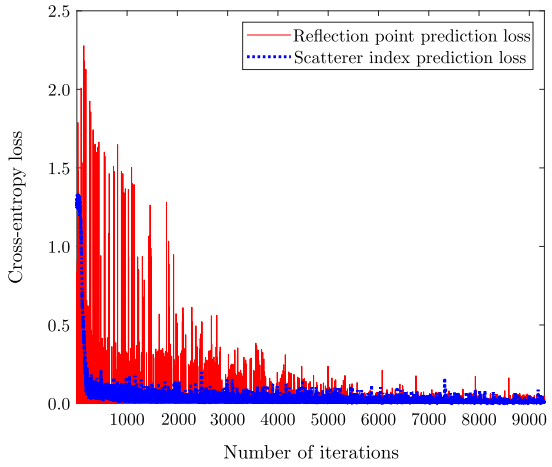


Fig. 5. Cross-entropy loss as a function of the number of iterations.

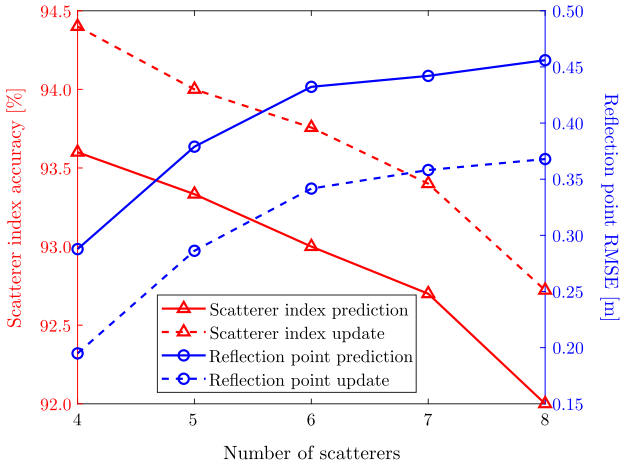


Fig. 6. Scatterer index and reflection point estimation accuracies as functions of the number of scatterers.

time with the beam coherence time representing the duration over which the beam remains aligned with the signal path [62]:

$$T_{\mathrm{c}} = \frac{D}{v \cos\theta} \arccos\left(\psi^2 \log \zeta + 1\right) \tag{75}$$

where D is the communication distance, θ is the beam angle, v is the speed of UE, ψ is the beamwidth, and $\zeta \in [0, 1]$ is a threshold for beam misalignment. For example, consider a BS equipped with $N = 64$ antenna array serving a UE moving at $v = 60\,\mathrm{km/h}$ at $D = 100\,\mathrm{m}$ and $\theta = 60°$. Then, the beamwidth is around $\psi \approx \frac{4}{N} \approx 7.2°$ and the corresponding beam coherence time with $\zeta = 0.5$ is $T_{\mathrm{c}} \approx 0.9\,\mathrm{s}$, which is clearly longer than the inference time of LMM-BM. Furthermore, considering ongoing advancements in fast and low-power LMMs and GPU architectures, together with the model compression techniques (e.g., pruning and quantization), we expect that the LMM inference time can be further reduced down the road. We also compare the performance with different numbers of LMM parameters. We observe that while LLaVA-NeXT-Interleave 13B model performs slightly better than LLaVA-NeXT-Interleave 7B model, the gap is marginal. This implies that LMM-BM can effectively be implemented with lighter LMMs, reducing inference time and resource usage.

Fig. 4 illustrates the visual comparison of the predicted (red dot) and the ground truth reflection points (blue dot). It shows that the proposed LMM-based reflection tracking technique can accurately predict VCPs by capturing both intra- and cross-modal correlations in pilot and sensing measurements. Also, Fig. 5 shows the cross-entropy losses during the fine-tuning process for the scatterer index and reflection point predictions. We see that the losses of both scatter index and reflection point predictions decrease significantly with the number of iterations. This indicates that the LoRA-based fine-tuning can effectively enhance the prediction accuracies with minimal parameter updates.

Fig. 6 shows the reflection tracking performances in terms of scatterer index classification accuracy and reflection point tracking accuracy as functions of the number of scatterers. The scatterer index classification accuracy is defined as the ratio of the number of scatterers with correctly classified

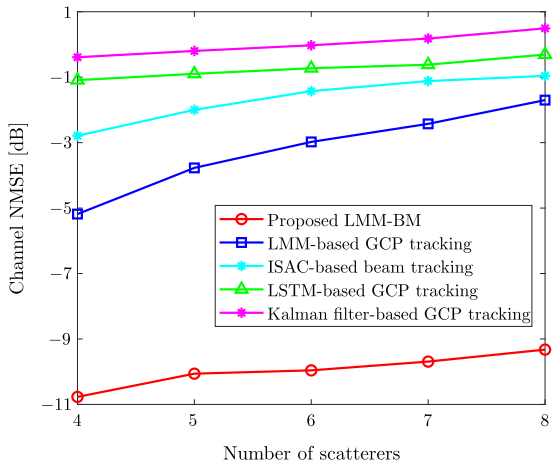


Fig. 7. Channel prediction NMSE as a function of the number of scatterers.

scatterers over the total number of samples. Additionally, the reflection point tracking accuracy is defined as the root mean squared error (RMSE) of the predicted reflection points over all samples. We observe that the proposed reflection tracking technique can effectively predict both scatterer indices and reflection points. For example, the LMM-BM achieves more than 92% in scatterer index classification accuracy while maintaining a reflection point tracking accuracy of less than 45 cm. We also observe that the reflection tracking performance slightly decreases with the number of scatterers. However, since the number of dominant paths in mmWave/THz band is typically fewer than 6, this degradation is unlikely to affect overall beam management performance.

Fig. 7 presents the channel prediction NMSE performance as a function of the number of scatterers. We observe that the proposed LMM-BM outperforms the conventional GCP-based beam tracking schemes by a large margin. For example, when the number of scatterers is 8, LMM-BM achieves more than 9 dB and 10.5 dB NMSE gains over the LSTM-based and Kalman filter-based schemes, respectively. Even when compared with the LMM-based scheme, LMM-BM achieves the NMSE gain of 7 dB. While the GCP tracking

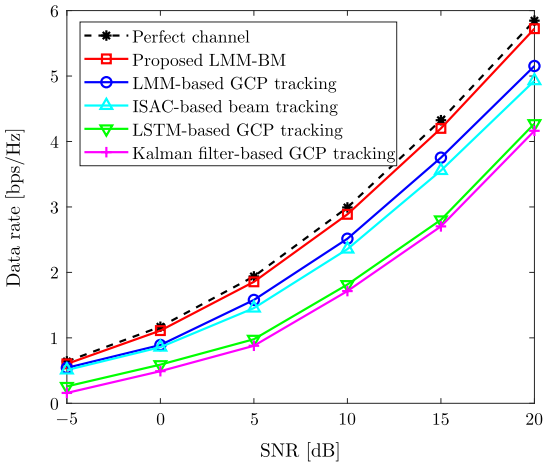


Fig. 8. Downlink data rate as a function of SNR.

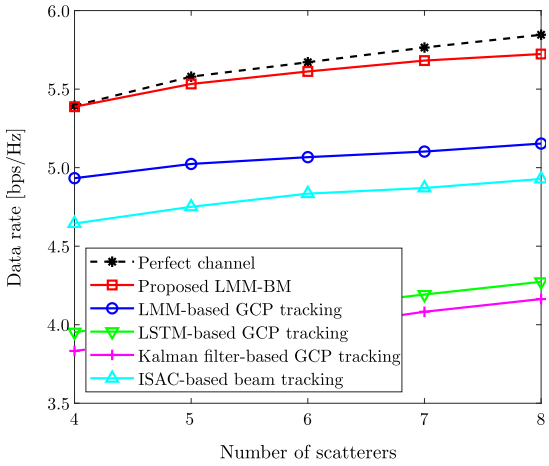


Fig. 9. Downlink data rate as a function of the number of scatterers.

TABLE II
PERFORMANCE OF LMM-BM FOR DIFFERENT LANE NUMBERS

Lanes	Reflection point RMSE	Channel NMSE	Data rate
2 lanes	0.31 m	−12.7 dB	5.72 bps/Hz
4 lanes	0.46 m	−9.23 dB	5.68 bps/Hz
6 lanes	0.50 m	−8.82 dB	5.62 bps/Hz

schemes struggle to handle the piecewise continuity of channel parameters due to the lack of contextual information about the scatterers, LMM-BM leverages environmental information extracted from sensing and pilot measurements and then sequentially predicts the scatterer indices and reflection points. In doing so, LMM-BM can effectively track the VCPs.

Fig. 8 shows the data rate as a function of the number of SNR. We observe that the proposed LMM-BM achieves a substantial data rate gain over the conventional beam tracking techniques. For example, when SNR $= 20$ dB, LMM-BM achieves around 16.1% data rate enhancement over the ISAC-based scheme. This is because the ISAC-based scheme tracks only the LOS component using the radar signal while neglecting the NLOS components. In contrast, LMM-BM simultaneously tracks the VCPs of all propagation paths by leveraging LMM. We note that the data rate gain of LMM-BM increases with the SNR. In the high SNR regime, the effect of noise is negligible so the channel prediction improvements of LMM-BM are directly reflected in the data rate.

Fig. 9 shows the downlink data rate as a function of the number of scatterers. Interestingly, we observe that the data rate gain of the proposed LMM-BM increases with the number of scatterers. For example, when the number of scatterers is 4, the data rate gain of LMM-BM over the LMM-based GCP tracking scheme is 8.7%, but it increases to 11.1% when the number of scatterers is 8. When the number of scatterers is large, the scatterers change more frequently, leading to a degradation in the prediction performance of the conventional approaches. However, such is not the case for LMM-BM since it tracks both the scatterer indices and reflection points.

Table II shows the performance of LMM-BM for a different number of lanes. We observe that LMM-BM shows robust performance even in the six-lane environment. Note that while the UE moves linearly along the line in the two-lane environment, the UE randomly changes lanes in the six-lane environment, so both the UE trajectory and reflection points change nonlinearly. By tracking the normal vector and offset of the reflecting surface along using the intrinsic relationship between the UE position and reflection points identified in Lemma 2 via LMM, LMM-BM effectively manages these nonlinear variations and precisely forecasts the VCPs.

To demonstrate the efficacy of the proposed blockage prediction and detection techniques, Fig. 10 plots the blockage detection probability as a function of the UE speed. We observe that the proposed techniques effectively detect blockage with more than 92% probability. This means that LMM-BM can effectively identify blockage caused by dynamic obstacles and exclude the obstructed path from the candidate beam directions. Additionally, Fig. 11 presents the data rate performance across various environments. We observe that LMM-BM shows consistently strong performance across all evaluated environments. Notably, even in a challenging multi-user UC environment involving nonlinear turning trajectories and dynamic blockage caused by a moving bus at an intersection, LMM-BM maintains robust performance. This robustness is primarily attributed to three factors: 1) VCP tracking is performed independently for each UE; 2) dynamic blockages are effectively handled through the proposed blockage prediction and detection techniques (see Fig. 10); and 3) the ability to handle nonlinear trajectories arises from the large network size and extensive pretraining of the LMM.

Table III presents the ablation study on the impact of UE position, scatterer indices, and reflection points. We observe that due to the power disparity between the LOS and NLOS components in mmWave/THz bands, the accuracy of UE position has the most pronounced effect on beam management performance. Furthermore, since the reflection points cannot be reliably determined when the associated scatterer indices are incorrect, the accuracy of the scatterer indices has a more detrimental effect than that of the reflection points.

V. CONCLUSION

In this paper, we proposed an LMM-based environment-aware beam management scheme called LMM-BM that

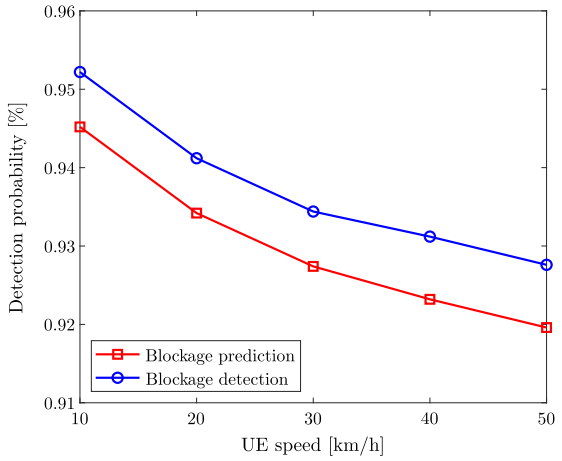


Fig. 10. Blockage detection probability as a function of the UE speed.

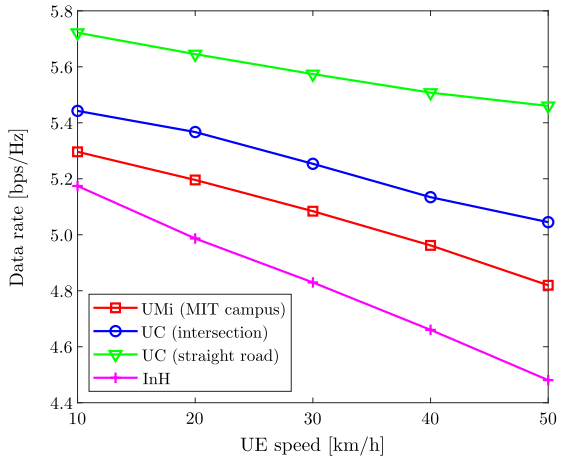


Fig. 11. Data rate as a function of the UE speed in various environments.

exploits the environmental information extracted from the sensor data (i.e., RGB-D images) and the pilot measurements to optimize beam directions. The key ingredients of LMM-BM are VCPs which include the positions of UE, reflection points, and scatterers. Unlike GCPs which lack contextual awareness of the propagation environment, VCPs facilitate the analysis of physical interactions with the surrounding objects such as scatterers and obstacles. Specifically, we developed a reflection tracking technique that tracks VCPs from sensing and pilot measurements using LMM. By leveraging its strong generalization ability along with NLP capabilities to interpret contextual task instructions, LMM can perform both CV tasks (e.g., object detection) and numerical tasks (e.g., multivariate tracking) required for VCP tracking. From the numerical evaluations, we demonstrated that LMM-BM can accurately predict VCPs and enhance the data rate significantly.

APPENDIX A
PROOF OF LEMMA 1

Using the BS and UE array response matrices, the channel matrix $\boldsymbol{H}^{(t)}[s]$ can be rewritten as

$$\boldsymbol{H}^{(t)}[s] = \boldsymbol{A}_{\text{bs}}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)})\, \text{diag}(\tilde{\boldsymbol{\beta}}^{(t)}[s])\, \big(\boldsymbol{A}_{\text{ue}}^{(t)}(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)})\big)^{\text{T}}. \tag{76}$$

Then, the vectorization of $\boldsymbol{H}^{(t)}[s]$ is computed as

$$\begin{aligned}&\text{vec}(\boldsymbol{H}^{(t)}[s])\\ &= \big(\boldsymbol{A}_{\text{ue}}^{(t)}(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}) \otimes \boldsymbol{A}_{\text{bs}}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)})\big)\, \text{vec}\big(\text{diag}(\tilde{\boldsymbol{\beta}}^{(t)}[s])\big) \qquad (77)\\ &= \big(\boldsymbol{A}_{\text{ue}}^{(t)}(\boldsymbol{\vartheta}^{(t)}, \boldsymbol{\varphi}^{(t)}) * \boldsymbol{A}_{\text{bs}}(\boldsymbol{\theta}^{(t)}, \boldsymbol{\phi}^{(t)})\big)\, \tilde{\boldsymbol{\beta}}^{(t)}[s] \qquad (78)\end{aligned}$$

where $\tilde{\boldsymbol{\beta}}^{(t)}[s]$ is a L-dimensional vector such that $\big[\tilde{\boldsymbol{\beta}}^{(t)}[s]\big]_l = \beta_l^{(t)}[s]\, e^{-j2\pi\Delta f \tau_l^{(t)}}$ for $l = 1, 2, \ldots, L$.

APPENDIX B
PROOF OF LEMMA 2

Note that (61) and (62) directly follow the law of reflection. Now, to prove (63), we first compute the image $\bar{\boldsymbol{p}}_{\text{bs}}^{(t)}$ of the BS position vector $\boldsymbol{p}_{\text{bs}}$ over the tangent space $\mathcal{T}_{\boldsymbol{p}_{\text{refl},i}^{(t)}}\mathcal{M}$ as

$$\bar{\boldsymbol{p}}_{\text{bs}}^{(t)} = \boldsymbol{p}_{\text{bs}} + 2\left(o_i^{(t)} - \big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{bs}}\right)\boldsymbol{n}_i^{(t)}. \tag{79}$$

Then, $\boldsymbol{p}_{\text{refl},i}^{(t)}$ is determined as the intersection of $\mathcal{T}_{\boldsymbol{p}_{\text{refl},i}^{(t)}}\mathcal{M}$ and the line connecting $\bar{\boldsymbol{p}}_{\text{bs}}^{(t)}$ and $\boldsymbol{p}_{\text{ue}}^{(t)}$:

$$\boldsymbol{p}_{\text{refl},i}^{(t)} = \bar{\boldsymbol{p}}_{\text{bs}}^{(t)} + v\left(\boldsymbol{p}_{\text{ue}}^{(t)} - \bar{\boldsymbol{p}}_{\text{bs}}^{(t)}\right) \tag{80}$$

for some $v \in \mathbb{R}$. Since $\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\boldsymbol{p}_{\text{refl},i}^{(t)} = o_i^{(t)}$, we obtain

$$v = \frac{o_i^{(t)} - \big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\bar{\boldsymbol{p}}_{\text{bs}}^{(t)}}{\big(\boldsymbol{n}_i^{(t)}\big)^{\text{T}}\big(\boldsymbol{p}_{\text{ue}}^{(t)} - \bar{\boldsymbol{p}}_{\text{bs}}^{(t)}\big)}. \tag{81}$$

By plugging (79) and (81) into (80), we get the desired result.

TABLE III
ABLATION STUDY ON THE IMPACT OF UE POSITION, SCATTERER INDICES, AND REFLECTION POINTS

Variables	Channel NMSE	Data rate
UE position	−11.5 dB	5.39 bps/Hz
Scatterer indices	−12.9 dB	5.60 bps/Hz
Reflection points	−13.0 dB	5.89 bps/Hz

ACKNOWLEDGMENT

The fundamental research described in this article has benefited from platforms provided by NVIDIA and SMC to WINS Lab at MIT.

REFERENCES

[1] S. Kim, J. Wu, B. Shim, and M. Z. Win, "Cell-free massive non-terrestrial networks," *IEEE J. Sel. Areas Commun.*, vol. 43, no. 1, pp. 201–217, Jan. 2025.
[2] H. Lee et al., "Towards 6G hyper-connectivity: Vision, challenges, and key enabling technologies," *J. Commun. Netw.*, vol. 25, no. 3, pp. 344–354, Jun. 2023.
[3] S. Dang, O. Amin, B. Shihada, and M.-S. Alouini, "What should 6G be?," *Nature Electron.*, vol. 3, no. 1, pp. 20–29, Jan. 2020.
[4] Q. Xue et al., "A survey of beam management for mmWave and THz communications towards 6G," *IEEE Commun. Surveys Tuts.*, vol. 26, no. 3, pp. 1520–1559, 3rd Quart., 2024.

[5] G. J. Foschini, G. D. Golden, R. A. Valenzuela, and P. W. Wolniansky, "Simplified processing for high spectral efficiency wireless communication employing multi-element arrays," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 11, pp. 1841–1852, Nov. 1999.

[6] A. Conti, W. M. Gifford, M. Z. Win, and M. Chiani, "Optimized simple bounds for diversity systems," *IEEE Trans. Commun.*, vol. 57, no. 9, pp. 2674–2685, Sep. 2009.

[7] J. Winters, "On the capacity of radio communication systems with diversity in a Rayleigh fading environment," *IEEE J. Sel. Areas Commun.*, vol. SAC-5, no. 5, pp. 871–878, Jun. 1987.

[8] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, Aut. 1996.

[9] J. H. Winters, J. Salz, and R. D. Gitlin, "The impact of antenna diversity on the capacity of wireless communication systems," *IEEE Trans. Commun.*, vol. 42, no. 2, pp. 1740–1751, Feb. 1994.

[10] P. F. Driessen and G. J. Foschini, "On the capacity formula for multiple input-multiple output wireless channels: A geometric interpretation," *IEEE Trans. Commun.*, vol. 47, no. 2, pp. 173–176, Feb. 1999.

[11] M. Z. Win, N. C. Beaulieu, L. A. Shepp, B. F. Logan, and J. H. Winters, "On the SNR penalty of MPSK with hybrid selection/maximal ratio combining over i.i.d. Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 51, no. 6, pp. 1012–1023, Jun. 2003.

[12] J. Winters, "Optimum combining for indoor radio systems with multiple users," *IEEE Trans. Commun.*, vol. COM-35, no. 11, pp. 1222–1230, Nov. 1987.

[13] M. Chiani, M. Z. Win, and H. Shin, "MIMO networks: The effects of interference," *IEEE Trans. Inf. Theory*, vol. 56, no. 1, pp. 336–349, Jan. 2010.

[14] *Study on New Radio Access Technology—Physical Layer Aspects*, document TR 38.802, Aug. 2017.

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

[16] Z. Xiao, T. He, P. Xia, and X.-G. Xia, "Hierarchical codebook design for beamforming training in millimeter-wave communication," *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3380–3392, May 2016.

[17] M. Giordani, M. Polese, A. Roy, D. Castor, and M. Zorzi, "A tutorial on beam management for 3GPP NR at mmWave frequencies," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 173–196, 1st Quart., 2019.

[18] V. Va, H. Vikalo, and R. W. Heath Jr., "Beam tracking for mobile millimeter wave communication systems," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2016, pp. 743–747.

[19] S. Jayaprakasam, X. Ma, J. W. Choi, and S. Kim, "Robust beam-tracking for mmWave mobile communications," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2654–2657, Dec. 2017.

[20] F. Liu, P. Zhao, and Z. Wang, "EKF-based beam tracking for mmWave MIMO systems," *IEEE Commun. Lett.*, vol. 23, no. 12, pp. 2390–2393, Dec. 2019.

[21] S. G. Larew and D. J. Love, "Adaptive beam tracking with the unscented Kalman filter for millimeter wave communication," *IEEE Signal Process. Lett.*, vol. 26, no. 11, pp. 1658–1662, Nov. 2019.

[22] P. Zhu, H. Lin, J. Bao, J. Li, and D. Wang, "Beam tracking for distributed millimeter-wave massive MIMO systems based on the unscented Kalman filter," *IEEE Wireless Commun. Lett.*, vol. 11, no. 4, pp. 712–716, Apr. 2022.

[23] K. Ma, Z. Wang, W. Tian, S. Chen, and L. Hanzo, "Deep learning for mmWave beam-management: State-of-the-art, opportunities and challenges," *IEEE Wireless Commun.*, vol. 30, no. 4, pp. 108–114, Aug. 2023.

[24] J. Zhang and C. Masouros, "Learning-based predictive transmitter-receiver beam alignment in millimeter wave fixed wireless access links," *IEEE Trans. Signal Process.*, vol. 69, pp. 3268–3282, 2021.

[25] J. Zhang, Y. Huang, C. Masouros, X. You, and B. Ottersten, "Hybrid data-induced Kalman filtering approach and application in beam prediction and tracking," *IEEE Trans. Signal Process.*, vol. 72, pp. 1412–1426, 2024.

[26] S. H. Lim, S. Kim, B. Shim, and J. W. Choi, "Deep learning-based beam tracking for millimeter-wave communications under mobility," *IEEE Trans. Commun.*, vol. 69, no. 11, pp. 7458–7469, Nov. 2021.

[27] S. H. A. Shah and S. Rangan, "Multi-cell multi-beam prediction using auto-encoder LSTM for mmWave systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10366–10380, Dec. 2022.

[28] J. Ye and X. Ge, "Beam management optimization for V2V communications based on deep reinforcement learning," *Sci. Rep.*, vol. 13, no. 1, p. 20440, Nov. 2023.

[29] S. Kim et al., "Role of sensing and computer vision in 6G wireless communications," *IEEE Wireless Commun.*, vol. 31, no. 5, pp. 264–271, Oct. 2024.

[30] G. R. Muns, K. V. Mishra, C. B. Guerra, Y. C. Eldar, and K. R. Chowdhury, "Beam alignment and tracking for autonomous vehicular communication using IEEE 802.11ad-based radar," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 535–540.

[31] Z. Du et al., "Integrated sensing and communications for V2I networks: Dynamic predictive beamforming for extended vehicle targets," *IEEE Trans. Wireless Commun.*, vol. 22, no. 6, pp. 3612–3627, Jun. 2023.

[32] X. Meng, F. Liu, C. Masouros, W. Yuan, Q. Zhang, and Z. Feng, "Vehicular connectivity on complex trajectories: Roadway-geometry aware ISAC beam-tracking," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 7408–7423, Nov. 2023.

[33] W. Xu, F. Gao, X. Tao, J. Zhang, and A. Alkhateeb, "Computer vision aided mmWave beam alignment in V2X communications," *IEEE Trans. Wireless Commun.*, vol. 22, no. 4, pp. 2699–2714, Apr. 2023.

[34] T. Zhang, J. Liu, and F. Gao, "Vision aided beam tracking and frequency handoff for mmWave communications," in *Proc. IEEE Conf. Comput. Commun. Workshops*, May 2022, pp. 1–2.

[35] Y. Cui et al., "Seeing is not always believing: ISAC-assisted predictive beam tracking in multipath channels," *IEEE Wireless Commun. Lett.*, vol. 13, no. 1, pp. 14–18, Jan. 2024.

[36] K. Chen, C. Qi, O. A. Dobre, and G. Y. Li, "Simultaneous beam training and target sensing in ISAC systems with RIS," *IEEE Trans. Wireless Commun.*, vol. 23, no. 4, pp. 2696–2710, Apr. 2024.

[37] J. Zhang, J. Huang, S. Jin, and S. Lu, "Vision-language models for vision tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 8, pp. 5625–5644, Aug. 2024.

[38] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inform. Process. Syst. (NIPS)*, 2017, pp. 5998–6008.

[39] *Study on Channel Model for Frequencies From 0.5 to 100 GHz*, document TR 38.901, 3GPP, Sep. 2024.

[40] Y. Han, Q. Liu, C.-K. Wen, M. Matthaiou, and X. Ma, "Tracking FDD massive MIMO downlink channels by exploiting delay and angular reciprocity," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 1062–1076, Sep. 2019.

[41] 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 Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 870–913, 2nd Quart., 2018.

[42] S. Kim, J. Moon, J. Wu, B. Shim, and M. Z. Win, "Vision-aided positioning and beam focusing for 6G terahertz communications," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 9, pp. 2503–2519, Sep. 2024.

[43] C. Lin and G. Y. Li, "Adaptive beamforming with resource allocation for distance-aware multi-user indoor terahertz communications," *IEEE Trans. Commun.*, vol. 63, no. 8, pp. 2985–2995, Aug. 2015.

[44] R. Piesiewicz, C. Jansen, D. Mittleman, T. Kleine-Ostmann, M. Koch, and T. Kurner, "Scattering analysis for the modeling of THz communication systems," *IEEE Trans. Antennas Propag.*, vol. 55, no. 11, pp. 3002–3009, Nov. 2007.

[45] K. Dovelos, M. Matthaiou, H. Q. Ngo, and B. Bellalta, "Channel estimation and hybrid combining for wideband terahertz massive MIMO systems," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 6, pp. 1604–1620, Jun. 2021.

[46] S. Kim, A. Lee, H. Ju, K. A. Ngo, J. Moon, and B. Shim, "Transformer-based channel parameter acquisition for terahertz ultra-massive MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 72, no. 11, pp. 15127–15132, Nov. 2023.

[47] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang, and H. Chen, "A survey on trajectory-prediction methods for autonomous driving," *IEEE Trans. Intell. Vehicles*, vol. 7, no. 3, pp. 652–674, Sep. 2022.

[48] Z. Lan, L. D. Liu, B. Fan, Y. Lv, Y. Ren, and Z. Cui, "Traj-LLM: A new exploration for empowering trajectory prediction with pre-trained large language models," *IEEE Trans. Intell. Veh.*, vol. 10, no. 2, pp. 794–807, Feb. 2025.

[49] Q. Dong et al., "A survey on in-context learning," in *Proc. Conf. Empirical Methods Natural Lang. Process (EMNLP)*, 2023, pp. 1107–1128.

[50] Z. Ling et al., "Deductive verification of chain-of-thought reasoning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2023, pp. 36407–36433.

[51] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proc. IEEE*, vol. 111, no. 3, pp. 257–276, Mar. 2023.

[52] M. Z. Win et al., "Network localization and navigation via cooperation," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 56–62, May 2011.

[53] J. A. Del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1G to 5G," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 1124–1148, 2nd Quart., 2018.
[54] A. Conti, G. Torsoli, C. A. Gómez-Vega, A. Vaccari, G. Mazzini, and M. Z. Win, "3GPP-compliant datasets for xG location-aware networks," *IEEE Open J. Veh. Technol.*, vol. 5, pp. 473–484, 2024.
[55] A. Conti, G. Torsoli, C. A. Gómez-Vega, A. Vaccari, and M. Z. Win, "XG-loc: 3GPP-compliant datasets for xG location-aware networks," *IEEE Dataport*, Tech. Rep., Dec. 2023, doi: 10.21227/rper-vc03.
[56] H. K. Dureppagari, C. Saha, H. S. Dhillon, and R. M. Buehrer, "NTN-based 6G localization: Vision, role of LEOs, and open problems," *IEEE Wireless Commun.*, vol. 30, no. 6, pp. 44–51, Dec. 2023.
[57] M. Z. Win, Y. Shen, and W. Dai, "A theoretical foundation of network localization and navigation," *Proc. IEEE*, vol. 106, no. 7, pp. 1136–1165, Jul. 2018.
[58] H. Chen, H. Sarieddeen, T. Ballal, H. Wymeersch, M. Alouini, and T. Y. Al-Naffouri, "A tutorial on terahertz-band localization for 6G communication systems," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 3, pp. 1780–1815, 3rd Quart., 2022.
[59] M. Z. Win, W. Dai, Y. Shen, G. Chrisikos, and H. V. Poor, "Network operation strategies for efficient localization and navigation," *Proc. IEEE*, vol. 106, no. 7, pp. 1224–1254, Jul. 2018.
[60] F. Li et al., "LLaVA-NeXT-interleave: Tackling multi-image, video, and 3D in large multimodal models," 2024, *arXiv:2407.07895*.
[61] J. E. Hu et al., "LoRA: Low-rank adaptation of large language models," in *Proc. Int. Conf. Learn. Represent (ICLR)*, Apr. 2021.
[62] V. Va, J. Choi, and R. W. Heath Jr., "The impact of beamwidth on temporal channel variation in vehicular channels and its implications," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 5014–5029, Jun. 2017.

Seungnyun Kim (Member, IEEE) received the B.S. (Hons.) and Ph.D. degrees in electrical and computer engineering from Seoul National University (SNU), Seoul, South Korea, in 2016 and 2023, respectively.

He is currently a Post-Doctoral Fellow with the Wireless Information and Network Sciences Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include information theory, optimization methods, and machine learning with applications to real-world problems, including wireless communications, network localization and navigation, and non-terrestrial networks.

Dr. Kim was a recipient of the Sejong Science Fellowship from Korean Government in 2023, the Best Ph.D. Dissertation Award from SNU in 2023, the Qualcomm Innovation Fellowship Finalist in 2021, and the Samsung Humantech Paper Award Gold Prize in 2019.

Subham Saha (Student Member, IEEE) received the B.Tech. degree in electronics and communication engineering from Indian Institute of Space Science and Technology (IIST), India, in 2020. He is currently pursuing the degree with the Wireless Information and Network Sciences Laboratory, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA.

From 2021 to 2023, he worked as an Engineer at the U. R. Rao Satellite Center, Indian Space Research Organization, Bengaluru, India. Since 2024, he has been a Research Assistant with the Wireless Information and Network Sciences Laboratory, MIT. His research interests include stochastic processes, optimization methods, and statistical learning, with applications in wireless network management, digital twins, and massive machine-type communications.

Mr. Saha was awarded the Director's Gold Medal for Best Academic Performer at IIST in 2020.

Seokhyun Jeong (Student Member, IEEE) received the B.S. degree from the Department of Electrical and Computer Engineering, Seoul National University, Seoul, South Korea, in 2022, where he is currently pursuing the Ph.D. degree in electrical and computer engineering. His main areas of research interests include artificial intelligence for wireless communications.

Byonghyo Shim (Fellow, IEEE) received the B.S. and M.S. degrees in control and instrumentation engineering from Seoul National University, South Korea, in 1995 and 1997, respectively, and the M.S. degree in mathematics and the Ph.D. degree in electrical and computer engineering from the University of Illinois Urbana–Champaign (UIUC), Champaign, IL, USA, in 2004 and 2005, respectively. From 1997 to 2000, he was an Officer (First Lieutenant) and an Academic Full-Time Instructor with the Department of Electronics Engineering, Korean Air Force Academy. From 2005 to 2007, he was a Staff Engineer with Qualcomm Inc., San Diego, CA, USA. From 2007 to 2014, he was an Associate Professor with the School of Information and Communication, Korea University, Seoul. Since 2014, he has been with Seoul National University (SNU), where he is currently a Professor with the Department of Electrical and Computer Engineering and the Vice Dean of the Engineering College.

His research interests include wireless communications, machine learning, and statistical signal processing. He was an Elected Member of the Signal Processing for Communications and Networking (SPCOM) Technical Committee of the IEEE Signal Processing Society. He was a recipient of the M. E. Van Valkenburg Research Award from the ECE Department, University of Illinois, in 2005, the Haedong Young Engineer Award from IEIE in 2010, the Irwin Jacobs Award from Qualcomm and KICS in 2016, the Shinyang Research Award from the Engineering College of SNU in 2017, the Okawa Foundation Research Award in 2020, the IEEE ComSoc AP Outstanding Paper Award in 2021, and the JCN Best Paper Award in 2024. He has been served as an Associate Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (TWC), IEEE TRANSACTIONS ON COMMUNICATIONS (TCOM), IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY (TVT), IEEE TRANSACTIONS ON SIGNAL PROCESSING (TSP), IEEE WIRELESS COMMUNICATIONS LETTERS (WCL), and *Journal of Communications and Networks* (JCN), and a Guest Editor for IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC).

Moe Z. Win (Fellow, IEEE) is the Robert R. Taylor Professor at the Massachusetts Institute of Technology (MIT) and the founding director of the Wireless Information and Network Sciences Laboratory. Prior to joining MIT, he was with AT&T Research Laboratories and with the NASA Jet Propulsion Laboratory.

His research encompasses fundamental theories, algorithm design, and network experimentation for a broad range of real-world problems. His current research topics include ultra-wideband systems, network localization and navigation, network interference exploitation, and quantum information science. He has served the IEEE Communications Society as an elected Member-at-Large on the Board of Governors, as elected Chair for the Radio Communications Committee, and as an IEEE Distinguished Lecturer. Over the last two decades, he held various editorial positions for IEEE journals and organized numerous international conferences. He has served on the SIAM Diversity Advisory Committee.

Dr. Win is an elected Fellow of the AAAS, the EURASIP, and the IET. He was honored with two IEEE Technical Field Awards: the IEEE Kiyo Tomiyasu Award (2011) and the IEEE Eric E. Sumner Award (2006, jointly with R. A. Scholtz). His publications, co-authored with students and colleagues, have received several awards. Other recognitions include the MIT Frank E. Perkins Award (2024), the MIT Everett Moore Baker Award (2022), the IEEE Vehicular Technology Society James Evans Avant Garde Award (2022), the IEEE Communications Society Edwin H. Armstrong Achievement Award (2016), the Cristoforo Colombo International Prize for Communications (2013), the Copernicus Fellowship (2011) and the *Laurea Honoris Caus* (2008) from the Università degli Studi di Ferrara, and the U.S. Presidential Early Career Award for Scientists and Engineers (2004).