GNN-Enabled Deep Unfolding for Precoding in Massive MIMO LEO Satellite Communications

Huibin Zhou*†, Xinrui Gong†, Christos G. Tsinos‡, Li You*†, Xiqi Gao*†, and Björn Ottersten¶
*National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China
†Purple Mountain Laboratories, Nanjing 211100, China
‡Department of Digital Industry Technologies, National and Kapodistrian University of Athens, Evripus Campus, 344 00 Athens, Greece
¶Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, 2721 Luxembourg City, Luxembourg
Email: {zhouhb, lyou, xqgao}@seu.edu.cn, gongxinrui@pmlabs.com.cn, ctsinos@uoa.gr, bjorn.ottersten@uni.lu

***Abstract*—Low Earth Orbit (LEO) satellite communication is crucial for developing sixth-generation (6G) networks. The integration of massive multiple-input multiple-output (MIMO) technology is being actively researched to enhance the performance of LEO satellite communication systems. However, the limited power resources of LEO satellites pose significant challenges to improving energy efficiency (EE) under power-constrained conditions. Typical optimization-based methods often lack real-time adaptability and computational efficiency. This paper proposes innovative solutions to address the challenges of precoding in massive MIMO LEO satellite communications. Specifically, we introduce a deep unfolding of the Dinkelbach algorithm and the weighted minimum mean square error (WMMSE) approach to achieve enhanced EE. This transformation of iterative optimization procedures into a graph neural network (GNN) leads to faster convergence and improved computational efficiency. Furthermore, we apply the Taylor expansion method to approximate matrix inversion within the GNN framework. Numerical experiments demonstrate the superiority of our proposed method in terms of complexity and robustness, achieving significant improvements over other state-of-the-art methods.**

I. Introduction

The rapid increase in data traffic, driven by the increasing use of smart devices, highlights the need to evolve communication networks towards sixth-generation (6G) technology [1]. However, the limitations of traditional terrestrial networks, such as uneven infrastructure distribution and unstable wireless environments, hinder their ability to handle the growing data demands [2]. 6G is envisioned as a fully interconnected and autonomous network, offering seamless coverage and ubiquitous services by integrating terrestrial networks with satellite systems, particularly low earth orbit (LEO) satellites [3]. Positioned at altitudes ranging from 500 km to 2000 km, LEO satellites have garnered significant attention due to their advantages over geostationary earth orbit (GEO) satellites, such as reduced round-trip latency, lower path loss, and decreased launch expenses [4]. Numerous initiatives, including Telesat and Amazon's Project Kuiper, are actively working to advance LEO satellite communication systems [5].

To enhance LEO satellite communication performance, massive multiple-input multiple-output (MIMO) technology is being investigated for its ability to improve link budget, key for 6G applications [6]. However, massive MIMO poses challenges such as higher computational complexity, increased power usage, and concerns related to hardware and weight, especially given the limited solar power and payload capacity of LEO satellites [7]. Typical optimization-based methods for precoding are computationally intensive and lack real-time adaptability in resource-constrained environments [8]. To address he computational challenges of massive MIMO, researchers are developing energy-efficient system designs and low-complexity algorithms optimized for satellite hardware, enabling LEO networks to meet 6G demands while adapting to dynamic conditions for reliable performance.

Optimizing LEO satellite systems remains a challenging task despite advancements. Traditional methods often lack real-time adaptability and computational efficiency, particularly in improving energy efficiency (EE) under power constraints [9]. Artificial intelligence (AI) offers promising solutions, enabling real-time optimization of precoding techniques and enhancing system adaptability . Machine learning, especially deep learning, can predict interference, optimize power distribution, and manage satellite resources [10]. These AI-driven approaches have the potential to reduce computational load and improve network performance, enhancing EE, coverage, and bandwidth utilization to meet the demands of future communication technologies [11].

Most deep learning-based methods operate end-to-end, making them easy to train but lacking interpretability due to their "black-box" nature. This limits their efficiency, especially in dynamic, large-scale wireless networks where

This work was supported by the National Natural Science Foundation of China for Outstanding Young Scholars under Grant 62322104, the Jiangsu Province Major Science and Technology Project under Grant BG2024005, the Natural Science Foundation of Jiangsu Province under Grant BK20231415, and the Fundamental Research Funds for the Central Universities under Grant 2242022k60007.

2025 IEEE Wireless Communications and Networking Conference (WCNC) | 979-8-3503-6836-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/WCNC61545.2025.10978590

979-8-3503-6836-9/25/$31.00 ©2025 IEEE

training data, like channel state information (CSI), may be scarce in 6G [12]. Generic neural networks (NNs) often require extensive data and computational resources, reducing their practical application. Algorithm unfolding addresses these issues by linking classic iterative algorithms with NNs, converting algorithm parameters into trainable ones, improving layer interpretability, and offering theoretical guarantees for better reliability and predictability [13].

This paper presents an innovative approach using deep-unfolding graph neural network (GNN) to address precoding in massive MIMO LEO satellite communications. The deep unfolding network method transforms each iteration of a conventional iterative algorithm into a layer-like structure within a NN [14], incorporating learnable parameters to enhance performance. The goal is to enhance the precoding design for massive MIMO LEO satellite communications, leading to better system performance and EE.

II. System Model and Problem Formulation

In this section, we introduce the channel model for the massive MIMO LEO communication system. Next, we present an overview of the downlink transmission power consumption model. Finally, we formulate the questions.

A. Channel Model

Consider a downlink communication system involving a LEO satellite and K user terminals (UTs), each equipped with a single antenna. Mounted on the satellite is a large-scale uniform planar array (UPA), consisting of elements arranged at half-wavelength spacing along both the X-axis and Y-axis. This configuration includes a total of $N_{\mathrm{t}}^{\mathrm{x}}$ elements along the X-axis and $N_{\mathrm{t}}^{\mathrm{y}}$ elements along the Y-axis. We denote the total number of antennas on the satellite as $N_{\mathrm{t}} \triangleq N_{\mathrm{t}}^{\mathrm{x}} N_{\mathrm{t}}^{\mathrm{y}}$. Let the number of UT paths for the k-th UT be L_k. At time t, the frequency response of the complex baseband channel at frequency f can be expressed as

$$\mathbf{h}_k(t,f) = \sum_{l=1}^{L_k} \alpha_{k,l} \exp\{\jmath 2\pi [t\nu_{k,l} - f\tau_{k,l}]\} \mathbf{v}_{k,l} \in \mathbb{C}^{N_t \times 1}, \tag{1}$$

where $\alpha_{k,l}$, $\nu_{k,l}$, $\tau_{k,l}$ and $\mathbf{v}_{k,l}$ indicate the channel gain, Doppler shift, propagation delay, and array response vector, respectively. $\nu_{k,l}$ is mainly determined by the Doppler frequency shift resulting from the mobility of the satellite and the UT, and can be expressed as $\nu_{k,l} = \nu_{k,l}^{\mathrm{sat}} + \nu_{k,l}^{\mathrm{ut}}$. Assume that the satellite Doppler shift and array response vector corresponding to all propagation paths are the same that is $\nu_{k,l}^{\mathrm{sat}} = \nu_k^{\mathrm{sat}}$, $\mathbf{v}_{k,l} = \mathbf{v}_k$, $\forall k, l$. In addition, the propagation delay is expressed as $\tau_{k,l} = \tau_{k,l}^{\mathrm{ut}} + \tau_k^{\min}$, where $\tau_k^{\min}$ represents the shortest delay of all propagation paths of the k-th UT. Therefore, the channel response shown in (1) can be transformed into

$$\mathbf{h}_k(t,f) = \exp\{\jmath 2\pi [t\nu_k^{\mathrm{sat}} - f\tau_k^{\min}]\} g_k(t,f) \mathbf{v}_k \in \mathbb{C}^{N_t \times 1}. \tag{2}$$

B. Downlink Transmission and Power Consumption Model

The signal received by the k-th UT can be expressed as

$$y_k = \mathbf{h}_k^{\mathsf{H}} \sum_{i=1}^{K} \mathbf{b}_i s_i + n_k = \mathbf{h}_k^{\mathsf{H}} \mathbf{b}_k s_k + \mathbf{h}_k^{\mathsf{H}} \sum_{i \neq k} \mathbf{b}_i s_i + n_k, \tag{3}$$

where n_k represents the complex circularly symmetric additive white Gaussian noise at the k-th UT, with a mean of 0 and variance N_0. We assume that the noise variance is equal among all the UTs. The term $\mathbf{b}_k \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ represents the precoding vector for the k-th UT. The transmitted signal from the satellite to the k-th UT is denoted by s_k, with its power set to 1, i.e., $\mathbb{E}|s_k|^2 = 1$.

The total power consumption in the LEO satellite communication system under consideration can be modeled as [15]

$$P^{\mathrm{total}} = \sum_{k=1}^{K} \xi ||\mathbf{b}_k||_2^2 + P_{\mathrm{t}}, \tag{4}$$

where P_{t} signifies the power utilized by the satellite transmitter, and $1/\xi$ indicates the efficiency of the transmit power amplifier. The total power consumption for a fully digital transmitter at the LEO satellite is expressed as

$$P_{\mathrm{t}} = N_{\mathrm{t}} P_{\mathrm{RFC}} + P_{\mathrm{LO}} + P_{\mathrm{BB}}, \tag{5}$$

where P_{BB} is the power consumed by baseband digital precoder, and P_{LO} is the power utilized by the local oscillator. $P_{\mathrm{RFC}} = P_{\mathrm{DAC}} + P_{\mathrm{mixer}} + P_{\mathrm{LPF}} + P_{\mathrm{BBA}}$, where P_{DAC}, P_{mixer}, P_{LPF}, and P_{BBA} denote the power utilized by the digital-to-analog converter, mixer, low-pass filter, and baseband amplifier, respectively.

C. Problem Formulation

The signal-to-interference-plus-noise ratio (SINR) in (3) for the k-th UT during downlink transmission can be defined as

$$\mathsf{SINR}_k \triangleq \frac{\mathbf{b}_k^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{b}_k}{\sum_{\ell \neq k} \mathbf{b}_\ell^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{b}_\ell + N_0}. \tag{6}$$

The average data rate of the k-th UT is specified as $\mathcal{R}_k = \mathbb{E}_{\mathbf{h}_k}\{\log(1 + \mathsf{SINR}_k)\}$, and $\mathcal{R}_k$ can be as

$$\mathcal{R}_k = \mathbb{E}\left\{\log_2\left(1 + \frac{\mathbf{b}_k^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{b}_k}{N_0 + \sum_{\ell \neq k} \mathbb{E}_{\mathbf{h}_k}\left\{\mathbf{b}_\ell^{\mathsf{H}} \mathbf{h}_k \mathbf{h}_k^{\mathsf{H}} \mathbf{b}_\ell\right\}}\right)\right\}. \tag{7}$$

The average rate $\mathcal{R}_k$ generally lacks a clear analytical expression, making it challenging to manage challenging to optimize with respect to (7). One possible solution is to use a Monte Carlo method to estimate the required value. However, this approach has a high computational complexity. To address this, we substitute the ergodic data rate $\mathcal{R}_k$ with an upper bound in the optimization process [8]

$$\mathcal{R}_k \leq \bar{\mathcal{R}}_k \triangleq \log\left(1 + \frac{\gamma_k |\mathbf{v}_k^{\mathsf{H}} \mathbf{b}_k|^2}{\sum_{\ell \neq k} \gamma_k |\mathbf{v}_k^{\mathsf{H}} \mathbf{b}_\ell|^2 + N_0}\right). \tag{8}$$

To formulate maximize the total achievable system EE of the digital precoding in the LEO satellite communication system

$$\mathcal{P}_1: \max_{\{\mathbf{b}_k\}_{k=1}^K} \frac{B_{\mathrm{w}} \sum_{k=1}^K \bar{\mathcal{R}}_k}{P^{\mathrm{total}}}, \tag{9a}$$

$$\text{s.t.} \sum_{k=1}^K ||\mathbf{b}_k||_2^2 \leq P_{\max}, \tag{9b}$$

where $P_{\max}$ is the maximum allowed total transmit power. It is important to observe that problem $\mathcal{P}_1$E presented in (9) is categorized as a fractional nonconvex problem.

III. GNN-Enabled Deep-unfolding Precoding Designs

In this section, we delve into the advancements made in digital precoding for MIMO LEO satellite networks, focusing on the integration of innovative techniques that enhance both performance and interpretability. We explore how the deep unfolding of the Dinkelbach algorithm combined with the WMMSE approach, along with the incorporation of the Taylor expansion method, collectively contribute to overcoming the computational challenges posed by the complex channel conditions inherent in satellite communications.

A. Dinkelbach Algorithm and WMMSE Algorithm

Using the Dinkelbach algorithm [16], we iteratively address $\mathcal{P}_1$ by addressing a series of sub-problems. Let n represent the iteration index, and ρ_n be the auxiliary variable at the n-th iteration. The sub-problem for the n-th iteration is then formulated as follows [17]

$$\mathcal{P}_2^n: \underset{\{\mathbf{b}_k^n\}_{k=1}^K, \rho_n}{\text{maximize}} \quad F(\rho_n) = B_{\mathrm{w}} \sum_{k=1}^K \bar{\mathcal{R}}_k - \rho_n P^{\mathrm{total}}, \tag{10a}$$

$$\text{s.t.} \quad \sum_{k=1}^K ||\mathbf{b}_k^n||_2^2 \leq P_{\max}. \tag{10b}$$

The variables $\{\mathbf{b}_k^n\}_{k=1}^K$ and ρ_n are optimized in an iterative manner in subproblem $\mathcal{P}_2^n$. More specifically, the value of ρ_{n+1} for the subsequent subproblem P_2^{n+1} for a given $\{\mathbf{b}_k^n\}_{k=1}^K$ is determined by

$$\rho_{n+1} = \frac{B_{\mathrm{w}} \sum_{k=1}^K \bar{\mathcal{R}}_k}{P^{\mathrm{total}}}. \tag{11}$$

We omit the index n since the procedure is identical for each subproblem. As demonstrated in [18], the WMMSE problem in (12) is equivalent to the maximization problem in (10), meaning that the optimal solution $\{\mathbf{b}_k\}_{k=1}^K$ is the same for both problems,

$$\mathcal{P}_3: \underset{\{\mathbf{b}_k, \omega_k, u_k\}_{k=1}^K}{\text{minimize}} \; B_{\mathrm{w}} \sum_{k=1}^K (\omega_k e_k - \log \omega_k) + \rho P^{\mathrm{total}}, \tag{12a}$$

$$\text{s.t.} \sum_{k=1}^K ||\mathbf{b}_k||_2^2 \leq P_{\max}, \tag{12b}$$

where ω_k and u_k are that represent the receiving matrix and the weight matrix for the UT k, respectively, and $e_k \triangleq |u_k \sqrt{\gamma_k} \mathbf{v}_k^{\mathsf{H}} \mathbf{b}_k - 1|^2 + \sum_{i \neq k} \gamma_k |u_k \mathbf{v}_k^{\mathsf{H}} \mathbf{b}_i|^2 + N_0 |u_k|^2$.

Notably, the three variables in $\mathcal{P}_3$ in (12) are highly interdependent, making it difficult to solve directly. The third variable can be updated by fixing two of the three variables. The update expressions for ω_k, u_k, and $\mathbf{b}_k$ are provided as follow

$$u_k = \sqrt{\gamma_k} \mathbf{v}_k^{\mathsf{H}} \mathbf{b}_k \left(\sum_{i=1}^K \gamma_k |\mathbf{v}_k^{\mathsf{H}} \mathbf{b}_i|^2 + N_0 \right)^{-1}, \tag{13}$$

$$\omega_k = \Bigg(|u_k \sqrt{\gamma_k} \mathbf{v}_k^{\mathsf{H}} \mathbf{b}_k - 1|^2 + \sum_{i \neq k}^K \gamma_k |u_k \mathbf{v}_k^{\mathsf{H}} \mathbf{b}_i|^2 + N_0 |u_k|^2 \Bigg)^{-1}, \tag{14}$$

$$\mathbf{b}_k^{\mathrm{opt}} = \omega_k (\sqrt{\gamma_k}) u_k^* \Bigg(\sum_{k=1}^K B_{\mathrm{w}} \omega_k |u_k|^2 \gamma_k \mathbf{v}_k \mathbf{v}_k^{\mathsf{H}} + \left(\rho \xi + a^{\mathrm{opt}} \right) \mathbf{I} \Bigg)^{-1} \mathbf{v}_k. \tag{15}$$

In order to clearly describe our algorithm, we define $u_k^\ell = U_\ell(\mathbf{b}_k^{\ell-1}, \mathbf{v}_k, \gamma_k)$, $\omega_k^\ell = W_\ell(u_k^\ell, \mathbf{b}_k^{\ell-1}, \mathbf{v}_k, \gamma_k)$, $\mathbf{b}_k^\ell = B_\ell(u_k^\ell, \omega_k^\ell, \mathbf{v}_k, \gamma_k)$, where U_ℓ, W_ℓ, and B_ℓ denote the iterative mapping functions at the ℓ-th iteration. The architecture of a deep unfolding model is as shown in Fig. 1, which appears to be used for iterative optimization or inference tasks. It shows a series of iterative blocks U_ℓ, W_ℓ, and B_ℓ that process the input data $\mathbf{b}_k^\ell$ sequentially through multiple iterations. Each block likely represents a different operation or transformation applied to the data, with the iterations continuing until a final output $\mathbf{b}_k^L$ is produced.

B. Deep Unfolding of the Dinkelbach Algorithm Combined with the WMMSE Approach

The inverse of the matrix has a very high complexity, and we need to optimize the process of the matrix. Define $a_k = \sum_{i=1}^K \gamma_k |\mathbf{v}_k^{\mathsf{H}} \mathbf{b}_i|^2 + N_0$ and $\mathbf{M}_k = \sum_{k=1}^K B_{\mathrm{w}} \omega_k |u_k|^2 \gamma_k \mathbf{v}_k \mathbf{v}_k^{\mathsf{H}} + (\rho \xi + \lambda^{\mathrm{opt}}) \mathbf{I}$. Matrix inversion has a high complexity in iterative algorithms, which affects the application of precoding algorithms in MIMO LEO satellite communications. We introduce GNNs to achieve an end-to-end matrix inversion process to address it. A graph is defined as $G = (V, E, s, t)$, where V denotes the set of nodes, E represents the set of edges, $s: V \longrightarrow \mathbb{C}^{d_1}$ maps each node to its corresponding features, where d_1 is the node feature

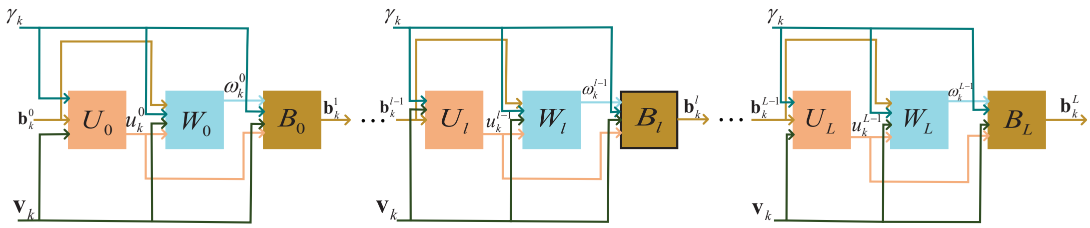

Fig. 1: The diagram illustrates the architecture of a deep unfolding model, which appears to be used for iterative optimization or inference tasks.

dimension and $t : E \longrightarrow \mathbb{C}^{d_2}$ maps edges to their features, where d_2 is the edge feature dimension.

For a satellite with N_t antennas, $\mathbf{M}_k$ is a complex matrix of size $N_t \times N_t$. Each antenna on the satellite is represented as a vertex, and the connections between antenna vertices are modeled as edges. The edge is represented as (v_i, v_j) between the v_i-th antenna vertex and the v_j-th antenna vertex, we apply a GNN with edge updates, where information from neighboring vertices connected by those edges is aggregated to update the hidden representation of each edge. In our algorithm, we need to invert a_k, e_k, and $\mathbf{M}_k$. Since a_k and e_k are scalar inversions, their inverses are straightforward, but $\mathbf{M}_k$ is an $N_t \times N_t$ matrix, making its inversion highly complex, requiring approximation. To address this, we use a GNN to approximate the matrix inversion of $\mathbf{M}_k$. Each row of $\mathbf{M}_k$ is distributed across different edges in the GNN, as illustrated in Fig. 2. The hidden representation $f{v_i v_j}^{(\ell)}$ of an edge (v_i, v_j) in layer ℓ depends on the specific edge features

$$f^{(\ell)}_{v_iv_j} = \mathsf{C}\Bigg(f^{(\ell-1)}_{v_i}, \underbrace{\mathsf{P_s}\Big(\mathsf{Q_s}(f^{(\ell-1)}_{iv_j}, s^{(\ell)}), i \in \mathcal{N}_{\mathsf{AN}}(v_i)\Big)}_{\text{(a)}}, \underbrace{\mathsf{P_d}\Big(\mathsf{Q_d}(f^{(\ell-1)}_{v_ij}, d^{(\ell)}), j \in \mathcal{N}_{\mathsf{AN}}(v_j)\Big)}_{\text{(b)}}, c^{(\ell)}\Bigg). \tag{16}$$

The terms (a) and (b) represent the aggregated information from the neighboring edges connected to edge (v_i, v_j) via AN_{v_i} and AN_{v_j}, respectively. The functions $\mathsf{P_s}(\cdot)$ and $\mathsf{P_d}(\cdot)$ denote the pooling functions utilized to aggregate the hidden representations of the neighboring edges related to edge (n, k). $\mathcal{N}_{\mathsf{AN}}(v_i) \triangleq 1, \cdots, v_i - 1, v_i + 1, \cdots, N_t$ and $\mathcal{N}_{\mathsf{AN}}(v_j) \triangleq 1, \cdots, v_j - 1, v_j + 1, \cdots, N_t$ denote the sets of indices without antenna v_i and without antenna v_j, respectively. The function $\mathsf{C}(\cdot, c^{(\ell)})$, $\mathsf{Q_s}(\cdot, s^{(\ell)})$ and $\mathsf{Q_d}(\cdot, d^{(\ell)})$ are a combination function with trainable parameters $c^{(\ell)}$ and processing functions with trainable parameters $s^{(\ell)}$ and $d^{(\ell)}$. Consequently, if both $\mathsf{Q_s}(\cdot, s^{(\ell)})$, $\mathsf{Q_d}(\cdot, d^{(\ell)})$ and $\mathsf{C}(\cdot, c^{(\ell)})$ are linear functions, the update equation for edge (v_i, v_j) in (16) can be rewritten as follows

$$f^{(\ell)}_{v_iv_j} = \sigma\Big(\underbrace{c \cdot f^{(\ell-1)}_{v_iv_j}}_{\text{(a)}} + \underbrace{\mathsf{P_s}\big(s \cdot f^{(\ell-1)}_{iv_j}, i \in \mathcal{N}_{\mathsf{AN}}(v_i)\big)}_{\text{(b)}} + \underbrace{\mathsf{P_d}\big(d \cdot f^{(\ell-1)}_{v_ij}, j \in \mathcal{N}_{\mathsf{AN}}(v_j)\big)}_{\text{(c)}} \Big). \tag{17}$$

C. The Taylor Expansion Method to Approximate Matrix Inversion within the GNN Framework

Since the matrix inversion has high computational complexity, employing Taylor expansion offers a promising approach for solving matrix inverses. Notably, the inverse matrix $\mathbf{P}^{-1}$ can be approximated by a first-order Taylor expansion around $\mathbf{P}_0$

$$\mathbf{P}^{-1} = 2\mathbf{P}_0^{-1} - \mathbf{P}_0^{-1}\mathbf{P}\mathbf{P}_0^{-1}. \tag{18}$$

The Taylor expansion can be utilized as an approximation for $\mathbf{P}^{-1}$, which is closer to $\mathbf{P}^{-1}$ than $\mathbf{P}_0$. We can iteratively employ the first-order Taylor expansion for a more precise approximation. Observing this iterative equation resembles the update equation of edge updates in GNNs. The iterative equation for the (v_i, v_j) element of $\mathbf{F}^{(\ell)}$ can be expressed as

$$f^{(\ell)}_{v_iv_j} = 2f^{(\ell-1)}_{v_iv_j} - \sum_{j=1}^{N_t} t^{(\ell-1)}_{jv_j} f^{(\ell-1)}_{v_ij}. \tag{19}$$

Compared to (17), (19) can also be seen as the update equation for edge (v_i, v_j), where $f^{(\ell)}_{v_iv_j}$ is the underlying representation of edge (v_i, v_j) in layer ℓ. The process begins by aggregating the foundational representations of adjacent edges in layer $(\ell - 1)$, which are then combined with the hidden representation of edge (v_i, v_j) from the same layer. While this approach is similar to edge updates, it does not represent the exact update equation. However, $t^{(\ell-1)}_{v_iv_j}$ can be embedded into the update equation. In layer ℓ, the hidden representation of edge (v_i, v_j) is computed in two steps. For edge (v_i, v_j), select $f^{(\ell-1)}_{v_iv_j}$ from the input and output the vector connected by the two terms in (17), i.e., $\mathbf{g}^{(\ell-1)}_{v_iv_j} = [f^{(\ell-1)}_{v_i,v_j}, \sum_{j=1}^{N_t} t^{(\ell-1)}_{jv_j} f^{(\ell-1)}_{v_ij}]^{\mathsf{T}}, v_i = 1, \cdots, N_t, v_j = 1, \cdots, N_t$.

The hidden representation of edge (v_i, v_j) is updated by

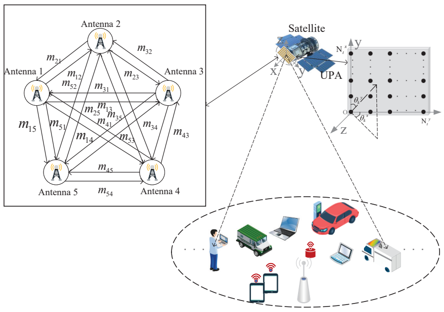

Fig. 2: Downlink LEO satellite communication system and its graphical representation with five antennas as an example.

replacing $f_{v_iv_j}^{(\ell-1)}$ with $\mathbf{g}_{v_iv_j}^{(\ell-1)}$

$$f_{v_iv_j}^{(\ell)} = \sigma\big(\mathbf{c}^{\mathsf{T}} \cdot \mathbf{g}_{v_iv_j}^{(\ell-1)} + \mathbf{s}^{\mathsf{T}} \cdot \sum_{i=1}^{N_t} \mathbf{g}_{iv_j}^{(\ell-1)} + \mathbf{d}^{\mathsf{T}} \cdot \sum_{j=1}^{N_t} \mathbf{g}_{v_ij}^{(\ell-1)}\big), \tag{20}$$

where $\mathbf{c} = [c_0, c_1]^{\mathsf{T}}$, $\mathbf{s} = [s_0, s_1]^{\mathsf{T}}$, and $\mathbf{d} = [d_0, d_1]^{\mathsf{T}}$ are trainable parameters. After substituting all parameters into (20), it can be rewritten as

$$\begin{aligned} f_{v_iv_j}^{(\ell)} = \sigma\Big(&\underbrace{\bar{c} f_{v_iv_j}^{(\ell-1)}}_{(a)} + \underbrace{\sum_{i=1, i\neq v_i}^{N_t} \bar{s} f_{iv_j}^{(\ell-1)}}_{(b)} + \underbrace{\sum_{j=1, j\neq v_j}^{N_t} \bar{d} f_{v_ij}^{(\ell-1)}}_{(c)} \\ &+ \underbrace{\sum_{i=1, i\neq v_i}^{N_t} \sum_{j=1, j\neq v_j}^{N_t} s_1 t_{jv_j}^{(\ell-1)} f_{ij}^{(\ell-1)}}_{(d)}\Big), \end{aligned} \tag{21}$$

where $\bar{c} = (c_0+s_0+d_0+c_1 t_{v_jv_j}^{(\ell-1)} + s_1 \sum_{j=1}^{N_t} t_{v_jj}^{(\ell-1)} + s_1 t_{v_jv_j}^{(\ell-1)})$, $\bar{s} = (s_0 + s_1 t_{v_jv_j}^{(\ell-1)})$, $\bar{d} = (c_1 t_{jv_j}^{(\ell-1)} + d_0 + s_1 t_{jv_j}^{(\ell-1)} + t_{jv_j}^{(\ell-1)})$.

In (21), compared with (17), terms (a), (b), and (c) correspond to each other, except that our trainable parameters have been replaced, with c replaced by $\bar{c}$, s replaced by $\bar{s}$, and d replaced by $\bar{d}$. As for term (d), it is an additional term. Since information from non-adjacent edges can be aggregated by layering multiple layers, we can eliminate term (d) in (21) to reduce computational complexity without sacrificing the expressive power of GNNs. We express the above method as $\mathbf{M}^{-1} = F(\mathbf{M})$, then our algorithm can be expressed as in **Algorithm 1**.

The network is trained in an unsupervised manner, as the optimal precoder remains unknown, it only requires channel-related samples without the need for acquiring the optimal precoder. The loss function for the network is designed to approximate the upper bound of system efficiency, which serves as the objective function upper bound for problem $\mathcal{P}_1$. Consequently, the loss function is defined as follows $\mathcal{L}(\mathbf{\Theta}) = -\frac{1}{\mathcal{D}} \sum_{d=1}^{\mathcal{D}} \frac{B_{\mathrm{w}} \sum_{k=1}^{K} \bar{\mathcal{R}}_k}{P^{\mathrm{total}}}$, where $\mathcal{D}$ is the size of the training batch.

Algorithm 1 GNN-enabled Deep Unfolding by Taylor Expansion Iterative WMMSE Precoding Algorithm.

Input: Thresholds $\epsilon_1, \epsilon_2 > 0; n = 0; \rho_n = 0$; Number of layers, L.
Output: Precoding vector $\mathbf{b}_k$.

while $F(\rho_n) > \epsilon_1$ **do**
  Initialize $\mathbf{b}_k$ such that $||\mathbf{b}_k||_2^2 = P/K$
  repeat
    $\omega_k' = \omega_k$,
    Calculate u_k by Eq. (13),
    Calculate ω_k by Eq. (14),
    $f_{v_iv_j}^{(0)} \leftarrow \sum_{k=1}^{K} B_{\mathrm{w}} \omega_k |u_k|^2 \gamma_k \mathbf{v}_k \mathbf{v}_k^{\mathsf{H}} + (\rho\xi + \lambda^{\mathrm{opt}})\,\mathbf{I}$,
    for 1 to L **do**
      Calculate $f_{v_iv_j}^{(\ell)}$ by Eq. (21)
    end for
    $F(M) \leftarrow f_{v_iv_j}^{(L)}$;
    $\mathbf{b}_k^{\mathrm{opt}} = \omega_k(\sqrt{\gamma_k}) u_k^* F(\mathbf{M}) \mathbf{v}_k$.
  until $|\sum_k \log \omega_k - \sum_k \log \omega_k'| < \epsilon_2$.
  $F(\rho_n) = (B_{\mathrm{w}} \sum_{k=1}^{K} \bar{R}_k - \rho_n P^{\mathrm{total}})|_{\mathbf{b}_k = \mathbf{b}_k^{\mathrm{opt}}}$.
  $\rho_{n+1} = \frac{B_{\mathrm{w}} \sum_{k=1}^{K} \bar{R}_k}{P^{\mathrm{total}}}\Big|_{\mathbf{b}_k = \mathbf{b}_k^{\mathrm{opt}}}$.
  n=n+1;
end while

TABLE I: Simulation Parameters

Parameter	Value
Bandwidth B_{w}	20 MHz
Number of antennas $N_{\mathrm{t}}^{\mathrm{x}} \times N_{\mathrm{t}}^{\mathrm{y}}$	8×8
Amplifier effectiveness $1/\xi$	0.5
Number of users K	10
Carrier frequency f_c	2 GHz
Signal to Noise Ratio	0 dB
Total transmission power constraint P_{max}	10

IV. Simulation Results

This section provides simulation results to assess the algorithm's performance concerning (9) in the context of an LEO satellite downlink communication system. Regarding the iterative algorithm, the iteration threshold is set to 10^{-5}. All matrix-form trainable parameters are initialized as zero matrices for the deep unfolding algorithm for massive MIMO LEO satellite precoding. Typical parameter settings are provided in Table I. In order to better reflect the good performance, compare our approach with the one in [9] named liner deep unfolding.

Fig. 3 showcases the EE efficiency achieved by both the iterative and deep unfolding algorithms as the number of antennas, N_t, increases. The graph demonstrates that with a higher count of LEO satellite antennas, there is a notable improvement in system energy efficiency for both the iterative LEO satellite precoding algorithm and the deep unfolding precoding algorithm. This enhancement is due to the increased antenna count, which boosts signal transmission

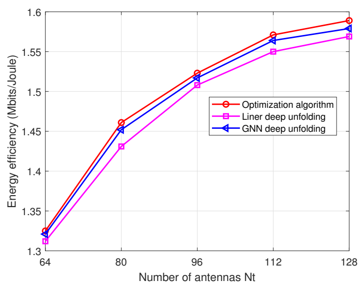

Fig. 3: Energy efficiency versus the number of antennas.

efficiency and extends signal coverage to accommodate more users. Remarkably, the deep unfolding precoding **Algorithm 1** achieves performance that is on par with the iterative algorithm across different antenna configurations.

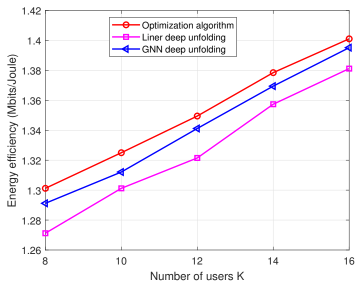

Fig. 4: Energy efficiency versus the number of users.

Fig. 4 displays the test results of a deep unfolding algorithm trained on datasets generated under specific system parameters and varying conditions. The model, once trained, was directly utilized in scenarios featuring the same number of antennas but varying numbers of users, eliminating the requirement for retraining. By comparing these results with those of a deep expansion algorithm whose parameters were adjusted, we observed that the performance of networks scaled to accommodate different numbers of users while maintaining the same number of antennas was nearly equivalent to that of networks specifically trained for varying user counts.

V. Conclusion

In this paper, we addressed the challenges in EE efficiency in LEO satellite communication for 6G networks, mainly focusing on integrating massive MIMO technology. Given the limited power of LEO satellites, optimizing energy consumption while maintaining high system performance was crucial. The paper proposed innovative AI approaches to tackle these challenges, specifically in precoding for massive MIMO LEO satellite networks. The paper developed a deep unfolding technique that combined the Dinkelbach algorithm with the WMMSE approach, enhancing convergence speed and computational efficiency. The Taylor expansion method was also incorporated to approximate matrix inversion within the GNN, further improving both the interpretability and performance of the proposed methods. Numerical experiments demonstrated that the performance of the GNN deep unfolding method in EE closely resembles that of optimization algorithms.

References

[1] C.-X. Wang, X. You, , and et al., “On the road to 6G: Visions, requirements, key technologies, and testbeds,” *IEEE Commun. Surv. Tutorials*, vol. 25, no. 2, pp. 905–974, 2st Quart. 2023.

[2] R. Deng, B. Di, and et al., “Holographic MIMO for LEO satellite communications aided by reconfigurable holographic surfaces,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 10, pp. 3071–3085, Oct. 2022.

[3] L. You, K.-X. Li, and et al., “Massive MIMO transmission for LEO satellite communications,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1851–1865, Aug. 2020.

[4] L. You, Y. Zhu, and et al., “Ubiquitous integrated sensing and communications for massive MIMO LEO satellite systems,” *IEEE Internet Things Mag.*, vol. 7, no. 4, pp. 30–35, Jul. 2024.

[5] E. Lagunas, S. Chatzinotas, and B. Ottersten, “Low-earth orbit satellite constellations for global communication network connectivity,” *Nat. Rev. Electr. Eng.*, 2024.

[6] Y. Huang, L. You, and et al., “Qos-aware precoding for dual-polarized downlink massive MIMO LEO satellite communications,” *Space Sci. Technol.*, vol. 4, p. 0178, Nov. 2024.

[7] K.-X. Li, L. You, and et al., “Downlink transmit design for massive MIMO LEO satellite communications,” *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 1014–1028, Feb. 2022.

[8] L. You, X. Qiang, and et al., “Hybrid analog/digital precoding for downlink massive MIMO LEO satellite communications,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 8, pp. 5962–5976, Jan. 2022.

[9] Q. Hu, Y. Cai, and et al., “Iterative algorithm induced deep-unfolding neural networks: Precoding design for multiuser MIMO systems,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1394–1410, Oct. 2021.

[10] Z. Jin, L. You, and et al., “An I2I inpainting approach for efficient channel knowledge map construction,” *IEEE Trans. Wirel. Commun.*, pp. 1–1, Dec. 2024.

[11] ——, “GDM4MMIMO: Generative diffusion models for massive MIMO communications,” 2024. [Online]. Available: https://arxiv.org/abs/2412.18281

[12] A. Chowdhury, G. Verma, C. Rao, A. Swami, and S. Segarra, “Unfolding WMMSE using graph neural networks for efficient power allocation,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 9, pp. 6004–6017, Sep. 2021.

[13] Q. Hu, Y. Liu, Y. Cai, G. Yu, and Z. Ding, “Joint deep reinforcement learning and unfolding: Beam selection and precoding for mmwave multiuser MIMO with lens arrays,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2289–2304, Aug. 2021.

[14] K. Kang, Q. Hu, Y. Cai, G. Yu, J. Hoydis, and Y. C. Eldar, “Mixed-timescale deep-unfolding for joint channel estimation and hybrid beamforming,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 9, pp. 2510–2528, Sep. 2022.

[15] L. You, X. Qiang, and et al., “Massive MIMO hybrid precoding for LEO satellite communications with twin-resolution phase shifters and nonlinear power amplifiers,” *IEEE Trans. Commun.*, vol. 70, no. 8, pp. 5543–5557, Aug. 2022.

[16] A. Zappone and E. A. Jorswieck, “Energy efficiency in wireless networks via fractional programming theory,” *Found. Trends Commun. Inf. Theory*, vol. 11, pp. 185–396, Jun. 2015.

[17] R. G. Ródenas, M. L. López, and D. Verastegui, “Extensions of Dinkelbach’s algorithm for solving non-linear fractional programming problems,” *Top*, vol. 7, pp. 33–70, 1999.

[18] Q. Shi, M. Razaviyayn, Z.-Q. Luo, and C. He, “An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel,” *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4331–4340, Jul. 2011.