Spatiotemporal-Attention Based Channel Prediction for UAV-RIS-Assisted LEO Satellite MIMO Communications

Spatiotemporal-Attention Based Channel Prediction for UAV-RIS-Assisted LEO Satellite MIMO Communications

Mingyi Wang, Yizhou Peng, *Student Member, IEEE,* Ruofei Ma, *Member, IEEE,* Gongliang Liu, *Member, IEEE,* Weixiao Meng, *Senior Member, IEEE,* Carla Fabiana Chiasserini, *Fellow, IEEE,* and Roberto Garello, *Senior Member, IEEE*

***Abstract*—Low earth orbit (LEO) satellite communications play a critical role in achieving global connectivity, yet they face significant challenges due to high satellite mobility and incomplete channel state information (CSI). Moreover, the integration of reconfigurable intelligent surfaces (RIS) in certain scenarios introduces additional complexities. In this paper, we propose a novel MIMO channel prediction framework tailored for LEO satellite communications involving unmanned aerial vehicle-mounted RIS (UAV-RIS), employing a spatiotemporal-attention (ST-attention) mechanism to capture both the spatial correlations among antennas and the temporal dynamics of rapidly varying channels. Furthermore, we leverage masked pretraining to enhance the model's robustness under scenarios of severe CSI incompleteness, enabling effective reconstruction of missing channel information. Comprehensive simulations demonstrate that our approach outperforms traditional model-based predictors, whether historical CSI is fully available or only partially observed.**

***Index Terms*—LEO satellite communications, MIMO channel prediction, reconfigurable intelligent surfaces (RIS), partial channel state information (pCSI), spatiotemporal-attention**

I. Introduction

LOW Earth orbit (LEO) satellite communication systems have attracted significant attention for providing seamless global connectivity and low-latency services [1], [2]. Compared to traditional geostationary Earth orbit (GEO) or medium Earth orbit (MEO) systems, LEO satellites operate at much lower altitudes, offering better link quality and reduced propagation delay.

This work is supported by Shandong Provincial Natural Science Foundation (ZR2023MF001, ZR2020MF141), the National Natural Science Foundation of China (61971156, 61801144). (*Corresponding authors: Ruofei Ma, Gongliang Liu)*

Mingyi Wang is with the Department of Communication Engineering, Harbin Institute of Technology, China, and also with the Department of Electronics and Telecommunications, Politecnico di Torino, Italy (e-mail: elewmy@163.com).

Yizhou Peng is with the College of Computing and Data Science, Nanyang Technological University, Singapore (e-mail: yizhou004@e.ntu.edu.sg).

Ruofei Ma, Gongliang Liu, and Weixiao Meng are with the Department of Communication Engineering, Harbin Institute of Technology, China (e-mail: maruofei@hit.edu.cn, liugl@hit.edu.cn, wxmeng@hit.edu.cn).

Carla Fabiana Chiasserini and Roberto Garello are with the Department of Electronics and Telecommunications, Politecnico di Torino, Italy (e-mail: carla.chiasserini@polito.it, roberto.garello@polito.it).

Integrating multiple-input multiple-output (MIMO) techniques and reconfigurable intelligent surfaces (RIS) into LEO satellite networks can further enhance communication quality, spectral efficiency, and coverage flexibility [3]. RIS modify the propagation environment by adjusting the phase shifts of incident signals, thereby enhancing signal strength, mitigating blockages, and improving link reliability. When RIS are dynamically deployed on mobile platforms, such as unmanned aerial vehicle-mounted RIS (UAV-RIS) [4], [5], system flexibility is further increased as their positions can be adapted to the prevailing channel conditions. Nonetheless, these benefits rely on continuously monitoring channel state information (CSI) [6] and accurately predicting future CSI based on historical observations, which is crucial for proactive resource allocation and robust link adaptation. However, achieving high-precision predictions is particularly challenging in dynamic LEO scenarios, where the positions of satellites, users, and mobile RIS fluctuate rapidly [7], [8].

Beyond the inherent challenges of channel prediction based on continuously observed CSI, highly dynamic conditions often lead to incomplete or "partial" CSI (pCSI), where channel measurements at certain time instances are either missing or inadequately captured [9]. As mentioned, the rapid orbital motion of LEO satellites and the mobility of UAV-RIS create rapidly time-varying channels, making it infeasible to obtain full CSI across all observation time slots. Moreover, limited uplink feedback, strict training overhead constraints, and sporadic measurement opportunities make the problem even more serious [10]. While classical methods typically assume sufficiently dense measurements or pilot signals [11], [12], the presence of pCSI in practice substantially complicates both channel estimation and prediction in LEO networks.

Given that channel prediction is crucial for optimizing resource allocation, reducing overhead, and ensuring robust communication performance in highly dynamic LEO satellite networks, it remains imperative to pursue accurate and proactive prediction strategies, particularly under pCSI conditions.

Early deep learning based time series models, such as Long-Short Term Memory (LSTM) networks [13], [14], can effectively capture temporal dependencies over short and medium ranges and have also been used for channel prediction [11], [12]. Zhang *et al.* [15] proposed a prediction framework that uses LSTM to predict dynamic interference periods and atmospheric attenuation, without estimating precise CSI values.

However, these models often struggle with sequences covering long durations and with the increased dimensionality inherent in MIMO channels. In particular, the integration of RIS not only increases the prediction dimension but also introduces complex spatial dependencies, which further complicate the accurate capture of channel characteristics. Recently, neural architectures incorporating attention mechanisms, such as the Transformer model [16], have been proposed to address these limitations by explicitly modeling dependencies across longer sequences. Unlike traditional recurrent networks, attention mechanisms enable the model to directly access and weigh relevant information from distant positions, effectively capturing complex temporal patterns and long-range relationships within the sequences. In [17], Transformer models were applied to channel prediction in terrestrial mobile networks, where the models can effectively capture the latent dynamics of the channel and mitigate the impact of user mobility on prediction accuracy. Nevertheless, conventional Transformer architectures primarily focus on the temporal dimension and do not explicitly account for spatial correlations, which makes them less effective for scenarios exhibiting rapid variations in both space and time, such as LEO satellite communications where spatial interactions are critical. The Spacetimeformer [18] addresses this limitation by extending the temporal-only Transformer architecture to jointly handle multiple dimensions, explicitly integrating attention mechanisms across both temporal and spatial domains. This multidimensional approach enhances forecasting capabilities, particularly in tasks involving complex interactions, such as traffic and weather prediction. However, applying joint spatial and temporal attention mechanisms to MIMO channel prediction remains unexplored. Another critical challenge arises from the high dimensionality of the channel data introduced by large-scale MIMO arrays and RIS deployments. As the number of satellite transmit antennas, RIS elements, or user antennas increases, the effective channel dimension can grow essentially quadratically, leading to a dimension explosion. This dramatically increases the memory footprint, computational resources, and inference latency required for both training and deployment, thereby severely undermining the timeliness of channel predictions.

Besides the general challenges of channel prediction in highly dynamic environments, pCSI is still a critical issue. Incomplete channel measurements disrupt the temporal structure of historical sequences, thereby degrading the performance of conventional deep learning architectures and limiting their effectiveness in practical LEO satellite scenarios. Furthermore, obtaining comprehensive and high-quality labeled datasets is particularly challenging in LEO satellite communications because rapid orbital motion and limited observation windows often result in channel data that are both incomplete and scarce [19]. The insufficient volume and diversity of data pose a major bottleneck for training machine learning models, especially when addressing the complex missing patterns inherent in pCSI scenarios, which require large-scale and diverse labeled datasets to achieve robust performance.

In summary, LEO satellite channel prediction faces three key challenges: (1) *Fast spatiotemporal variation*: the coupled motion of the satellite, UAV-RIS, and ground users introduces rapid Doppler shifts and angle drifts, significantly increasing the difficulty of accurate prediction; (2) *High-dimensional tensor structure*: the composite satellite–RIS–user link yields channel matrices whose size grows quadratically with the numbers of antennas and RIS elements, quickly exhausting computational resources and increasing the burden on model training; (3) *pCSI*: limited feedback bandwidth, link outages, and intentional undersampling create large temporal gaps in the observed CSI, severely degrading prediction performance.

Recent advances in representation learning have motivated the adoption of pretraining strategies in communication systems to address challenges such as pCSI. One popular approach of pretraining is self-supervised learning (SSL) [20], which projects input sequences into high-dimensional representations, capturing rich latent features without relying on labeled data. This strategy has been validated across various domains using models such as GPT [21], Hubert [22], and data2vec [23], which leverage extensive unsupervised data to enhance downstream tasks. By integrating this SSL-based pretraining technology into channel prediction frameworks, meaningful spatiotemporal features can be extracted even under conditions of incomplete input, thereby improving overall performance and generalizability.

To address pCSI conditions in satellite MIMO channel prediction and enhance both accuracy and robustness, we propose a novel spatiotemporal-attention (ST-attention) based architecture combined with an SSL pretraining strategy. Our main contributions can be summarized as follows:

- We propose a systematic modeling framework that unifies satellite mobility, UAV-RIS dynamics, ground user movement, and channel acquisition constraints under a single predictive model. By categorizing pCSI scenarios and addressing them with a holistic approach, this framework can robustly capture the spatiotemporal dependencies inherent in realistic LEO satellite communications. Furthermore, the framework is extensible to various orbital configurations and larger-scale MIMO/RIS systems.
- We introduce a ST-attention mechanism for channel prediction, going beyond conventional LSTM-based or temporal-only attention methods. Specifically, the proposed architecture jointly models spatial and temporal dependencies by decomposing the input sequence into spatial and temporal components and then applying dedicated attention modules. Through spatial embeddings (e.g., the relative positions of satellites and RIS, antenna array structures) and temporal features, the model effectively learns channel variations induced by satellite trajectories, RIS reconfigurations, and user mobility. This integrated approach enhances prediction accuracy for LEO satellite MIMO channels, even under continuous satellite motion and UAV-RIS mobility.
- Inspired by SSL techniques based on masked language modeling (e.g., BERT [24]), we adopt an SSL pretraining strategy that simulates pCSI by artificially masking random channel entries. The SSL model is trained to reconstruct these missing values based on the observed spatiotemporal context, thereby learning robust representa-

tions. Then, the learned parameters of the SSL model are used to initial the subsequent channel prediction model, providing a strong knowledge for fine-tuning on real-world scenarios where CSI may be partially available. This strategy is especially beneficial for small datasets, given that high-quality labeled data are particularly scarce in satellite communications, ultimately yielding more accurate and robust predictions under limited data conditions.

The remainder of this paper is organized as follows. Section II presents the system model. Section III describes the proposed ST-attention based prediction method for MIMO channel prediction. Then Section IV introduces the pretraining strategy designed to handle pCSI. Section VI discusses the simulation setup, experimental evaluations, and comparative results. Finally, Section VII concludes the paper.

II. System Model

This section presents the system model for a satellite MIMO communication network with a UAV-RIS. In Section II-A, we outline the overall architecture, including the LEO satellite, the UAV-RIS, and the ground users. Section II-B then describes the channel modeling and dataset construction process, covering path loss, small-scale fading, and Doppler effects. In Section II-C, we examine common pCSI outage patterns, providing insight for developing corresponding solutions.

A. System Architecture

The considered downlink communication scenario involves an LEO satellite equipped with multiple transmit antennas, a UAV-RIS, and multiple ground users each with receive antennas. The RIS comprises passive reflecting elements that adjust the phase and amplitude of incident signals rather than directly receiving them. The overall system model is illustrated in Fig. 1.

The LEO satellite employs a uniform planar array (UPA) consisting of N_S transmit antennas, and its motion is determined by its orbital trajectory. Let $\mathbf{p}_S(t)$ denote the satellite's position vector at time t. The UAV-RIS, composed of N_R reflecting elements, is capable of dynamically adjusting the phase of its incoming signals. Its position at time t is represented by $\mathbf{p}_R(t)$. On the ground, each user is equipped with a MIMO array comprising N_U receive antennas, and the position of the k-th user at time t is denoted by $\mathbf{p}_{U,k}(t)$.

Transmissions from the LEO satellite to each ground user occur via two distinct paths. One path is enhanced by the UAV-RIS, which reflects and intelligently modifies the signal to extend coverage and improve channel quality, while the other is the direct satellite-to-user link [25]. Both paths include line-of-sight (LOS) and non-line-of-sight (NLOS) components. As the LEO satellite follows its orbital trajectory, both the UAV-RIS and the ground users may move in different directions at varying velocities. This relative motion induces dynamic variations in the channel conditions, leading to time-varying and spatially diverse characteristics that complicate the accurate acquisition and prediction of the channel states.

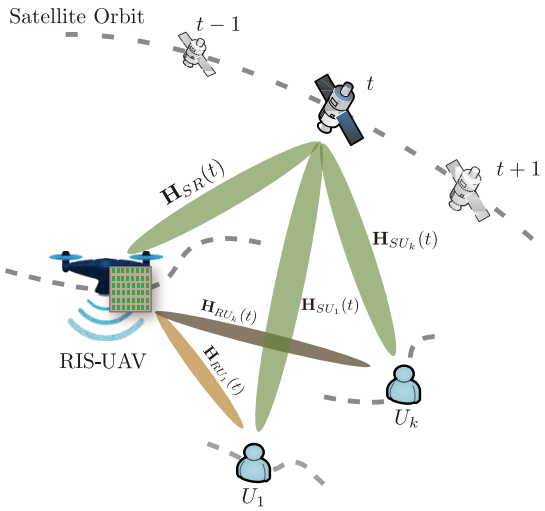


Fig. 1: System architecture and channel Prediction Model in LEO Satellite Communication with UAV-RIS

B. Channel Modeling and Dataset Construction

This subsection specifies the physical assumptions, mathematical formulation, and fully reproducible pipeline used to create the channel dataset. A compact symbol list is reported in Table I.

1) Geometry and Notation: At every discrete snapshot t, the Earth-centred–Earth-fixed (ECEF) positions $\mathbf{p}_S(t)$ of the LEO satellite, $\mathbf{p}_R(t)$ of the UAV-RIS, and $\mathbf{p}_{U,k}(t)$ of the k-th user are recorded. All nodes are treated as three-dimensional rigid bodies, and their instantaneous attitudes are forwarded to the Phased Array System Toolbox so that every planar array is rotated to its true orientation before the steering vector is computed. The satellite adopts the common nadir-pointing mode: its body z-axis is fixed toward the Earth's centre, the x-axis lies in the along-track direction, and no spin is introduced during the short simulation window. The UAV frame varies with its roll, pitch, and yaw angles $(\varphi_R, \theta_R, \psi_R)$ as dictated by its attitude controller. The hand-held user device may also rotate freely in roll, pitch, and yaw, but its antenna array is always kept oriented skyward.

2) Large-Scale Attenuation: The free-space path loss between any two nodes A and B is given by Equation

$$L_{\mathrm{FS},AB}(t) = \left(\frac{4\pi\, d_{AB}(t)}{\lambda}\right)^2, \quad d_{AB}(t) = \|\mathbf{p}_B(t) - \mathbf{p}_A(t)\|. \tag{1}$$

Additional large-scale losses include atmospheric absorption, denoted $L_{\mathrm{atm}}(f_c, \vartheta)$, where ϑ is the elevation angle and the attenuation is obtained from the ITU-R P.676 [30] gas-absorption curves, and rain attenuation $L_{\mathrm{rain}}(f_c, R_{0.01})$, derived from ITU-R P.618 [31] using the 0.01-percentile rain rate $R_{0.01}$ of the local climate. The overall large-scale power gain is therefore given by Equation

$$\beta_{AB}(t) = 10^{-\left(L_{\mathrm{FS},AB}(t) + L_{\mathrm{atm}}(f_c,\vartheta) + L_{\mathrm{rain}}(f_c,R_{0.01})\right)/10}. \tag{2}$$

3) Small-Scale Fading and Doppler: Each elementary link follows the 3GPP TR 38.901 UMi-LoS cluster model [32,

Tab. 7.7.1-1]: 12 clusters, 20 rays per cluster, root-mean-square (RMS) delay spread $\sigma_\tau = 30$ ns, azimuth/elevation angle spreads $\sigma_{\mathrm{AOD}} = 5^\circ$ and $\sigma_{\mathrm{AOA}} = 10^\circ$. The complex baseband channel of the ℓ-th ray is

$$\mathbf{h}_{AB}^{(\ell)}(t) = \sqrt{\frac{\kappa_\ell}{K_R+1}}\, e^{j2\pi f_D^{(\ell)} t}\, \mathbf{a}_B\big(\boldsymbol{\theta}_B^{(\ell)}\big)\, \mathbf{a}_A^H\big(\boldsymbol{\theta}_A^{(\ell)}\big), \quad (3)$$

where K_R is the Rician factor (10 dB for the two satellite-related links, 5 dB for the RIS–user link), $f_D^{(\ell)} = \frac{v_{AB}(t)}{\lambda}\cos\varphi^{(\ell)}$ is the ray-level Doppler shift, and κ_ℓ is a normaliser ensuring $\sum_\ell \kappa_\ell = 1$. The MATLAB `nrCDLChannel` object automatically generates the time evolution of all ray phases; its autocorrelation converges to the classical Clarke–Jake's form $R_{hh}(\Delta t) = J_0\big(2\pi f_D^{\max}\Delta t\big)$.

4) Time- and Space Correlation Models:

Temporal: Although `nrCDLChannel` already realises Clarke–Jake's fading, we explicitly denote the equivalent first-order Gauss–Markov recurrence for each tap $\mathbf{h}_{t+1} = \alpha_t \mathbf{h}_t + \sqrt{1-\alpha_t^2}\mathbf{w}_t$, with $\alpha_t = J_0(2\pi f_D^{\max}\Delta t)$ and $\mathbf{w}_t \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$, so that readers can map the toolbox output to the analytical model.

Spatial: The spatial correlation dictated by the Clustered Delay Line (CDL) angle spreads is retained without any extra exponential filtering. Under this setting the channel covariance follows the standard Kronecker model: it factorises into the outer product of a transmit-side and a receive-side covariance matrix, each fixed solely by the array geometry and the CDL angle statistics, and is therefore independent of any further spatial filtering or Tx–Rx coupling assumptions.

5) Dataset Generation Pipeline:

- **Trajectory synthesis**: Satellite positions are obtained with the Simplified General Perturbations Model 4 (SGP4) algorithm from the publicly available Two-Line Elements (TLEs) of the International Space Station (ISS) [35]. This yields a representative LEO orbit whose altitude is on the order of several hundred kilometres, with an orbital period of about 90–100 min. The UAV-RIS operates at an altitude of 50–100 m and follows a Dubins path [36], namely the shortest curvature-constrained trajectory connecting randomly chosen way-points; the horizontal speed is sampled uniformly from 0.5 m/s to 30 m/s, and the minimum turn radius is set to 200 m. Ground users move according to a Gauss–Markov mobility model [37] with memory parameter $\alpha = 0.8$ and instantaneous speeds drawn uniformly from 0 m/s to 100 m/s.
- **Large-scale parameters**: For every snapshot we compute L_{FS}, L_{atm}, L_{rain}, and an i.i.d. log-normal shadow term $\chi_{\mathrm{dB}} \sim \mathcal{N}(0, 5^2)$ dB.
- **Cluster initialisation**: Delay/angle spreads and the Rician K_R factor are drawn once per epoch, guaranteeing intra-window consistency.
- **Small-scale realization**: At each time step, we invoke `nrCDLChannel` to generate the three complex MIMO sub-channels $\mathbf{h}_{SR}(t)$, $\mathbf{h}_{RU_k}(t)$, and $\mathbf{h}_{SU_k}(t)$. Here, $\mathbf{h}_{SR}(t) \in \mathbb{C}^{N_R \times N_S}$ denotes the satellite–RIS link, $\mathbf{h}_{RU_k}(t) \in \mathbb{C}^{N_U \times N_R}$ the RIS–user k link, and $\mathbf{h}_{SU_k}(t) \in \mathbb{C}^{N_U \times N_S}$ the direct satellite–user k link. The toolbox internally applies Doppler shifts, angle dispersion, and spatial correlation according to the CDL parameters.
- **Storage**: Each snapshot is stored as a single row of the dataset matrix in its flattened form, as given by (4), which is shown at the bottom of this page, where

  $$N = 2N_R N_S \;+\; 2K\,N_U N_R \;+\; 2K\,N_U N_S$$

  is the total feature dimension of each flattened snapshot. Here, each real/imaginary block is obtained by column-major vectorisation of the corresponding complex matrix. The real and imaginary parts are stored separately so that the entire dataset is a single real-valued tensor—this avoids complex-number support issues in many machine-learning frameworks and simplifies normalization. All entries are saved as 32-bit floats.

MATLAB R2024a with the *Satellite Communications*, *Phased Array*, and *Communications* toolboxes generates the requisite snapshots, ensuring the dataset can be reproduced without specialised hardware.

C. pCSI in LEO Satellite Communications

In LEO satellite communication systems, pCSI often arises due to high mobility and limited feedback bandwidth. Measurements may be lost, corrupted, or deliberately omitted to reduce communication and computational overhead. As LEO satellites move on their orbits at high speeds, limitations in feedback or measurement frequencies and inherent propagation delays, contribute to an increasing prevalence of incomplete CSI. The scarcity of reliable measurements significantly complicates channel estimation and prediction, ultimately degrading overall communication performance.

To organize the discussion of pCSI, we identify three representative patterns, as illustrated in Fig. 2, where $\hat{\mathbf{h}}(t)$ denotes the historical channel observation at time t, whose precise definition is provided in Section III-B. The first two patterns correspond to passive, undesired losses frequently observed in actual deployments, while the third pattern involves deliberate undersampling designed to conserve resources.

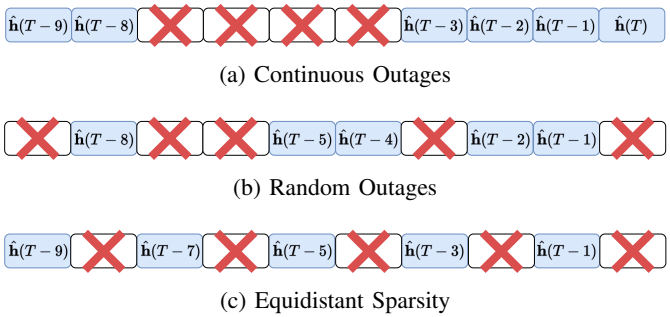
1) Continuous Outages: Extended disruptions in CSI acquisition can occur when satellites pass behind obstacles, during abrupt satellite handovers [38], or following a prolonged failure in the downlink feedback link. In these instances, consecutive time steps of CSI measurements are lost, resulting in contiguous gaps that may span a substantial portion of the observation window.

2) Random Outages: CSI measurements may be sporadically lost or corrupted due to sensor malfunctions, brief interference events, or transient communication errors. These intermittent disruptions break the temporal continuity of the

$$\begin{aligned}\mathbf{h}(t) = \big[\ &\mathrm{vec}\big(\Re\{\mathbf{h}_{SR}(t)\}\big)^{\mathsf{T}},\ \mathrm{vec}\big(\Im\{\mathbf{h}_{SR}(t)\}\big)^{\mathsf{T}},\ \big\{\mathrm{vec}\big(\Re\{\mathbf{h}_{RU_k}(t)\}\big)^{\mathsf{T}}\big\}_{k=1}^{K},\\ &\big\{\mathrm{vec}\big(\Im\{\mathbf{h}_{RU_k}(t)\}\big)^{\mathsf{T}}\big\}_{k=1}^{K},\ \big\{\mathrm{vec}\big(\Re\{\mathbf{h}_{SU_k}(t)\}\big)^{\mathsf{T}}\big\}_{k=1}^{K},\ \big\{\mathrm{vec}\big(\Im\{\mathbf{h}_{SU_k}(t)\}\big)^{\mathsf{T}}\big\}_{k=1}^{K}\big] \ \in\ \mathbb{R}^{1\times N},\end{aligned} \quad (4)$$

TABLE I: Physical-layer parameters used in the channel generator

Parameter	Value	Source	**Parameter**	Value	Source
Carrier frequency f_c	27 GHz	Ka band	Satellite array N_S	25/256 UPA	—
Bandwidth B	100 MHz	—	User antennas N_U	1 (handheld)	—
Element spacing	$\lambda/2$	Std. design	RIS panel N_R	9/81	—
Rician K_R (S–R, S–U)	10 dB	[33]	Rician K_R (R–U)	5 dB	[32]
RMS delay spread σ_τ	30 ns	[32]	Angle spreads $\sigma_{\mathrm{AOD}}/\sigma_{\mathrm{AOA}}$	$5^\circ/10^\circ$	[32]
Satellite velocity	7.4–7.6 km/s	LEO orbit	Max Doppler $f_D^{\max}$	6.8×10^5 Hz (@27 GHz)	computed
UAV speed v_R	0.5–30 m/s	Dubins path	Shadow fading σ_χ	5 dB	ITU-R P.1812 [34]



(a) Continuous Outages

(b) Random Outages

(c) Equidistant Sparsity

Fig. 2: Representative pCSI patterns in LEO satellite communication scenarios.

CSI data, complicating the application of standard reconstruction and making precise channel modeling and prediction more challenging.

3) Equidistant Sparsity: In satellite channel prediction, reducing the CSI sampling frequency conserves power and bandwidth. This deliberate undersampling creates uniformly spaced gaps in the measurement sequence and reduces data transmission and processing requirements. Although fewer measurements complicate reconstruction, the resource savings often justify the trade-off when structured sparsity is exploited.

These three pCSI patterns may occur individually or in combination. For example, a system that employs intentional undersampling can also suffer from unforeseen link failures. In all cases, missing or incomplete CSI degrades the performance of algorithms tracking time-varying channels. By classifying pCSI as continuous outages, random outages, and equidistant sparsity, we can more effectively address channel data losses under practical LEO communication constraints.

III. Proposed ST-attenton Based Channel Prediction Method

In this section, we introduce a ST-attention based framework for predicting future MIMO channel states in LEO satellite communications. The core idea is to employ a transformer-style encoder-decoder network that simultaneously models spatial correlations across multiple antennas and temporal dynamics driven by orbital motion, RIS reconfiguration, user mobility, and environmental variations. Unlike conventional interpolation or purely time-series approaches, the proposed method applies fine-grained attention over both antennas and time steps, which is particularly valuable in high-dimensional satellite communication scenarios. Specifically, in Section III-A we describe the transformer-based spatiotemporal modeling and highlight its advantages over traditional methods. Section III-B details the feature representation and input encoding strategy used to construct the channel observation tokens. Finally, the subsequent subsections present the design of the ST-attention mechanism and the training objective for network optimization.

A. Transformer-based Spatiotemporal Modeling

Transformers [16] were originally developed for sequence-to-sequence tasks in natural language processing (NLP) [39] tasks. They employ an attention mechanism to capture long-range dependencies by dynamically adjusting the weights assigned to different parts of the input. This attention mechanism contrasts with earlier recurrent architectures that process inputs sequentially and often fail to preserve long-range context.

Effective forecasting of LEO satellite channels benefits from simultaneously modeling temporal variations, such as satellite movement and user mobility, and spatial interactions across large antenna arrays. Relying solely on time-based attention (T-attention) may not adequately capture these high-dimensional dependencies. In contrast, ST-attention applies multi-head self-attention across both time steps and antennas, thereby revealing correlations among antennas at the same time as well as temporal dependencies within each antenna over different time instants.

Specifically, multi-head self-attention projects the input embeddings into query, key, and value representations for each head. By computing attention weights in parallel, each head learns distinct types of correlations, including dependencies among different antennas at a single time step and temporal correlations within each antenna across multiple time instants. The resulting attention weights are then combined to form a comprehensive representation of the input sequence that preserves both spatial and temporal structures. This joint attention mechanism across both temporal and spatial dimensions is particularly beneficial for LEO satellite channels as it enables the model to capture rapid dynamic channel variations caused by high-speed orbital motion, RIS reconfiguration, user mobility, and environmental changes.

B. Feature Representation and Input Encoding

In our framework, channel measurements are acquired at discrete time instants t. Let T be the current time slot, c the number of past observations provided to the model, and g the number of future slots to predict. When $t \in \{T-c', \ldots, T\}$ with $c'=c-1$, we write $\mathbf{h}(t)$ as $\hat{\mathbf{h}}(t)$ to indicate an observed CSI snapshot; when $t \in \{T+1, \ldots, T+g\}$, we write $\mathbf{h}(t)$ as

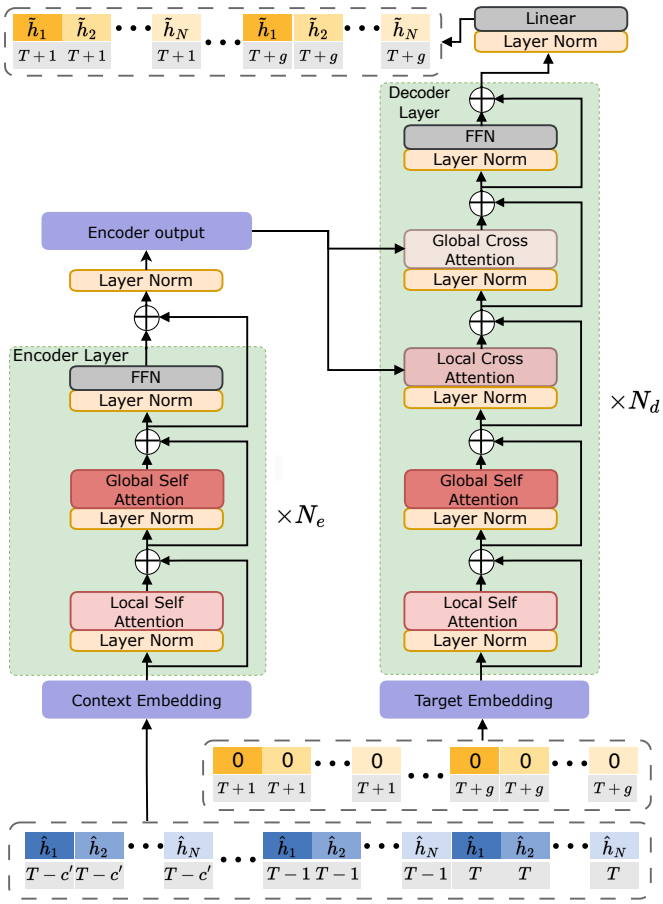


Fig. 3: Overall architecture of the proposed ST-attention based framework.

$\tilde{\mathbf{h}}(t)$ to indicate a CSI value to be predicted. Thus, $\hat{\mathbf{h}}(t)$ serves as model input, whereas $\tilde{\mathbf{h}}(t)$ is the network's prediction target.

A similar definition yields the predicted vectors $\tilde{\mathbf{h}}(t)$, which combine the real and imaginary parts of the satellite-to-RIS, satellite-to-user, and RIS-to-user channels into a unified representation. Our objective is to utilize the CSI from the past c time instants to construct $\hat{\mathbf{h}}(t)$ and feed it into the network shown in Fig. 1 in order to generate the predicted CSI $\tilde{\mathbf{h}}(t)$ for the subsequent g time instants. Because the transformer network is permutation-invariant, we incorporate temporal ordering via sinusoidal positional encodings [16]. In this way, the model can differentiate between earlier and later time steps, which is an essential feature for accurate time-series forecasting.

A key contribution of our method is the fine-grained attention mechanism. Rather than assigning a single attention weight per time step, each element of $\hat{\mathbf{h}}(t)$ is treated as an individual token in the transformer's self-attention module, as depicted in Fig. 3. Consequently, each attention head learns distinct weighting patterns over these tokens, thereby modeling dependencies both across time steps and among antennas within the same time step. This design enables the model to capture abrupt local changes (e.g., channel fades on specific antennas) as well as global trends (e.g., orbital motion).

C. ST-attention Mechanism

Fig. 3 depicts the overall structure of our proposed ST-attention based framework. In addition, Fig. 4 illustrates the contrast between an attention mechanism that solely focuses on the temporal dimension and one that simultaneously attends to both temporal and spatial dimensions. The temporal-only attention compresses the entire spatial slice of the channel tensor at each time instant into a single high-dimensional token, so the attention weights are distributed only along the temporal axis. By contrast, ST-attention treats each antenna–time pair as an independent token; this fine-grained representation allows the model to assign weights with per-antenna, per-time precision, enabling it to capture abrupt fades on individual antennas as well as the slow drifts induced by satellite motion.

1) Encoder: As illustrated in Fig. 3, the encoder processes past channel observations through stacked layers composed of:

- **Global Self-Attention:** Computes attention across the entire sequence (all time steps and antenna elements), enabling the model to learn broad spatiotemporal relationships.
- **Local Self-Attention:** Focuses on a smaller time window or antenna subset, capturing fine-grained variations (e.g., abrupt channel fades on specific antennas).
- **Position-Wise Feed-Forward Network (FFN):** The FFN is a dedicated non-linear module that processes each channel feature token independently. It transforms the raw channel embeddings into a richer representation by capturing complex non-linear propagation effects and subtle spatiotemporal variations.

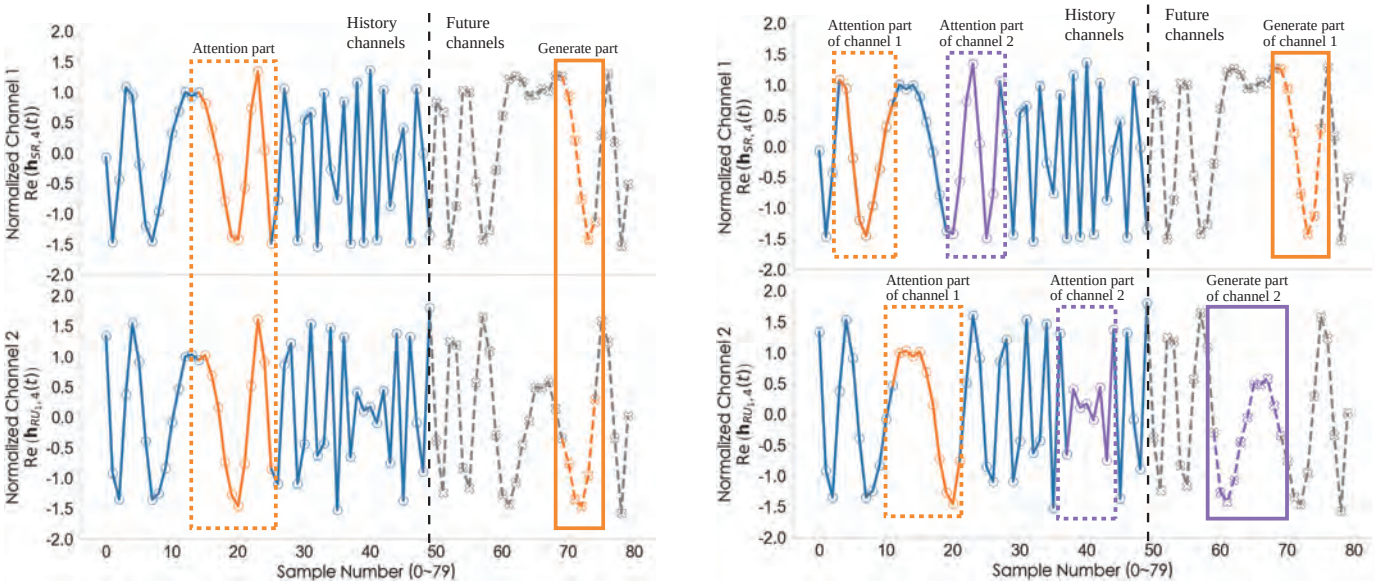
Let $X \in \mathbb{R}^{(c\times N)\times D}$ be the embedded input tokens, where $(c\times N)$ encompasses all channel pairs over the entire historical period, and D is the embedding dimension. In the multi-head attention mechanism, the input X is first projected into three distinct representations: the queries Q, the keys M, and the values V. These projections serve different roles:

- **Queries** Q**:** Represent the elements that seek relevant information from other tokens.
- **Keys** M**:** Encode the content of the tokens, acting as indices that the queries use to locate pertinent information.
- **Values** V**:** Contain the actual information that will be aggregated based on the attention weights.

For each attention head h, these projections are computed as $Q^h = XW_Q^h$, $M^h = XW_M^h$, $V^h = XW_V^h$, where W_Q^h, W_M^h, and W_V^h are trainable weight matrices. The attention function for each head is defined as

$$\text{Attention}(Q^h, M^h, V^h) = \text{softmax}\left(\frac{Q^h (M^h)^\top}{\sqrt{d_m}}\right) V^h, \quad (5)$$

where d_m is the key dimension and $(\cdot)^\top$ denotes transposition. Following the common Transformer design, we set $d_m = \frac{D}{H}$, with H the number of attention heads. Finally, the outputs from all H heads are concatenated and projected using a weight matrix W_O to yield the final multi-head attention



(a) T-attention (b) ST-attention

Fig. 4: Comparison of prediction results under different attention parts.

output:

$$\text{MultiHead}(X) = \bigoplus_{h=1}^{H} \text{Attention}(Q^h, K^h, V^h)\, W_O^h, \quad (6)$$

where $\oplus$ denotes concatenation.

2) Decoder: To generate channel predictions for future time steps $[T+1, \ldots, T+g]$, the decoder employs:

- **Masked Self-Attention:** Operates over future tokens in a causal manner, ensuring that information from later time steps does not influence earlier predictions.
- **Global and Local Cross-Attention:** Attends to the encoder outputs alongside the partially known future tokens. Global cross-attention captures large-scale dependencies, while local cross-attention refines local details for sudden channel variations.

Each decoder layer also includes a position-wise FFN and normalization layers.

By explicitly assigning attention weights to specific antennas at specific time instants, the ST-attention mechanism captures both large-scale orbital effects and localized fading phenomena. This multi-scale attention design is especially suitable for highly dynamic LEO satellite channels, where accurate forecasting hinges on understanding both global and fine-grained variations.

D. Prediction Head and Training Objective

The final decoder layer produces hidden states, which are then mapped to channel estimates by a fully connected prediction head. We train the model by minimizing the mean squared error (MSE) between the predicted and the ground-truth channels:

$$\mathcal{L}_{\text{pred}} = \frac{1}{gNB} \sum_{b=1}^{B} \sum_{i=1}^{N} \sum_{j=1}^{g} \left\| \hat{h}_i^{\text{pred}}(j) - \hat{h}_i^{\text{true}}(j) \right\|^2, \quad (7)$$

where B is the batch size. $\hat{h}_i^{\text{pred}}(j)$ and $\hat{h}_i^{\text{true}}(j)$ represent the predicted and ground-truth channel coefficients, respectively, for the i-th dimension at the j-th future time step. Minimizing $\mathcal{L}_{\text{pred}}$ encourages the predicted channel coefficients to align closely with their true counterparts, thereby improving forecasting accuracy.

IV. Pretraining Strategy for pCSI

In realistic LEO satellite communication scenarios, the high mobility of satellites, UAV-RISs, and ground users often leads to pCSI, where certain channel coefficients are intermittently missing or heavily corrupted. Such incomplete observations can significantly degrade prediction performance because many models are incapable of capturing essential spatiotemporal dependencies in sparse or irregular data. To address this challenge, we draw inspiration from the masked language modeling approach widely used in NLP [24] and propose a two-stage channel prediction scheme that can robustly handle missing channel entries. This approach enables accurate future channel predictions under pCSI conditions, particularly when high-quality labeled datasets are limited.

Specifically, in the first stage we train a network to recover masked channel elements. In this process, a subset of channel entries in the input sequence is deliberately masked and the model is optimized to infer these missing values. Through unsupervised training on a large dataset with actively masked entries, the model learns the underlying missing patterns. The loss function is computed solely on the masked entries, which encourages the model to learn robust and noise-resistant features and enhances its ability to handle the pCSI issue.

After pretraining converges, the model serves as a feature extractor that captures complex and diverse missing patterns in the satellite MIMO channel. Through such a parameter transfer, the predictor inherits the refined knowledge that can provide a better initialization for the subsequent prediction

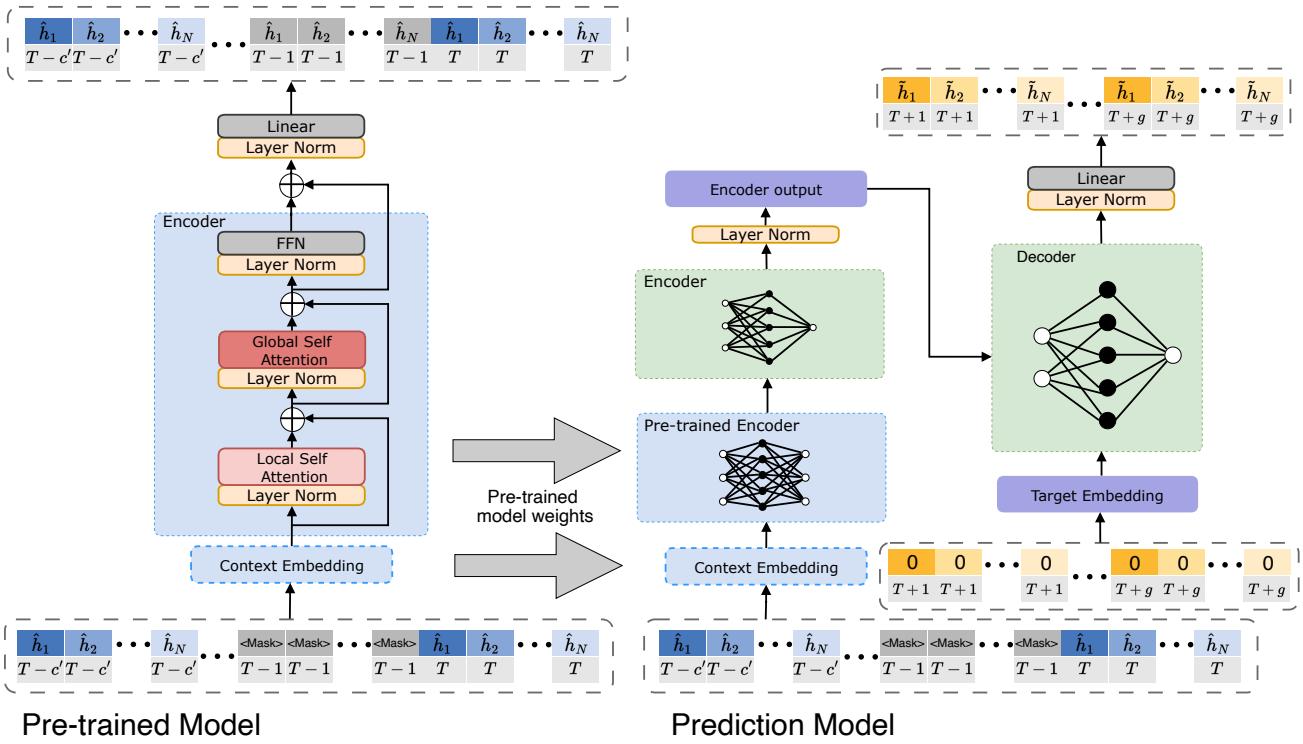


Fig. 5: Illustration of the proposed two-stage pretraining strategy for handling pCSI. The left module shows the masked reconstruction pretraining, while the right module presents the subsequent channel prediction network.

training, thereby improving accuracy and stability under realistic pCSI conditions.

In this section, we first present the masked reconstruction pretraining module and then describe how the pretrained layers are integrated into the channel prediction network. Finally, we outline the fine-tuning process under pCSI conditions and discuss the benefits of the proposed two-stage pretraining strategy.

A. Masked Reconstruction Pretraining

As shown on the left side of Fig. 5, the pretraining module consists of an embedding layer, multiple transformer encoder layers, and a linear reconstruction head dedicated to recovering masked entries. Let

$$\mathbf{H}_{\text{full}} = \Big\{ \mathbf{h}(t) \Big| t = T-c', \ldots, T \Big\} \in \mathbb{C}^{(cN)\times 1}, \quad (8)$$

denote the stacked channel observations over the past c time instants. To simulate partial observations, we define a binary mask

$$\mathbf{Z} = \Big\{ \mathbf{z}(t) \Big| t = T-c', \ldots, T \Big\} \in \{0,1\}^{(cN)\times 1}, \quad (9)$$

where each $\mathbf{z}(t) \in \{0,1\}^{N\times 1}$ satisfies

$$\mathbf{z}(t) = \alpha(t) \cdot \mathbf{1}, \quad \text{with } \alpha(t) \in \{0,1\}. \quad (10)$$

This indicates that at time t, the entire channel vector is fully visible when $\alpha(t)$=0 and completely masked when $\alpha(t)$=1.

To emulate the three pCSI patterns in Section II-C, a binary mask $\mathbf{Z} \in \{0,1\}^{cN\times 1}$ is generated for every training sample as follows:

1) Pattern selection: one of the three patterns, continuous outage, random outage, or equidistant sparsity, is selected according to the specified probability distribution;

2) Missing-ratio sampling: the target ratio $\rho \sim \mathcal{U}[\rho_{\min}, \rho_{\max}]$, where $(\rho_{\min}, \rho_{\max}) = (0.1, 0.9)$ by default;

3) Mask construction: according to the selected pattern, exactly $\lfloor \rho cN \rfloor$ entries of $\mathbf{Z}$ are set to 1.

The input to the pretraining network is then formed by applying element-wise multiplication

$$\widehat{\mathbf{H}} = (\mathbf{1} - \mathbf{Z}) \odot \mathbf{H}_{\text{full}}, \quad (11)$$

where $\odot$ denotes the element-wise product and $\mathbf{1}$ is a vector of ones with the same dimension as $\mathbf{Z}$. This operation preserves the observed entries while setting the masked entries to zero.

Given $\widehat{\mathbf{H}}$, the model outputs a reconstructed version $\widetilde{\mathbf{H}}$ that aims to fill in the missing components. The binary mask $\mathbf{Z}$ plays a crucial role by indicating which elements are masked, so that the reconstruction loss is computed only on these missing entries. The pretraining loss function is defined as

$$\mathcal{L}_{\text{recon}} = \frac{1}{B} \sum_{i=1}^{B} \Big\| \big(\widetilde{\mathbf{H}}^{(i)} - \mathbf{H}_{\text{full}}^{(i)} \big) \odot \mathbf{Z}^{(i)} \Big\|^2, \quad (12)$$

where B is the batch size for pretraining. In this context, each element $h_n(t)$ of $\mathbf{H}_{\text{full}}^{(i)}$ corresponds to $\tilde{h}_n(t)$ in $\widetilde{\mathbf{H}}$ and to $\hat{h}_n(t)$ in $\widehat{\mathbf{H}}$, respectively. Since the channel coefficients are continuous variables, we adopt MSE as the loss function rather than the cross-entropy (CE) loss used in masked language modeling tasks [24]. Minimizing (12) trains the model to exploit both local correlations among antenna elements and

global temporal dependencies induced by motion, thereby learning to reconstruct missing channel states from partial observations. This pretraining not only enhances robustness against incomplete CSI measurements but also fosters generalizable representations for subsequent channel prediction tasks.

B. Integration with the Prediction Model

After completing the masked reconstruction pretraining, we transfer the learned parameters to initialize the prediction model. In particular, the pretrained embedding layer replaces the original embedding layer, and the pretrained encoder layers are inserted ahead of the existing encoder in the prediction model. As illustrated on the right side of Fig. 5, this arrangement allows the prediction model to inherit the ability to handle the incomplete CSI, since the transferred layers have already captured how to reconstruct the missing channel entries. By integrating these pretrained layers, the model can acquire robust spatiotemporal representations from the outset, eliminating the need to learn such features from scratch. Consequently, the subsequent training phase can converge more rapidly and achieve a higher prediction accuracy than a random initialization.

C. Fine-Tuning with pCSI

After initialization with pretrained parameters, the model is refined end-to-end to accept incomplete channel observations and produce predictions for future time slots. During this fine-tuning stage, training focuses exclusively on the channel prediction loss $\mathcal{L}_{\text{pred}}$ defined in (7), aligning the learning objective directly with accurate forecasting. This approach allows the model to specialize in predicting future channel states under pCSI conditions, while utilizing the robust representations acquired during pretraining. As a result, the training process emphasizes the core forecasting task and benefits from the pretrained layers' ability to infer missing CSI entries.

D. Practical Considerations and Benefits

The proposed two-stage pretraining strategy is designed to robustly handle the various pCSI scenarios outlined in Section II-C. By training on extensive data that reflect a wide range of missing patterns and proportions, the model develops spatiotemporal representations that effectively mitigate the impact of channel interruptions regardless of the underlying missing data distribution. In addition, the pretrained network enables the use of low-frequency CSI measurements to accurately predict high-frequency channel states, thereby reducing feedback overhead while preserving acceptable accuracy. Initializing the prediction model with these pretrained weights accelerates convergence and enhances stability, as the network no longer needs to learn representations from the scratches on incomplete data. Overall, this two-stage pretraining strategy delivers robust performance under pCSI conditions and is well suited for practical satellite communication deployments.

V. Complexity Reduction and Scalability Analysis

The ST-attention mechanism captures both fine-grained spatial correlations among satellite, RIS, and user antennas and the rapid temporal dynamics of LEO channels, but its computation and memory grow quadratically with the product of spatial elements and time steps. In realistic UAV-RIS-assisted LEO systems, satellite arrays and large RIS panels can each comprise hundreds of elements, making full self-attention over all antenna–time tokens impractical for typical on-board or edge accelerators. To address this issue, we introduce a compact beamspace representation that maps the element-domain channel onto a sparse angular basis, thereby reducing the spatial token count and substantially lightening the attention workload without compromising prediction accuracy.

A. DFT-Based Beamspace Projection

Both the LOS satellite path and the RIS-reflected path are dominated by a small number of strong specular components. Consequently, the element-space MIMO channels $\mathbf{h}_{SR}(t)$, $\mathbf{h}_{RU_k}(t)$, and $\mathbf{h}_{SU_k}(t)$ exhibit intrinsic angle sparsity, with most array elements receiving highly correlated signals. By applying a two-dimensional discrete Fourier transform (DFT), most of the channel power is concentrated into a few dominant angular bins. The beamspace projection reduces the spatial token, enabling the attention mechanism to operate on a much smaller sequence and thereby dramatically lowering both computation and memory costs without sacrificing prediction accuracy.

Let $\mathbf{F}_{\mathcal{D}}$ be the unitary $\mathcal{D}\times\mathcal{D}$ DFT matrix. For each snapshot t we obtain the beam-domain representations

$$\mathbf{b}_{SR}(t) = \mathbf{F}_{N_R}\,\mathbf{h}_{SR}(t)\,\mathbf{F}_{N_S}^{H}, \tag{13a}$$

$$\mathbf{b}_{RU_k}(t) = \mathbf{F}_{N_U}\,\mathbf{h}_{RU_k}(t)\,\mathbf{F}_{N_R}^{H}, \tag{13b}$$

$$\mathbf{b}_{SU_k}(t) = \mathbf{F}_{N_U}\,\mathbf{h}_{SU,k}(t)\,\mathbf{F}_{N_S}^{H}. \tag{13c}$$

For every matrix in (13) we retain the P strongest coefficients; their linear indices define the set $\mathcal{P} = \{i_1, \ldots, i_P\}$. Taking the satellite–RIS link as an example, each index is mapped to a row–column coordinate on the 2-D DFT grid, denoted $(u_p, v_p) \in \{0, \ldots, N_R{-}1\} \times \{0, \ldots, N_S{-}1\}$, and normalised to $[0, 1]$: $\tilde{u}_p = \frac{u_p}{N_R-1}$ and $\tilde{v}_p = \frac{v_p}{N_S-1}$.

The p-th retained beam delivers

$$\mathbf{z}_{SR}^{(p)}(t) = \big[\underbrace{|b_{SR}^{(p)}(t)|}_{\text{magnitude}}, \underbrace{\angle b_{SR}^{(p)}(t)}_{\text{phase}}, \tilde{u}_p,\, \tilde{v}_p\big]^{T} \in \mathbb{R}^4. \tag{14}$$

The same procedure is applied to the RIS–user and satellite–user sub-links, yielding beam tokens $\mathbf{z}_{RU_k}^{(p)}(t)$ and $\mathbf{z}_{SU_k}^{(p)}(t)$. Stacking all P beams of the three sub-links gives a fixed–length real vector

$$\begin{aligned}\mathbf{s}(t) = \big[&\mathbf{z}_{SR}^{(1)}(t)^T, \ldots, \mathbf{z}_{SR}^{(P)}(t)^T, \mathbf{z}_{RU_1}^{(1)}(t)^T, \ldots, \\ &\mathbf{z}_{RU_K}^{(P)}(t)^T, \mathbf{z}_{SU_1}^{(1)}(t)^T, \ldots, \mathbf{z}_{SU_K}^{(P)}(t)^T\big]^T,\end{aligned} \tag{15}$$

with the dimensionality is now $N_{\text{tok}} = 4P\,(1 + 2K)$.

For each sub-link, the decoder outputs P beam tokens $\tilde{\mathbf{z}}_*^{(p)}(t) = [\tilde{m}^{(p)}, \tilde{\varphi}^{(p)}, \tilde{u}_p, \tilde{v}_p]$. We form the complex coefficient $\tilde{b}^{(p)}(t) = \tilde{m}^{(p)} e^{j\tilde{\varphi}^{(p)}}$ and write it at grid index (u_p, v_p); all other beamspace entries are zero. The element-domain channel is then recovered by the inverse DFT, e.g. $\tilde{\mathbf{h}}_{SR}(t) = \mathbf{F}_{N_R}^H \tilde{\mathbf{B}}_{SR}(t) \mathbf{F}_{N_S}$. The same step is applied to RIS–user and satellite–user sub-links, keeping the sequence length limited to P while preserving perfect invertibility.

B. Complexity Analysis

This subsection evaluates the computational cost of the network with and without beam-domain compression. The FLOP count includes only the matrix multiplications and softmax operations that dominate self-attention. Point-wise functions such as LayerNorm and nonlinear activations are not considered, because their complexity is $O(LD)$, where L denotes the token-sequence length, defined as the product of the history depth c and the number of tokens per snapshot, and is therefore negligible.

As described in Section II-B5 and Section V-A, a full snapshot comprises $N_{\text{tok}}^{\text{full}} = 2N_R N_S + 2K\,N_U\big(N_R + N_S\big)$, whereas beam-domain sparsification reduces the token count to $N_{\text{tok}}^{\text{beam}} = 4P\big(1 + 2K\big)$.

For a token sequence of length L, the global self-attention branch incurs $F_g(L) = 2L^2 d_m + 4L\,D$, while the local branch incurs $F_l(L) = 2L\,w\,d_m + 4w\,D$, where w denotes the size of the local-attention window. With $L = cN_{\text{tok}}^{\text{full}}$, the total cost without compression is

$$F_{\text{full}} = (N_e + N_d)\Big[2\big(cN_{\text{tok}}^{\text{full}}\big)^2 d_m + 4\big(cN_{\text{tok}}^{\text{full}}\big)D + 2\big(cN_{\text{tok}}^{\text{full}}\big)w\,d_m + 4wD\Big], \tag{16}$$

Replacing $N_{\text{tok}}^{\text{full}}$ by $N_{\text{tok}}^{\text{beam}}$ and adding the FFT overhead $c\,O\big(N_{\text{tok}}^{\text{full}} \log N_{\text{tok}}^{\text{full}}\big)$ yields

$$F_{\text{beam}} = (N_e + N_d)\Big[2\big(cN_{\text{tok}}^{\text{beam}}\big)^2 d_m + 4\big(cN_{\text{tok}}^{\text{beam}}\big)D + 2\big(cN_{\text{tok}}^{\text{beam}}\big)w\,d_m + 4wD\Big] + c\,O\big(N_{\text{tok}}^{\text{full}} \log N_{\text{tok}}^{\text{full}}\big). \tag{17}$$

Because the quadratic term $2\big(cN_{\text{tok}}^{\text{full}}\big)^2 d_m$ dominates, compressing each snapshot from $N_{\text{tok}}^{\text{full}}$ to $N_{\text{tok}}^{\text{beam}}$ yields a marked reduction in both FLOPs and memory.

VI. Performance Evaluation

In this section, the performance of the proposed method is evaluated via simulations. We will first introduce he simulation environment and experimental setup, and then make a comparative analysis of our proposal against the selected baselines. Key observations and performance trends under various experimental conditions will be discussed in detail.

A. Simulation Setup

We evaluate two array scales: a small setup with a satellite array of $N_S = 25$ antennas and a UAV-RIS of $N_R = 9$ elements, serving $K = 4$ single-antenna users ($N_U = 1$); and a large setup with $N_S = 256$ satellite antennas and $N_R = 81$ RIS elements, while K and N_U remain unchanged.

Two configurations are considered: a satellite array with $N_S = 25/9$ antennas, an RIS with $N_R = 9/4$ elements, and support for $K = 4/2$ users, each equipped with a single antenna ($N_U = 1$).

CSI matrices $\mathbf{h}_{SR}(t)$, $\mathbf{h}_{RU_k}(t)$, and $\mathbf{h}_{SU_k}(t)$ are generated over distinct orbital time windows in August 2024 to prevent data leakage. The dataset is partitioned by temporal segments: 60% of 10,000 samples are used for training (collected between August 1 and August 2, 2024), 20% for validation (August 3–4, 2024), and the remaining 20% for testing (August 5–6, 2024). This temporal separation ensures independence among the subsets. To verify that the time-based split does not introduce distributional bias, we computed descriptive statistics of four key channel metrics (path loss, Rician K-factor, Doppler shift, and SNR) for each segment and performed two–sample Kolmogorov–Smirnov (KS) tests. As summarised in Table III, all KS p-values exceed 0.35, indicating that the three segments are statistically consistent.

All models are trained on eight NVIDIA H100 GPUs [40], each with 80 GB of memory. A constant effective batch size is maintained by fixing the product of the per-GPU batch size B, the number of gradient accumulation steps S, and the number of GPUs G such that

$$B \times S \times G = \mathcal{K}, \tag{18}$$

with $\mathcal{K}$ set to 40. This configuration balances memory usage and computational throughput, ensuring stable convergence and fair comparisons across experiments [41].

B. Simulation Results

As illustrated in Fig. 6, we compare the multi-step prediction performance of LSTM [12], LSTNet [42], Linear [43], T-attention Transformer [17], and the proposed ST-attention based channel prediction scheme. The channel acquisition (sampling) interval is set to 10^{-3} s and the history length is 30. The figure reports the normalized MSE (NMSE) for prediction horizons of 2, 8, 14, 20, and 26 steps.

It can be observed that all methods exhibit increasing NMSE as the prediction horizon grows, indicating a degradation in accuracy over longer future intervals. Nevertheless, the proposed ST-attention based scheme achieves consistently lower NMSE than the other approaches at every horizon, demonstrating its superior ability to learn both temporal dependencies and spatial correlations. It is also seen that the Linear model performs well at shorter horizons (e.g., c=2), but its accuracy declines rapidly as the horizon increases, reflecting the limitations of an autoregressive linear approach. For longer-range predictions (e.g., beyond 14 steps), LSTM exhibits relatively strong performance, but it remains notably outperformed by our proposed ST-attention based scheme.

TABLE II: Computational Efficiency of Different Models at Various Prediction Steps. Parameter count (Params) is measured in thousands (K) and FLOPs is measured in millions (M)

Model	2 Steps		8 Steps		14 Steps		20 Steps		26 Steps	
	Params	FLOPs	Params	FLOPs	Params	FLOPs	Params	FLOPs	Params	FLOPs
LSTM	1270	38.09	1270	46.28	1270	54.47	1270	62.66	1270	70.85
LSTNet	655.07	25.13	655.07	100.53	655.07	175.93	655.07	251.33	655.07	326.73
Linear	22.40	0.88	22.40	3.52	22.40	6.17	22.40	8.81	22.40	11.50
T-attention Transformer	525.47	5.78	526.63	6.87	527.78	7.96	528.93	9.04	530.08	10.13
ST-attention Transformer	565.06	4260	566.21	4880	567.36	5490	568.51	6100	569.66	6710

TABLE III: Statistical consistency between the three data splits

Segment	Mean (dB)	Std (dB)	KS p-value
Train	16.976	0.314	–
Validation	16.987	0.318	0.372 (Train vs. Validation)
Test	16.988	0.320	0.536 (Train vs. Test)
			0.902 (Validation vs. Test)

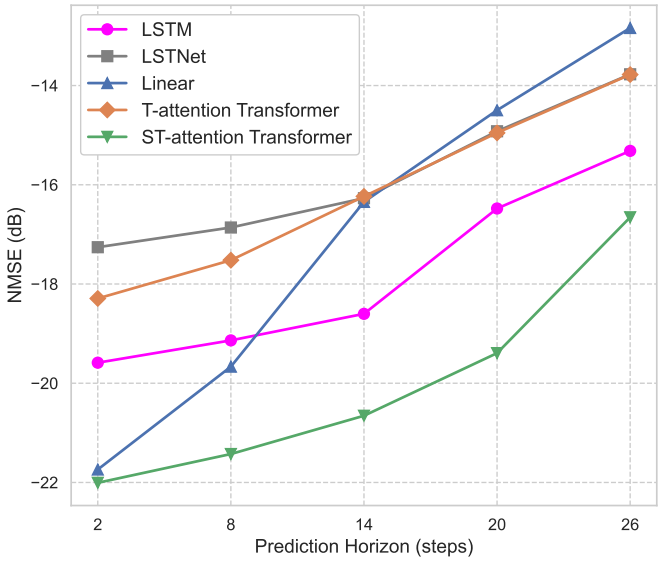


Fig. 6: Multi-step prediction performance for various methods, where the sampling interval is 1 ms, the history length is c=30, and the numbers of antennas and users are N_S=25, N_R=9, K=4, and N_U=1.

To demonstrate the fairness of the comparisons in Fig. 6, we also summarise all baseline configurations for the two-step prediction in Table V and report model size and computational efficiency in Table II. In Table V, LSTNet's "64/100" denotes 64 convolutional filters in its convolutional neural network (CNN) block and a 100-unit hidden state in its recurrent neural network (RNN) block; the LSTM is implemented as a two-layer bidirectional network; and both the T-attention and ST-attention Transformers employ a 2+2 architecture, comprising two encoder layers and two decoder layers.

In Table II, although the parameter count for each model remains unchanged, the computational load (in FLOPs) increases with a longer prediction horizon. The proposed ST-attention based scheme has a model size that is smaller than

TABLE IV: Model Configurations and Prediction Performance

Metric	Config 1	Config 2	Config 3
Params	568.51 K	1.72 M	3.84 M
Model Dim.	64	96	128
Fwd. Dim.	128	192	256
Enc. Layers	2	2	3
Dec. Layers	2	2	3
Attn. Heads	2	4	4
QK Dim.	32	48	64
V Dim.	32	48	64
FLOPs	6.1 GMac	21.09 GMac	53.63 GMac
Inf. Time (NX)	0.24 ms	0.84 ms	2.15 ms
Inf. Time (AGX)	0.09 ms	0.31 ms	0.79 ms
NMSE	-19.4 dB	-19.83 dB	-19.97 dB

Notes: **Model Dim.** and **Fwd. Dim.** are the model and feedforward dimensions. **Enc. Layers** and **Dec. Layers** are the numbers of encoder and decoder layers. **Attn. Heads** is the count of attention heads. **QK Dim.** and **V Dim.** are query/key and value dimensions. **FLOPs** is measured in GMac (Giga MACs). **NX** and **AGX** refer to Jetson Orin NX and Jetson AGX Orin; inference times assume 25 and 68.75 TFLOPS, respectively.

TABLE V: Hyper-parameters of baseline models

Model	Hidden size	Layers	Dropout	Params
LSTM	128	2 (Bi)	0.1	1.27 M
LSTNet (CNN/RNN)	64/100	1/1	0.2	0.66 M
Linear	–	–	0	22 k
T-attention Transformer	d_{model}=64	2+2	0.1	0.53 M
ST-attention Transformer	d_{model}=64	2+2	0.1	0.57 M

that of LSTM [12] and LSTNet [42], and is comparable to that of the T-attention Transformer [17]. Its size is larger than that of the Linear model [43]; however, this is inherent to the characteristics of the Linear model, which cannot further improve performance by simply increasing its capacity. This indicates that our comparison is fair and does not rely on deliberately reducing the model sizes of other approaches. Notably, our proposed ST-attention based scheme requires a higher FLOPs count compared to other models. This increased cost is primarily due to its ability to jointly capture spatial and temporal attentions, which can be regarded as the price for achieving higher precision in channel prediction. In scenarios where precise channel prediction is critical for link adaptation and resource allocation, the performance gains justify the additional computational expense. Moreover, as hardware performance and optimization techniques continue to improve, the relative computational overhead of our scheme is expected to decrease, further enhancing its feasibility for practical deployment.

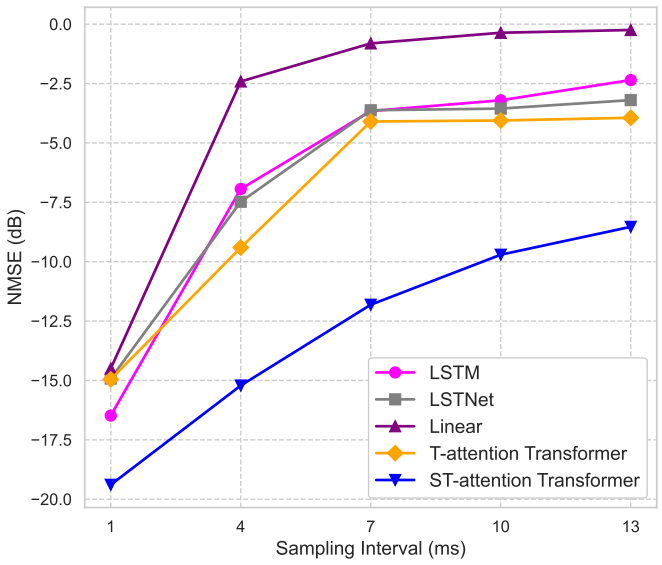


Fig. 7: Prediction performance under different sampling intervals, where the history length is c=30, the prediction horizon is g=20, and the numbers of antennas and users are N_S=25, N_R=9, K=4, and N_U=1.

To illustrate the impact of model size on prediction performance, Table IV summarizes the trade-off between model complexity and prediction accuracy for a history length of 30 and a prediction horizon of 20 steps, evaluated across three configurations. The table reports key metrics and the achieved NMSE. In addition, we report the inference times on the Jetson Orin NX [44] and the Jetson AGX Orin [45] platforms. The Jetson Orin NX is a low-power platform designed for compact and energy-efficient deployments and is well suited for satellite applications with strict power and space constraints. In contrast, the Jetson AGX Orin delivers higher computational performance and serves as a benchmark for scenarios with relaxed power constraints.

As the model scale increases from Config 1 to Config 3, the parameter count, FLOPs, and inference times increase substantially. For example, FLOPs rise from 6.1 GMac in Config 1 to 53.63 GMac in Config 3, with the inference time on the Jetson Orin NX increasing from 0.24 ms to 2.15 ms and on the Jetson AGX Orin from 0.09 ms to 0.79 ms. Meanwhile, the NMSE improves modestly from -19.4 dB to -19.97 dB, indicating that larger models can better capture complex channel dynamics and yield improved prediction accuracy. However, it should be noted that the benefits of increasing model size are subject to diminishing returns, as evidenced by the marginal NMSE improvement despite a substantial increase in computational cost.

These findings underscore the trade-off between computational resource consumption and prediction performance. The inference times on both platforms illustrate the range of hardware environments available for deployment and provide guidance for selecting a configuration that balances efficiency with accuracy in practical satellite communication applications.

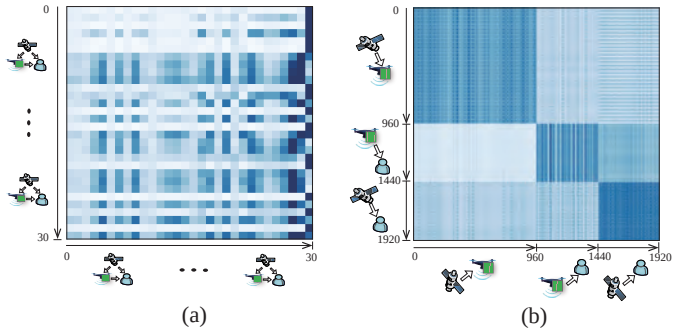


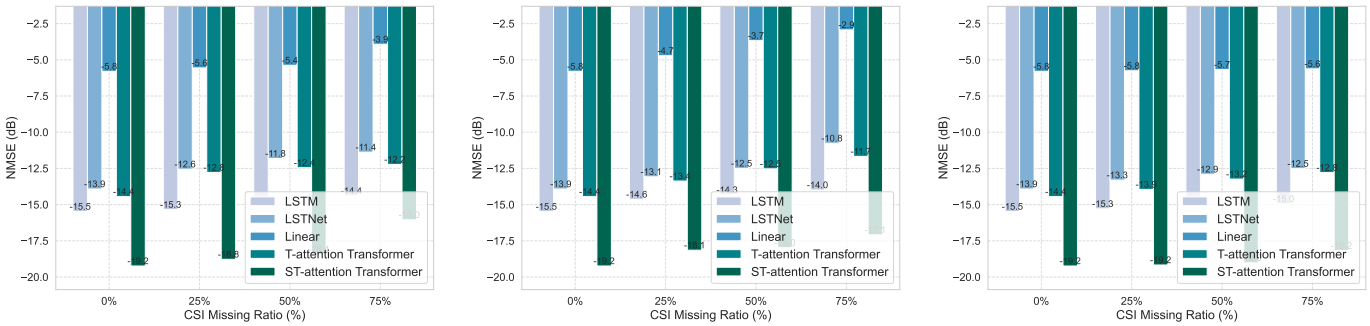
Fig. 8: Visualization of attention mechanisms: (a) T-attention and (b) ST-attention, where the history length is c=30, the prediction horizon is g=30, and the numbers of antennas and users are N_S=4, N_R=4, K=2, and N_U=1.

As shown in Fig. 7, we further evaluate the NMSE performance of five models at various sampling intervals while keeping the history length fixed at 30 and predicting 20 future steps. Although all methods exhibit increased errors as the sampling interval grows, the proposed ST-attention based scheme shows a significantly slower rise in NMSE. This performance is primarily due to its integrated spatiotemporal attention mechanism, which effectively captures fine-grained channel dynamics even under sparse data conditions.

Furthermore, when the sampling interval exceeds 4 ms, the T-attention based method outperforms the other approaches except for our proposed scheme. This observation confirms that temporal attention alone is effective in extracting critical dependencies from sparser data, but the combination of spatial and temporal attention can further enhance prediction accuracy. These results underscore the advantages of a holistic spatiotemporal attention design for robust channel forecasting in dynamic environments.

To further illustrate why ST-attention outperforms purely T-attention, Fig. 8 compares the two mechanisms by visualising their attention maps. Under T-attention, the model attends almost exclusively to the temporal dimension, overlooking spatial interactions among antennas or sub-channels and thus missing subtle high-dimensional variations. By contrast, ST-attention performs joint spatio-temporal reasoning, allowing the network to exploit cross-antenna dependencies in addition to temporal evolution. Concretely, on the same validation split, the average row entropy increases from 1.46 to 2.31 bits, indicating a much broader receptive field. Moreover, ST-attention allocates approximately 37 % of its mass to cross-antenna links, whereas T-attention assigns virtually none. Therefore, the model can now distribute its focus across both time and space, thereby reconstructing missing channel entries more effectively and maintaining higher prediction accuracy over long horizons. This richer spatial modelling ultimately delivers greater robustness in dynamic satellite environments.

Table VI contrasts the element-domain results shown earlier for the small array with a beamspace variant, then extends the same comparison to a large array. For the small geometry, keeping only $P = 9$ beams cuts the FLOPs from 6.06 GMac to 1.52 GMac while incurring less than 0.1 dB NMSE loss.



(a) Continuous Outages (b) Random Outages (c) Equidistant Sparsity

Fig. 9: Prediction performance under three pCSI scenarios after beamspace compression, where the history length is c=30, the prediction horizon is g=20, the retained-beam count is P=26, and the antenna configuration is N_S=256, N_R=81, K=4, N_U=1.

TABLE VI: Prediction Accuracy and Complexity With and Without Beamspace Compression

Scenario	Full Element Domain		Beamspace Compression	
	NMSE (dB)	FLOPs (GMac)	NMSE (dB)	FLOPs (GMac)
Small array[a] P=9	−19.40	6.06	−19.35	1.52
Large array[b] P=26	—	4.50×10^5	−19.24	12.63

[a] c=30, g=20, and the numbers of antennas and users are N_S=25, N_R=9, K=4, and N_U=1.
[b] c=30, g=20, and the numbers of antennas and users are N_S=256, N_R=81, K=4, and N_U=1.

This is because the most of channel power concentrates in the dominant angular bins captured by those beams, and the discarded beams mainly carry noise and minor scattering components. In the large-array case ($N_S = 256,\ N_R = 81$), an element-domain implementation would require an impractical 4.5×10^5 GMac and terabytes of intermediate activations, so we report it only as an analytical estimate. Applying beamspace compression with $P = 26$ reduces the cost to 12.6 GMac while preserving excellent NMSE performance, thereby enabling scalable training and real-time inference even for very large arrays.

Fig. 9 shows the impact of pCSI on prediction accuracy under the three outage patterns introduced in Section II-C. The curves are obtained for the large-array setting after beamspace compression with the retained-beam count P=26. In each scenario, the outage ratio is varied from 0% to 75%, with a history length of 30 and a prediction horizon of 20. As the outage ratio increases, all methods exhibit an increase in NMSE.

The ST-attention based method consistently achieves the lowest NMSE, highlighting its robustness to incomplete CSI observations. By exploiting global context from both spatial and temporal dimensions, the proposed model can more effectively reconstruct and "fill in" the missing CSI entities. In contrast, conventional time-series models rely primarily on temporal correlations and lack the ability to capture spatial interactions, making them more susceptible to performance degradation at high outage ratios. Moreover, the linear autoregressive baseline suffers the largest drop because its coefficients are fitted in the element domain. After the channel is projected and truncated in beamspace, much of the statistical structure it relies on is removed, which leads to a sharp NMSE increase.

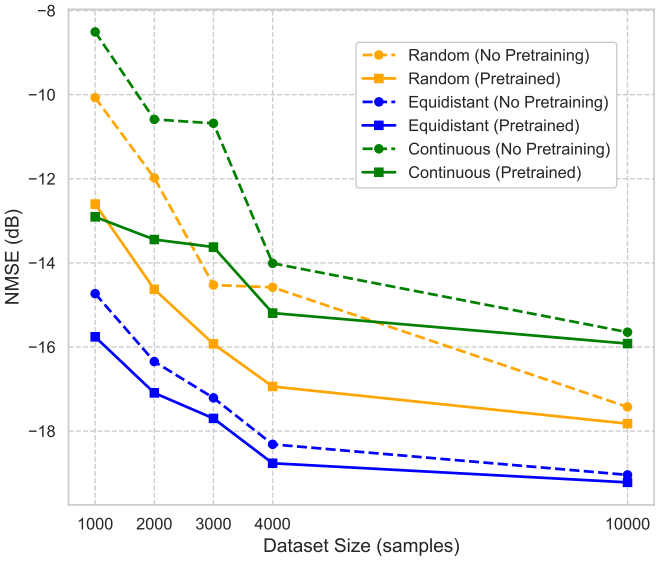


Fig. 10: Effect of pretraining on pCSI conditions performance improvement under varying dataset sizes after beamspace compression, where the history length is c=30, the prediction horizon is g=20, the retained-beam count is P=26, and the numbers of antennas and users are N_S=256, N_R=81, K=4, and N_U=1.

Although the proposed ST-attention based scheme already achieves superior prediction performance under pCSI conditions compared to other benchmark methods, further improvements can be obtained by using a two-stage channel prediction scheme based on pretraining. Fig. 10 illustrates, under beamspace compression, how pretraining improves CSI prediction accuracy across different dataset sizes for the three pCSI patterns. For each pre-training sequence we first draw a masking pattern according to the probabilities associated with the corresponding curve in Fig. 10. A missing ratio is then sampled from the uniform law $\rho \sim \mathcal{U}[0.10, 0.90]$; exactly $\lfloor \rho\, c \rfloor$ time steps are masked and zero-filled. The identical

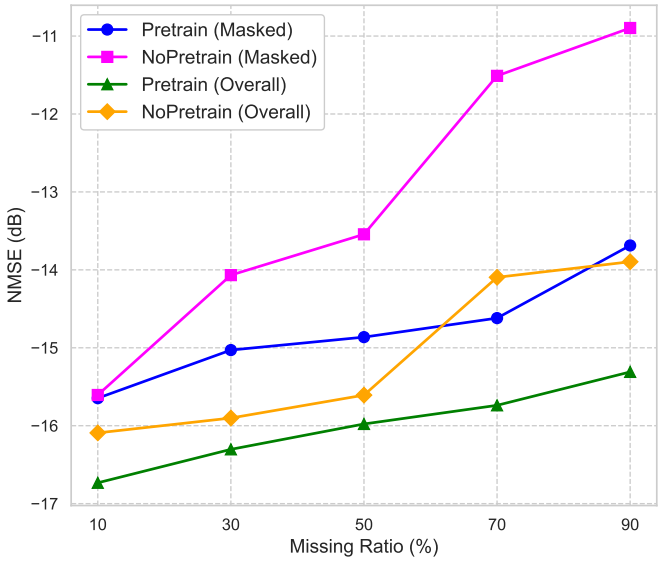


Fig. 11: Comparison of prediction performance under various mixed missing ratios after beamspace compression, where the history length is c=30, the prediction horizon is g=20, the retained-beam count is P=26, and the numbers of antennas and users are N_S=256, N_R=81, K=4, and N_U=1.

masking protocol is applied during fine-tuning so that the encoder is exposed to statistically consistent inputs throughout training.

The horizontal axis indicates the dataset size, while the vertical axis shows the NMSE. For each pCSI pattern, we compare the prediction performance of models trained with and without pretraining. It is worth noting that the dataset comprises 10% samples with channel outages and 90% samples with complete continuous inputs, enabling the trained model to handle both normal CSI and pCSI conditions.

The experimental results indicate that when the dataset is small (e.g., 1,000 samples), the pretraining processing can significantly reduce the NMSE, demonstrating that the pretrained representations effectively facilitate the reconstruction of missing channel information and capture essential spatiotemporal dependencies under limited data conditions. As the dataset size increases, the performance gap between the pretrained and non-pretrained models gradually narrows, suggesting that with sufficient training data the model can learn near-optimal representations directly from the data. However, even when the dataset size reaches 10,000 samples, the pretraining processing still provides measurable performance gains. In addition, the extent of the improvement brought in by the pretraining processing varies among the different outage scenarios. In particular, under limited data conditions, the performance gains offered by pretraining are more pronounced for Continuous Outages and Random Outages compared to Equidistant Sparsity. This difference is due to the regularity of the missing pattern in the Equidistant Sparsity, which enables the prediction model to inherently capture more of the missing structure.

To better reflect realistic operating conditions, Fig. 11 shows, under beamspace compression, the prediction performance in mixed outage scenarios with varying maximum CSI missing ratios. In a mixed outage scenario, the masking pattern may correspond to any of the three outage types. Moreover, the maximum missing ratio indicates that the probability of an outage occurring in the historical sequence is uniformly distributed between 0 and the specified maximum value.

Specifically, we investigate four configurations: Pretrain (Masked), NoPretrain (Masked), Pretrain (Overall), and NoPretrain (Overall). The first two curves evaluate prediction accuracy exclusively under pCSI conditions, while the latter two assess the overall performance across all scenarios, including both pCSI and fully observed CSI cases.

It is observed that as the maximum missing ratio increases, the NMSE becomes larger across all configurations, reflecting the growing challenge of reconstructing the channel with fewer available measurements. Nonetheless, the pretraining-based approach consistently achieves lower NMSE than the non-pretrained model. In particular, the improvement in the masked conditions is more significant than that in capturing the overall performance encompassing both complete and pCSI cases, confirming that the pretrained representations are especially effective in reconstructing missing CSI entries. This result further validates the effectiveness of our framework in scenarios with severe measurement outages or sparse sampling, which are common challenges in dynamic satellite communication systems. The ability to maintain high prediction accuracy even at high outage ratios further highlights the robustness of the proposed ST-attention based scheme combined with pretraining processing.

VII. Conclusion

In this paper, we proposed a novel MIMO channel prediction framework for LEO satellite communications involving UAV-RIS. Our proposal leverages a transformer-based ST-attention mechanism to capture both long-range temporal dependencies and spatial correlations. Additionally, we designed a two-stage self-supervised pretraining strategy which uses masked channel observations to reconstruct missing CSI entries and learn robust spatiotemporal features, thereby enhancing the system's resilience to incomplete CSI. Simulation results show that the proposed ST-attention based approach consistently outperforms conventional methods, including T-attention transformer and LSTM networks, under both perfect and pCSI conditions. Moreover, the designed pretraining strategy plays a critical role in mitigating the adverse effects of severe outages on CSI acquisition, validating the significance of integrating SSL into channel prediction frameworks for dynamic LEO satellite communication environments. Overall, the combination of ST-attention and pretraining offers a promising direction for robust channel prediction in challenging pCSI scenarios.

Acknowledgment

The authors gratefully acknowledge the National Supercomputing Centre Singapore (NSCC) for providing the computational resources used in this work.

REFERENCES

[1] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M. Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis, "Satellite Communications in The New Space Era: A Survey and Future Challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 70–109, 2021.

[2] X. Luo, H.-H. Chen, and Q. Guo, "LEO/VLEO Satellite Communications in 6G and Beyond Networks – Technologies, Applications, and Challenges," *IEEE Network*, vol. 38, no. 5, pp. 273–285, Sep. 2024.

[3] K. Tekbıyık, G. K. Kurt, A. R. Ekti, and H. Yanikomeroglu, "Reconfigurable Intelligent Surfaces in Action for Nonterrestrial Networks," *IEEE Vehicular Technology Magazine*, vol. 17, no. 3, pp. 45–53, Sep. 2022.

[4] T. Sun, S. Yin, L. Deng, and F. Richard Yu, "Reinforcement-Learning-Based Trajectory Design and Phase-Shift Control in UAV-Mounted-RIS Communications," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 3, pp. 163–175, 2025.

[5] P. S. Bithas, G. A. Ropokis, G. K. Karagiannidis, and H. E. Nistazakis, "UAV-Assisted Communications With RIS: A Shadowing-Based Stochastic Analysis," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 7, pp. 10000–10010, Jul. 2024.

[6] L. You, K.-X. Li, J. Wang, X. Gao, X.-G. Xia, and B. Ottersten, "Massive MIMO Transmission for LEO Satellite Communications," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 8, pp. 1851–1865, 2020.

[7] K. Feng, T. Zhou, T. Xu, X. Chen, H. Hu, and C. Wu, "Reconfigurable Intelligent Surface-Assisted Multisatellite Cooperative Downlink Beamforming," *IEEE Internet of Things Journal*, vol. 11, no. 13, pp. 23222–23235, Jul. 2024.

[8] J. Shi, Z. Li, J. Hu, Z. Tie, S. Li, W. Liang, and Z. Ding, "OTFS Enabled LEO Satellite Communications: A Promising Solution to Severe Doppler Effects," *IEEE Network*, vol. 38, no. 1, pp. 203–209, Jan. 2024.

[9] C.-H. Lin, S.-C. Lin, and L. C. Chu, "A Low-Overhead Dynamic Formation Method for LEO Satellite Swarm Using Imperfect CSI," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 6923–6936, May 2024.

[10] M. Alsenwi, E. Lagunas, and S. Chatzinotas, "Robust Beamforming for Massive MIMO LEO Satellite Communications: A Risk-Aware Learning Framework," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 6560–6571, May 2024.

[11] Y. Zhang, Y. Wu, A. Liu, X. Xia, T. Pan, and X. Liu, "Deep Learning-Based Channel Prediction for LEO Satellite Massive MIMO Communication System," *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1835–1839, Aug. 2021.

[12] M. Ying, X. Chen, Q. Qi, and W. Gerstacker, "Deep Learning-based Joint Channel Prediction and Multibeam Precoding for LEO Satellite Internet of Things," *IEEE Transactions on Wireless Communications*, vol. 23, no. 10, pp. 13946–13960, 2024.

[13] S. Hochreiter, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[15] H. Zhang, W. Song, X. Liu, M. Sheng, W. Li, K. Long, and O. A. Dobre, "Intelligent Channel Prediction and Power Adaptation in LEO Constellation for 6G," *IEEE Network*, vol. 37, no. 2, pp. 110–117, Mar. 2023.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.

[17] H. Jiang, M. Cui, D. W. K. Ng, and L. Dai, "Accurate Channel Prediction Based on Transformer: Making Mobility Negligible," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 9, pp. 2717–2732, Sep. 2022.

[18] J. Grigsby, Z. Wang, N. Nguyen, and Y. Qi, "Long-Range Transformers for Dynamic Spatiotemporal Forecasting," 2023, *arXiv:2109.12218*.

[19] Z. Lin, Y. Zhang, Z. Chen, Z. Fang, C. Wu, X. Chen, Y. Gao, and J. Luo, "LEO-Split: A Semi-Supervised Split Learning Framework over LEO Satellite Networks," 2025, *arXiv:2501.01293*.

[20] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised Representation Learning by Predicting Image Rotations," 2018, *arXiv:1803.07728*.

[21] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[22] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[23] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A General Framework for Self-Supervised Learning in Speech, Vision and Language," *The 39th International Conference on Machine Learning*, PMLR, 2022, pp. 1298–1312.

[24] J. Devlin, "Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding," 2018, *arXiv:1810.04805*.

[25] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[26] R. Wang, M. A. Kishk, and M.-S. Alouini, "Ultra-Dense LEO Satellite-Based Communication Systems: A Novel Modeling Technique," *IEEE Communications Magazine*, vol. 60, no. 4, pp. 25–31, Apr. 2022.

[27] D. A. Vallado, *Fundamentals of Astrodynamics and Applications*, New York, NY, USA: Springer Science & Business Media, 2001.

[28] J. Zhu, "Conversion of Earth-Centered Earth-Fixed Coordinates to Geodetic Coordinates," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 30, no. 3, pp. 957–961, Jul. 1994.

[29] International Telecommunication Union, "Propagation Data and Prediction Methods Required for the Design of Earth-Space Telecommunication Systems," Tech. Rep. ITU-R P.618-13, Geneva, Switzerland, 2017.

[30] International Telecommunication Union, Radiocommunication Sector (ITU-R), *Recommendation ITU-R P.676-13: Attenuation by Atmospheric Gases and Related Effects*, Aug. 2022.

[31] International Telecommunication Union, Radiocommunication Sector (ITU-R), *Recommendation ITU-R P.618-15: Propagation Data and Prediction Methods Required for the Design of Earth–Space Telecommunication Systems*, Sep. 2024.

[32] 3rd Generation Partnership Project (3GPP), "Study on Channel Model for Frequencies from 0.5 to 100 GHz," Tech. Rep. TR 38.901, 2019.

[33] A. Abdi, C. Tepedelenlioglu, M. Kaveh, and G. Giannakis, "On The Estimation of The K Parameter for The Rice Fading Distribution," *IEEE Communications Letters*, vol. 5, no. 3, pp. 92–94, 2002.

[34] International Telecommunication Union, Radiocommunication Sector (ITU-R), *Recommendation ITU-R P.1812-7: A Path-Specific Propagation Prediction Method for Point-to-Area Terrestrial Services in the Frequency Range 30 MHz to 6 000 MHz*, Aug. 2023.

[35] NASA, "International Space Station (ISS) Facts and Figures," 2019.

[36] L. E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, no. 3, pp. 497–516, Jul. 1957.

[37] B. Liang and Z. J. Haas, "Predictive distance-based mobility management for PCS networks," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 718–732, Oct. 2003.

[38] P. K. Chowdhury, M. Atiquzzaman, and W. Ivancic, "Handover Schemes in Satellite Networks: State-of-The-Art and Future Research Directions," *IEEE Communications Surveys & Tutorials*, vol. 8, no. 4, pp. 2–14, 2006.

[39] A. Gillioz, J. Casas, E. Mugellini, and O. A. Khaled, "Overview of The Transformer-Based Models for NLP Tasks," *15th Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2020, pp. 179–183.

[40] NVIDIA Corporation, "NVIDIA H100 Tensor Core GPUs," 2024.

[41] P. Goyal, P. Dollár, R. Girshick, P. Noordhuis, L. Wesolowski, A. Kyrola, A. Tulloch, Y. Jia, and K. He, "Accurate, large minibatch SGD: Training ImageNet in 1 hour," 2017, *arXiv:1706.02677*.

[42] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling Long- and Short-Term Temporal Patterns with Deep Neural Networks," 2018, *arXiv:1703.07015*.

[43] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are Transformers Effective for Time Series Forecasting?" 2022, *arXiv:2205.13504*.

[44] NVIDIA Corporation, "NVIDIA Jetson Orin NX Series Data Sheet," 2022.

[45] NVIDIA Corporation, "NVIDIA Jetson AGX Orin Developer Kit," 2025.