Transformer-based Pilot-to-Prediction for Frequency-Selective Channels in OFDM Systems

1st Louis Lagona, 2nd MohammadAmin Vakilifard, 3rd Carsten Bockelmann, 4th Armin Dekorsy *Department of Communications Engineering*
University of Bremen, Germany
Email: {lagona, vakilifard, bockelmann, dekorsy}@ant.uni-bremen.de

***Abstract*—5G-Beyond (5G-B)/6G require higher spectral efficiency (SE) and data throughput (DT), motivating reduced pilot overhead in orthogonal frequency-division multiplexing (OFDM) systems. Traditional methods target temporal prediction and address frequency selectivity via pilot-tone interpolation, leaving time–frequency prediction unexplored. To address this gap, we propose a Transformer-based pilot-to-prediction (P2P) neural network (NN) for joint channel estimation and prediction. Exploiting the Transformer's self-attention, our method captures temporal dynamics and inter-subcarrier correlations to predict the frequency-selective channel across all subcarriers using only past pilot observations, requiring no pilots in the symbols being predicted. Simulation results show that our approach closely matches the accuracy of a genie-aided receiver with perfect channel knowledge and outperforms Wiener filtering and long short-term memory (LSTM) baselines. Consequently, pilot overhead is substantially reduced, enhancing SE by 33.3%. Achievable data rates are thus improved, underscoring the suitability of Transformer-based channel prediction for future wireless standards.**

***Index Terms*—Channel prediction, Transformer, frequency-selective fading, OFDM, pilot overhead**

I. Introduction

Next-generation 5G-Beyond (5G-B)/6G systems demand higher spectral efficiency (SE) and data throughput (DT), making pilot overhead a practical bottleneck in orthogonal frequency-division multiplexing (OFDM) links. Pilot-aided estimation occupies time–frequency resources by design, limiting SE and DT [1]. Classical pilot-interpolation schemes are widely used, but still incur this overhead [2]. This motivates channel prediction: leveraging past pilots to supply timely channel state information (CSI) while omitting pilots in the predicted symbols [3], [4].

Classical predictors (e.g. MMSE/Wiener) require second-order statistics, matrix operations, and are sensitive to model mismatch [5], [6]. Deep predictors improve robustness: long short-term memory (LSTM) models capture temporal dynamics [7], while Transformer predictors offer parallel, multi-step forecasting and reduced error propagation [4]. However, most prior prediction works emphasize temporal evolution, often per subcarrier; frequency selectivity is handled via per-tone processing or interpolation, whereas few jointly model time–frequency structure with no pilots in the predicted OFDM symbols [4], [8]. Channelformer (attention-based) [9] and ChannelNet (CNN-based) [10] are intra-frame channel frequency response (CFR) estimators, not forecasting models. Using either for prediction entails (i) per-symbol full-grid estimation for past OFDM symbols, (ii) causal temporal extrapolation (e.g., Wiener/LSTM/Transformer), and (iii) inter-stage error accumulation because they target spatial super-resolution/denoising rather than temporal dynamics.

To address these limitations, we propose a Transformer-based pilot-to-prediction (P2P) neural network (NN) that performs joint channel estimation and prediction. Leveraging the Transformer's self-attention, the P2P model simultaneously captures both temporal channel dynamics and inter-subcarrier correlations. Unlike interpolation-based schemes, this predictor reconstructs the complete frequency-selective channel for forthcoming OFDM symbols from previously received pilots, enabling a substantial reduction in pilot density—and thus overhead—throughout the prediction horizon. Consequently, the method improves SE and DT, which is pivotal for latency-sensitive OFDM systems. These efficiency gains underscore the suitability of the Transformer-based P2P design for future wireless standards that demand minimal overhead and maximal data rates.

Notations: Lowercase (uppercase) italics denote time (frequency) domain variables, with bold letters indicating matrices, e.g., x, $\mathbf{x}$, X, and $\mathbf{X}$. For matrices $\mathbf{X}$, $\mathbf{X}[k,:]$ and $\mathbf{X}[:,n]$ represent the kth row and nth column vectors, respectively.

II. System Model

The transmitter and receiver are assumed to be equipped with single antennas. Each OFDM subframe consists of N subcarriers and T OFDM symbols. The relationship between transmitted and received signal can be represented by

$$\mathbf{Y} = \mathbf{X} \odot \mathbf{H} + \mathbf{U} \tag{1}$$

where $\odot$ stands for the Hadamard product. $\mathbf{H} \in \mathbb{C}^{T\times N}$ represents the channel matrix. $\mathbf{X} \in \mathbb{C}^{T\times N}$ and $\mathbf{Y} \in \mathbb{C}^{T\times N}$ are transmitted signal and received signal respectively. Let $\mathbf{U} \in \mathbb{C}^{T\times N}$ denote the additive Gaussian noise matrix with i.i.d. entries $U[k,n] \sim \mathcal{CN}(0,\sigma^2)$. The data subcarriers employ 16-QAM modulation, while pilot subcarriers use QPSK modulation as a balance between SE for data transmission and robustness for channel estimation. If nth subcarrier and

This research was supported in part by the German Federal Ministry of Research, Technology and Space (BMFTR) within the projects Open6GHub under grant number 16KISK016

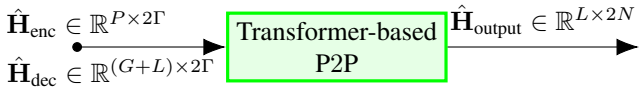

Fig. 1. Schematic structure of a Transformer-based P2P.

kth symbol are allocated to transmit pilot symbol, it can be written as

$$Y_p[k,n] = H_p[k,n]X_p[k,n] + U[k,n] \tag{2}$$

the subscript p indicates that subcarrier at $[k,n]$ is a pilot symbol. Channel estimation on the pilot position can be obtained using least squares (LS):

$$\hat{H}_p^{\text{LS}}[k,n] = \arg\min_{H_p[k,n]} \|Y_p[k,n] - H_p[k,n]X_p[k,n]\|^2 \tag{3}$$

$$= \frac{Y_p[k,n]}{X_p[k,n]}. \tag{4}$$

Each OFDM symbol k comprises Γ pilot subcarriers and Λ data subcarriers (i.e. $\Gamma + \Lambda = N$). All OFDM symbols use the same comb-type pilot grid, placing pilots on subcarriers $n = 1 + \lfloor (i-1)(N-1)/(\Gamma-1) \rfloor$, $i = 1, \ldots, \Gamma$, thus including both $n = 1$ and $n = N$.

III. Transformer-based Pilot-to-Prediction of OFDM Symbols

The input and output data of the Transformer are illustrated in Fig. 1. The encoder input $\hat{\mathbf{H}}_{\text{enc}} \in \mathbb{R}^{P\times 2\Gamma}$ comprises channel estimates at pilot subcarriers from the last $P \leq T$ OFDM symbols and defines the total temporal context.

$$\hat{\mathbf{H}}_{\text{enc}}[k,:] = \left[\Re(\hat{\mathbf{H}}_p^{\text{LS}}[k,:]), \Im(\hat{\mathbf{H}}_p^{\text{LS}}[k,:])\right], \tag{5}$$

where $\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary components of the complex-valued channel estimates $\hat{\mathbf{H}}_p^{\text{LS}} \in \mathbb{C}^{P\times\Gamma}$ at pilot positions. In contrast, the decoder input $\hat{\mathbf{H}}_{\text{dec}} \in \mathbb{R}^{(G+L)\times 2\Gamma}$ consists of the most recent $G \leq P$ pilot-based estimates, followed by L zero-padded rows that reserve positions for autoregressive prediction. The output $\hat{\mathbf{H}}_{\text{output}} \in \mathbb{R}^{L\times 2N}$ corresponds to the predicted CFR for the next L OFDM symbols. The Transformer employs an encoder-decoder structure (Fig. 2) based on the classical architecture [11], adapted here for the P2P channel prediction task. Let Φ_e and Φ_d denote the number of encoder and decoder layers, respectively.

A. Encoder of Transformer Model

Following standard Transformer practice, each 2Γ-dimensional pilot vector is projected into a d_{model}-dimensional embedding via

$$\hat{\mathbf{H}}_{\text{enc}}^{\text{embedded}} = \hat{\mathbf{H}}_{\text{enc}}\mathbf{W}_{\text{enc}}^{\text{embed}}, \quad \mathbf{W}_{\text{enc}}^{\text{embed}} \in \mathbb{R}^{2\Gamma\times d_{\text{model}}} \tag{6}$$

where $\mathbf{W}_{\text{enc}}^{\text{embed}}$ contains learnable weights. To convey the OFDM-symbol order otherwise lost in the permutation-invariant attention the embedded sequence $\hat{\mathbf{H}}_{\text{enc}}^{\text{embedded}} \in \mathbb{R}^{P\times d_{\text{model}}}$ is additively fused with a learnable sinusoidal positional encoding $\mathbf{E}_{\text{enc}}^{\text{pos}} \in \mathbb{R}^{P\times d_{\text{model}}}$:

$$\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}} = \hat{\mathbf{H}}_{\text{enc}}^{\text{embedded}} + \mathbf{E}_{\text{enc}}^{\text{pos}}. \tag{7}$$

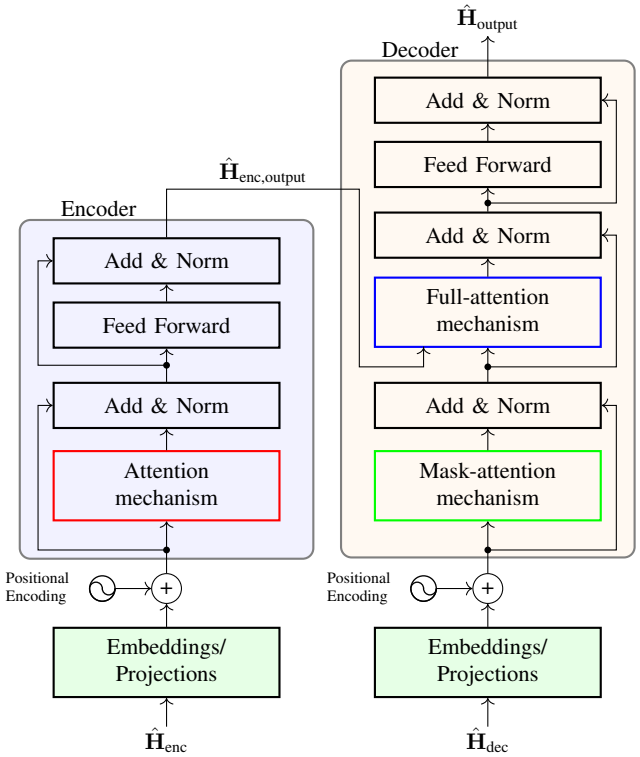

Fig. 2. The schematic structure of the proposed Transformer consists of an Encoder-Decoder.

Here, $\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}} \in \mathbb{R}^{P\times d_{\text{model}}}$ denotes the positionally encoded input sequence. The positional matrix $\mathbf{E}_{\text{enc}}^{\text{pos}}$ is initialized with sinusoidal values and treated as trainable during optimization:

$$\mathbf{E}_{\text{enc}}^{\text{pos}}[t, 2m] = \sin\left(\frac{t}{10000^{2m/d_{\text{model}}}}\right), \tag{8}$$

$$\mathbf{E}_{\text{enc}}^{\text{pos}}[t, 2m+1] = \cos\left(\frac{t}{10000^{2m/d_{\text{model}}}}\right), \tag{9}$$

where $m \in \{0, \ldots, \lfloor \frac{d_{\text{model}}}{2} \rfloor - 1\}$ indexes the dimension, $t \in \{0, \ldots, P-1\}$ represents the temporal position of the input sequence, and d_{model} is chosen as an even integer to ensure complete coverage of sine/cosine pairs. Each row $\mathbf{E}_{\text{enc}}^{\text{pos}}[t,:]$ encodes absolute position information while allowing adaptation to channel temporal characteristics.

The core components of both the encoder and decoder are the multi-head attention (MHA) layers, which implement the attention mechanism. These layers consist of multiple attention heads Z operating in parallel, accelerating training and capturing relationships between different channel values. Each attention head ζ_i computes the output of the ith head using the attention mechanism (Fig. 2 red-boxed):

$$\zeta_i^{\text{enc}} = \text{Attention}(\mathbf{Q}_i^{\text{enc}}, \mathbf{K}_i^{\text{enc}}, \mathbf{V}_i^{\text{enc}}), \tag{10}$$

$$\mathbf{K}_i^{\text{enc}} = \mathbf{W}_{K,i}^{\text{enc}}\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}}, \quad \mathbf{W}_{K,i}^{\text{enc}} \in \mathbb{R}^{d_{\text{model}}\times d_k} \tag{11}$$

$$\mathbf{Q}_i^{\text{enc}} = \mathbf{W}_{Q,i}^{\text{enc}}\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}}, \quad \mathbf{W}_{Q,i}^{\text{enc}} \in \mathbb{R}^{d_{\text{model}}\times d_k} \tag{12}$$

$$\mathbf{V}_i^{\text{enc}} = \mathbf{W}_{V,i}^{\text{enc}}\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}}, \quad \mathbf{W}_{V,i}^{\text{enc}} \in \mathbb{R}^{d_{\text{model}}\times d_v} \tag{13}$$

where $\mathbf{W}_{K,i}^{\text{enc}}$, $\mathbf{W}_{Q,i}^{\text{enc}}$, $\mathbf{W}_{V,i}^{\text{enc}}$ are learnable weight matrices for the encoder. The projections ensure dimensional consistency:

$\mathbf{K}_i^{\text{enc}}, \mathbf{Q}_i^{\text{enc}} \in \mathbb{R}^{P\times d_k}$ and $\mathbf{V}_i^{\text{enc}} \in \mathbb{R}^{P\times d_v}$ with $d_k = d_v = \frac{f_a \cdot d_{\text{model}}}{Z}$, where $f_a \geq 1$ scales the attention head dimension. This definition of d_k and d_v applies uniformly to all attention heads in both encoder and decoder modules. The scaled dot-product attention is then computed as:

$$\text{Attention}(\mathbf{Q}_i^{\text{enc}}, \mathbf{K}_i^{\text{enc}}, \mathbf{V}_i^{\text{enc}}) = \text{softmax}\left(\frac{\mathbf{Q}_i^{\text{enc}}(\mathbf{K}_i^{\text{enc}})^\top}{\sqrt{d_k}}\right)\mathbf{V}_i^{\text{enc}}, \tag{14}$$

where softmax is defined as $\text{softmax}(\mathbf{x})_j = \frac{\exp(x_j)}{\sum_k \exp(x_k)}$ and applied row-wise. $(\cdot)^\top$ denotes the matrix transpose. The scaling factor $\sqrt{d_k}$ ensures numerical stability. Outputs of Z attention heads are concatenated and linearly transformed:

$$\text{MultiHead}^{\text{enc}}(\mathbf{Q}^{\text{enc}}, \mathbf{K}^{\text{enc}}, \mathbf{V}^{\text{enc}}) = \text{Concat}(\zeta_1^{\text{enc}}, \ldots, \zeta_Z^{\text{enc}})\mathbf{W}_O^{\text{enc}}, \tag{15}$$

where $\mathbf{W}_O^{\text{enc}} \in \mathbb{R}^{(Zd_k)\times d_{\text{model}}}$ maps the concatenated outputs to the d_{model} dimension. This design enables multi-head attention to learn diverse dependencies, enhancing the model's representational power. The MHA output $\hat{\mathbf{H}}_{\text{out, MHA}} \in \mathbb{R}^{P\times d_{\text{model}}}$ is added to the positionally encoded input, followed by layer normalization (LN) to stabilize training:

$$\hat{\mathbf{H}}_{\text{out, MHA}} = \text{LN}\left(\hat{\mathbf{H}}_{\text{enc}}^{\text{pos}} + \text{MultiHead}^{\text{enc}}(\mathbf{Q}^{\text{enc}}, \mathbf{K}^{\text{enc}}, \mathbf{V}^{\text{enc}})\right), \tag{16}$$

where $\text{LN}(\cdot)$ is the LN function:

$$\text{LN}(\mathbf{X}[i,:]) = \frac{\mathbf{X}[i,:] - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}, \tag{17}$$

with μ_i and σ_i^2 as the mean and variance of $\mathbf{X}[i,:]$, respectively:

$$\mu_i = \frac{1}{d_{\text{model}}}\sum_{j=1}^{d_{\text{model}}} X[i,j], \quad \sigma_i^2 = \frac{1}{d_{\text{model}}}\sum_{j=1}^{d_{\text{model}}} \left(X[i,j] - \mu_i\right)^2, \tag{18}$$

where $\epsilon = 10^{-5}$ ensures numerical stability. The normalized output is passed through a position-wise feed-forward network (FFN):

$$\text{FFN}(\hat{\mathbf{H}}_{\text{out, MHA}}) = \left(\text{ReLU}\left(\hat{\mathbf{H}}_{\text{out, MHA}}\mathbf{W}_{\text{input}} + \mathbf{b}_{\text{input}}\right)\right)\mathbf{W}_{\text{hidden}} + \mathbf{b}_{\text{output}}, \tag{19}$$

where ReLU is defined as $\text{ReLU}(x) = \max(0, x)$. Here, $\mathbf{W}_{\text{input}} \in \mathbb{R}^{d_{\text{model}}\times d_{\text{ff}}}$, $\mathbf{W}_{\text{hidden}} \in \mathbb{R}^{d_{\text{ff}}\times d_{\text{model}}}$, $\mathbf{b}_{\text{input}} \in \mathbb{R}^{d_{\text{ff}}}$, and $\mathbf{b}_{\text{output}} \in \mathbb{R}^{d_{\text{model}}}$ are learnable weights and biases, where $d_{\text{ff}} = 4 \cdot d_{\text{model}}$ follows the standard Transformer configuration. Finally, another residual connection and LN yield the encoder output:

$$\hat{\mathbf{H}}_{\text{enc,output}} = \text{LN}\left(\hat{\mathbf{H}}_{\text{out, MHA}} + \text{FFN}(\hat{\mathbf{H}}_{\text{out, MHA}})\right). \tag{20}$$

This final output $\hat{\mathbf{H}}_{\text{enc,output}} \in \mathbb{R}^{P\times d_{\text{model}}}$ is the real-valued matrix representing the encoder's output.

B. Decoder of Transformer Model

The decoder input $\hat{\mathbf{H}}_{\text{dec}} \in \mathbb{R}^{(G+L)\times 2\Gamma}$ comprises G historical OFDM symbols at pilot subcarriers concatenated with a zero matrix $\mathbf{0}_{L\times 2\Gamma}$. The zero-padded block enables autoregressive prediction by reserving space for L future symbols, which are iteratively populated during inference. The embedding operation and positional encoding follow the same procedure as in the encoder (cf. (6)–(9)), using a dedicated learnable projection matrix $\mathbf{W}_{\text{dec}}^{\text{embed}} \in \mathbb{R}^{2\Gamma\times d_{\text{model}}}$ and positional matrix $\mathbf{E}_{\text{dec}}^{\text{pos}}$. Subsequent masked self-attention is structurally identical to the encoder's MHA (cf. (10)–(13)), but employs decoder-specific weights and applies causal masking to restrict attention to past and current positions only (Fig. 2 green-boxed).

The masked multi-head output $\mathbf{Z}_{\text{dec}} \in \mathbb{R}^{(G+L)\times d_{\text{model}}}$ combines all heads through:

$$\mathbf{Z}_{\text{dec}} = \text{LN}\left(\hat{\mathbf{H}}_{\text{dec}}^{\text{pos}} + \text{Concat}(\zeta_1^{\text{dec}}, \ldots, \zeta_Z^{\text{dec}})\mathbf{W}_O^{\text{dec}}\right), \tag{21}$$

where $\mathbf{W}_O^{\text{dec}} \in \mathbb{R}^{(Zd_v)\times d_{\text{model}}}$ and $\text{Mask}(\cdot)$ applies an upper triangular mask:

$$\text{Attention}(\mathbf{Q}_i^{\text{dec}}, \mathbf{K}_i^{\text{dec}}, \mathbf{V}_i^{\text{dec}}) = \text{softmax}\left(\frac{\text{Mask}(\mathbf{Q}_i^{\text{dec}}(\mathbf{K}_i^{\text{dec}})^\top)}{\sqrt{d_k}}\right)\mathbf{V}_i^{\text{dec}}, \tag{22}$$

where $\text{Mask}(\mathbf{X}) = \mathbf{X} + \mathbf{\Delta}_{\text{upper}}(-\infty)$ applies a causal mask. Here $\mathbf{\Delta}_{\text{upper}}(-\infty)$ is an upper strictly triangular matrix defined as:

$$\Delta_{\text{upper}}(-\infty)[i,j] = \begin{cases} 0, & \text{if } j \leq i \\ -\infty, & \text{if } j > i \end{cases}. \tag{23}$$

This ensures $\text{softmax}(-\infty) = 0$ for future positions ($j > i$), blocking information flow from future tokens while allowing attention to past/current positions. This is essential for autoregressive prediction.

For cross-attention (Fig. 2 blue-boxed) with encoder outputs, key/value projections become:

$$\mathbf{K}_i^{\text{cross}} = \mathbf{W}_{K,i}^{\text{cross}}\hat{\mathbf{H}}_{\text{enc,output}}, \quad \mathbf{W}_{K,i}^{\text{cross}} \in \mathbb{R}^{d_{\text{model}}\times d_k} \tag{24}$$

$$\mathbf{V}_i^{\text{cross}} = \mathbf{W}_{V,i}^{\text{cross}}\hat{\mathbf{H}}_{\text{enc,output}}, \quad \mathbf{W}_{V,i}^{\text{cross}} \in \mathbb{R}^{d_{\text{model}}\times d_v} \tag{25}$$

$$\mathbf{Q}_i^{\text{cross}} = \mathbf{W}_{Q,i}^{\text{cross}}\mathbf{Z}_{\text{dec}}. \quad \mathbf{W}_{Q,i}^{\text{cross}} \in \mathbb{R}^{d_{\text{model}}\times d_k} \tag{26}$$

The final cross-attention $\mathbf{Z} \in \mathbb{R}^{(G+L)\times d_{\text{model}}}$ output combines encoder-decoder information:

$$\mathbf{Z} = \text{LN}\left(\mathbf{Z}_{\text{dec}} + \text{Concat}(\zeta_1^{\text{cross}}, \ldots, \zeta_Z^{\text{cross}})\mathbf{W}_O^{\text{cross}}\right), \tag{27}$$

where $\zeta_i^{\text{cross}} = \text{Attention}(\mathbf{Q}_i^{\text{cross}}, \mathbf{K}_i^{\text{cross}}, \mathbf{V}_i^{\text{cross}})$. A position-wise FFN identical to the encoder (cf. (19), (20)) completes processing, resulting in $\mathbf{Z}_{\text{final}} \in \mathbb{R}^{(G+L)\times d_{\text{model}}}$. Finally, an output projection reduces dimensionality to $2N$ and selects only the last L rows:

$$\hat{\mathbf{H}}_{\text{output}} = \left(\mathbf{Z}_{\text{final}}\mathbf{W}_{\text{out}}\right)[-L:,:], \quad \mathbf{W}_{\text{out}} \in \mathbb{R}^{d_{\text{model}}\times 2N} \tag{28}$$

producing the predicted channel $\hat{\mathbf{H}}_{\text{output}} \in \mathbb{R}^{L\times 2N}$ for the next L OFDM symbols.

C. Loss Function

The model is trained to minimize the mean squared error (MSE) between predicted and true channel responses:

$$\mathcal{L}_{\text{MSE}}(\Theta) = \frac{1}{2NL}\left\|\hat{\mathbf{H}}_{\text{output}} - \mathbf{H}_{\text{true}}\right\|_F^2, \tag{29}$$

where $\mathbf{H}_{\text{true}} \in \mathbb{R}^{L\times 2N}$ contains real/imaginary components of future channels. The loss depends on $|\Theta| = 158{,}336$ trainable parameters.

TABLE I
TRANSFORMER HYPERPARAMETERS

Variable	*Value*	*Variable*	*Value*	*Variable*	*Value*
d_{model}	64	d_{ff}	256	Z	8
Ψ	200	Φ_e	1	Φ_d	1

TABLE II
SYSTEM PARAMETERS

Variable	*Value*	*Variable*	*Value*	*Variable*	*Value*
T	30	N	128	Γ	32
Λ	96	Δf	15 kHz	f_c	2.5 GHz
L	5	P	30	G	10

IV. SIMULATION RESULTS

We use a single-layer Transformer ($\Phi_e = \Phi_d = 1$). Training uses AdamW [12] ($\eta = 10^{-3}$, weight-decay $\lambda = 4 \times 10^{-3}$) with $\eta_{\text{epoch}} = \eta, e^{-0.1\max(\text{epoch}-9,0)}$ and early stopping (patience 20). We train on $5{\times}10^4$ random channels (80/20 split) for $\Psi = 200$ epochs, batch size 64, at fixed SNR $\rho_{\text{train}} = 10$ dB. The attention factor $f_a = 5$ scales the projection dimensions; fixed positional embeddings encode temporal order. The encoder uses full attention for long-range dependencies. Hyperparameters are summarized in Table I.

Frequency-selective channels are generated using 3GPP COST 259 typical urban (CTU) [13] with $f_c = 2.5$ GHz, $\Delta f = 15$ kHz, and various user velocities v, resulting in corresponding Doppler shifts f_D. The Transformer employs $P = 30$ (subframe length $T = 30$) encoder inputs and $G = 10$ decoder history symbols to predict $L = 5$ future OFDM symbols (≈ 333 μs ahead). Prediction horizon $L = 5$ trades off accuracy and prediction length; shorter horizons enhance precision, while longer horizons degrade due to generalized trend prediction. The P2P LSTM from [7], comprising (Dense, 2×LSTM, Dense; $|\Theta|_{\text{LSTM}} = 157{,}908$), has a comparable complexity to our Transformer model. The Wiener filter uses $P_{\text{Wiener}} = 4$ past OFDM symbols where LS-based pilot estimates are linearly interpolated across subcarriers before temporal prediction. Simulations utilize an RTX 4090, Ryzen 9 7950X3D, and 64 GB RAM. System parameters are listed in Table II.

BER/MSE vs. SNR are averaged over 150,000 channels, each containing L OFDM symbols (128 data subcarriers/symbol); for MSE/SE over predicted symbols at 10 dB we use 10^4 channels, with separate train/test seeds. Figs. 3 and 4 illustrate BER/MSE vs. SNR for predictions of L OFDM symbols, demonstrating the effectiveness of our Transformer-based approach. Our Transformer slightly outperforms P2P LSTM (≈ 0.5 dB MSE at 10 dB) and clearly beats Wiener (BER gain 0.5–1.5 dB; MSE ≈ 4.2 dB). Fig. 4 shows an increase in MSE as velocity rises from 3 km/h to 100 km/h, reflecting the key performance-limiting effect of Doppler shifts. At 3 km/h and 0–5 dB, prediction is noise-limited; both models saturate at the pilot-noise floor, yielding comparable MSE. At 100 km/h, doubling pilot density to $\Gamma = 64$ yields $\approx 1.5\times$ lower MSE than $\Gamma = 32$, validating that increased pilot density effectively mitigates high-Doppler degradation. Fig. 5 shows stable MSE across the prediction horizon for both NNs, whereas Wiener

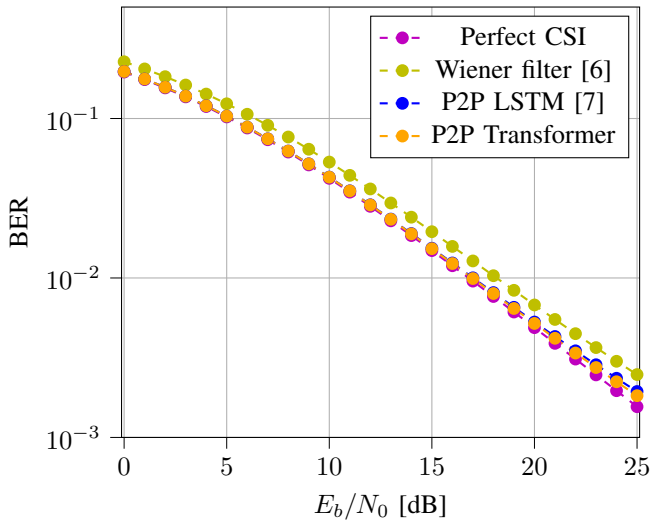

Fig. 3. BER for 16-QAM over CTU channel at 2.5 GHz.

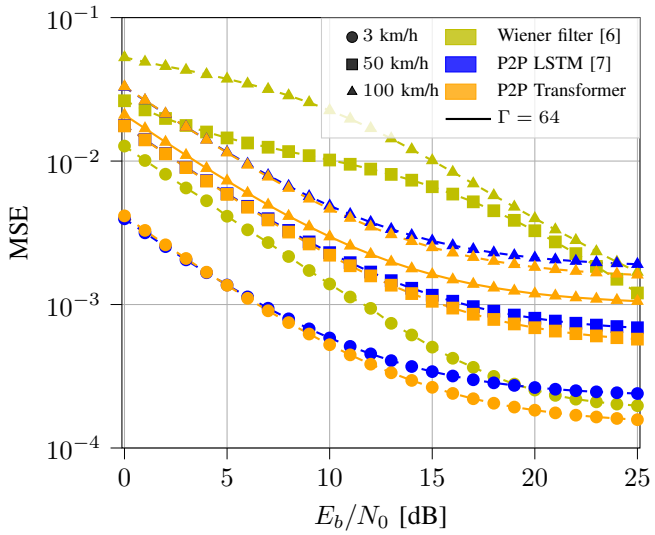

Fig. 4. MSE for 16-QAM over COST259 Urban at 2.5 GHz.

filter performance rapidly deteriorates. At 100 km/h, MSE rises to 10^{-3}–10^{-2}, remains structurally stable across the prediction horizon, and degrades gracefully with Doppler. Fig. 6 shows the SE over predicted OFDM symbols. Transformer- and LSTM-based methods closely approach the perfect-CSI performance, while Wiener filtering lags. The upper dashed line denotes the SE limit with no pilots (128 data subcarriers), and the lower dashed line reflects ideal SE with 32 pilots per symbol. These bounds indicate the maximum SE under perfect prediction with and without pilots.

Under identical training and testing conditions—differing only in architecture—the Transformer averaged 3.59 s per epoch (45% faster than the LSTM's 6.53 s), reached its minimum validation loss 537 s earlier, and achieved a 10.3% lower final validation loss. At inference, the Transformer required 2.39 ms per batch of 64 channels (49% faster than the LSTM's 4.66 ms). For hardware-agnostic complexity, we report parameters and FLOPs per sample forward pass (one

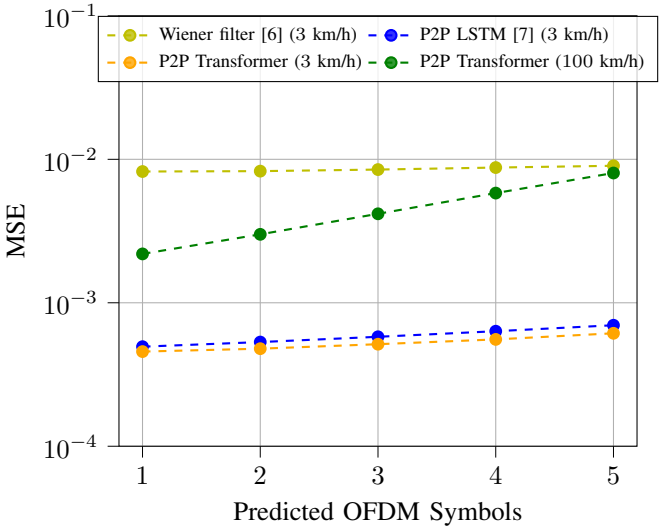

Fig. 5. MSE for predicted OFDM Symbols over CTU channel at 2.5 GHz.

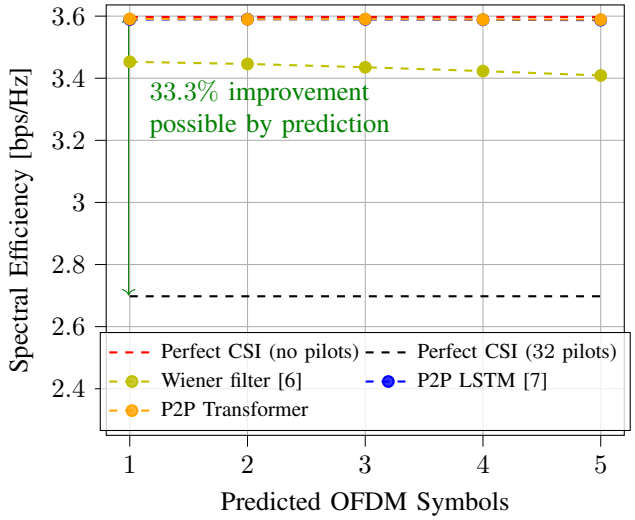

Fig. 6. SE for predicted Symbols over CTU channel at 2.5 GHz.

frame predicting L symbols; multiply–add = 2 FLOPs): Transformer $|\Theta| = 158{,}336$ ($\approx$ 7.1 MFLOPs) vs. LSTM $|\Theta|_{\text{LSTM}} = 157{,}908$ ($\approx$ 9.9 MFLOPs, $\approx$ 40% higher). These outcomes substantiate the benefits of Transformer-based predictors, enabling parallel predictions, mitigating error propagation, and accelerating inference.

V. Conclusion

This work introduced a Transformer-based P2P architecture for accurate prediction of frequency-selective fading channel in OFDM systems. Unlike prior methods that predict each subcarrier independently over time, our model jointly captures temporal and inter-subcarrier correlations to predict the full frequency-selective channel. Against a genie-aided receiver (perfect CSI), LSTM, and Wiener, our model nearly attains genie-level accuracy and outperforms the baselines. While the results show the P2P framework is effective for both NNs in this low-mobility scenario, the Transformer primarily distinguished itself by its superior runtime performance, a key practical advantage. By substantially reducing pilot overhead, the proposed technique significantly enhances SE and DT, addressing critical demands of future 5G-B and 6G wireless networks. Future work will study a non-P2P Transformer for high mobility that uses full context to prioritize prediction and mitigate high-velocity degradation.

References

[1] P. Tarafder, C. Chun, A. Ullah, Y. Kim, and W. Choi, "Channel estimation in 5g-and-beyond wireless communication: A comprehensive survey," *Electronics*, vol. 14, no. 4, 2025. [Online]. Available: https://www.mdpi.com/2079-9292/14/4/750

[2] X. Dong, W.-s. Lu, and A. C. Soong, "Linear interpolation in pilot symbol assisted channel estimation for ofdm," *IEEE Transactions on Wireless Communications*, vol. 6, no. 5, pp. 1910–1920, 2007.

[3] B. D. Filippo, C. Amatetti, and A. Vanelli-Coralli, "Uplink ofdm channel prediction with hybrid cnn-lstm for 6g non-terrestrial networks," 2025. [Online]. Available: https://arxiv.org/abs/2502.09326

[4] 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, 2022.

[5] D. Schafhuber and G. Matz, "Mmse and adaptive prediction of time-varying channels for ofdm systems," *IEEE Trans. Wireless Commun.*, vol. 4, no. 2, pp. 593–602, 2005.

[6] I. Wong, A. Forenza, R. Heath, and B. Evans, "Long range channel prediction for adaptive ofdm systems," in *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004.*, vol. 1, 2004, pp. 732–736 Vol.1.

[7] W. Jiang and H. D. Schotten, "Deep learning for fading channel prediction," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 320–332, 2020.

[8] Y. Zhang, A. Alkhateeb *et al.*, "Predicting future csi feedback for highly-mobile massive mimo systems," arXiv:2202.02492, 2022.

[9] D. Luan and J. Thompson, "Channelformer: Attention based neural solution for wireless channel estimation and effective online training," 2023. [Online]. Available: https://arxiv.org/abs/2302.04368

[10] M. Soltani, V. Pourahmadi, A. Mirzaei, and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Communications Letters*, vol. 23, no. 4, pp. 652–655, 2019.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[12] I. Loshchilov and F. Hutter, "Fixing weight decay regularization in adam," *CoRR*, vol. abs/1711.05101, 2017. [Online]. Available: http://arxiv.org/abs/1711.05101

[13] "Universal mobile telecommunications system (umts); deployment aspects (3gpp tr 25.943 version 17.0.0 release 17)," ETSI, Technical Report TR 125 943 V17.0.0, April 2022. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/125900_125999/125943/17.00.00_60/tr_125943v170000p.pdf