# Continuous-Time Channel Prediction Based on Tensor Neural Ordinary Differential Equation

**Mingyao Cui[1], Hao Jiang[1], Yuhao Chen[1], Yang Du[2], Linglong Dai[1,*]**

[1] The Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

[2] Huawei Technology Co., Ltd. Chengdu Research Institute, Chengdu 611730, China

[*] The corresponding author, email: daill@tsinghua.edu.cn

**Abstract:** Channel prediction is critical to address the channel aging issue in mobile scenarios. Existing channel prediction techniques are mainly designed for discrete channel prediction, which can only predict the future channel in a fixed time slot per frame, while the other intra-frame channels are usually recovered by interpolation. However, these approaches suffer from a serious interpolation loss, especially for mobile millimeter-wave communications. To solve this challenging problem, we propose a tensor neural ordinary differential equation (TN-ODE) based continuous-time channel prediction scheme to realize the direct prediction of intra-frame channels. Specifically, inspired by the recently developed continuous mapping model named neural ODE in the field of machine learning, we first utilize the neural ODE model to predict future continuous-time channels. To improve the channel prediction accuracy and reduce computational complexity, we then propose the TN-ODE scheme to learn the structural characteristics of the high-dimensional channel by low-dimensional learnable transform. Simulation results show that the proposed scheme is able to achieve higher intra-frame channel prediction accuracy than existing schemes.

**Keywords:** channel prediction; massive multiple-input-multiple-output; millimeter-wave communications; ordinary differential equation

## I. INTRODUCTION

Millimeter-wave (mmWave) massive multiple-input multiple-output (MIMO) has been a critical technology for boosting data transmission speed in 5G communication networks [1]. By deploying a large number of antennas at the base station (BS), massive MIMO can achieve several orders of magnitude improvements in beamforming gain [2]. To fully realize this potential, accurate channel state information (CSI) is required at the BS for the efficient design of precoding. According to the current 5G standard [3], each frame in 5G wireless communication systems contains multiple time slots, while only the first time slot of each frame is used to estimate the CSI through the predefined sounding reference signal (SRS). Then, the subsequent time slots within the same frame perform precoding design according to the CSI estimated in the first slot.

However, since the channel is time varying in mobile scenarios, the CSI in the first time slot may significantly differ from the actual channels in the subsequent time slots. This is called channel aging in the literature [4, 5]. Specifically, the channel coherence time is inversely proportional to the carrier frequency and user mobile speed, which could be shorter than the channel estimation period or SRS period in mobile scenarios. For example, for the case of 28 GHz carrier

frequency and 60 km/h user mobile speed, the channel coherence time is about 0.32 ms, which is smaller than the shortest SRS period of 0.625 ms defined by the 5G standard [3]. In this case, the estimated CSI in the first time slot becomes outdated, which could cause a serious spectral efficiency loss of about 30% [6]. Therefore, the channel aging problem has to be carefully addressed to enable fast user mobility in mmWave massive MIMO systems.

## 1.1 Prior Works

To address the channel aging problem, channel prediction techniques have been widely studied to predict the future channels by exploring the channel correlation in the time domain [6–13]. There are two typical categories of channel prediction techniques, i.e., model-based and data-based channel prediction. For the first category [6–8], some classical models are utilized to characterize the time-varying channels, such as the linear extrapolation model [6], the autoregressive (AR) model [7], and the spatio-temporal auto-regressive (ST-AR) model [8]. However, since the actual mobile channels simultaneously suffer from the multi-path effect and the Doppler effect, the time-varying characteristics of actual channels are complicated. Thus, for this category of channel prediction techniques, the fossilized models are difficult to match the time-varying channels, resulting in the unreliable performance in mobile scenarios.

To deal with this problem, data-based channel prediction techniques have been recently proposed to match the time-varying channels in the data-driven way [9–13]. Since the neural network models are able to learn the intrinsic complicated feature from data, which could be exploited to improve the channel prediction accuracy. Specifically, in [9], a fully-connected (FC) network was utilized to predict future channels according to the input of high-dimensional historical channels. Then, to decrease the training complexity caused by high-dimensional historical inputs, the recurrent neural network (RNN) like architectures, such as RNN, gate recurrent unit (GRU), and long-short term memory (LSTM), were trained to iteratively process historical channels [10–12]. Furthermore, to avoid the prediction error propagation problem of the sequential prediction of future channels, the transformer model was used to predict future channels

in parallel in [13].

However, the existing channel prediction techniques [6–13] were designed for discrete channel prediction, while they fail to directly predict the channels in all time slots of each frame. To be more specific, as we discussed before, the channels can only be estimated in the first time slot of each frame through the transmission of SRS. Based on these discretely estimated historical channels, the future channels with the same time interval are predicted by existing channel prediction techniques. Then, the channels in other time slots between two adjacent SRS could be recovered by using interpolation methods. Unfortunately, there exists a serious interpolation loss for these discrete channel prediction techniques in mobile scenarios. One possible solution is continuous-time channel prediction for all time slots of each frame. Unfortunately, to the best of our knowledge, none of the existing methods can achieve continuous-time channel prediction.

## 1.2 Our Contributions

To fill in this gap, we propose a tensor neural ordinary differential equation (TN-ODE) based continuous-time channel prediction scheme in this paper. Specifically, inspired by the recently developed continuous-time signal processing technology named neural ODE in the field of machine learning [14], we adopt the neural ODE architecture proposed in [14] to model the continuous-time channel prediction problem. In the above architecture, a GRU-based encoder is used to preprocess the discretely sampled historical channels, then a neural ODE-based decoder is used to predict future channels in consecutive time slots. Furthermore, to improve the channel prediction accuracy and reduce the computational complexity of the neural ODE, we propose the TN-ODE to exploit the structural characteristics of channels in multiple domains by a series of low-dimensional learnable transforms. To be more specific, in the antenna domain, the channel model is described by different angles of arrival (AoAs) and angles of departure (AoDs), while in the frequency domain, the channel model is mainly determined by multiple times of arrival (ToAs). Thanks to these structural characteristics, the proposed TN-ODE allows us to decouple the complicated high-dimensional channel prediction into efficient low-dimensional channel predic-

tion in multiple domains. Simulation results show that the proposed TN-ODE based continuous-time channel prediction technique can effectively mitigate the interpolation loss and improve the channel prediction performance in all time slots of each frame.

## 1.3 Organization and Notation

The remainder of this paper is organized as follows. In Section II, the system model of the mmWave massive MIMO is introduced, and the continuous-time channel prediction problem in this system is then formulated. After that, we elaborate on the proposed TN-ODE based continuous-time channel prediction model in Section III. Section IV illustrates the simulation results. Finally, conclusions are drawn in Section V.

*Notation*: We denote the column vector $\mathbf{a}$ and matrix $\mathbf{A}$ by boldface lower-case and upper-case letters, respectively; $\mathbf{A}^T$, $\mathbf{A}^H$, and $\mathbf{A}^{-1}$ are the transpose, conjugate transpose, and inverse of the matrix $\mathbf{A}$, respectively; $\mathbf{A} \otimes \mathbf{B}$ is the Kronecker product of the matrix $\mathbf{A}$ and matrix $\mathbf{B}$; $\mathbf{A} \circ \mathbf{B}$ is the Hadamard product of $\mathbf{A}$ and $\mathbf{B}$; $\mathbf{I}_N$ denotes an $N \times N$ identity matrix. $\mathcal{CN}\left(\mu, \sigma^2\right)$ is the probability density function of the circularly symmetric complex Gaussian distribution with mean $\mu$ and variance $\sigma^2$. $\mathbb{E}\{\cdot\}$ denotes the statistical expectation. We use $\text{vec}(\mathbf{A})$ to denote the vectorization of matrix $\mathbf{A}$. $\sigma(x) = \frac{1}{1+e^{-x}}$ and $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ represent the Sigmoid function and hyperbolic tangent function, respectively. We denote $h[n], n \in \mathbb{Z}$ as a discrete-time sequence and $h(t), t \in \mathbb{R}$ as a continuous-time sequence.

## II. SYSTEM MODEL

In this section, we will first introduce the signal model of the mmWave massive MIMO system. Then, the continuous-time channel prediction is formulated to avoid the interpolation loss problem in existing discrete channel prediction schemes.

## 2.1 Signal Model

In this paper, we consider an uplink time division duplexing (TDD) based mmWave massive MIMO system with orthogonal frequency division multiplexing (OFDM). The base station (BS) equipped with an $N_{\text{T}}$-antenna uniform linear array (ULA) [15] serves an $N_{\text{R}}$-antenna user with $M$ subcarriers. To reduce energy consumption, hybrid precoding structure is employed in the BS [16], where the number of radio frequency (RF) chains is $N_{\text{RF}}$, as indicated in Figure 1. According to the 5G standard [3], the time resources for communication are divided into frames and each frame consists of $Q$ time slots. As shown in Figure 2, the $Q$ time slots could be further divided into three parts, i.e., uplink pilots, uplink data transmission, and downlink data transmission. For the $q$-th time slot, $N_q$ symbols are included and the channel remains unchanged during the $N_q$ symbols based on the block-fading assumption [17], where the channel remains time-invariant within each time slot and changes between different time slots.
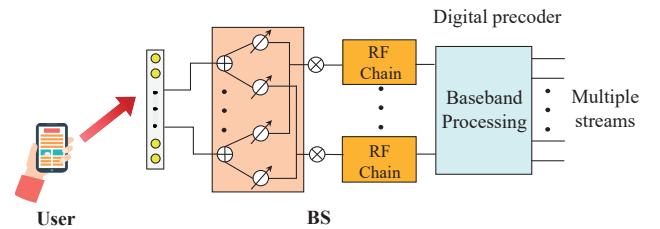


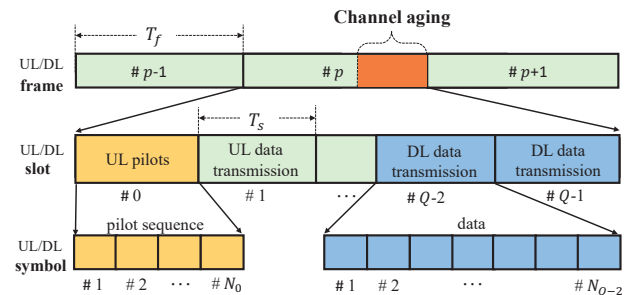**Figure 1.** *Hybrid precoding for mmWave massive MIMO.*



**Figure 2.** *The 5G frame structure defined by 3GPP [3].*

Let $\mathbf{H}_m(t) \in \mathbb{C}^{N_{\text{T}} \times N_{\text{R}}}$ denote the channel at the time $t$. Due to the limited number of scattering clusters in the mmWave propagation environment, we adopt the widely used geometric Saleh-Valenzuela multipath channel model [16] to characterize the mmWave channel. Under this model, $\mathbf{H}_m(t)$ can be denoted as

$$\mathbf{H}_m(t) = \sum_{l=1}^{L} \alpha_l e^{-j2\pi(v_l t + f_m \tau_l)} \mathbf{a}_{\text{T}}(\phi_{l,\text{T}}) \mathbf{a}_{\text{R}}^H(\phi_{l,\text{R}}),$$

$$(1)$$

where $L$ is the number of the paths, $\alpha_l$, $v_l$, $\tau_l$, $\phi_{l,\text{T}}$, and $\phi_{l,\text{R}}$ are the complex path gain, Doppler shift, ToA, AoA, and AoD of the $l$-th path, respectively. For $m \in \{1, 2, \cdots, M\}$, $f_m = f + \frac{B}{2}(m - \frac{M}{2})$ denotes the $m$-th subcarrier frequency, with $f$, $B$, and $M$ being the carrier frequency, bandwidth, and the number of subcarriers. Since the ULA is considered in this paper, the array steering vector $\mathbf{a}_{\text{T}}(\phi_{l,\text{T}})$ and $\mathbf{a}_{\text{R}}(\phi_{l,\text{R}})$ could be represented by [18]

$$\mathbf{a}_{\text{T}}(\phi_{l,\text{T}}) = \frac{1}{\sqrt{N_T}}[e^{-j\frac{2\pi}{\lambda}d\sin(\phi_{l,\text{T}})\mathbf{n}_{\text{T}}}], \quad (2)$$

$$\mathbf{a}_{\text{R}}(\phi_{l,\text{R}}) = \frac{1}{\sqrt{N_R}}[e^{-j\frac{2\pi}{\lambda}d\sin(\phi_{l,\text{R}})\mathbf{n}_{\text{R}}}], \quad (3)$$

where $\mathbf{n}_{\text{T}} = [0, 1, \cdots, N_{\text{T}} - 1]^T$ and $\mathbf{n}_{\text{R}} = [0, 1, \cdots, N_{\text{R}} - 1]^T$, $\lambda$ is the carrier wavelength, and $d$ is the antenna spacing usually set as $d = \lambda/2$.

We denote $T_f$ and $T_s$ as the duration time of one frame and one time slot, where $T_f = QT_s$. Accordingly, we can use $\mathbf{H}_m^{(p,q)} = \mathbf{H}_m(t_{p,q})$ to denote the channel at the $q$-th time slot of the $p$-th frame and the $m$-th subcarrier, where $t_{p,q} = pT_f + qT_s$. Then, the received signal $\mathbf{Y}_m^{(p,q)} \in \mathbb{C}^{N_{\text{RF}} \times N_q}$ at the $q$-th time slot of the $p$-th frame and the $m$-th subcarrier in BS can be expressed by

$$\begin{aligned}\mathbf{Y}_m^{(p,q)} &= \mathbf{A}^{(p,q)}\mathbf{H}_m^{(p,q)}\mathbf{S}_m^{(p,q)} + \mathbf{A}^{(p,q)}\mathbf{N}_m^{(p,q)} \\ &= \overline{\mathbf{H}}_m^{(p,q)}\mathbf{S}_m^{(p,q)} + \mathbf{A}^{(p,q)}\mathbf{N}_m^{(p,q)}, \quad (4)\end{aligned}$$

where $\mathbf{A}^{(p,q)} \in \mathbb{C}^{N_{\text{RF}} \times N_{\text{T}}}$ is the frequency-independent combining matrix, $\mathbf{S}_m^{(p,q)} \in \mathbb{C}^{N_{\text{R}} \times N_q}$ denotes the transmitted signal, $\mathbf{N}_m^{(p,q)} \in \mathbb{C}^{N_{\text{RF}} \times N_q}$ is the Gaussian noise and each element following the distribution $\mathcal{CN}(0, \sigma^2)$ with $\sigma^2$ being the noise power, and $\overline{\mathbf{H}}_m^{(p,q)} \in \mathbb{C}^{N_{\text{RF}} \times N_{\text{R}}}$ is the effective channel matrix in the $q$-th time slot of the $p$-th frame and the $m$-th subcarrier.

We utilize the discrete Fourier transmission (DFT) codebook to design the analog combining $\mathbf{A}^{(p,q)}$ [19]. In the DFT codebook, each codeword points to a specific azimuth AoA and all codewords will cover the entire beam space. By traversing all codewords, the strongest $N_{\text{RF}}$ codewords could be selected to construct $\mathbf{A}^{(p,q)}$. Benefiting from the fact that the time-varying channel is mainly caused by the Doppler effect, while the AoA and AoD are time-invariant in

several frames during tens of milliseconds [20], the optimal combining matrix stays unchanged in several frames. In this case, we suppose $\mathbf{A}^{(p,q)} = \mathbf{A}, \forall p \in \{0, 1, \cdots, P - 1\}, \forall q \in \{0, 1, \cdots, Q - 1\}$, where $P$ is the number of frames in the order of tens of milliseconds.

In particular, when $q = 0$, the effective channel $\overline{\mathbf{H}}_m^{(p,0)}$ of the first time slot in the $p$-th frame is estimated according to the predefined pilot sequence $\mathbf{S}_m^{(p,0)}$ and received signal $\mathbf{Y}_m^{(p,0)}$. Generally, we use the least square (LS) channel estimation method to recover the effective channel, which could be represented by

$$\text{vec}(\hat{\mathbf{H}}_m^{(p,0)}) = (\mathbf{S}_m^{(p,0)^T} \otimes \mathbf{I}_{N_{\text{RF}}})^{-1}\text{vec}(\mathbf{Y}_m^{(p,0)}), \quad (5)$$

where $\text{vec}(\hat{\mathbf{H}}_m^{(p,0)})$ is the vectorization of LS channel estimation $\hat{\mathbf{H}}_m^{(p,0)}$. When $q \neq 0$, the $\mathbf{S}_m^{(p,q)}$ is the transmitted signal and the achievable average rate $R$ could be written as

$$R = \quad (6)$$

$$\frac{1}{M}\sum_{m=1}^{M}\log_2\left|\mathbf{I}_{N_{\text{R}}} + \frac{1}{N_{\text{R}}\sigma^2}\mathbf{D}_m^{(p,q)}\overline{\mathbf{H}}_m^{(p,q)}\overline{\mathbf{H}}_m^{(p,q)^H}\mathbf{D}_m^{(p,q)^H}\right|.$$

We utilize the classical zero-forcing method [2] to design the digital precoding $\mathbf{D}_m^{(p,q)} \in \mathbb{C}^{N_{\text{R}} \times N_{\text{RF}}}$ in the $q$-th time slot of the $p$-th frame and the $m$-th subcarrier as:

$$\mathbf{D}_m^{(p,q)} = (\hat{\mathbf{H}}_m^{(p,q)^H}\hat{\mathbf{H}}_m^{(p,q)})^{-1}\hat{\mathbf{H}}_m^{(p,q)^H}. \quad (7)$$

## 2.2 Problem Formulation

To calculate the digital precoding $\mathbf{D}_m^{(p,q)}$, the estimated instantaneous $\hat{\mathbf{H}}_m^{(p,q)}$ is required according to (7). Whereas, since only the CSI at the first time slot of each frame $\hat{\mathbf{H}}_m^{(p,0)}$ is available, we usually use the $\hat{\mathbf{H}}_m^{(p,0)}$ to perform precoding for the subsequent time slots, i.e., $\mathbf{D}_m^{(p,q)} = \mathbf{D}_m^{(p,0)}$. Unfortunately, due to the channel aging issue induced by mobility, the outdated CSI $\hat{\mathbf{H}}_m^{(p,0)}$ has a significant change compared with the actual effective channel $\overline{\mathbf{H}}_m^{(p,q)}$, which results in a sever performance loss for mmWave MIMO in mobile scenarios.

To mitigate the performance loss caused by chan-

nel aging, some channel prediction techniques [6–13] have been proposed to deal with the channel aging issue by exploring the temporal correlation of the time-varying channel. Specifically, the existing channel prediction schemes could predict future channels in discrete frames, i.e., $\hat{\mathbf{H}}_m^{(p+1,0)}, \cdots, \hat{\mathbf{H}}_m^{(p+K,0)}$, based on the historical channels $\hat{\mathbf{H}}_m^{(p-J,0)}, \cdots, \hat{\mathbf{H}}_m^{(p,0)}$ with the same time interval. Since these channel prediction methods are designed for discrete channel prediction, which only predict the channel in the first time slot of each frame, they can not realize the direct prediction of the channels for all time slots in future frames. Thus, the interpolation method has to be utilized to recover the channels $\hat{\mathbf{H}}_m^{(p+k,q)}$ with $q > 0$ as

$$\hat{\mathbf{H}}_m^{(p+k,q)} = (1 - \frac{q}{Q})\hat{\mathbf{H}}_m^{(p+k,0)} + \frac{q}{Q}\hat{\mathbf{H}}_m^{(p+k+1,0)}, \quad (8)$$

where $k = 0, 1, \cdots, K-1$. However, due to the complicated change of the channel, simple interpolation is difficult to describe the actual change of the channel. Therefore, there is an interpolation loss for the existing discrete channel prediction schemes.

Unlike the existing discrete channel prediction schemes, we reformulate the channel prediction problem as a continuous-time channel mapping problem to avoid interpolation loss. Specifically, we utilize the historical discrete channels from the past $J$ frames to predict the future continuous-time channels in the next $K$ frames, which could be formulated as

$$\min_{\boldsymbol{\theta}} \sum_{k=0}^{K-1} \sum_{q=0}^{Q-1} \sum_{m=1}^{M} \mathbb{E}\left\{ \frac{\|\overline{\mathbf{H}}_m^{(p+k,q)} - \hat{\mathbf{H}}_m^{(p+k,q)}\|^2}{\|\overline{\mathbf{H}}_m^{(p+k,q)}\|^2} \right\}, \tag{9a}$$

$$\text{s.t. } (\hat{\mathbf{H}}_m^{(p,1)}, \hat{\mathbf{H}}_m^{(p,2)}, \cdots, \hat{\mathbf{H}}_m^{(p+K-1,Q-1)})$$
$$= f(\hat{\mathbf{H}}_m^{(p-J,0)}, \cdots, \hat{\mathbf{H}}_m^{(p,0)}; \boldsymbol{\theta}), \tag{9b}$$

where $f(\cdot)$ is the proposed continuous-time channel prediction model and $\boldsymbol{\theta}$ is the parameters of the model. Since the normalized mean square error (NMSE) is not affected by the amplitude of the channel, we adopt the NMSE as the minimization target to realize stable convergence. It is worth noting that the estimated historical channels are discretely sampled at the first time slot of each frame. Correspondingly, the predicted channels are continuously distributed at any time slot of each future frame. By contrast, the existing discrete

channel prediction schemes only predict the channel at the first time slot of the future frames. Thus, the proposed continuous-time channel prediction scheme realizes the direct prediction of the future channel in any time slot so that the interpolation loss can be avoided.

## III. PROPOSED METHOD

In this section, we first introduce the background of neural ODE and elaborate on the framework of neural ODE based channel prediction. Then, we propose the TN-ODE to explore the mmWave channel structure to improve the channel prediction performance.

### 3.1 Background of Neural ODE

To achieve continuous-time channel prediction, it is crucial to find an appropriate technique to process continuous-time signals. Recently, with the rapid advancement in the field of dynamical systems, neural ODE becomes an attractive technology for modeling continuous-time sequences [14, 21, 22]. Neural ODEs use first-order differential equations to fit the hidden state of time sequences, so it is capable of handling continuous-time signals. To make this paper self-contained, we provide a brief background of neural ODE. Specifically, classical RNN-like architectures, including RNN, GRU, and LSTM, build complicated networks to encode time sequences into a series of hidden states:

$$\mathbf{h}[n] = \mathbf{h}[n-1] + g(\mathbf{h}[n-1], \boldsymbol{\theta}). \tag{10}$$

Here, $\mathbf{h}[n]$ represents the hidden state at the $n$-th discrete time, $g(\cdot)$ denotes the state transition function realized by neural networks, and $\boldsymbol{\theta}$ is the network parameters. The transition in (10) is built on a discrete difference equation, which is awkward to deal with signals not belonging to discrete time samples. On the contrary, neural ODEs define a continuous-time hidden state $\mathbf{h}(t)$, which can be formulated as a time-invariant differential equation:

$$\frac{\mathrm{d}\mathbf{h}(t)}{\mathrm{d}t} = f(\mathbf{h}(t), \boldsymbol{\theta}). \tag{11}$$

Besides, (11) is equivalent to the following integral form:

$$\mathbf{h}(t) = \int_{t_0}^{t} f(\mathbf{h}(\tau), \boldsymbol{\theta}) \mathrm{d}\tau + \mathbf{h}(t_0). \qquad (12)$$

Here, $\mathbf{h}(t_0)$ is the initial hidden state, and function $f(\mathbf{h}(t), \boldsymbol{\theta})$ describes the dynamic of hidden state $\mathbf{h}(t)$. One can acquire the hidden state $\mathbf{h}(t)$ at an arbitrary time $t$ by solving problem (12) through an ODE solver:

$$\mathbf{h}(t) = \mathrm{ODESolver}(f(\cdot, \boldsymbol{\theta}), \mathbf{h}(t_0), t_0, t). \qquad (13)$$

As indicated in [21], such an ODE solver can be implemented by various numerical schemes, including the forward and backward Euler methods, the Runge-Kutta method, and the linear multi-step method. As a consequence, applying neural ODE models (12) and solvers (13) allows us to deal with continuous-time sequences, so as to achieve continuous-time channel prediction.

## 3.2 Framework of Neural ODE Based Channel Prediction

Based on the above background, the framework of neural ODE-based channel prediction is presented in this subsection. Our aim is to predict the channels for all time slots of the future $K$ frames by processing those historical $J$ channels. The Latent ODE architecture introduced in [22] is adopted to model this process. For expression clarity, we denote $\hat{\mathbf{H}}[n] = [\mathrm{vec}(\hat{\mathbf{H}}_1^{(n,0)}), \mathrm{vec}(\hat{\mathbf{H}}_2^{(n,0)}), \cdots, \mathrm{vec}(\hat{\mathbf{H}}_M^{(n,0)})]$ and $\hat{\mathbf{H}}(t) = [\mathrm{vec}(\hat{\mathbf{H}}_m(t)), \mathrm{vec}(\hat{\mathbf{H}}_2(t)), \cdots, \mathrm{vec}(\hat{\mathbf{H}}_M(t))]$.

As shown in Figure 3, the neural ODE-based channel prediction is composed of two modules, i.e., an encoder and a decoder [22]. Generally speaking, the encoder is responsible for extracting features from the historical channels $\hat{\mathbf{H}}[n]$ for $n = \{0, -1, \cdots, -J + 1\}$. The output of the encoder serves as the initial state of the decoder. Correspondingly, the decoder exploits a neural ODE to infer future continuous-time channels $\hat{\mathbf{H}}(t)$ for $t > 0$.

Specifically, the encoder's role is to extract the features from historical channels. Since the SRS signals are transmitted and received with equally-sized time interval $T_f$, RNN-like architectures are enough to deal

with these sequences. We denote the hidden state of $\hat{\mathbf{H}}[n]$ as $\mathbf{R}[n]$. Then, based on the Markov property of RNN models, the map from $\mathbf{R}[n-1]$ to $\mathbf{R}[n]$ can be written as

$$\mathbf{R}[n] = \mathrm{EncoderCell}(\mathbf{R}[n-1], \hat{\mathbf{H}}[n], \boldsymbol{\theta}_E), \qquad (14)$$

where EncoderCell$(\cdot)$ is the transition function of the RNN-like network with $\boldsymbol{\theta}_E$ being the learnable parameters.

For the decoder, a neural ODE model is deployed to specific the dynamic of the future channel's hidden state. This hidden state is defined as $\mathbf{O}(t)$. Besides, the final output $\mathbf{R}[0]$ of the encoder is regarded as the initial state $\mathbf{O}(0)$ of decoder. Therefore, for any times $t > 0$, the hidden state $\mathbf{O}(t)$ can be presented as

$$\frac{\mathrm{d}\mathbf{O}(t)}{\mathrm{d}t} = \mathrm{DecoderCell}(\mathbf{O}(t), \boldsymbol{\theta}_D), \qquad (15)$$

where DecoderCell$(\cdot)$ denotes the transition function of the neural ODE network with $\boldsymbol{\theta}_D$ being its learnable parameters. Note that (15) can be solved by the ODESolver as presented in (13). After that, one layer neural network Pred$(\cdot)$ is built to output the predicted channel $\hat{\mathbf{H}}(t)$ from the hidden state $\mathbf{O}(t)$:

$$\hat{\mathbf{H}}(t) = \mathrm{Pred}(\mathbf{O}(t), \boldsymbol{\theta}_P), \qquad (16)$$

where $\boldsymbol{\theta}_P$ denotes its parameters. All in all, following this neural ODE framework, we are capable of extracting features from the previous channels and then predicting future continuous-time channels for any $t$.

## 3.3 TN-ODE Based Channel Prediction

In this subsection, we elaborate on the idea of tensor neural ODE for designing the three crucial transition functions: EncoderCell$(\cdot)$, DecoderCell$(\cdot)$, and Pred$(\cdot)$.

We commence our discussion by briefly introducing the transition functions widely used in classical neural ODE framework [22]. The authors in [22] deployed a GRU model as its encoder transition function and modified the GRU model to act as the decoder transition function. To fit in our channel prediction framework, the inputs, hidden states, and outputs should be first vectorized as the following col-
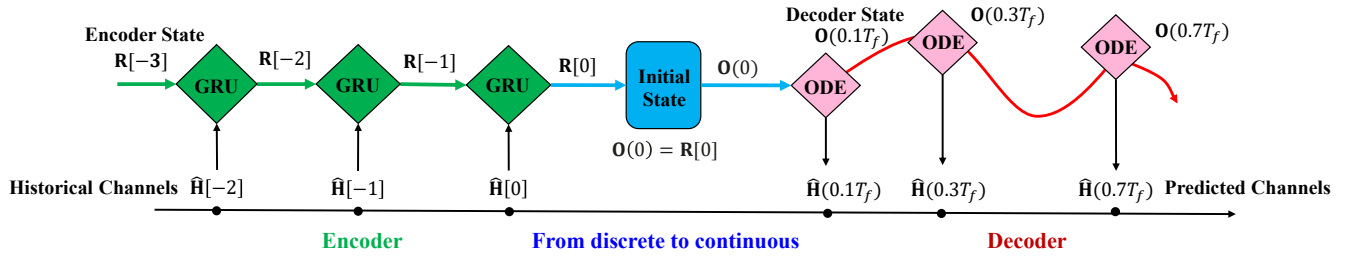
**Figure 3.** *The framework of neural ODE-based channel prediction.*

umn vectors: $\hat{\mathbf{h}}[n] = \text{vec}(\hat{\mathbf{H}}[n])$, $\mathbf{r}[n] = \text{vec}(\mathbf{R}[n])$, $\mathbf{o}(t) = \text{vec}(\mathbf{O}(t))$, and $\hat{\mathbf{h}}(t) = \text{vec}(\hat{\mathbf{H}}(t))$. Then, according to the GRU architecture [22], EncoderCell($\cdot$) consists of the following modules:

$$\mathbf{z} = \sigma\left(\mathbf{U}^z\hat{\mathbf{h}}[n] + \mathbf{W}^z\mathbf{r}[n-1]\right), \tag{17a}$$

$$\mathbf{x} = \sigma\left(\mathbf{U}^x\hat{\mathbf{h}}[n] + \mathbf{W}^x\mathbf{r}[n-1]\right), \tag{17b}$$

$$\mathbf{u} = \tanh\left(\mathbf{U}^u\hat{\mathbf{h}}[n] + \mathbf{W}^u(\mathbf{r}[n-1] \circ \mathbf{x})\right), \tag{17c}$$

$$\mathbf{r}[n] = (\mathbf{1} - \mathbf{z}) \circ \mathbf{u} + \mathbf{z} \circ \mathbf{r}[n-1], \tag{17d}$$

where matrices $\{\mathbf{U}, \mathbf{W}\}$ are the learnable parameters. As for the decoder, it is different from the encoder which can receive external stimulus $\hat{\mathbf{h}}[n]$ to update its states. The hidden state transition of DecoderCell($\cdot$) is an auto-regressive process without external stimulus. Thereby, to fit GRU model in this decoder, we can carry out the steps below to modify GRU: remove $\hat{\mathbf{h}}[n]$ from (17a)-(17d); replace $\mathbf{r}[n-1]$ and $\mathbf{r}[n]$ with $\mathbf{o}(t)$ and $\frac{d\mathbf{o}(t)}{dt}$, respectively. Finally, function Pred($\cdot$) can be realized by a fully connected layer, i.e. $\hat{\mathbf{h}}(t) = \mathbf{W}^h\mathbf{o}(t)$. As a result, the entire neural ODE-based channel prediction is successfully established based on the classical GRU model.

There is no denying that the above transition functions have the ability to process continuous-time sequences. However, they will suffer from two serious problems when applied to channel prediction. First, these transition functions fail to exploit the underlying channel structure. As shown in (1), mmWave channels exhibit obvious correlations in multiple domains. For example, the antenna-domain channel is constructed by the superposition of multiple array steering vectors with different AoAs and AoDs. Besides, in the frequency domain, the channel structure can be captured by several ToAs. However, simply vectorizing

channels to fit in the GRU model will undermine such regular structures. Second, the computational complexity of these transition functions is also unaffordable. Take the function Pred($\cdot$) as an example, we suppose $\mathbf{W}^h$ is a square matrix. As the dimension of $\hat{\mathbf{h}}(t)$ is $N_{\text{RF}}N_{\text{R}}M \times 1$, then matrix $\mathbf{W}^h$ will contain $2N_{\text{RF}}^2N_{\text{R}}^2M^2$ floating points. If $N_{\text{RF}} = N_{\text{R}} = 4$ and $M = 256$, then the number of floating points of $\mathbf{W}^h$ is $2N_{\text{RF}}^2N_{\text{R}}^2M^2 = 33,554,432$, which costs unacceptable computational resources.

To address these two critical problems, we propose the TN-ODE by exploiting the channel correlation. Our scheme is inspired by the tensor decomposition based signal processing algorithms [23], which extract the information of channels from different domains and process them separately. In our model, we preserve the matrix form of $\hat{\mathbf{H}}[n]$, $\mathbf{R}[n]$, $\mathbf{O}(t)$, and $\hat{\mathbf{H}}(t)$, and use different learnable transforms to independently extract the antenna-domain and frequency-domain information from historical channels. We take the matrix product $\mathbf{U}^z\hat{\mathbf{h}}[n]$ in (17a) as an example. The classical GRU model (17a) vectorizes $\hat{\mathbf{H}}[n]$ as $\hat{\mathbf{h}}[n] \in \mathbb{C}^{N_{\text{RF}}N_{\text{R}}M \times 1}$ and uses a high-dimensional matrix $\mathbf{U}^z$ to process $\hat{\mathbf{h}}[n]$. Instead, we keep the shape of $\hat{\mathbf{H}}[n]$ as $N_{\text{RF}}N_{\text{R}} \times M$ and use two independent low-dimensional matrices $\mathbf{U}_l^z$ and $\mathbf{U}_r^z$ to separately work on the antenna domain and frequency domain of $\hat{\mathbf{H}}[n]$, which gives rise to $\mathbf{U}_l^z\hat{\mathbf{H}}[n]\mathbf{U}_r^z$. Similarly, we modify all modules in (17a)-(17d) by the same means to construct the tensor-inspired EncoderCell($\cdot$) as

$$\mathbf{Z} = \sigma\left(\mathbf{U}_l^z\hat{\mathbf{H}}[n]\mathbf{U}_r^z + \mathbf{W}_l^z\mathbf{R}[n-1]\mathbf{W}_r^z\right), \tag{18a}$$

$$\mathbf{X} = \sigma\left(\mathbf{U}_l^x\hat{\mathbf{H}}[n]\mathbf{U}_r^x + \mathbf{W}_l^x\mathbf{R}[n-1]\mathbf{W}_r^x\right), \tag{18b}$$

$$\mathbf{U} = \tanh\left(\mathbf{U}_l^u\hat{\mathbf{H}}[n]\mathbf{U}_r^u + \mathbf{W}_l^u(\mathbf{R}[n-1] \circ \mathbf{X})\mathbf{W}_r^u\right), \tag{18c}$$

$$\mathbf{R}[n] = (\mathbf{1} - \mathbf{Z}) \circ \mathbf{U} + \mathbf{Z} \circ \mathbf{R}[n-1], \qquad (18d)$$

where matrices $\{\mathbf{U}_l, \mathbf{U}_r, \mathbf{W}_l, \mathbf{W}_r\}$ are all learnable parameters. Here, matrices $\{\mathbf{U}_l^z, \mathbf{U}_l^x, \mathbf{U}_l^u\}$ have a size of $F_l \times N_{\mathrm{RF}} N_{\mathrm{R}}$, matrices $\{\mathbf{U}_r^z, \mathbf{U}_r^x, \mathbf{U}_r^u\}$ have a size of $M \times F_r$, matrices $\{\mathbf{W}_l^z, \mathbf{W}_l^x, \mathbf{W}_l^u\}$ have a size of $F_l \times F_l$, and matrices $\{\mathbf{W}_r^z, \mathbf{W}_r^x, \mathbf{W}_r^u\}$ have a size of $F_r \times F_r$. $F_l$ and $F_r$ denote the feature dimensions. Notice that the computations in (18a)-(18d) are all complex-valued multiplications, which are realized by the complex-valued neural network (CVNN) proposed in [8]. The Sigmoid and Tanh functions work on the real and imaginary parts respectively. Moreover, we can use the same way to transform the classical DecoderCell($\cdot$) and Pred($\cdot$) to their tensor forms. To be specific, the transition function of the ODE decoder can be written as

$$\mathbf{Z}(t) = \sigma\left(\mathbf{V}_l^z \mathbf{O}(t) \mathbf{V}_r^z\right), \qquad (19a)$$

$$\mathbf{X}(t) = \sigma\left(\mathbf{V}_l^x \mathbf{O}(t) \mathbf{V}_r^x\right), \qquad (19b)$$

$$\mathbf{U}(t) = \tanh\left(\mathbf{V}_l^u (\mathbf{O}(t) \circ \mathbf{X}(t)) \mathbf{V}_r^u\right), \qquad (19c)$$

$$\frac{\mathrm{d}\mathbf{O}(t)}{\mathrm{d}t} = (\mathbf{1} - \mathbf{Z}(t)) \circ \mathbf{U}(t) + \mathbf{Z}(t) \circ \mathbf{O}(t), \quad (19d)$$

where the size of matrices $\{\mathbf{V}_l^z, \mathbf{V}_l^x, \mathbf{V}_l^u\}$ is $F_l \times F_l$ and the size of matrices $\{\mathbf{V}_r^z, \mathbf{V}_r^x, \mathbf{V}_r^u\}$ is $F_r \times F_r$. Then, the function of Pred($\cdot$) can be given by

$$\hat{\mathbf{H}}(t) = \mathbf{W}_l^h \mathbf{O}(t) \mathbf{W}_r^h, \qquad (20)$$

with $\mathbf{W}_l^h \in \mathbb{C}^{N_{\mathrm{RF}} N_{\mathrm{R}} \times F_l}$ and $\mathbf{W}_r^h \in \mathbb{C}^{F_r \times M}$.

Our proposed TN-ODE enjoys two crucial merits compared to the classical one [22]. To begin with, it preserves the structural features of multi-domain channels in the entire procedure, so our scheme is specific for predicting wireless continuous-time channels. Moreover, its computational complexity is much lower than that of [22]. We still take the function Pred($\cdot$) as an example. As shown in (20), we suppose both $\mathbf{W}_l^h$ and $\mathbf{W}_r^h$ are square matrices. Since the shape of $\hat{\mathbf{H}}(t)$ is $N_{\mathrm{RF}} N_{\mathrm{R}} \times M$, matrices $\mathbf{W}_l^h$ and $\mathbf{W}_r^h$ have sizes of $N_{\mathrm{RF}} N_{\mathrm{R}} \times N_{\mathrm{RF}} N_{\mathrm{R}}$ and $M \times M$, which gives rise to $2(N_{\mathrm{RF}}^2 N_{\mathrm{R}}^2 + M^2)$ floating points. Therefore, if $N_{\mathrm{RF}} = N_{\mathrm{R}} = 4$ and $M = 256$, the number of floating points is decreasing from $33,554,432$ in $\mathbf{W}^h$ to $131,574$ in $\mathbf{W}_l^h$ and $\mathbf{W}_r^h$. The computational complexity is significantly improved.

As a consequence, our proposed TD-ODE takes advantage of the continuous-time signal processing capability of ODE and the multi-domain structure of mmWave channels, so it is promising to achieve efficient continuous-time channel prediction, which will be demonstrated in the simulation section.

## 3.4 Training and Testing Details

In this subsection, we supplement some training and testing details. To begin with, we adopt an offline training and online testing strategy. In the offline training stage, we use the clustered delay line (CDL) channel model to randomly generate $N_{\mathrm{train}}$ time-varying channel samples. We divide these samples into $\frac{N_{\mathrm{train}}}{BS}$ batches, with $BS$ being the batch size. We consider the $b$-th batch. Each sample of this batch is a time-varying channel sequence, which is divided into two periods. The first period corresponds to the historical channels. In this period, we sample $J$ time slots with an equal time interval of $T_f$. The corresponding historical channels are $\mathbf{H}^{\mathrm{input}} = \{\hat{\mathbf{H}}[-J + 1], \cdots, \hat{\mathbf{H}}[-1], \hat{\mathbf{H}}[0]\}$. The second period is regarded as the future time, where $P$ time slots are randomly sampled from the time duration $[0, KT_f]$. We use $t_1^b \leq t_2^b \leq \cdots \leq t_P^b$ to index these sampled times in the $b$-th batch. Therefore, the corresponding noise-free channels are $\{\overline{\mathbf{H}}(t_1^b), \overline{\mathbf{H}}(t_2^b), \cdots, \overline{\mathbf{H}}(t_P^b)\}$, which are working as the training labels. Then, we use the proposed TN-ODE model to process $\mathbf{Y}^{\mathrm{input}}$ and predict $\{\hat{\mathbf{H}}(t_1^b), \hat{\mathbf{H}}(t_2^b), \cdots, \hat{\mathbf{H}}(t_P^b)\}$. Finally, the NMSE is used for the loss function:

$$\mathrm{Loss} = \frac{1}{P} \sum_{i=1}^{P} \mathbb{E}\left\{ \frac{\|\hat{\mathbf{H}}(t_i^b) - \overline{\mathbf{H}}(t_i^b)\|^2}{\|\overline{\mathbf{H}}(t_i^b)\|^2} \right\}. \qquad (21)$$

Based on this loss function, the Adam optimizer is adopted to update the network parameters using their gradients. Notice that adjoint sensitivities proposed in [14] are used to efficiently compute the ODE's gradients. The above procedure is carried out batch by batch until convergence.

The data size in the testing stage is $N_{\mathrm{test}}$, where each channel sample is still divided into two periods. The first period is the same as that in the training stage. Regarding the second period, our target is to predict channels for future $KQ$ time slots (or $K$ frames). Therefore, we sample $KQ$ slots with an equal

time interval of $T_s$, which are denoted by $t_i = iT_s$, $i = 1, \cdots, KQ$. Then, we use the well-trained TN-ODE model to predict $\hat{\mathbf{H}}(t_i)$, $i = 1, \cdots, KQ$. Finally, these predicted channels are used for precoding.

## 3.5 Computational Complexity

In this subsection, we provide a detailed computational complexity analysis of the proposed scheme in the testing stage. Here, we mainly count the number of complex-valued multiplications.

For a sequence of historical channels $\{\hat{\mathbf{H}}[-J + 1], \cdots, \hat{\mathbf{H}}[-1], \hat{\mathbf{H}}[0]\}$, the total $J$ channels are processed by the EncoderCell (18a)-(18d) sequentially. Steps (18a)-(18c) have a complexity in the order of $\mathcal{O}(F_l N_{\text{RF}} N_{\text{R}} M + F_l M F_r)$, and step (18d) has a complexity of $\mathcal{O}(F_l F_r)$. Therefore, taking into account the $J$ channels, the computational complexity of the encoder is $\mathcal{O}(J F_l N_{\text{RF}} N_{\text{R}} M + J F_l M F_r) + \mathcal{O}(J F_l F_r) = \mathcal{O}(J F_l N_{\text{RF}} N_{\text{R}} M + J F_l M F_r)$.

As for the decoder, we can similarly derive that the computational complexities of calculating the functions DecoderCell$(\cdot)$ and Pred$(\cdot)$ are $\mathcal{O}(F_l^2 F_r + F_l F_r^2)$ and $\mathcal{O}(N_{\text{RF}} N_{\text{R}} F_l F_r + N_{\text{RF}} N_{\text{R}} F_r M)$, respectively. Moreover, the ODESolver$(\cdot)$ in (13) needs to calculate the DecoderCell$(\cdot)$ for $G$ times, where $G$ is proportional to $KQ$. Therefore, the computational complexity of the decoder is $\mathcal{O}(G F_l^2 F_r + G F_l F_r^2)$. Finally, as $KQ$ future channels are predicted, the overall number of complex-valued multiplications of the function Pred$(\cdot)$ is $\mathcal{O}(KQ N_{\text{RF}} N_{\text{R}} F_l F_r + KQ N_{\text{RF}} N_{\text{R}} F_r M)$.

As a consequence, the computional complexity of the proposed TN-ODE model is

$$\mathcal{O}(J F_l M (N_{\text{RF}} N_{\text{R}} + F_r)) + \mathcal{O}(G F_l F_r (F_l + F_r))$$
$$+ \mathcal{O}(KQ N_{\text{RF}} N_{\text{R}} F_r (F_l + M)). \quad (22)$$

## IV. SIMULATION RESULTS

In this section, simulation results are provided to demonstrate the superiority of our scheme. The CDL-B channel model in the Matlab 5G toolbox [13] is utilized to generate the data set. For each channel sample, the velocity of user is randomly generated from the uniform distribution $\mathcal{U}(30 \text{ km/h}, 60 \text{ km/h})$ and the delay spread is randomly chosen from the uniform distribution $\mathcal{U}(50 \text{ ns}, 200 \text{ ns})$. The simulation configura-

tions are presented in Table 1. The compared benchmarks are as follows: 1) the perfect CSI; 2) the classical AI-based algorithms, including the GRU-based channel prediction [11] and the FC network based algorithm [9]; 3) the classical model-based techniques, including the prony-based angular-delay domain channel prediction (PAD) [6] and ST-AR [8] algorithms; 4) utilizing the outdated channels without prediction.

**Table 1.** *Simulation configurations.*

| Parameter | Value | Parameter | Value |
|-----------|-------|-----------|-------|
| $N_{\text{T}}$ | 128 | $N_{\text{RF}}$ | 4 |
| $N_{\text{R}}$ | 4 | $f$ | 28 GHz |
| $B$ | 100 MHz | $M$ | 256 |
| SNR | 10 dB | $N_{\text{train}}$ | 1000 |
| $N_{\text{test}}$ | 200 | $BS$ | 32 |
| $T_f$ | 0.625 ms | $T_s$ | 0.125 ms |
| $J$ | 10 | $P$ | 5 |
| $K$ | 2 | $Q$ | 5 |
| $F_l$ | 64 | $F_r$ | 128 |

In Figure 4, the average rate performance is evaluated. We follow the 5G standard and set $T_f$ as 0.625 ms and $T_s$ as 0.125 ms. Therefore, the classical GRU, FC, ST-AR, and PAD algorithms predict the channels at the 5-th and 10-th time slots, and then recover the channels at other time slots through linear interpolation. It is clear from Figure 4 that the average rate performance of classical algorithms degrades at the interpolated channels. Fortunately, our proposed scheme is able to avoid interpolation loss by predicting the future channels at all time slots with the assistance of TN-ODE. Additionally, the proposed TN-ODE exploits the multi-domain channel structure, so it can even achieve higher average rate than classical algorithms at the 5-th and 10-th time slots.

In Figure 5, the real part of the true future channels and the predicted channels for an arbitrary antenna index and subcarrier are presented. We can observe from this figure that the existing discrete-time channel prediction techniques can only accurately predict the future channels at SRS positions, while the interpolated channels considerably deviate from the true channels. On the contrary, the proposed TN-ODE scheme well captures the dynamic of continuous-time channels. The simulation result in Figure 6 further supports our discussion, where the NMSE performance against
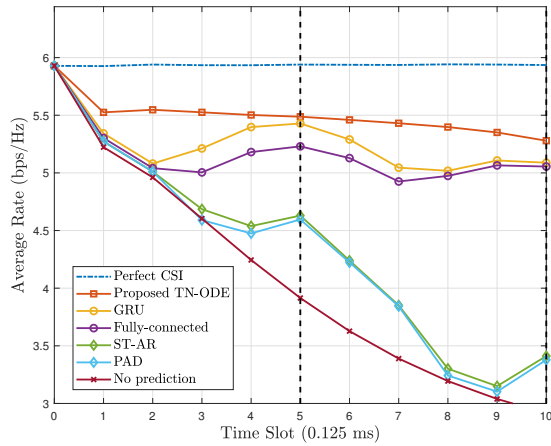
**Figure 4.** *Average rate performance against time slots.*



**Figure 6.** *NMSE performance against time slots.*

time slots is illustrated. It is obvious that the achieved NMSE of classical algorithms intensively fluctuates with respect to time slots, which is induced by the error of interpolation. On the contrary, the NMSE performance of our scheme smoothly deteriorates over time, and it is always lower than -10 dB. As a result, we can conclude that our TN-ODE based approach accomplishes accurate continuous-time channel prediction.



**Figure 5.** *The real part of the true future channels and the predicted channels.*

## V. CONCLUSION

In this paper, we have investigated the essential problem of continuous-time channel prediction in mobile mmWave massive MIMO systems. At first, we adopted the neural ODE to model the temporal corre-
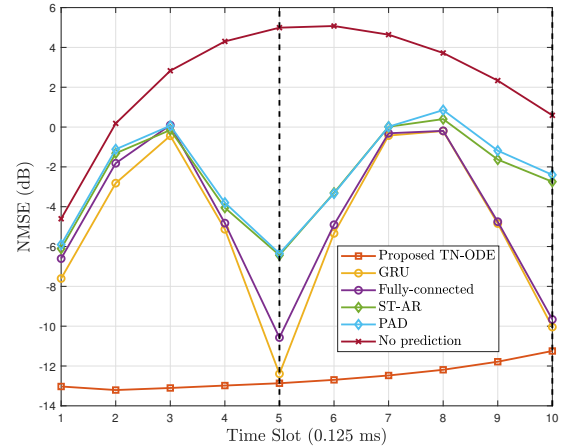
lation of mmWave channels, and then we introduced the neural ODE based channel prediction framework. This framework deployed a GRU-based encoder to extract features from historical channels and used a neural ODE based decoder to predict future continuous-time channels. After that, a TN-ODE model was proposed to improve this framework, which makes full use of the multi-domain channel structure. Simulations demonstrated that our scheme accomplished accurate channel prediction in all time slots of several future frames. The proposed TN-ODE model can be potentially extended to various continuous-time channel prediction scenarios, such as cell-free communication scenarios and RIS-aided communication scenarios. In the future, we will investigate the multi-user continuous-time channel prediction.

## References

[1] T. S. Rappaport, S. Sun, *et al.*, "Millimeter wave mobile communications for 5g cellular: It will work!" *IEEE Access*, 2013, vol. 1, pp. 335–349.

[2] T. L. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, 2010, vol. 9, no. 11, pp. 3590–3600.

[3] 3GPP TS 38.331 v15.6.0, "Radio resource control (rcc) protocol specification," 2019.

[4] K. T. Truong and R. W. Heath, "Effects of channel aging in massive mimo systems," *J. Commun. Netw.*, 2013, vol. 15, no. 4, pp. 338–351.

[5] J. Hu, Y. Hou, *et al.*, "Multi-sinusoidal waveform shaping for integrated data and energy transfer in aging channels," *China Communications*, 2023, vol. 20, no. 4, pp. 243–258.

[6] H. Yin, H. Wang, *et al.*, "Addressing the curse of mobility in massive mimo with prony-based angular-delay domain channel predictions," *IEEE J. Sel. Areas Commun.*, 2020, vol. 38, no. 12, pp. 2903–2917.

[7] K. Baddour and N. Beaulieu, "Autoregressive modeling for fading channel simulation," *IEEE Trans. Wireless Commun.*, 2005, vol. 4, no. 4, pp. 1650–1662.

[8] C. Wu, X. Yi, *et al.*, "Channel prediction in high-mobility massive mimo: From spatio-temporal autoregression to deep learning," *IEEE J. Sel. Areas Commun.*, 2021, vol. 39, no. 7, pp. 1915–1930.

[9] H. Kim, S. Kim, *et al.*, "Massive mimo channel prediction: Kalman filtering vs. machine learning," *IEEE Trans. Commun.*, 2021, vol. 69, no. 1, pp. 518–528.

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

[11] W. Jiang and H. D. Schotten, "Deep learning for fading channel prediction," *IEEE Open J. Commun. Soc.*, 2020, vol. 1, pp. 320–332.

[12] C. Liu, X. Liu, *et al.*, "Deep learning-empowered predictive beamforming for irs-assisted multi-user communications," in *Proc. IEEE Global Communications Conference (GLOBECOM'21)*, 2021, pp. 1–7.

[13] H. Jiang, M. Cui, *et al.*, "Accurate channel prediction based on transformer: Making mobility negligible," *IEEE J. Sel. Areas Commun.*, 2022, vol. 40, no. 9, pp. 2717–2732.

[14] R. T. Q. Chen, Y. Rubanova, *et al.*, "Neural ordinary differential equations," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2018, pp. 6571–6583.

[15] G. Interdonato, E. Bjrnson, *et al.*, "Ubiquitous cell-free massive mimo communications," *EURASIP Journal on Wireless Communications and Networking*, 2019, vol. 40, no. 1, pp. 1–13.

[16] O. E. Ayach, S. Rajagopal, *et al.*, "Spatially sparse precoding in millimeter wave mimo systems," *IEEE Trans. Wireless Commun.*, 2014, vol. 13, no. 3, pp. 1499–1513.

[17] Z. Zhou, J. Fang, *et al.*, "Channel estimation for millimeter-wave multiuser mimo systems via parafac decomposition," *IEEE Trans. Wireless Commun.*, 2016, vol. 15, no. 11, pp. 7501–7516.

[18] H. Mi, B. Ai, *et al.*, "Multi-scenario millimeter wave wireless channel measurements and sparsity analysis," *China Communications*, 2022, vol. 19, no. 11, pp. 16–31.

[19] J. Brady, N. Behdad, *et al.*, "Beamspace mimo for millimeter-wave communications: System architecture, modeling, analysis, and measurements," *IEEE Trans. Antennas Propag.*, 2013, vol. 61, no. 7, pp. 3814–3827.

[20] S. H. Lim, S. Kim, *et al.*, "Deep learning-based beam tracking for millimeter-wave communications under mobility," *IEEE Trans. Commun.*, 2021, vol. 69, no. 11, pp. 7458–7469.

[21] Y. Lu, A. Zhong, *et al.*, "Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations," in *Proc. the 35th International Conference on Machine Learning*, 2018, vol. 80, pp. 3276–3285.

[22] Y. Rubanova, R. T. Q. Chen, *et al.*, "Latent odes for irregularly-sampled time series," in *Proc. the 33rd International Conference on Neural Information Processing Systems*, 2019, pp. 5320–5330.

[23] Z. Zhou, J. Fang, *et al.*, "Low-rank tensor decomposition-aided channel estimation for millimeter wave mimo-ofdm systems," *IEEE J. Sel. Areas Commun.*, 2017, vol. 35, no. 7, pp. 1524–1538.

## Biographies

***Mingyao Cui*** received the B.E. and M.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2020 and 2023, respectively. He is currently pursuing the Ph.D. degree in the University of Hong Kong, Hong Kong, China. His research interests include massive MIMO, millimeter-wave communications, and near field communications. He received the IEEE ICC Outstanding Demo Award and the National Scholarship in 2022, and the HKPF Scholarship in 2023.

***Hao Jiang*** received the B.S. degree in physics from Tsinghua University, Beijing, China, in 2018 and the Ph.D. degree from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2023. He is currently a Research Scientist with Huawei Technology. His research interests include mmWave communications, machine learning for wireless communications, and reconfigurable intelligent surface.

***Yuhao Chen*** received the B.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 2021, where he is currently pursuing the M.S. degree in electronic engineering. His research interests include massive MIMO, reconfigurable intelligent surface (RIS), and near-field communcations. He received the IEEE ICC Outstanding Demo Award in 2022 and the National Scholarship in 2023.

**Yang Du** received the Ph.D. degree in communication and information systems with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu, China. From 2017 to 2018, he was a Visiting Ph.D. Student with the Electrical Engineering Department, Columbia University, New York, NY, USA. In April 2019, he joined Huawei Technologies Company Ltd.

He has authored or co-authored over 30 journal and conference papers. His research interests include wireless communications and signal processing, with a focus on massive MIMO, multiple access, multiuser detection, wireless caching, anti-interference technology, compressive sensing, and machine learning. He was a recipient of the National Ph.D. Scholarship in 2015, 2016, and 2017.

**Linglong Dai** (Fellow, IEEE) received the B.S. degree from Zhejiang University, Hangzhou, China, in 2003, the M.S. degree (with the highest honor) from the China Academy of Telecommunications Technology, Beijing, China, in 2006, and the Ph.D. degree (with the highest honor) from Tsinghua University, Beijing, China, in 2011. From 2011 to 2013, he was a Postdoctoral Research Fellow with the Department of Electronic Engineering, Tsinghua University, where he was an Assistant Professor from 2013 to 2016, an Associate Professor from 2016 to 2022, and has been a Professor since 2022. His current research interests include massive MIMO, reconfigurable intelligent surface (RIS), millimeter-wave and Terahertz communications, near-field communications, machine learning for wireless communications, and electromagnetic information theory.

He has coauthored the book *MmWave Massive MIMO: A Paradigm for 5G* (Academic Press, 2016). He has authored or coauthored over 100 IEEE journal papers and over 60 IEEE conference papers. He also holds over 20 granted patents. He has received five IEEE Best Paper Awards at the IEEE ICC 2013, the IEEE ICC 2014, the IEEE ICC 2017, the IEEE VTC 2017-Fall, the IEEE ICC 2018, and the IEEE GLOBECOM 2023. He has also received the Tsinghua University Outstanding Ph.D. Graduate Award in 2011, the Beijing Excellent Doctoral Dissertation Award in 2012, the China National Excellent Doctoral Dissertation Nomination Award in 2013, the URSI Young Scientist Award in 2014, the *IEEE Transactions on Broadcasting* Best Paper Award in 2015, the *Electronics Letters* Best Paper Award in 2016, the National Natural Science Foundation of China for Outstanding Young Scholars in 2017, the IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award in 2017, the IEEE ComSoc Asia-Pacific Outstanding Paper Award in 2018, the *China Communications* Best Paper Award in 2019, the *IEEE Access* Best Multimedia Award in 2020, the IEEE Communications Society Leonard G. Abraham Prize in 2020, the IEEE ComSoc Stephen O. Rice Prize in 2022, the IEEE ICC Outstanding Demo Award in 2022, and the National Science Foundation for Distinguished Young Scholars in 2023. He was listed as a Highly Cited Researcher by Clarivate Analytics from 2020 to 2023. He was elevated as an IEEE Fellow in 2022.