The following publication C. Ding and I. W. -H. Ho, "Digital-Twin-Enabled City-Model-Aware Deep Learning for Dynamic Channel Estimation in Urban Vehicular Environments," in IEEE Transactions on Green Communications and Networking, vol. 6, no. 3, pp. 1604-1612, Sept. 2022 is available at https://dx.doi.org/10.1109/TGCN.2022.3173414.

Digital-Twin-Enabled City-Model-Aware Deep Learning for Dynamic Channel Estimation in Urban Vehicular Environments

Cao Ding, Ivan Wang-Hei Ho, *Senior Member, IEEE,*

***Abstract*—Most of the existing works on vehicle-to-everything (V2X) communications assume some deterministic or stochastic channel models, which is unrealistic for highly-dynamic vehicular channels in urban environments under the influence of high-speed vehicle motion, intermittent connectivity, and signal attenuation in urban canyon. Enabled by the concept of digital twin, the digital replica of a real-world physical system, this paper proposes a city-model-aware deep learning algorithm for dynamic channel estimation in urban vehicular environments. Specifically, the digital twin simulation allows us to accurately model radio ray reflection and attenuation in urban canyon, and the data can be supplemented and validated with empirical measurements. Our results indicates that the city-model-aware deep neural network (CMA DNN) estimator performs much better than conventional methods and has more than 32% improvement in BER when compared with generic DNN approaches that do not take the 3D city model into account. Since some geometry-based models like ray-tracing techniques used in the digital twin simulation for dynamic channel modeling could be computational expensive, we also propose a basis expansion model (BEM) approach to simplify the computation load of the overall methodology to gain a good balance between accuracy and timeliness.**

***Index Terms*—Digital Twin, Vehicular Networks, Deep Learning, Channel Estimation.**

I. Introduction

Channel estimation has a pivotal role in signal detection and data recovery in Orthogonal Frequency Division Multiplexing (OFDM) wireless communication systems. The existing literature on channel estimation is roughly divided into two categories: the conventional algorithms that are based on channel statistics [1], such as the least square (LS) and linear minimum mean square error (LMMSE) methods; the others are deep neural network (DNN) based approaches [2], [3], [4]. Recently, a considerable amount of literature has been published on deep learning for wireless communications. These studies include channel estimation and signal detection [5], network security problems [6], and channel state information (CSI) feedback [7]. In the literature on channel estimation, the DNN is trained as an estimator that performs as the receiver in the communication system to retrieve the transmitted data. To achieve high-quality estimation, both types of estimators rely on inserting pilots [8] according to a pre-set format (e.g., block, comb, lattice, etc.), especially the conventional ones. However, the use of a large number of pilot tones will critically consume the energy and bandwidth for the transmission of data symbols. To address this issue, Abdelhamid *et al.* [9] investigated an LS-based decision feedback channel estimation that reduces 76% of the pilots' power compared to the conventional LS approach. However, several studies [10] [11] indicated that the wireless channels for vehicular ad-hoc networks (VANETs) are far more problematic to estimate due to the high-speed motion and intermittent connectivity within vehicles, which may lead to the failure of meeting various green communication requirements. In general, highly accurate channel estimation and signal detection will facilitate the implementation of various cooperative driving strategies (e.g., smart platooning, intelligent ramp merging), so as to enable a smarter and safer transportation system for smart cities.

The existing literature on channel estimation is extensive and most of the works focus particularly on static OFDM systems [12]. However, in highly-mobile vehicular networks, the high-speed movement of vehicles causes Doppler frequency shift as well as carrier frequency offsets (CFOs) [13], resulting in inter-carrier interference (ICI) in addition to multipath transmissions in urban canyon environments [14]. Furthermore, a large body of work in the VANET literature is based on link-layer simulations [15], and most of them assume some deterministic or stochastic radio propagation models like Rayleigh fading in wireless communications [16]. There is a relatively small body of literature that considers realistic building effects in urban areas. The novelty of this paper is to consider the high mobility and 3D urban canyon effect for V2X communication channels and customize a channel estimation methodology combining the digital twin (DT) and deep learning concepts. With the aid of the complex-exponential basis expansion model (CE-BEM) [17], a better trade-off between BER and computational complexity can be achieved to allow the DT to run efficiently.

The digital twin has emerged as a cyber 3D platform for representing a physical system, which takes realistic data of physical entities in terms of their properties and parameters and various sensor data as inputs, and predicts the future states of the system based on simulation tools and machine learning methods. The concept of DT was first articulated by Jiaju *et al.* in the field of aerospace engineering [18] and popularised in Industry 4.0 for various manufacturing processes. This paper

This work was supported in part by the General Research Fund (Project No. 15201118) established under the University Grant Committee (UGC) of the Hong Kong Special Administrative Region (HKSAR), China; and by The Hong Kong Polytechnic University (Project No. 1-ZVTJ).

C. Ding and I. W.-H. Ho (corresponding author) are with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: caosnow.ding@connect.polyu.hk; ivanwh.ho@polyu.edu.hk).

combine the concept of DT with various vehicular networking and traffic modeling techniques and develop a platform for joint transport and communications simulation in a 3D virtual game environment that can accept real-time human driver inputs.

Overall, the main contributions of this work are as three-fold:

i) We present the utilization of a digital twin system that integrates the 3D virtual urban environment, traffic simulation and network simulation in the cyber world, and real automobile or driving simulator in the physical world. It serves as a test bench for the dynamic vehicular channel estimation problem in this work, and such a cyber-physical system (CPS) can be used to facilitate the design and optimization of different models and algorithms in vehicular networks and intelligent transportation systems (ITS).

ii) We propose a dynamic channel estimation approach for V2X communications based on city-model-aware (CMA) DNN, which takes into account the influence of building blockage and attenuation in urban canyons via a realistic 3D city model, ray-tracing technique, and building penetration loss model in the digital twin. A digital twin-enabled DNN channel estimation framework is also presented. Our result shows that under the signal-to-noise ratio (SNR) of 15 dB, the BER of the CMA DNN estimator is reduced by more than 32% as compared with the ordinary DNN estimator.

iii) We introduce the BEM for approximating high-speed dynamic wireless channels to reduce the computational and time complexity of CMA DNN and other channel estimation algorithms, which can greatly reduce the complexity of the digital twin simulation without affecting the BER performance much. The proposed BEM CMA DNN estimator reduces the simulation runtime by more than 33% by sacrificing about 6.5% of BER performance.

The structure of the paper is as follows. Section II introduces the system model. Section III presents the implementation of the digital-twin-enabled CMA DNN channel estimation. Section IV compares the estimation results and computational complexity. Finally, Section V concludes the paper.

II. System Models

We consider a single-input single-output (SISO) vehicle-to-infrastructure (V2I) communication scenario as shown in Fig. 1. The base station (BS) is the transmitter and continuously transmits signals to the moving vehicle i, which is the mobile receiver.

A. Multipath Propagation Channel

In the dynamic wireless channel, the signals sent by the BS experience multiple paths, each path encounters different path gain and delay spread, then finally converges at the receiver vehicle i. In Fig. 1, paths L_1 and L_2 shown in the figure are the line of sight (LOS) path and building reflected paths (NLOS), respectively. The transmitted signals suffer high delay

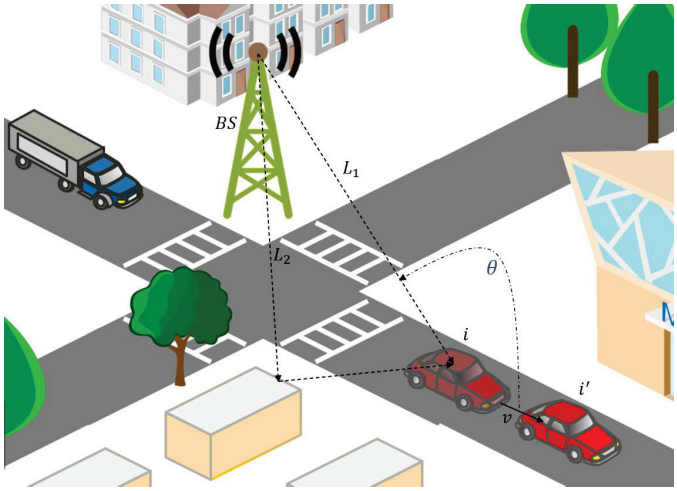


Fig. 1. The SISO V2I communication scenario

spread and doppler frequency shift at a scale of hundreds of nanoseconds [19]. In this work, we consider a SISO V2I scenario according to the wide-sense stationary uncorrelated scattering (WSSUS) [20] channel of VANETs. Specifically, the channel impulse response for node i can be described as

$$h(\tau,t)=\sum_{m=1}^{M(t)}\delta(\tau-\tau_m(t))G_m(t), \tag{1}$$

where τ is the delay spread, t denotes time. $M(t)$ is the total number of propagation paths. For the m-th propagation path, the complex channel gain is $G_m(t)$, which can be expressed as

$$G_m(t)=\sqrt{\frac{P_m(t)}{R}}\sum_{r=1}^{R}e^{j\Phi_{m,r}}e^{j|\vec{\nu}|\cos(\psi_{m,r}(t)-\theta)t}, \tag{2}$$

where $P_m(t)$ is the time-varying power, R is the total number of radio ray components in a cluster. $\Phi_{m,r}$ is the random phase parameter, which obeys the uniform distribution of $[0,2\pi]$. $\vec{\nu}$ is the relative speed between i and BS, $\psi_{m,r}(t)$ is the angle of arrival (AOA) of the r-th ray component, θ is the angle between the main path and the driving direction. In this paper, we adopt the IMT-Advanced high-speed railway channel standard [21] for setting the above parameters (i.e., $P_m(t)$, $\tau_m(t)$ and $M(t)$).

B. Doppler Frequency Shift

The Doppler shift parameter is introduced in the simulation. Assumed that in a very short time interval Δt, vehicle i moves to position $i^{'}$, the BS transmits signals to i through radio ray, which has the electromagnetic frequency f and wavelength λ. The phase change between i and $i^{'}$ can be expressed as

$$\Delta\phi=-\frac{2\pi|\vec{\nu}|\Delta t}{\lambda}\cos(\theta). \tag{3}$$

The Doppler frequency shift is

$$f_{ds}=-\frac{|\vec{\nu}|\cos(\theta)}{\lambda} \tag{4}$$

C. Conventional Channel Estimation Schemes

Conventional algorithms (e.g., LS, LMMSE) are considered as the benchmarks. For the dynamic wireless channel in (1), after performing the discrete Fourier transform (DFT) at the receiver, the received signals can be represented as

$$\boldsymbol{Y}_p(k) = \boldsymbol{H}_p(\tau, k)\boldsymbol{X}_p(k) + \boldsymbol{\Omega}(k), \tag{5}$$

where $\boldsymbol{X}_p(k)$, $\boldsymbol{H}_p(\tau, k)$, and $\boldsymbol{\Omega}(k)$ are the transmitted signals, channel response, and the DFT of additive white Gaussian noise (AWGN) respectively. The LS estimates the channel without considering the noise given in (5), and the LMMSE further minimizes the mean squar error (MSE) by the linear weighting method based on the LS estimation.

III. Digital-Twin-Enabled City-Model-Aware DNN Channel Estimation

In this section, we present the implementation of the city-model-aware DNN (CMA DNN) dynamic channel estimation approach based on a digital twin platform, where the digital twin is customized according to the traffic experiment on PolyU campus.

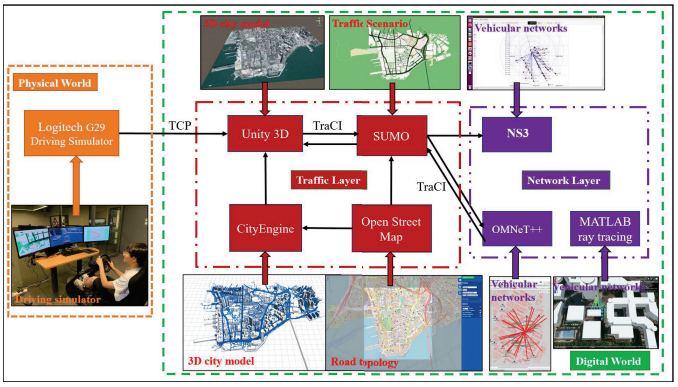


Fig. 2. The Digital Twin for Vehicular Networks and ITS

A. Digital Twin for Vehicular Networks and Intelligent Transportation Systems (ITS)

To the best of our knowledge, this is one of the first studies that utilizes a digital twin paradigm for vehicular networks and ITS. The proposed framework is illustrated in Fig.2.

1) The physical world: we employ a driving simulator to imitate the ego vehicle, due to the cost for realistic on-road experiments are expensive and it is hard to evaluate the effect of individual parameters in different scenarios via experiments. The driving simulator realizes the real-time connection between the real world and the virtual world through the game engine (i.e., Unity 3D). Besides, the experimental data of traffic (e.g., vehicle status, sensor data) can also be imported into the digital twin platform as reference data for the simulation.

2) The digital (cyber) world: we use the open street map (OSM) database to reconstruct the realistic street layout and building information of Tsim Sha Tsui, Hong Kong, and utilize this information to generate the stochastic traffic scenario (e.g., stochastic vehicle types, traffic light system (TLS), and automated driving) in the simulation of urban mobility (SUMO). The vehicle trajectories and traffic information is imported into the network simulation (based on OMNET++, NS3, MATLAB, etc.) in real-time for simulating the vehicular networks.

3) Simulation Loop empowered by the Traffic Control Interface (TraCI): In the digital twin, the port that connects the 3D virtual environment, traffic simulation and network simulation is the TraCI, which uses the client/server architecture based on transmission control protocol (TCP) to provide access to SUMO in real-time. In our simulation loop, TraCI is utilized to construct the bridges among the traffic simulation, the network simulation, and the Unity 3D engine.

4) The Usage of Simulation Results: In this paper, the digital twin is used as a tool to collect training datasets for DNN channel estimators. By collecting simulated channel data under different conditions (i.e., with building loss and without building loss) and training corresponding DNN models, we aim to improve the accuracy and efficiency of dynamic channel estimation.

B. City-Model-Aware Channel Modeling

In this section, we propose a city-model-aware (CMA) channel modeling approach that quantitatively considers the reflection loss and penetration loss with respect to the buildings on PolyU campus, Tsim Sha Tsui, Hong Kong.

1) Building Reflection Loss: In the network simulation, we divide signal paths into LOS and NLOS in the building reflection loss model. We apply different loss calculation models according to different radio propagation situations. In the LOS situation, the two-ray ground reflection model is used for signal loss. In the NLOS situation, the reflection model uses the two-ray ground reflection model and the knife-edge diffraction model.

The overall CMA loss model can be mathematically expressed as

$$\begin{aligned} |U_{tot}| = &\left(\frac{U_0 d_0}{d_{LOS}} \cos(\omega(t - \frac{d_{LOS}}{c})\right) \\ &+ \sum_a R_a \frac{U_0 d_0}{d_a} \cos(\omega(t - \frac{d_a}{c})) \\ &+ \sum_e D_e \frac{U_0 d_0}{d_e} \cos(\omega(t - \frac{d_e}{c}))) \end{aligned} \tag{6}$$

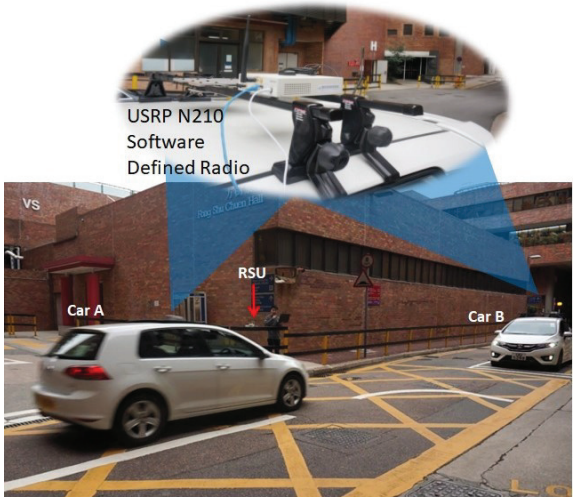
$$P_r = \frac{|U_{tot}|^2 \lambda^2}{480\pi^2}, \tag{7}$$

where U_{tot} is the resultant electric-field envelope, R_a is the reflection coefficient of the a-th path, D_e is the diffraction coefficient of the e-th path, and P_r is the received power.

We implement the network simulation accordingly, the location is the intersection of Fong Shu Chuen Hall on PolyU campus, which the simulation result is consistent with the

empirically collected dataset in our previous work [22]. As shown in Fig. 3(a), the red label is the *BS*, and the blue labels denote the trajectory of a moving vehicle. By collecting the transmitted and received signals along the trajectory, we obtain the dataset for the CMA model, which contains the building loss feature on the PolyU campus.

(a) Ray-tracing V2I scenario on PolyU campus



(b) On-road V2I experiment with USRP on PolyU campus

Fig. 3. City-Model-Aware Simulation and Experiment

TABLE I
PENETRATION VALUE OF DIFFERENT BUILDING MATERIALS

No.	Building Material	Penetration Loss Value (dB)	Height Gain	2 dB/Floor
1	Wood	4	Outdoor shadowing	7 dB
2	Concrete and Windows	7-15	Indoor shadowing	10 dB
3	Stone	12	External wall	5 dB

2) Building Penetration Loss: In addition to the reflection at buildings, the penetration effect also causes signal fading, and the signal loss varies with respect to different materials [23]. In the network simulation, we take into account the building penetration loss based on a module in NS3. We summarize the building penetration value of different types of building materials and antenna height gain in Table I, the building penetration loss can be expressed as

$$Loss_{(Pen)} = \begin{cases} ItuR1411(\boldsymbol{C}_B, \boldsymbol{C}_V), LOS \\ ItuR1411(\boldsymbol{C}_B, \boldsymbol{C}_V) + N_{wall} B_{Pen}, NLOS \end{cases} \tag{8}$$

where $\boldsymbol{C}_B$ and $\boldsymbol{C}_V$ denote the coordinates of the *BS* and vehicle i, respectively. $ItuR1411(X, Y)$ is a propagation loss model defined in NS3, N_{wall} denotes the number of walls of the considered building, B_{Pen} is the corresponding building penetration loss as illustrated in TABLE I.

As shown in Fig. 3(b), on-road V2I experiments based on universal software radio peripheral (USRP) were conducted on PolyU campus to collect experimental vehicular channel data [22]. One set of the data is used in the training dataset for the CMA channel modeling approach, and another set is used as testing data to verify the performance of the proposed models.

C. Basis Expansion Model

Complex-exponential BEM (CE-BEM) has the advantages of fast generation of basis vectors, independent of channel statistics, and the orthogonality of basis vectors. Therefore, this paper exploits CE-BEM for modeling high-speed vehicular channels, the channel impulse response of the s-th symbol at the n-th sampling point on the u-th tap in (1) can be expressed as

$$\boldsymbol{h}_s(n,u) = \sum_{q=0}^{Q-1} b_{n,q} \boldsymbol{c}_{s,u,q} \tag{9}$$

$$= \boldsymbol{b}_n^T \boldsymbol{c}_{s,u}, \tag{10}$$

where $\boldsymbol{h}_s(n,u)$ denotes the channel impulse response, Q is the dimension of the compressed basis vector, $\boldsymbol{b}_n = [\boldsymbol{b}_{n,0}, ..., \boldsymbol{b}_{n,Q-1}]^T$ is the basis vector, and $\boldsymbol{c}_{s,u} = [\boldsymbol{c}_{s,u,0}, ..., \boldsymbol{c}_{s,u,(Q-1)}]^T$ is the coefficient vector of contractive basis. Since the CE-BEM is used, we have $b_{n,q} = e^{\frac{2j\pi(q-Q)n}{N_{sub}}}$, where N_{sub} is the number of subcarriers. Let $\boldsymbol{g}_{s,u} = [\boldsymbol{h}_s(0,u), ..., \boldsymbol{h}_s(N-1,u)]^T$ denote the channel response vector of the s-th OFDM symbol on the u-th tap, $\boldsymbol{B} = \boldsymbol{I}_U \otimes [\boldsymbol{b}_0, ..., \boldsymbol{b}_{N-1}]^T$ and $\boldsymbol{c}_s = [\boldsymbol{c}_{s,0}^T, ..., \boldsymbol{c}_{s,U-1}^T]$, then we have

$$\boldsymbol{g}_s = [\boldsymbol{g}_{s,0}^T, ..., \boldsymbol{g}_{s,U-1}^T]^T \tag{11}$$

$$= \boldsymbol{B}\boldsymbol{c}_n + \boldsymbol{\zeta}, \tag{12}$$

where $\boldsymbol{g}_s$ denotes the channel impulse response of the s-th OFDM symbol, and $\boldsymbol{\zeta}$ is the compression error of the BEM model. The mathematical analysis in [24] shows that it is reasonable to ignore the BEM compression error from the perspective of receiver design. Because with the increase of the dimension Q, the compression error constantly decreases.

By replacing the channel impulse response $h_p(\tau, t)$ in (5) in the OFDM system transmission model with BEM, we have the baseband OFDM transmission model based on the BEM channel as follows

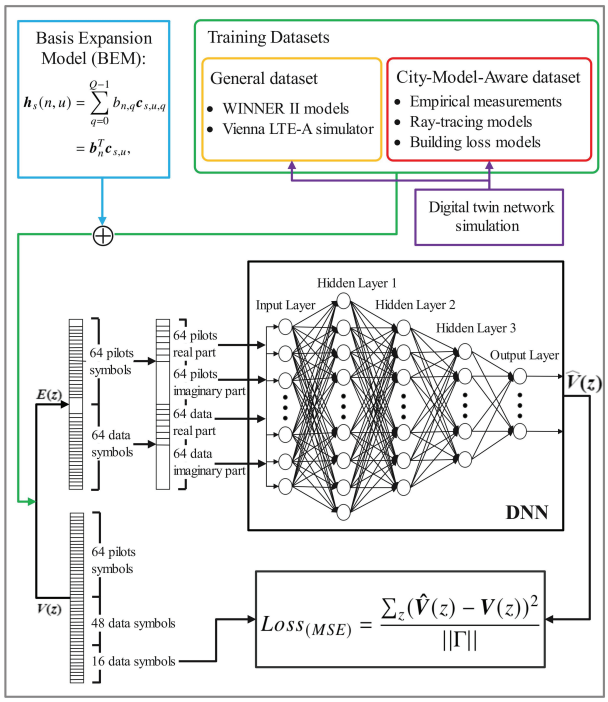
$$y_s(t) = \boldsymbol{A}_s(t)\boldsymbol{c}_s(t) + \omega(t), \tag{13}$$

where $\boldsymbol{A}_s(t) = \boldsymbol{F}\tilde{\boldsymbol{X}}_s\boldsymbol{B}$ is the observation matrix, $\boldsymbol{F}$ is the discrete Fourier transform (DFT) matrix,

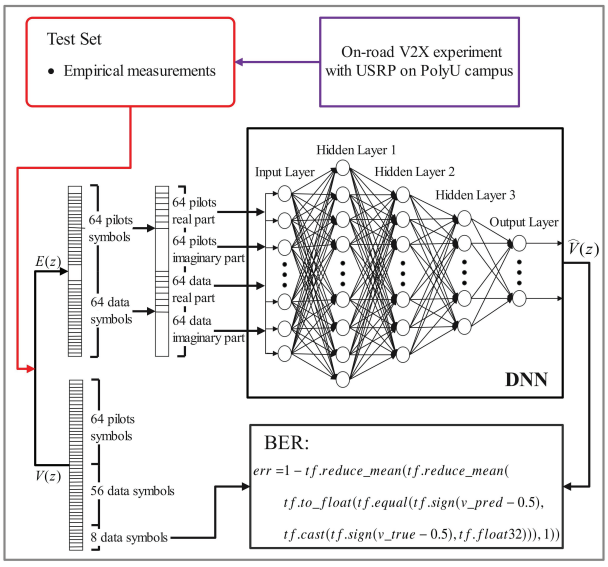
$\tilde{\boldsymbol{X}}_s = [\boldsymbol{x}_s^{(0)}, ..., \boldsymbol{x}_s^{(U-1)}]$ is the set of diagonal matrices of all transmission symbols in the s-th symbol with $\boldsymbol{x}_s^{(l)} = diag\{[\boldsymbol{x}(N_{sub}-u), \boldsymbol{x}(N_{sub}-u+1), ..., \boldsymbol{x}(0), ..., \boldsymbol{x}(s-u-1)]\}$.

Our experimental results indicate that the proposed approach can effectively reduce the complexity while greatly maintain BER performance. The reader is referred to Section IV. B for the channel estimation performance and complexity analysis of various BEM approximation approaches.

D. DNN-based Channel Estimation



(a) The Training Process of the DNN Estimator



(b) The Test Process of the DNN Estimator

Fig. 4. The DNN-based Channel Estimation Framework

In this paper, DNN is utilized to estimate the channel and recover the data of dynamic vehicular networks. The training and test process of DNN-based channel estimation is shown in Fig. 4(a) and Fig. 4(b), respectively.

1) Relationships Between Datasets: The training datasets include the city-model-aware dataset and general dataset. In addition to simulation data from the digital twin, the CMA dataset is enriched with one set of previous empirical measured data on PolyU campus, while the general dataset is obtained based on the WINNER II model [25] and the Vienna LTE-A simulator by Soltani *et al* [26]. The test data is another set of empirical measurements on PolyU campus. The training sets are used to train DNN estimators and save them as Keras models. We then load these Keras models to perform data recovery and calculate the BER based on the same test set.

2) The Training Process: We assume the OFDM communication process as a black box, and the inputs to the DNN model includes original transmitted signals $\boldsymbol{V}(z)$ and received signals $\boldsymbol{E}(z)$ (for supervised learning) which are composed of two OFDM blocks: a pilot block and a data block. The received signals can be expressed as

$$\boldsymbol{E}(z) = \boldsymbol{H}(z)\boldsymbol{V}(z) + \boldsymbol{\Omega}(z), \tag{14}$$

The output of the neural network is the estimated data $\hat{\boldsymbol{V}}(z)$, which is a cascade of nonlinear transformations of the input data via different layers in the DNN, it can be expressed as

$$\hat{\boldsymbol{V}}(z) = f(\boldsymbol{V}(z), \boldsymbol{\varpi}) = f^{(J_{tot}-1)}(f^{(J_{tot}-2)}(...f^{(1)}(\boldsymbol{V}(z)))), \tag{15}$$

where J_{tot} and $\boldsymbol{\varpi}$ are the total number of layers and the weights of the DNN, respectively. $f(\beth)$ represents the nonlinear function which may be the ReLu function or Sigmoid function, defined as $f_{Relu}(\beth) = max(0, \beth)$, $f_{Sigmoid}(\beth) = \frac{1}{1+e^{-\beth}}$, respectively. The DNN model consists of the input layer, three hidden layers that all applied the ReLu function and the output layer that applied the Sigmoid function.

In the training implementation of DNN, we input two OFDM blocks to the DNN at a time. We consider two pilot configurations with 8 pilots and 64 pilots in the first OFDM block.

i) For the case of 64 pilots, as shown in Fig. 4(a), we divide the 64 pilot symbols plus 64 data symbols into the real parts (of 128 symbols) and imaginary parts (of 128 symbols). At the input layer of DNN, we set 256 neurons to accept the 256 real and imaginary parts. At the DNN output layer, we have 16 sigmoid function neurons to recover 16 bits and adopt eight DNN networks in parallel to recover all of the 128 bits in the data block, which can reduce the complexity of DNN training and enhance the BER performance.

ii) For the case of 8 pilots, the 8 pilots are evenly inserted into the first OFDM block, and data symbols are transmitted on the other subcarriers (i.e., 8 pilots symbols plus 120 data symbols in the two OFDM blocks). Instead of changing the number of neurons in each layer of DNN, we use 15 DNN networks in parallel to restore the 240 bits of data.

iii) For the loss function, we use the mean square error (MSE) as the loss function of the DNN, mathematically expressed as

$$Loss_{(MSE)} = \frac{\sum_z (\hat{\boldsymbol{V}}(z) - \boldsymbol{V}(z))^2}{||\Gamma||} \tag{16}$$

where $||\Gamma||$ denotes the total number of elements in the training dataset, $\hat{\boldsymbol{V}}(z)$ is the recovered data (i.e., the prediction), and $\boldsymbol{V}(z)$ is the originally transmitted data (i.e., the label). For example, in Fig. 4(a), the 16-bit prediction is evaluated against the originally transmitted data.

3) The Test Process: We load trained DNN channel estimators, input the test set for data recovery and calculating the BER with respect to the original data. Specifically, in Keras, we set the BER as a metrics, and its BER is calculated as follows

$$\begin{aligned} err =& 1 - tf.reduce\_mean(tf.reduce\_mean(tf.to\_float(\\ & tf.equal(tf.sign(v_{pred} - 0.5), tf.cast(tf.sign(v_{true} \\ & - 0.5), tf.float32))), 1)) \end{aligned} \tag{17}$$

Where v_{pred} is each bit of DNN output, v_{true} is each original bit in the test set, $tf.*$ are respective mathematical functions in Tensorflow. Since the estimated v_{pred} contains decimals and v_{true} contains only 0 and 1, we only compare if the signs of $(v_{pred} - 0.5)$ and $(v_{true} - 0.5)$ are consistent to determine whether the estimation is correct. We then compare all samples of the test set and calculate the average BER.

IV. Numerical Results and Performance Evaluation

In this section, we compare the numerical simulation results with multiple benchmarks for verifying the performance of the digital-twin-enabled CMA DNN channel estimation approach.

A. Simulation Parameters

For a fair comparison of different methods, we have the same initial parameters in these experiments. According to the IEEE 802.11p physical layer standard, the transmission bandwidth is set as 10 MHz, the number of subcarriers is 64, the length of cyclic prefix is 16 bits, the number of resource blocks is 50, the number of multiple transmission paths is 24, and the maximum delay in a sampling period is 16 μs. We consider additive white Gaussian noise (AWGN) in the OFDM system with QPSK modulation, and the relative speed between the transmitter and receiver is 88 km/h.

The DNN model is a fully connected network that consists of five layers: the input layer, three hidden layers, and the output layer. The number of neurons in each layer is 256, 512, 256, 128, and 16 respectively. We utilize Keras optimized based on TensorFlow deep-learning framework for DNN model training and testing, and use NVIDIA CUDA for the GPU back-end to shorten the training and testing time. The number of training epochs is set to 10,000, the learning rate is set to 0.001 and the batch size is 64. The training and test datasets include the received symbol set and the original transmitted symbol set. Assume that every 128 received symbols and the corresponding 16 bits original data as one sample, the number of samples in each training set is 25,000, while the number of samples in the test set is 10,000.

B. Performance Evaluation

Fig. 5 compares the BER curves of the proposed CMA DNN estimator and other conventional algorithms. Noted that the 64-pilot cases serve as benchmarks here for evaluating the performance of 8-pilot cases, since it represents an ideal case with half of the transmitted symbols as pilots. In normal practice, it is not possible to have 64 pilots per two OFDM blocks as it will severely reduce the data efficiency. For the 64-pilot cases, at 15 dB SNR, the BER of CMA DNN is 74.68%, 8.50%, and 13.28% lower than that of LS, LMMSE, and DNN, respectively. For the 8-pilot cases, the BER improvement is even better, the BER is 92.23%, 88.50%, and 32.38% lower than that of LS, LMMSE, and DNN, respectively. Fig. 5 also compares the BER of the 8-pilot estimators after BEM compression. At 15 dB SNR, the BER of the BEM CMA DNN is 33.64%, 31.20%, 29.65%, and 24.77% lower than that of BEM LS, BEM LMMSE, BEM DNN, and DNN, respectively. One major finding is that the BER of BEM CMA DNN is only 6.5% higher than that of CMA DNN, but it can greatly reduce the computational complexity and speed up the simulations of the digital twin.

Fig. 6 shows the normalized mean square error (NMSE) performance of different 8-pilot channel estimators. At 15 dB SNR, the NMSE of the CMA DNN is 92.99%, 91.78%, 78.51%, 73.10%, 63.05%, 49.13%, and 19.48% lower than that of LS, LMMSE, BEM LS, BEM LMMSE, BEM DNN, DNN, and BEM CMA DNN respectively. The mathematical formula for calculating the NMSE in the simulation is as follows

$$NMSE = \frac{\sum_{j=1}^{J} (\boldsymbol{h}_{est} - \boldsymbol{h}_{em})^2}{\sum_{j=1}^{J} (\boldsymbol{h}_{em})^2}, \tag{18}$$

where $\boldsymbol{h}_{em}$ is the theoretical transfer function generated from the test set, $\boldsymbol{h}_{est}$ is the estimated transfer function for each channel estimator, and J is the number of estimations.

Fig. 7 and Fig. 8 compare the BER and NMSE performance of the proposed 8-pilot CMA DNN estimator with other related V2X channel estimators introduced in [27]. These related channel estimators include the spectral temporal averaging (STA) channel estimator, the constructed data pilots (CDP) channel estimator, the time-domain reliability test frequency-domain interpolation (TRFI) channel estimator, the minimum MSE using virtual pilots (MMSE-VP) channel estimator, the auto-encoder DNN (AE-DNN), and the STA-DNN, respectively. For a fair comparison, we adjust the simulation parameters accordingly: the relative speed is set to 40 km/h, the number of multi-paths is set to 12, and the other parameters are consistent with the VTV Urban Canyon (VTV-UC QPSK) scenario in [27]. As shown in Fig. 7, at 15 dB SNR, the BER of the proposed 8-pilot CMA DNN is 21.28%, 19.90%, 83.65%, 91.01%, and 94.19% lower than that of STA-DNN(15-15-15), STA-DNN(15-10-15), AE-DNN, MMSEVP, and TRFI (CDP), respectively. In Fig. 8, at 15 dB SNR, the NMSE of

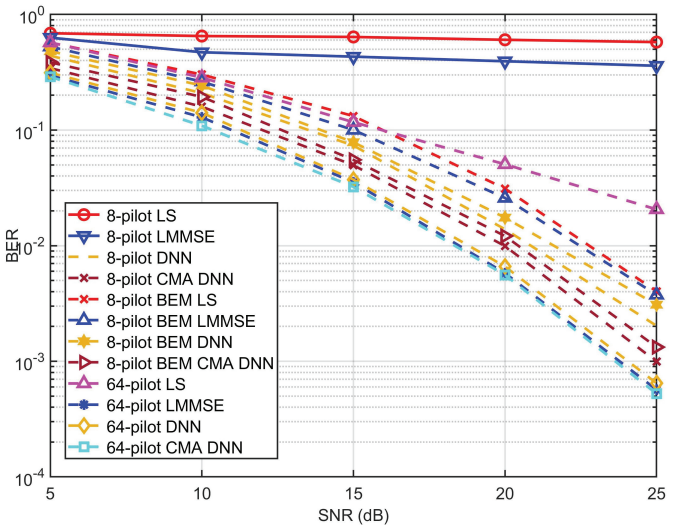


Fig. 5. Comparison of BER Performance at 88 km/h

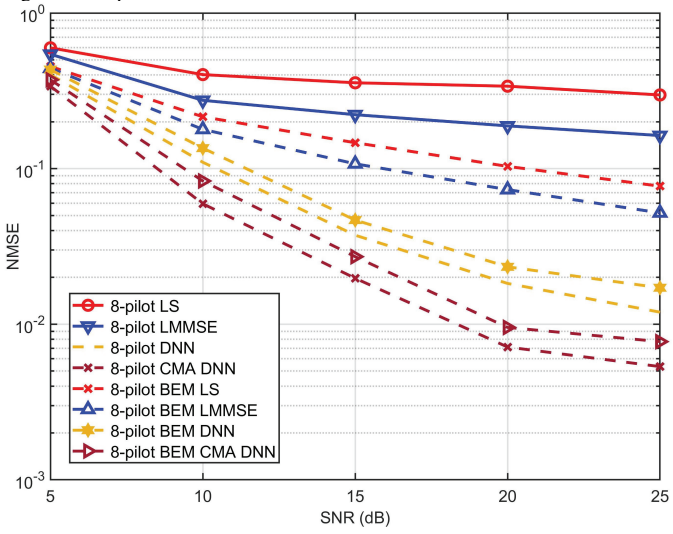


Fig. 6. Comparison of NMSE Performance at 88 km/h

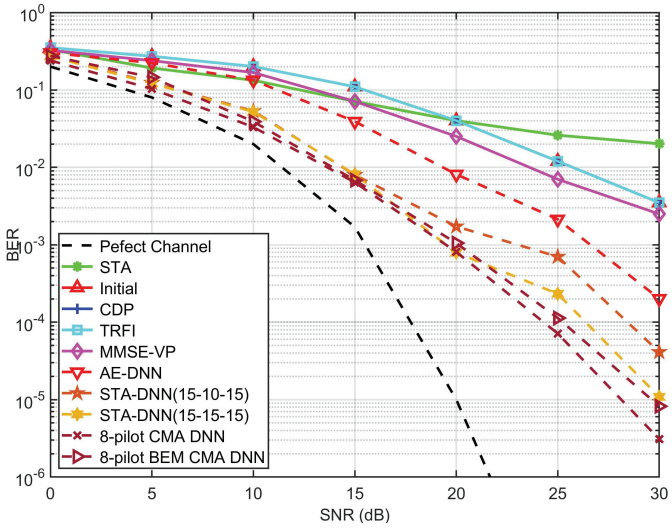


Fig. 7. Comparison of BER Performance at 40 km/h

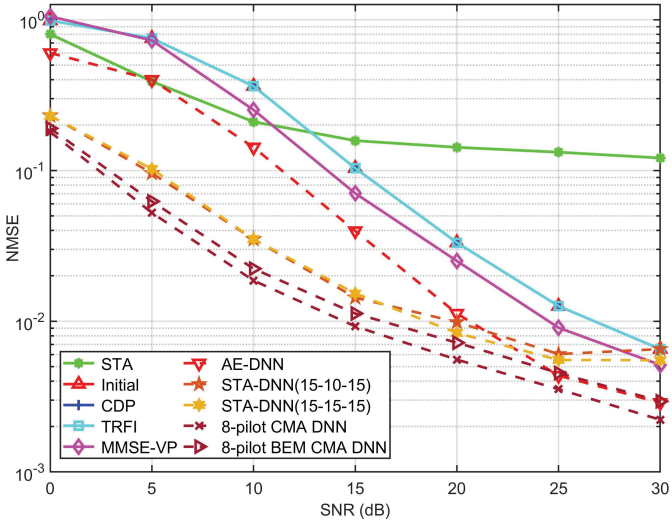


Fig. 8. Comparison of NMSE Performance at 40 km/h

the proposed 8-pilot CMA DNN is 39.34%, 35.54%, 76.69%, 86.88%, and 91.06% lower than that of STA-DNN(15-15-15), STA-DNN(15-10-15), AE-DNN, MMSEVP, and TRFI (CDP), respectively.

C. Computational Complexity Analysis

Since the channel estimation duration varies with the performance of different CPUs and GPUs, we compare the computational and time complexity of the proposed and other estimators in Table II and rank them from low to high complexity level.

For the calculation of the time complexity of the CMA DNN estimator, the time complexity of handling building reflection loss by ray tracing is much greater than that of the penetration loss model. Therefore, we primarily consider the worst-case due to ray tracing here. We built the cost equation referring to the previous study [28], which is expressed as follows

$$E_{CMA} = T_0 + p_{total}T_s + p_{avg}\Theta_{total}\kappa T_h, \qquad (19)$$

TABLE II
COMPUTATIONAL COMPLEXITY COMPARISON

No.	Channel Estimator	Computational Complexity (Times of Multiplying Operations)	Time Complexity (From low to high)
1	LS	N_{sub}	$O(N_{sub})$
2	DNN and BEM DNN	$J_1J_2 + J_2J_3 + J_3J_4 + J_1J_4$	$O(J_1J_2)$
3	BEM CMA DNN	$(QL_d) + p_{total}(QL_d) + p_{avg}\Theta_{total}\kappa(QL_d) + J_1J_2 + J_2J_3 + J_3J_4 + J_1J_4$	$O(p_{avg}\Theta_{total} \cdot \kappa(QL_d))$
5	CMA DNN	$T_0 + p_{total}T_s + p_{avg}\Theta_{total}\kappa T_h + J_1J_2 + J_2J_3 + J_3J_4 + J_1J_4$	$O(p_{avg}\Theta_{total}\kappa T_h)$
6	BEM LS	$4(QL_d)^2N_{sub}$	$O((QL_d)^2N_{sub})$
7	BEM LMMSE	$2N_{sub}^2(QL_d) + 18N_{sub}(QL_d)^2 + 3(QL_d)^3 + (QL_d)^2$	$O(N_{sub}^2(QL_d))$
8	LMMSE	$3N_{sub}^3 + N_{sub}^2$	$O(N_{sub}^3)$

where p_{total} and p_{avg} represent the total and average cell probabilities, and Θ_{total} represents the total occupancy. T_0, T_s and T_h represent the time to initialize a ray, the time to step to the next cell, and the time to test an object for intersection with the ray. κ is the collusion factor between the distributions of objects and rays. We adopt the default settings for these parameters with respect to the ray-tracing function in MATLAB.

As shown in Table II, J_1, J_2, J_3, and J_4 represent the number of neurons in the input layer and three hidden layers of the DNN, respectively. Q represents the dimension of the BEM compression basis, and L_d represents the vector of transmitted symbols. We can see from the table that using BEM to model corresponding channels can effectively reduce the computational and time complexity of various estimators. Specifically, for the CMA DNN method, reducing the complexity can reduce the simulation delay of the digital twin. Hence, large-scale training data for the DNN can be obtained in a more efficient manner.

TABLE III
SIMULATION RUN TIME COMPARISON

Channel Estimator	**Total Runtime (ms)**
CMA DNN	213,845,693
BEM CMA DNN	142,458,157
Computer Configuration	**Parameter**
CPU	Intel core i7-10870H
GPU	NVIDIA GeForce RTX 3060 Laptop (6 GB)
RAM	16 GB DDR4 3200MHz

For a practical runtime evaluation, we collected the runtime of CMA DNN and BEM CMA DNN under the 8-pilot situation as shown in Table III. Note that the total runtime here includes the CMA, BEM CMA simulation time, as well as the corresponding DNN training and testing time. Our computer configuration is also given as a reference. The total runtime of the CMA DNN integrated with BEM is 33.38% lower than that of the CMA DNN method.

V. CONCLUSION

This paper proposed a channel estimation method based on a realistic city model and radio propagation in digital twin simulation and deep neural networks. Compared with traditional algorithms such as LS and LMMSE that use 8 pilots at 15 dB SNR, the BER performance of the proposed CMA DNN method is greatly improved by more than 92.23% and 88.50%, respectively. In order to highlight the influence of hybrid building loss on channel estimation in urban canyon environments, we compared the BER performance of CMA DNN and generic DNN estimator that does not consider building loss. Our results indicate that the BER performance of CMA DNN at 15 dB SNR is improved by more than 13.28% when 64 pilots are used. When fewer pilot tones are used (e.g., 8 pilots), the BER performance of the CMA DNN estimator is much better than all other methods (e.g., more than 32.38% better than the generic DNN estimator). While improving the BER performance of the channel estimator, this paper also considered the use of BEM to simplify the simulation process of dynamic vehicular channels in the digital twin so that large-scale training data for the deep learning model can be acquired efficiently. Our results show that the CMA DNN method with BEM-approximated dynamic channels sacrifices a small level (i.e., 6.50%) of BER performance, but with greatly reduced computational complexity and runtime (i.e., 33.38%) for efficient operation of the digital twin. The proposed digital twin and deep learning investigation framework in this pioneering work will open up many other research opportunities for connected autonomous vehicles and cooperative driving in future smart cities.

REFERENCES

[1] Y. Li, L. J. Cimini, and N. R. Sollenberger, "Robust channel estimation for ofdm systems with rapid dispersive fading channels," *IEEE Transactions on Communications*, vol. 46, no. 7, pp. 902–915, 1998.

[2] H. Ye, G. Y. Li, and B. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 114–117, 2018.

[3] W. Zhang, X.-G. Xia, and P. C. Ching, "Optimal training and pilot pattern design for ofdm systems in rayleigh fading," *IEEE Transactions on Broadcasting*, vol. 52, no. 4, pp. 505–514, 2006.

[4] G. B. Jency and S. J. Gladwin, "Channel estimation using optimized pilot placement in ofdm system," in *2014 International Conference on Communication and Signal Processing*, 2014, pp. 1529–1534.

[5] H. He, C.-K. Wen, S. Jin, and G. Y. Li, "Deep learning-based channel estimation for beamspace mmwave massive mimo systems," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 852–855, 2018.

[6] M. Kang and J. Kang, "A novel intrusion detection method using deep neural network for in-vehicle network security," in *2016 IEEE 83rd Vehicular Technology Conference (VTC Spring)*, 2016, pp. 1–5.

[7] C. Wen, W. Shih, and S. Jin, "Deep learning for massive mimo csi feedback," *IEEE Wireless Communications Letters*, vol. 7, no. 5, pp. 748–751, 2018.

[8] S. Coleri, M. Ergen, A. Puri, and A. Bahai, "Channel estimation techniques based on pilot arrangement in ofdm systems," *IEEE Transactions on Broadcasting*, vol. 48, no. 3, pp. 223–229, 2002.

[9] A. Ladaycia, A. Mokraoui, K. Abed-Meraim, and A. Belouchrani, "Toward green communications using semi-blind channel estimation," in *2017 25th European Signal Processing Conference (EUSIPCO)*, 2017, pp. 2254–2258.

[10] D. Feng, C. Jiang, G. Lim, L. J. Cimini, G. Feng, and G. Y. Li, "A survey of energy-efficient wireless communications," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 167–178, 2013.

[11] K. Qian and W.-Q. Wang, "Energy-efficient antenna selection in green mimo relaying communication systems," *Journal of Communications and Networks*, vol. 18, no. 3, pp. 320–326, 2016.

[12] S. Omar, A. Ancora, and D. T. Slock, "Performance analysis of general pilot-aided linear channel estimation in lte ofdma systems with application to simplified mmse schemes," in *2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2008, pp. 1–6.

[13] Z. Wang, J. Huang, S. Zhou, and Z. Wang, "Iterative receiver processing for ofdm modulated physical-layer network coding in underwater acoustic channels," *IEEE Transactions on Communications*, vol. 61, no. 2, pp. 541–553, 2013.

[14] G. Acosta-Marum and M. A. Ingram, "Six time- and frequency- selective empirical channel models for vehicular wireless lans," *IEEE Vehicular Technology Magazine*, vol. 2, no. 4, pp. 4–11, 2007.

[15] J. Cabrejas, D. Martín-Sacristán, S. Roger, D. Calabuig, and J. F. Monserrat, "On the integration of grassmannian constellations into lte networks: A link-level performance study," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2017, pp. 1824–1829.

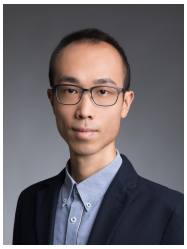
[16] V. Tarokh, N. Seshadri, and A. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Transactions on Information Theory*, vol. 44, no. 2, pp. 744–765, 1998.

[17] G. Leus, "On the estimation of rapidly time-varying channels," in *2004 12th European Signal Processing Conference*, 2004, pp. 2227–2230.

[18] J. Wu, Y. Yang, X. Cheng, H. Zuo, and Z. Cheng, "The development of digital twin technology review," in *2020 Chinese Automation Congress (CAC)*, 2020, pp. 4901–4906.

[19] A. F. Molisch, F. Tufvesson, J. Karedal, and C. F. Mecklenbrauker, "A survey on vehicle-to-vehicle propagation channels," *IEEE Wireless Communications*, vol. 16, no. 6, pp. 12–22, 2009.
[20] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Transactions on Communications Systems*, vol. 11, no. 4, pp. 360–393, 1963.
[21] Series, M., "Guidelines for evaluation of radio interface technologies for imt-advanced," *Report ITU*, vol. 638, pp. 1–72, 2009.
[22] Z. Situ, I. W.-H. Ho, T. Wang, S. C. Liew, and S. C.-K. Chau, "Ofdm modulated pnc in v2x communications: An ici-aware approach against cfos and time-frequency-selective channels," *IEEE Access*, vol. 7, pp. 4880–4897, 2019.
[23] M. M. Soliman, M. Maruful Hasan Sabbir, M. Alkaeed, M. R. Ahmed, I. Ahmed Rafi, and M. M. Hasan Mahfuz, "Investigation of signal penetration loss variation on different building components and disparity of receiver location on 3g/4g network in the context of bangladesh," in *2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC)*, 2020, pp. 53–58.
[24] F. Hlawatsch and G. Matz, *Wireless Communications Over Rapidly Time-Varying Channels*, 1st ed. USA: Academic Press, Inc., 2011.
[25] P. Kyösti, J. Meinilä, L. Hentila, X. Zhao, T. Jämsä, C. Schneider, M. Narandzic, M. Milojević, A. Hong, J. Ylitalo, V.-M. Holappa, M. Alatossava, R. Bultitude, Y. Jong, and T. Rautiainen, "Winner ii channel models," *IST-4-027756 WINNER II D1.1.2 V1.2*, 02 2008.
[26] 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.
[27] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Deep learning based channel estimation schemes for ieee 802.11p standard," *IEEE Access*, vol. 8, pp. 113 751–113 765, 2020.
[28] J. Cleary and G. Wyvill, "Analysis of an algorithm for fast ray tracing using uniform space subdivision," *The Visual Computer*, vol. 4, pp. 65–83, 03 1988.

Cao Ding received the B.Eng. degree in automation from Beijing University of Chemical Technology, Beijing, China, and the M.Sc. degree in electronic and information engineering from The Hong Kong Polytechnic University, Hong Kong. He is now a Research Assistant of The Hong Kong Polytechnic University. His research interests include vehicular networks and intelligent transport systems (ITS), specifically in vehicular ad hoc network (VANET). His research focuses most on the digital twin of vehicular networks and intelligent transport systems.



Ivan Wang-Hei Ho (M'10–SM'18) received the B.Eng. and M.Phil. degrees in information engineering from The Chinese University of Hong Kong, Hong Kong, in 2004 and 2006, respectively, and the Ph.D. degree in electrical and electronic engineering from the Imperial College London, London, U.K., in 2010. He was a Research Intern with the IBM Thomas J. Watson Research Center, Hawthorne, NY, USA, and a Postdoctoral Research Associate with the System Engineering Initiative, Imperial College London. In 2010, he cofounded P2 Mobile Technologies Ltd., where he was the Chief Research and Development Engineer. He is currently an Associate Professor with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong. His research interests include wireless communications and networking, specifically in vehicular networks, intelligent transportation systems (ITS), and Internet of things (IoT). He primarily invented the MeshRanger series wireless mesh embedded system, which received the Silver Award in Best Ubiquitous Networking at the Hong Kong ICT Awards 2012. His work on indoor positioning and IoT also received the Gold Medal at the International Trade Fair Ideas and Inventions New Products (iENA) in Germany in 2019, and the Gold Medal with the Organizer's Choice Award in the International Invention Innovation Competition in Canada (iCAN) in 2020. He is currently an Associate Editor for the IEEE Access and IEEE Transactions on Circuit and Systems II, and was the TPC Co-Chair for the PERSIST-IoT Workshop in conjunction with ACM MobiHoc 2019 and IEEE INFOCOM 2020.