A Machine Learning-Based Channel Data Enhancement Platform for Digital Twin Channels

AI Bo[1], ZHANG Yuxin[1], YANG Mi[1], HE Ruisi[1], GUO Rongge[2]

(1. School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China;
2. School of Traffic and Transportation, Beijing Jiaotong University, Beijing 100044, China)

DOI: 10.12142/ZTECOM.202502004

https://kns.cnki.net/kcms/detail/34.1294.TN.20250523.1012.002.html, published online May 23, 2025

Manuscript received: 2025-03-17

Abstract: Reliable channel data helps characterize the limitations and performance boundaries of communication technologies accurately. However, channel measurement is highly costly and time-consuming, and taking actual measurement as the only channel data source may reduce efficiency because of the constraints of high testing difficulty and limited data volume. Although existing standard channel models can generate channel data, their authenticity and diversity cannot be guaranteed. To address this, we use deep learning methods to learn the attributes of limited measured data and propose a generative model based on generative adversarial networks to rapidly synthesize data. A software simulation platform is also established to verify that the proposed model can generate data that are statistically similar to the measured data while maintaining necessary randomness. The proposed algorithm and platform can be applied to channel data enhancement and serve channel modeling and algorithm evaluation applications with urgent needs for data.

Keywords: channel measurement; channel modeling; deep learning; data enhancement

Citation (Format 1): AI B, ZHANG Y X, YANG M, et al. A machine learning-based channel data enhancement platform for digital twin channels [J]. ZTE Communications, 2025, 23(2): 20 – 30. DOI: 10.12142/ZTECOM.202502004

Citation (Format 2): B. Ai, Y. X. Zhang, M. Yang, et al., "A machine learning-based channel data enhancement platform for digital twin channels," *ZTE Communications*, vol. 23, no. 2, pp. 20 – 30, Jun. 2025. doi: 10.12142/ZTECOM.202502004.

1 Introduction

In recent years, the 6G wireless communication technology has attracted widespread attention, and many institutes have officially started the 6G research[1]. With the expansion of 6G to full-scenario, multi-frequency, and wide-coverage applications, the demands for 6G mobile communications are becoming more diversified and complicated. As a signal transmission medium, wireless channels are an indispensable part of communication links, and their characteristics determine the upper limit of communication system performance. A channel model is a mathematical description of the key channel characteristics, so channel modeling is a basis for the design, simulation, and planning of wireless communication systems.

The goal of channel research is to provide a model that can generate channel parameters according to the input parameter set. This model can be a mathematical model based on statistical fitting, such as the common empirical statistical model[2] and the geometric stochastic model[3]. For example, Ref. [4] conducted statistical modeling of features such as arrival time and power of multipath components, ensuring they conform to specific distributions. Ref. [5] introduced a geometric multiple-input multiple-output (MIMO) channel model for millimeter-wave (mmWave) mobile-to-mobile (M2M) applications, using a few clusters placed on two rings centered on the transmitter and receiver. In addition, the deterministic model based on numerical analysis and simulation is another channel modeling idea[6]. For example, the classic Longly-Rice model[7] uses a two-ray interference approach from geometric optics to predict radio wave propagation characteristics within the line-of-sight region. Ref. [8] investigated the channel characteristics of massive MIMO systems in the 26 GHz mmWave band for indoor scenarios using ray-tracing (RT). The simulation results are consistent with the measured results. With the

This work was supported by the National Key R&D Program of China under Grant No. 2023YFB2904802 , National Natural Science Foundation of China under Grant Nos. 62301022, 62221001, 62431003, and 62101507, Young Elite Scientists Sponsorship Program by CAST under Grant No. 2022QNRC001, and Program for Science & Technology R&D Plan Joint Fund of Henan Province under Grant No. 225200810112.
The corresponding author is YANG Mi.

expansion and application of artificial intelligence (AI) methods such as deep learning, researchers have proposed some AI-based channel models that use neural networks instead of traditional mathematical formulas and numerical simulations to generate channel parameters. Typical examples are found in Refs. [9 – 11]. Specifically, Ref. [9] used the convolutional autoencoder to extract 3D-building information to assist path loss prediction in street canyon scenarios. Ref. [10] employed convolutional neural networks to predict channel path loss using receiver-centric satellite maps as environmental features. Ref. [11] used a multilayer perceptron and long short-term memory (LSTM) to estimate real-time channel attenuation at Q-band. For a further overview of the existing classical modeling methods, please refer to Refs. [12 – 14].

No matter what the specific modeling method is, a consensus is that the channel model is essentially a mapping relationship. Although the model input attributes can be heterogeneous data such as scene category labels, antenna heights, three-dimensional models, and satellite images, the mapping relationship is generally between the environment and the corresponding channel parameters. The question worth considering here is whether these channel models, which we can collectively call environment-driven models, are the only solutions to channel research, in other words, whether these models can solve all the demands for channel data at present. For most application requirements, such as network deployment and coverage prediction, it is meaningful to input the necessary environmental characteristics to get the channel parameters of the corresponding input scene. However, it should not be forgotten that channel data are not only applied to environment-related applications. In other words, the existence of environment input should not be a prerequisite for generating channel data. For example, after obtaining some measured channel data through expensive and time-consuming actual measurement, researchers want to get more data under the same conditions conveniently. Another similar situation is that an algorithm needs to use a lot of real channel data to evaluate its performance, but the existing data are insufficient. The above two hypothetical situations are real cases in research and engineering practice. At the moment, the classic environment-driven model cannot meet all the requirements. Faced with these situations, researchers may need a so-called data-driven channel model, which can learn the characteristics of a small number of existing data and output a large number of similar data. Alternatively, it can be interpreted as a digital twin model, which constructs a virtual copy of the real physical channel, and this "copy" has the same statistical characteristics as the original data. In a word, this data-based modeling process, which does not depend on environmental input, can be called Channel Data Enhancement. It has significant practical value in some application scenarios.

At present, there have been several studies on data-driven channel models. As the groundbreaking work, Ref. [15] introduced the use of generative adversarial networks (GAN) to address autonomous channel modeling. Building on this, the GAN model was utilized to learn the distribution of additive white Gaussian noise channels. Ref. [16] developed a link-level MIMO channel generation method named ChannelGAN to support deep learning-based channel state information (CSI) feedback research. For different scenarios, Ref. [17] proposed a GAN-based channel data augmentation algorithm for communication systems in industrial Internet of Things (IIoT) scenarios to address the issue of insufficient data. Ref. [18] performed the GAN model to generate channel responses to address the issue of inadequate channel estimation performance in high-speed train scenarios. However, despite these efforts, some shortcomings still exist. Most studies rely on idealized simulated channel data, whereas measured data can more accurately capture various interference factors presented in real-world environments. Developing channel models based on measurement can enhance their credibility. Additionally, whether the channel characteristics described by these models are consistent with real data has not yet been comprehensively validated. Therefore, this paper proposes a channel data enhancement platform, the core capability of which is to quickly generate a large number of simulation data with similar characteristics based on a small number of data. Specifically, the platform consists of three subsystems: the channel measurement subsystem, which is used to collect the measured data and construct the basic data set; the data enhancement algorithm, which provides a model that can learn the characteristics of the data set and output the simulation data; the application software, which integrates the algorithm and necessary control functions to provide a convenient interface for users.

The rest of this paper is organized as follows. Section 2 describes the proposed channel data enhancement platform design and architecture. Section 3 is about the subsystems related to channel measurement and the data set in the platform. Following that, we explain the proposed data enhancement algorithm in Section 4. The algorithm verification and application software are described in Section 5. Finally, Section 6 concludes the paper.

2 Platform Design and Architecture

In this paper, a channel data enhancement platform is implemented, which can complete channel measurement in a high dynamic scene and then use the proposed algorithm to learn and measure channel characteristics, greatly expanding the number of channel data. The overall design and architecture of the proposed platform are shown in Fig. 1.

The platform is divided into three subsystems:

1) Channel measurement subsystem

Based on the software-defined radio instrument, this subsystem realizes broadband channel sounding. The subsystem can be applied to dynamic scenarios covering the sub-6 GHz frequency band. The measurement subsystem contains a sepa-

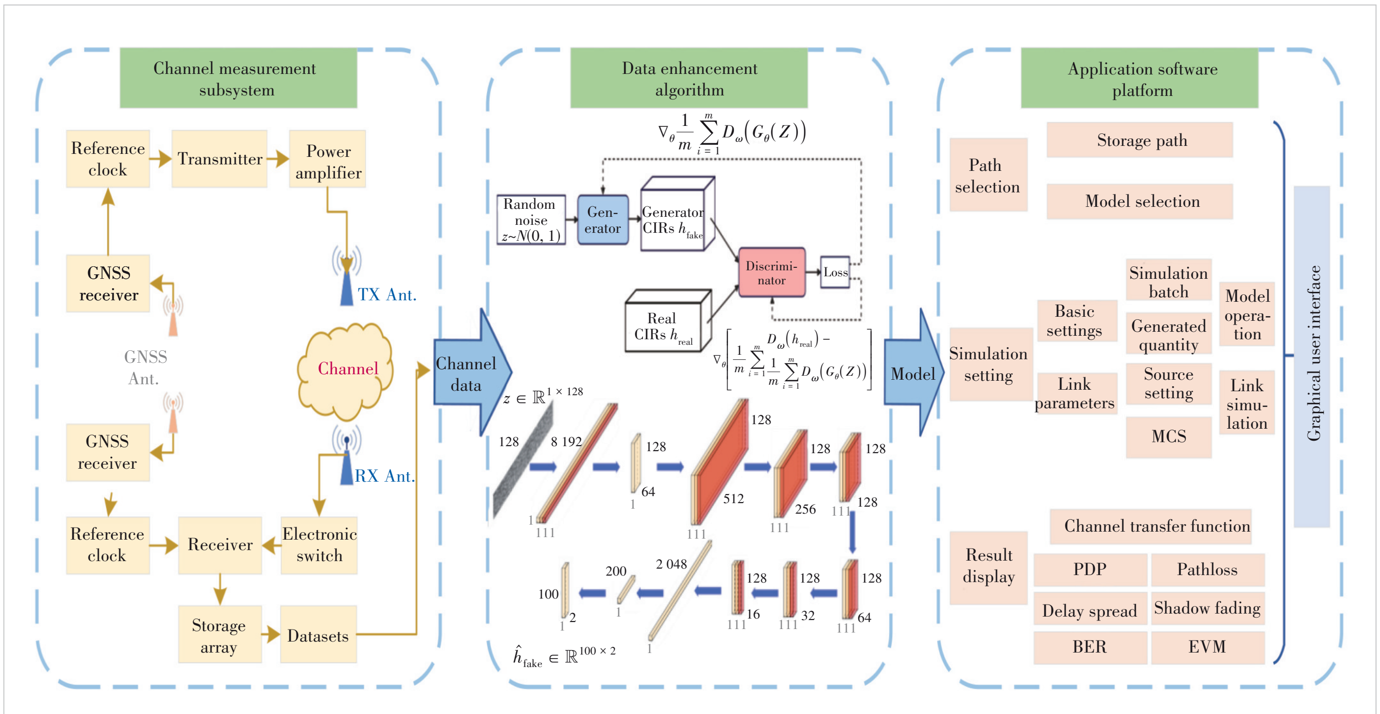


Figure 1. Proposed channel data enhancement platform architecture

rate transmitter and a receiver, which can process and display the collected signals in real time. In the dynamic scene, the back-to-back calibration can eliminate the influence of system response errors of cables and transceivers.

Due to the diversity of measurement scenarios, the core functional indicators of the measurement subsystem need to be defined by software. This can be scalable enough to meet the needs of different measurement environments. Specifically, the subsystem transmitter needs to complete baseband signal generation, power amplification, signal processing, and visual display. The receiver needs to complete signal reception, baseband signal processing, channel coefficient extraction, channel parameter analysis, visual display, and others. The overall structural design is complex and needs to be adapted to the cooperation on different hardware devices. Therefore, the software and hardware design and development of the measurement subsystem is one of the main difficulties in the whole platform implementation process. The measurement system program is flexible and can be migrated to different SDR hardware. The hardware configuration can be flexibly combined according to the requirements of the actual environment.

2) Data enhancement algorithm

The data enhancement algorithm needs to use the measured channel impulse response (CIR) obtained by the measurement subsystem. Then, the channel simulator based on GAN is trained to learn the intrinsic characteristics of measured data. The proposed method reduces the complexity of channel simulation and can quickly generate channel data by using the trained model. In addition, the accuracy of generated channel data is verified by channel high-order statistical characteristics, such as power delay profile, shadow fading, and delay spread.

The key point of subsystem algorithm design is to learn the characteristics of measured data. However, with the increase of measurement bandwidth, the time delay resolution of data becomes higher. In addition, CIRs are composed of multiple ray clusters, which contain a lot of noise signals. Therefore, the prime difficulty in data preprocessing is to denoise and reduce the dimension of the CIR matrix while retaining effective information as much as possible. Furthermore, the model network structure includes the number and types of networks, the logical relationship between networks, and others. These architectures directly affect the complexity and, more importantly, the accuracy. In addition, the appropriate training algorithm should be carefully selected for the specific network structure. Architecture and training are the key control factors of model performance.

3) Application software

After verifying the channel simulation ability of the model through experiments, the focus shifts to building a convenient software platform. This paper designs an easy-to-operate simulation application program based on MATLAB, which can complete the functions of model loading, simulation data generation, real-time verification, display, and data storage. In addition, a visual interface is designed.

3 Channel Measurement and Datasets

The broadband wireless channel measurement subsystem includes a separate transmitter and a separate receiver. The transmitter consists of a baseband signal source, a local oscillator, an up-converter, a power amplifier, a filter, a signal processing unit, and an antenna. The receiver is composed of a baseband signal source, a local oscillator, a down-converter, a low noise amplifier, an electronic switch, a data storage unit, and an antenna[19]. The transmitter sends a signal at a specific carrier frequency to act as a sounding signal, and the receiver can identify and detect the signal after channel attenuation and distortion.

The channel measurement subsystem uses multi-carrier signals as sounding signals, as shown in Eq. (1).

$$s_k = \sum_{i=0}^{L-1} d_i \exp\left(\frac{j2i\pi k}{L}\right) \quad (0 \leqslant k \leqslant L-1) \tag{1}$$

where L represents the number of subcarriers and d_i represents the symbol of each subcarrier. The out-of-band power is reduced by rectangular window function filtering. At the receiver, the received signal is shown in Eq. (2).

$$Y(f) = X(f)H_{TX}(f)H(f)H_{RX}(f) \tag{2}$$

where $X(f)$ and $Y(f)$ represent the transmitted and received signals in the frequency domain respectively. $H(f)$ represents the channel transfer function, and $H_{TX}(f)$ and $H_{RX}(f)$ are the transfer functions of equipment and cables at the transmitter and receiver respectively. The transmitter and receiver are directly connected by cables for back-to-back calibration, so the influence of equipment and cables on the measurement results can be eliminated.

The measurement subsystem takes the signal transceiver based on software-defined radio (SDR) as core hardware. The transmitter implements the loading and generation of baseband sounding signals and the up-conversion of the baseband signals through secondary frequency conversion (baseband to intermediate frequency and intermediate frequency to radio frequency). The receiver samples and down-converts the signals captured by the antennas to obtain the baseband signals and stores them in the local disk. This subsystem realizes hardware device driving and signal processing, and finally obtains key channel parameters and displays them visually. The receiver and transmitter of this subsystem use a rubidium atomic clock calibrated by the global navigation satellite system (GNSS) as the reference clock source to ensure the consistency of the 10 MHz reference clock[20]. Main parameters of the subsystem are shown in Table 1, and the equipment is shown in Fig. 2.

Table 1. Parameters of measurement subsystem

Parameter	Value
Carrier frequency	5.9 GHz
Bandwidth	Max to 160 MHz
Transmit power	Max to 55 dBm
Transmit signal type	Multi-carrier signals
Transmit signal samples	1 024
Snapshot interval	6.4 μs

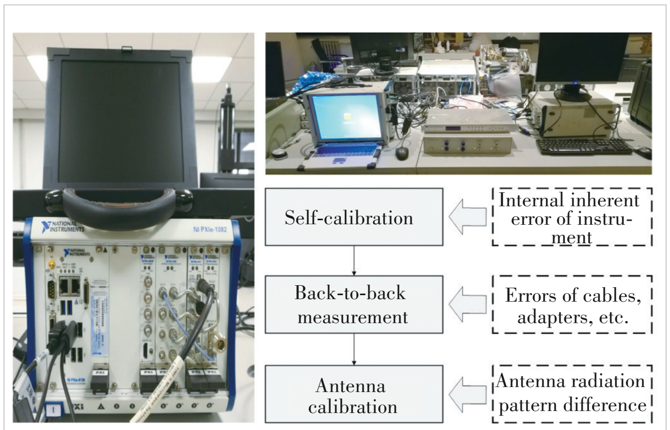
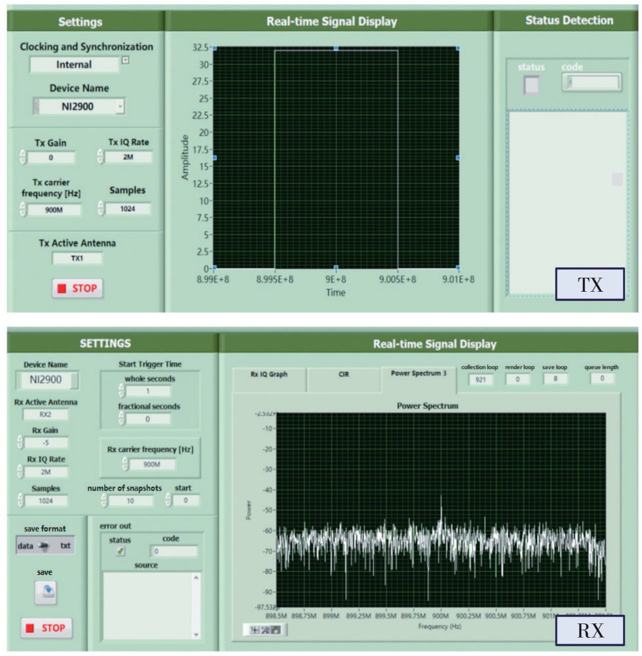


Figure 2. Vector signal transceiver equipment

The software of the channel measurement subsystem is developed based on LabVIEW. LabVIEW is a program development environment developed by National Instruments (NI), which is well compatible with SDR-based signal transceivers used in subsystems and can also easily establish a visual interface. The main purpose of subsystem software is to drive and control the hardware. The software design should be able to call the hardware equipment, configure the measurement parameters such as frequency, bandwidth, clock, and sampling rate of the equipment, and ensure that the received signal data can be stored[21].

The subsystem also provides a visual user interface for testers, as shown in Fig. 3. The interface includes the configuration of various parameters, system running state detection, and error reporting. To observe the channel state in real time during the measurement process, the subsystem also processes some collected data in real time and gets typical channel parameters. In Fig. 3, the receiver interface shows the CIR at the current time. Besides CIR, the current time domain waveform chart and frequency spectrum chart can be displayed in real time.

The original response obtained by the subsystem includes the channel response, the inherent response of the measurement system, and the antenna radiation characteristics. Therefore, system calibration verification is needed to eliminate the errors caused by these factors. As shown in Fig. 2, the calibration verification of the subsystem is divided into three parts:



RX: receiver TX: transmitter

Figure 3. Visual interface of channel measurement subsystem

instrument self-calibration, back-to-back measurement calibration, and antenna system calibration.

The purpose of instrument self-calibration is to make the performance and output of the instrument conform to the nominal value. The transmitter used in this paper has a self-calibration function, and subsequent measurements can only be started after the self-calibration has passed before each measurement. Back-to-back measurement can eliminate the errors caused by cables and adapters. The specific method can be summarized as follows. The reference measurement is conducted when the channel response is known by connecting the attenuator directly between the transmitter and the receiver. Thus, the system's inherent response is obtained. During the actual measurement data processing, the collected data are processed using these reference measurement results to eliminate the inherent response of the system and then get the accurate channel response. Antenna calibration refers to the measurement of antenna gain in all propagation directions in an anechoic chamber, which is an important prerequisite to ensure the accuracy of test results. The measurement error from antenna radiation can be eliminated when processing the received data.

The experimental study on channel measurement in this paper was carried out in Beijing, China. During the field measurements, the transmitter and receiver vehicles moved in the same direction and kept an interval of 20 – 40 m. During the measurement period, the maximum vehicle speed was no more than 70 km/h, and the system acquired 16-channel snapshots per second. To reduce the influence of random backscatters, measurement routes were restricted to empty road sections. Both the transmitting and receiving antennas were installed on the roof, and the antenna heights were about 1.8 m. The total number of measured channel snapshots was about 7 000 groups.

4 Proposed Data Enhancement Algorithm

4.1 GAN-Based Algorithm

GAN is a kind of deep generation model, which can implicitly learn the probability distribution of input images to generate identically distributed images. Initially developed for image generation, GAN is not a simple method for copying or imitating reality, nor does it merely blend or average multiple real samples. Instead, it uses two game-theoretic neural networks, namely the generator (G-network) and the discriminator (D-network), to learn intrinsic statistical patterns of real data, without direct objective functions.

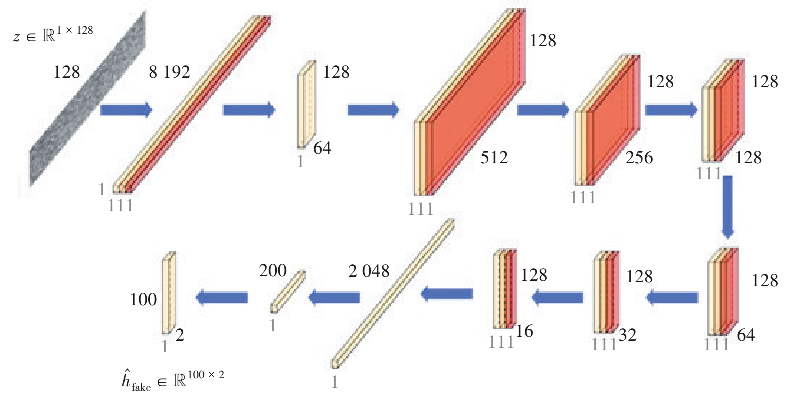
G-network is used to learn the distribution of real data to generate identically distributed data, and D-network judges the probability whether its input data comes from reality or generation. Through training, the purpose of the generator is to gradually generate realistic data to deceive the discriminator. Discriminators want to always be able to distinguish between real and generated data. Therefore, the essence of GAN is to make the generator learn the approximate value of real data distribution through antagonistic learning.

GAN usually has some problems in training, such as mode collapse, unstable optimization, gradient disappearance, and non-convergence. To avoid the above problems, this paper uses Wasserstein GAN with gradient penalty (WGAN-GP) as the network framework, which is an improved version of GAN. Wasserstein distance, also known as the Earth-Mover (EM) distance, is used to evaluate the similarity between two distributions, which can provide a relatively stable gradient relative to Jensen-Shannon (JS) divergence. GP can avoid the problem of gradient disappearance caused by large model weights. Therefore, WGAN-GP is more stable and converges faster in training and can significantly improve the training speed and address the slow convergence issue in original WGAN.

4.2 Algorithm-Based Model Design

4.2.1 Generator Design

Fig. 4 illustrates the network architecture and detailed parameters of the generator in this algorithm. The model takes noise vector as input and generates CIR through the generator that uses one-dimensional convolution to extract features. The convolution layer can create a convolution kernel, and the input of this layer is rolled up in a single space (or time) dimension to produce the output. The convolution kernel size in the generator is set to 3. Subsequently, the batch normalization layer is added behind each convolution layer, which acceler-



Layer Type	Output Shape
Input layer	(128, 1)
Dense & batche normalization & LeakyReLU	(8 192, 1)
Reshape	(128, 64)
Conv ID & batche normalization & LeakyReLU	(128, 512)
Conv ID & batche normalization & LeakyReLU	(128, 256)
Conv ID & batche normalization & LeakyReLU	(128,128)
Conv ID & batche normalization & LeakyReLU	(128, 64)
Conv ID & batche normalization & LeakyReLU	(128, 32)
Conv ID & batche normalization & LeakyReLU	(128, 16)
Flatten	(2 048, 1)
Dense	(200, 1)
Reshape	(100, 2)

LeakyReLU: Leaky Rectified Linear Unit

Figure 4. Generator network design and detailed parameters

ates the convergence speed of model training. It also makes the model training process more stable to avoid gradient explosion or gradient disappearance. In addition, this paper chooses Leaky Rectified Linear Unit (LeakyReLU) as the activation function to alleviate the problem of gradient disappearance. The expression of LeakyReLU is shown in Eq. (3).

$$\mathrm{LeakyReLU}(x) = \begin{cases} x, \ x \geqslant 0 \\ \alpha \cdot x, \ \text{otherwise} \end{cases} \tag{3}$$

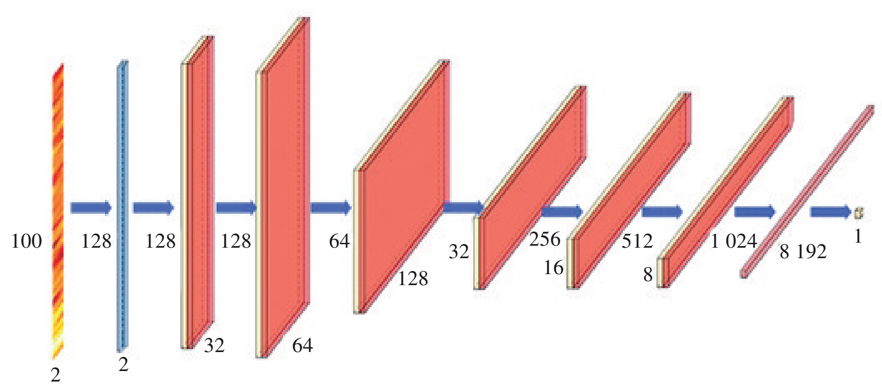
where x is the input of LeakyReLU. When $x < 0$, LeakyReLU gives x a slope α. Parameter α is an adjustable superparameter, and the value set in this paper is 0.2. Because Tanh can limit the output to $[-1, 1]$, the generated CIR better matches with the real CIR amplitude. Therefore, Tanh is selected as the activation function after the last convolution layer, and its expression is shown in Eq. (4).

$$\mathrm{Tanh}(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{4}$$

where x is the input of Tanh. When the input noise passes through six convolution layers, it will pass through the Flatten layer, and the result will be mapped into a separable space in combination with the fully connected layer. The fully connected layer maps the learned features to the sample label space. Since the generator finally outputs the CIR, it is necessary to reshape the samples passing through the fully connected layer.

4.2.2 Discriminator Design

Fig. 5 shows the network architecture and detailed configuration of the discriminator. The input of the discriminator is the CIR sample generated by the generator or the real CIR sample. The input channel samples are first zero-padded to facilitate the subsequent convolution process. Similar to the generator, the discriminator mainly uses one-dimensional convolution and LeakyReLU activation function. The convolution kernel size of the one-dimensional convolution is 5. Finally, it is output through the Flatten and fully connected layers. The output of the discriminator is the probability while the input is a



Layer Type	Output Shape
Input layer	(100, 2)
Zero padding ID	(128, 2)
Conv ID & LeakyReLU	(128, 32)
Conv ID & LeakyReLU	(128, 64)
Conv ID & LeakyReLU	(64,128)
Conv ID & LeakyReLU	(32, 256)
Conv ID & LeakyReLU	(16, 512)
Conv ID & LeakyReLU	(8, 1 024)
Flatten	8 192
Dense	1

LeakyReLU: Leaky Rectified Linear Unit

Figure 5. Discriminator network design and detailed parameters

real channel sample or a generated channel sample.

5 Algorithm Verification and Application Software

5.1 Algorithm Verification

Algorithm implementation consists of model design, model training, and CIR sample generation. The training process follows an alternating scheme, where the discriminator is updated multiple times per generator update to ensure stable convergence. The Adam optimization algorithm is employed to update the parameters of the GAN network with a learning rate of 0.000 05. Upon completing 2 500 training epochs, the trained model is saved. Then, in the generation process, the saved model is used to generate CIR by inputting the desired number of CIR samples along with a 128-dimensional random noise vector.

In this section, the similarity between the real and generated channels is demonstrated by comparing the distribution performance of the power delay profile (PDP), path loss, and root mean square (RMS) delay spread between the measurement and generated data. To facilitate accurate evaluation against real channels, this paper generates channel samples equal in number to the real ones.

Figs. 6a and 6b illustrate the channel PDP obtained through actual measurements and GAN generation, respectively. It can be seen that GAN-generated PDP closely matches the measured data in terms of morphology, especially aligning with the peak positions in the delay domain observed in the measurements. Additionally, the generated channels preserve the diversity, randomness, and noise-affected characteristics of real channels, demonstrating high fidelity. Fig. 6c presents a comparison of the averaged PDP. Specifically, when calculating the PDP, the samples are averaged according to the number of samples, as shown in Eq. (5).

$$\mathrm{PDP} = \frac{1}{N}\sum_{N}\left|h(n,\tau)\right|^2 \tag{5}$$

where N is the total number of channel samples, h represents the measured or generated CIR, n is the sample index corresponding to the number of delay points, and τ refers to the delay points.

For the real channel, the average PDP is depicted by the black curve in Fig. 6c. The average PDP of the channels generated by the AI model after 2 500 training iterations is shown by the blue dashed line with square markers. For comparison, channels generated by the model after training for 20 epochs are included, with PDP illustrated by the purple solid line

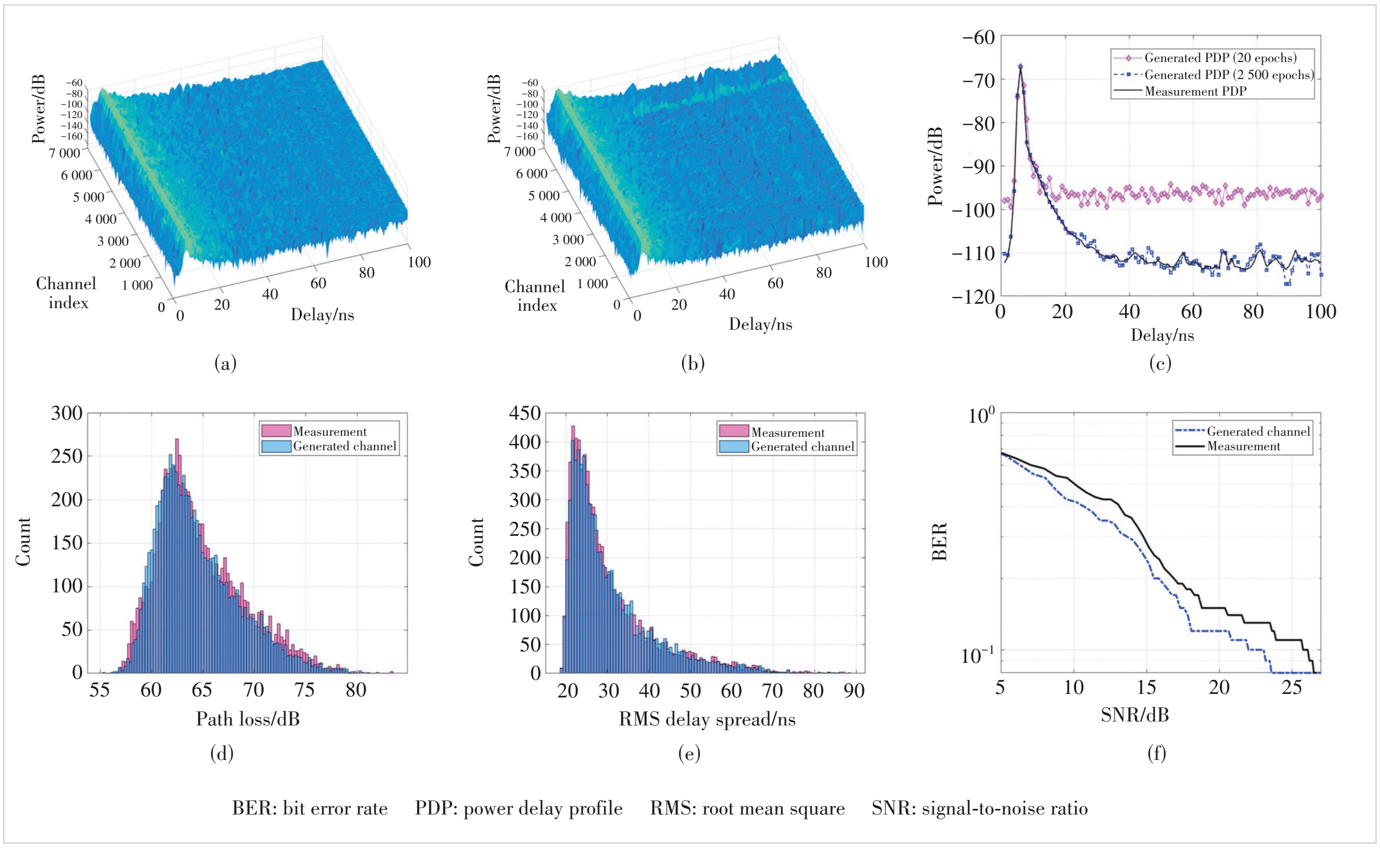


Figure 6. Algorithm verification results: (a) measured PDP; (b) generated PDP; (c) PDP comparison; (d) path loss; (e) RMS delay spread; (f) BER performance

with diamond markers in Fig. 6c. It can be observed that the channel power distribution generated by the model trained for 20 epochs ranges in [−100 dB, −60 dB], whereas the power distribution of both the real channel and the channel generated by the model trained for 2 500 epochs spans from −120 dB to −60 dB. This discrepancy arises from insufficient training, which prevents the model from fully capturing channel characteristics and distribution. As a result, the generated channel data lack multipath details and exhibit higher noise power. Channels generated by a high-performing GAN model closely resemble the real channels, including the transition of the PDP from peak values to a gradual stabilization.

Further validating the distribution of channel parameters is crucial for evaluating model performance. Path loss is used to characterize the power loss that occurs during signal transmission, which is an important parameter for evaluating signal coverage area and quality in wireless communication systems. It can be calculated using PDP, as shown in Eq. (6).

$$\mathrm{PL} = \frac{1}{N_\tau}\left(\sum_{N_\tau}\left|h(n,\tau)\right|^2\right) \tag{6}$$

where N_τ denotes the number of delay points, and h represents the measured or generated channel. Fig. 6d illustrates the path loss distributions for both the measured and generated data. It is evident that the generated data (blue histogram) exhibits a high degree of overlap with the measurement (red histogram) in terms of path loss. Meanwhile the mean path loss values for the measured and generated channels are 64.80 dB and 64.47 dB, further demonstrating the high similarity between the generated and real channels.

RMS delay spread is used to describe the degree of delay dispersion in a channel, which reflects the impact of the delay distribution of each propagation path on the received signal in a multipath propagation environment. RMS delay spread can be calculated as follows:

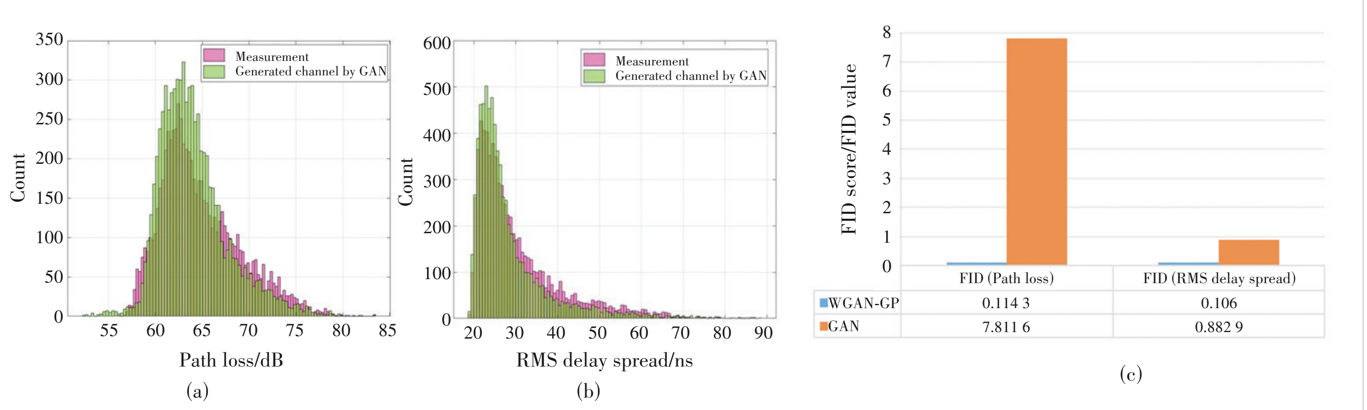
$$\tau_{\mathrm{RMS}}(n) = \sqrt{\frac{\sum_{\tau_N}\tau(n)^2\mathrm{PDP}(n,\tau)}{\sum_{\tau_N}\mathrm{PDP}(n,\tau)} - \bar{\tau}(n)^2} \tag{7}$$

where τ_N represents the delay component of the N-th channel sample, $\bar{\tau}(n)$ refers to the average delay. $\bar{\tau}(n)$ is calculated as:

$$\bar{\tau}(n) = \frac{\sum_{\tau_N}\tau(n)\mathrm{PDP}(n,\tau)}{\sum_{\tau_N}\mathrm{PDP}(n,\tau)} \tag{8}$$

Comparing the histograms displaying the RMS delay spread distributions of the measured and GAN-generated channels, Fig. 6e shows that both channels exhibit a high degree of consistency in their distribution shapes and ranges. Additionally, the mean RMS delay spreads for the measured and the generated channels are 30.65 ns and 30.74 ns, further validating the similarity between the two channel distributions. This also confirms the strong performance of the GAN model in capturing the channel delay characteristics.

Furthermore, the generation performance of the standard GAN model is further compared and evaluated. Figs. 7a and 7b present the statistical distributions of path loss and RMS delay spread from the generated channel by the GAN model. By comparing these results with the WGAN-GP performance in Fig. 6, it is evident that WGAN-GP achieves better alignment between the statistical characteristics of the generated channel data and those of the measured channel. Fig. 7c provides a quantitative assessment of the fidelity of the generated channel data using the Frechet Inception Distance (FID) metric. The results indicate that the WGAN-GP model achieves significantly lower FID scores (0.114 3 for path loss and 0.106 for RMS delay spread) compared to the standard GAN model



FID: Frechet Inception Distance GAN: generative adversarial network RMS: root mean square WGAN-GP: Wasserstein GAN with gradient penalty

Figure 7. Standard GAN model generation performance: (a) path loss; (b) RMS delay spread; (c) FID comparison

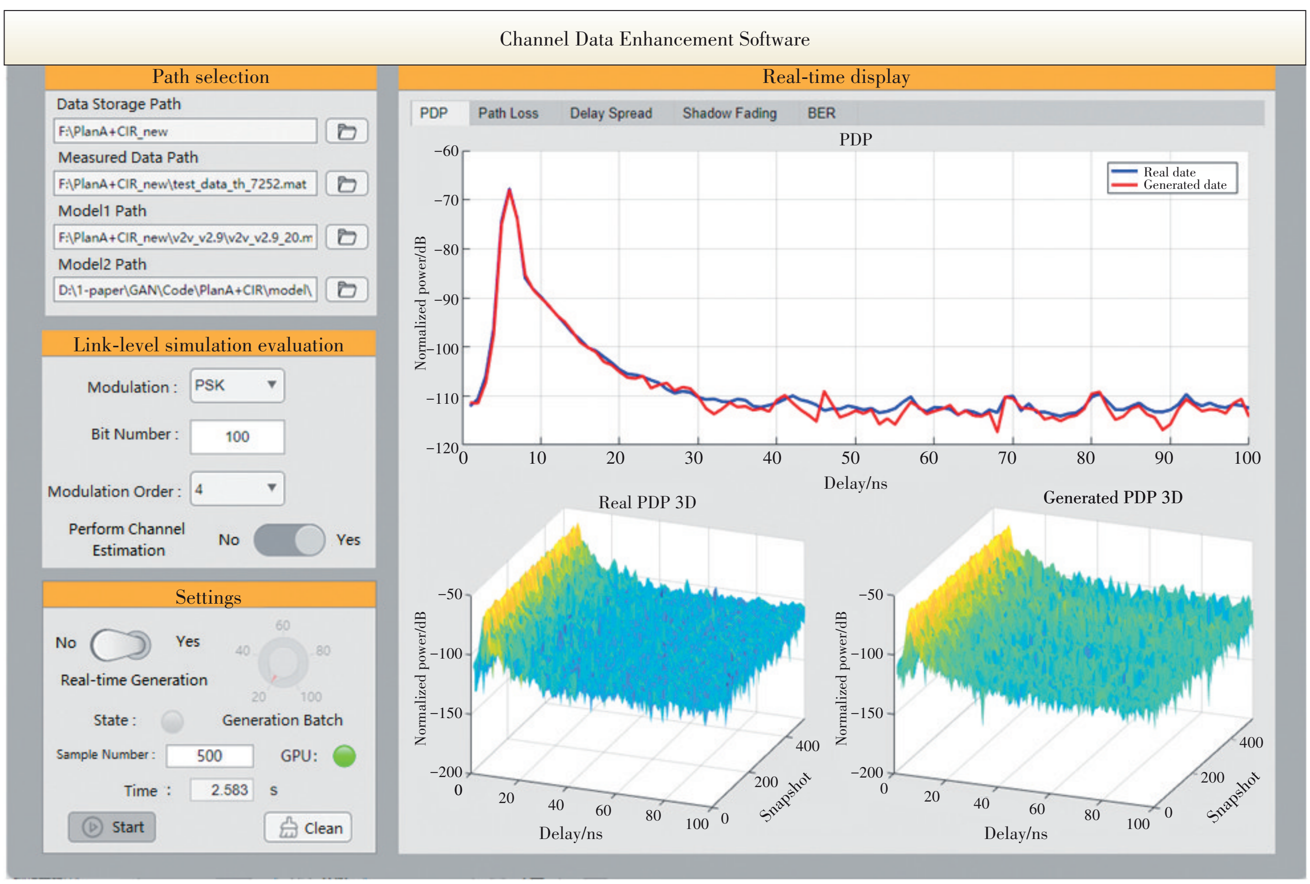
(7.811 6 and 0.882 9, respectively). This demonstrates that WGAN-GP is capable of generating channel data with higher fidelity, ensuring a closer match to the statistical characteristics of measurements.

To validate the effectiveness of GAN-generated channels, a simplified link-level simulation was conducted for evaluation, which employed phase shift keying (PSK) modulation with 100 transmitted bits and a modulation order set to 16. Fig. 6f compares the bit error rate (BER) of the real channel with that of the GAN-generated channel, where the BER curves of the real and GAN-generated channel are highly consistent, exhibiting similar BER trends under different signal-to-noise ratio (SNR) conditions. This high level of similarity indicates that the GAN-generated channel can effectively simulate the real channel in terms of error performance.

5.2 Application Software

The main function of the application software is to generate channels by using the previous algorithm, and the visual interface is shown in Fig. 8. The software can be divided into two sub-functions: one-time channel generation and uninterrupted real-time channel generation. The former can generate a specified number of channel data at one time. In addition, the software can track the duration record generated by the channel. When generating channels in real time, the function of selecting generation batches is added. If the generation batch is selected, the channel can be generated in real time according to the batch size, and the dynamic generation process of the channel and the dynamic distribution of the channel parameters can also be seen on the visualization panel. After the dynamic generation of the channel is completed, the software will detect the end of the generation and turn the indicator light green as a prompt.

Fig. 9 shows the operation flow of the software. First, the path needs to be set, including selecting measurement data and generating models. The path to store the generated channel data should also be configured. Next, the options of link simulation are configured. The modulation mode can be PSK or quadrature amplitude modulation (QAM), and the modulation order can be 4 or 16. In link simulation, we can choose



BER: bit error rate PDP: power delay profile

Figure 8. Channel data enhancement application software interface

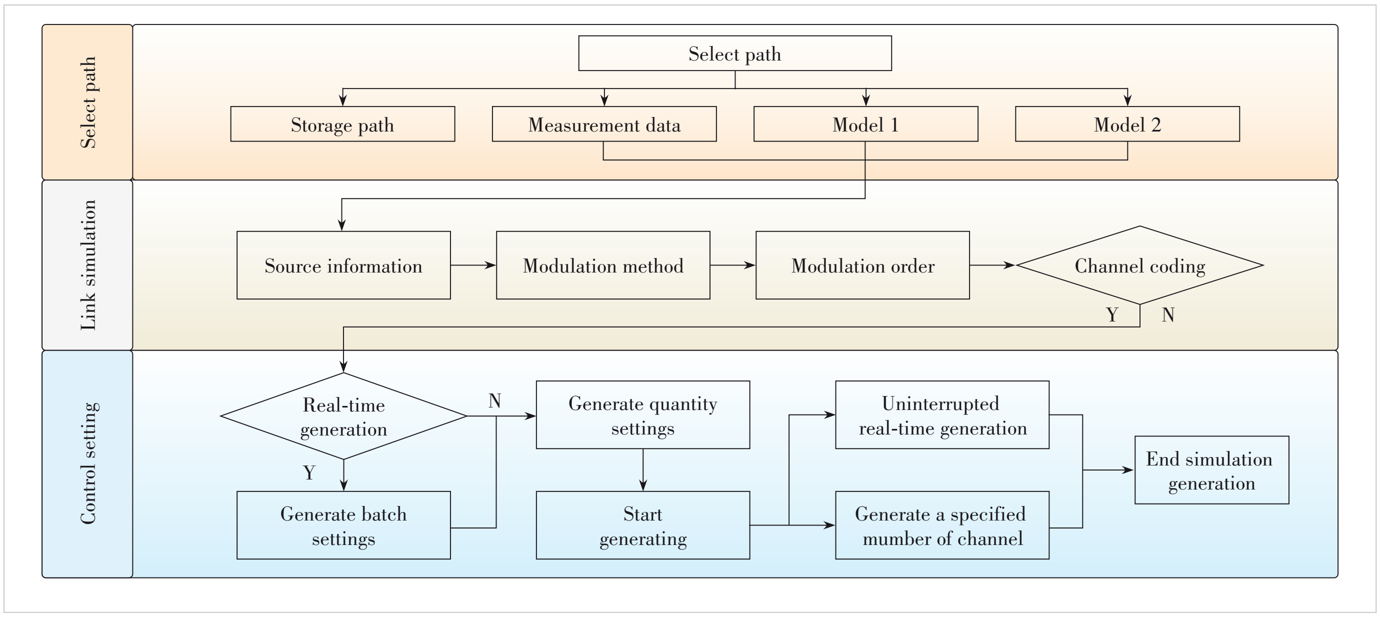


Figure 9. Operation flow chart of channel enhancement software

whether to perform channel estimation or not. Then, we should choose whether to generate channels in real time in the basic settings. If not, the software will generate all the channel data with the set number of channels at one time. If it is real-time generation, we need to further set the generation batch; the software will subsequently generate channel data according to the set generation batch, and the display window will dynamically display the whole channel generation process. Finally, after clicking the Start button, the software will initiate the generation of channel data based on the specified configuration. Clicking the Clear button will then clear the contents of the display window, allowing the settings to be reset for generating channel data under the new configuration.

6 Conclusions

Channel characteristics and models are the basis of communication system design and evaluation. Meanwhile, it has been a consensus that channel data is the support of channel research and modeling. To address the current issue of challenging channel data acquisition, this paper proposes a channel data enhancement platform based on the idea of a digital twin channel. The platform includes three key subsystems: channel measurement, enhancement algorithm, and application software. The measurement subsystem is a broadband dynamic channel measurement system based on the SDR architecture, which can complete channel data acquisition in the sub-6 GHz frequency. The channel enhancement algorithm, the core of the proposed platform, is a neural network based on the GAN architecture. It can learn the intrinsic characteristics of real channel data and quickly generate a large number of highly similar simulation channels. We verify and evaluate the generated channel under the high-order characteristics of power delay profile, path loss, shadow fading, and root mean square delay spread. The results show that the generated channel is similar to the original channel in statistical characteristics and has sufficient randomness. Finally, the platform includes integrated software for engineers and researchers, which can call the above algorithm and generate channel data in real time. The result of this paper is a potential channel modeling and simulation methodology.

References

[1] LU K, HAN B M, ZHOU X S. Smart urban transit systems: from integrated framework to interdisciplinary perspective [J]. Urban rail transit, 2018, 4(2): 49–67. DOI: 10.1007/s40864-018-0080-x
[2] HATA M. Empirical formula for propagation loss in land mobile radio services [J]. IEEE transactions on vehicular technology, 1980, 29(3): 317–325. DOI: 10.1109/T-VT.1980.23859
[3] ZAJIC A G, STUBER G L, PRATT T G, et al. Wideband MIMO mobile-to-mobile channels: geometry-based statistical modeling with experimental verification [J]. IEEE transactions on vehicular technology, 2009, 58(2): 517–534. DOI: 10.1109/TVT.2008.928001
[4] SALEH A A M, VALENZUELA R. A statistical model for indoor multipath propagation [J]. IEEE journal on selected areas in communications, 1987, 5(2): 128–137. DOI: 10.1109/JSAC.1987.1146527
[5] HE R S, AI B, STÜBER G L, et al. Geometrical-based modeling for millimeter-wave MIMO mobile-to-mobile channels [J]. IEEE transactions on vehicular technology, 2018, 67(4): 2848–2863. DOI: 10.1109/TVT.2017.2774808
[6] PANG L H, ZHANG J, ZHANG Y, et al. Investigation and comparison of 5G channel models: from QuaDRiGa, NYUSIM, and MG5G perspectives [J]. Chinese journal of electronics, 2022, 31(1): 1–17. DOI: 10.1049/cje.2021.00.103
[7] WEINER M. Use of the Longley-Rice and Johnson-Gierhart tropospheric radio propagation programs: 0.02–20 GHz [J]. IEEE journal on selected

areas in communications, 1986, 4(2): 297 – 307. DOI: 10.1109/JSAC.1986.1146313

[8] AI B, GUAN K, HE R S, et al. On indoor millimeter wave massive MIMO channels: measurement and simulation [J]. IEEE journal on selected areas in communications, 2017, 35(7): 1678 – 1690. DOI: 10.1109/JSAC.2017.2698780

[9] GUPTA A, DU J F, CHIZHIK D, et al. Machine learning-based urban canyon path loss prediction using 28 GHz Manhattan measurements [J]. IEEE transactions on antennas and propagation, 2022, 70(6): 4096 – 4111. DOI: 10.1109/TAP.2022.3152776

[10] THRANE J, ARTUSO M, ZIBAR D, et al. Drive test minimization using deep learning with Bayesian approximation [C]//Proc. IEEE 88th Vehicular Technology Conference (VTC-Fall). IEEE, 2018: 1 – 5. DOI: 10.1109/VTCFall.2018.8690911

[11] BAI L, XU Q, HUANG Z W, et al. An atmospheric data-driven Q-band satellite channel model with feature selection [J]. IEEE transactions on antennas and propagation, 2022, 70(6): 4002 – 4013. DOI: 10.1109/TAP.2021.3137285

[12] YANG M, AI B, HE R S, et al. Dynamic V2V channel measurement and modeling at street intersection scenarios [J]. IEEE transactions on antennas and propagation, 2023, 71(5): 4417 – 4432. DOI: 10.1109/TAP.2023.3249101

[13] HUANG C, HE R S, AI B, et al. Artificial intelligence enabled radio propagation for communications—part II: scenario identification and channel modeling [J]. IEEE transactions on antennas and propagation, 2022, 70(6): 3955 – 3969. DOI: 10.1109/TAP.2022.3149665

[14] WANG C L, AI B, HE R S, et al. Channel path loss prediction using satellite images: a deep learning approach [J]. IEEE transactions on machine learning in communications and networking, 2024, 2: 1357 – 1368

[15] YANG Y, LI Y, ZHANG W X, et al. Generative-adversarial-network-based wireless channel modeling: challenges and opportunities [J]. IEEE communications magazine, 2019, 57(3): 22 – 27. DOI: 10.1109/MCOM.2019.1800635

[16] XIAO H, TIAN W Q, LIU W D, et al. ChannelGAN: deep learning-based channel modeling and generating [J]. IEEE wireless communications letters, 2022, 11(3): 650 – 654. DOI: 10.1109/LWC.2021.3140102

[17] LIANG X, LIU Z Y, CHANG H R, et al. Wireless channel data augmentation for artificial intelligence of things in industrial environment using generative adversarial networks [C]//Proc. IEEE 18th International Conference on Industrial Informatics (INDIN). IEEE, 2020, 1: 502 – 507. DOI: 10.1109/indin45582.2020.9442206

[18] ZHANG D Y, ZHAO J H, YANG L H, et al. Generative adversarial network-based channel estimation in high-speed mobile scenarios [C]//Proc. 13th International Conference on Wireless Communications and Signal Processing (WCSP). IEEE, 2021: 1 – 5. DOI: 10.1109/WCSP52459.2021.9613362

[19] ZHAO H, LYU Q, LIU Y C, et al. Wireless channel measurements and modeling of LTE broadband system for high-speed railway scenarios [J]. Chinese journal of electronics, 2018, 27(5): 1092 – 1097

[20] ZHANG Z Y, HE R S, AI B, et al. A cluster-based statistical channel model for integrated sensing and communication channels [J]. IEEE transactions on wireless communications, 2024, 23(9): 11597 – 11611. DOI: 10.1109/TWC.2024.3383594

[21] YANG M, AI B, HE R S, et al. Measurements and cluster-based modeling of vehicle-to-vehicle channels with large vehicle obstructions [J]. IEEE transactions on wireless communications, 2020, 19(9): 5860 – 5874. DOI: 10.1109/TWC.2020.2997808

Biographies

AI Bo received his MS and PhD degrees from Xidian University, China in 2002 and 2004, respectively. He was an Excellent Postdoctoral Research Fellow with Tsinghua University, China in 2007. He is currently a professor with the State Key Laboratory of Advanced Rail Autonomous Operation and School of Electronic and Information Engineering, Beijing Jiaotong University, China. He has authored or co-authored eight books and published over 300 academic research papers. His five papers have been recognized as ESI highly cited papers. His research interests include channel measurement and channel modeling, and dedicated mobile communications for rail traffic systems.

ZHANG Yuxin received her MS degree in information and communication engineering from Beijing Jiaotong University (BJTU), China in 2024. She is currently pursuing a PhD degree at the School of Electronics and Information Engineering, BJTU. Her current research focuses on artificial intelligence-based wireless channel modeling.

YANG Mi (myang@bjtu.edu.cn) received his MS and PhD degrees from Beijing Jiaotong University, China in 2017 and 2021, respectively. He is currently an associate professor with the State Key Laboratory of Advanced Rail Autonomous Operation and the School of Electronic and Information Engineering, Beijing Jiaotong University. His research interests include wireless channel measurement and modeling.

HE Ruisi received his BE and PhD degrees from Beijing Jiaotong University (BJTU), China in 2009 and 2015, respectively. He is currently a professor with the School of Electronics and Information Engineering, BJTU. He has been a visiting scholar at Georgia Institute of Technology, USA, University of Southern California, USA, and Université Catholique de Louvain (UCLouvain), Belgium. His research interests include wireless propagation channels, 5G, and 6G communications. He has authored/co-authored eight books, five book chapters, more than 200 journal and conference papers, and several patents.

GUO Rongge received her BS degree from Sun Yat-sen University, China in 2014, and MS and PhD degrees from Beijing Jiaotong University, China in 2017 and 2021, respectively. She is currently a lecturer with the School of Traffic and Transportation, Beijing Jiaotong University. Her research interests include public transport, intelligent transport systems (ITS), transport planning, transport management, and traffic control.