U.P.B. Sci. Bull., Series C, Vol. 87, Iss. 3, 2025 ISSN 2286 – 3540

FAST SHORT-TERM CHANNEL STATE PREDICTION BASED ON DEEP TRUST BLOCKCHAIN NETWORKS

Jun ZHU[1,*], Guoyin ZHANG[2]

As wireless communication technology evolves, precise and swift channel fading prediction is crucial for enhancing communication quality. Traditional methods, however, struggle with computational complexity and accuracy, failing to meet modern systems' real-time and reliability demands. Recently, deep learning has emerged as a promising solution to address the challenge of channel fading prediction. This paper introduces a novel approach, the Blockchain-based Deep Belief Network (BDBN), which leverages blockchain's decentralization, transparency, and security to ensure data integrity and trust during transmission. The BDBN aims to achieve rapid short-term channel fading prediction in wireless communications, thereby optimizing system performance. Meanwhile, the advantages of DBN in dealing with complex nonlinear problems are combined in order to model and predicti wireless channel fading. By training a large amount of historical channel data, the BDBN is able to learn the potential patterns of channel changes, thus realizing highly accurate short-term prediction. Experimental results demonstrate that the BDBN method outperforms traditional prediction methods, enhancing both accuracy and speed. In particular, BDBN exhibits stronger robustness and adaptability when dealing with bursty channel fading. In addition, the introduction of blockchain technology also effectively improves the level of data security and privacy protection, providing a strong guarantee for the secure transmission of wireless communication systems.

Keywords: channel prediction, deep learning, deep trust networks, blockchain

1. Introduction

In wireless communications, accurate channel state information (CSI) is vital for reliable data transmission. However, CSI is often difficult to obtain accurately due to multipath effects, noise, and other interfering factors. Although traditional channel estimation methods can provide a certain degree of CSI, the complex computation process and feedback delay limit their performance in practical applications. Hence, channel prediction techniques have been developed to anticipate future channel states, enabling the preemptive adjustment of transmission parameters and enhancing overall communication system security [1].

[1]* Harbin Engineering University, Harbin 150001, Heilongjiang, China, Corresponding author: e-mail: zj1qaz2wsx3edc@163.com
[2] Harbin Engineering University, Harbin 150001, Heilongjiang, China, e-mail: zhangguoyin@hrbeu.edu.cn

Recently, deep learning has shown exceptional performance across various domains, attributed to its powerful nonlinear processing capabilities. The Deep Belief Network (DBN), a significant model in deep learning, features multiple hidden layers and excels at capturing complex data features [2]. However, DBN has seen limited application in wireless communications, particularly in channel prediction. Additionally, existing channel prediction methods often lack transparency and security, making them susceptible to attacks and tampering.

Research has been conducted to obtain Channel State Information (CSI), leading to the introduction of several valid channel prediction algorithms in recent years. Tugay Eyceöz et al. combined autoregression (AR) with fading channels to propose a linear fading channel prediction method based on AR. As a classic algorithm, it calculates the next channel sample using linear regression with previous samples, resulting in a series of channel samples. However, errors accumulate in the regression calculations, potentially limiting its application in complex scenarios such as high-speed and underwater communications.

Another effective method is the nonlinear prediction mechanism. For example, Zhao et al. [3] combined support vector machines (SVM) with wireless signals to propose an effective channel prediction method. Zhao et al. [4] also introduced a novel short-term channel prediction method based on Echo State Networks (ESN), achieving satisfactory performance due to its numerous neurons in the hidden layer. However, this also increases its computational complexity.

Introduced by Hinton et al. [5] in 2006, the deep belief network (DBN) represents a highly effective variant of neural networks. DBN comprises multiple hidden layers, each considered as a restricted Boltzmann machine (RBM). In the training section, every hidden layer is trained by unsupervised learning model, then it is retrained using the output above with unsupervised learning model. In this way, weights in hidden layers are adjusted well, which has ability to complete more complex nonlinear issues than traditional neural network (NN). Since its inception, DBN has garnered significant attention and been utilized in diverse fields, including image classification [6,7], and fault diagnosis [8]. In the wireless communication field [9], Nie et al. [10] used deep neural networks to obtain network traffic. However, network traffic is not adequate for CSI. To our knowledge, no literature exists on leveraging blockchain-based DBNs for CSI channel prediction in wireless communications, prompting us to explore this research area.

To address the above issues, this paper proposes a blockchain-based deep trust network (BDBN) method for realizing fast short-term prediction of fading channels in wireless communications [11]. This method integrates the decentralization, transparency, and security features of blockchain technology with DBN's strengths in handling intricate nonlinear problems. Its goal is to enhance the precision and efficiency of channel predictions while safeguarding the authenticity and integrity of the forecasted data.

2. Principles of blockchain technology

Blockchain is a decentralized database technology that facilitates the storage and transmission of data in a distributed network. Blockchain technology is characterized by decentralization, transparency, security and immutability, thus ensuring the authenticity and integrity of the data it contains [12]. Blockchain uses decentralized ledger technology to distribute data across numerous nodes, each of which keeps a complete record of transactions [13].

In a blockchain network, each block contains data, a timestamp, and a hash of the previous block, resulting in a tamper-proof chain structure [14]. Each time a block is added to the blockchain, a new block is automatically created and a nonce is added to the hash of the previous block, thus protecting the integrity of the previous block's data. Added to the blockchain, a new block is automatically created and a nonce is added to the hash of the previous block, thus protecting the integrity of the previous block's data. In addition, blockchain protects the confidentiality and integrity of data through encryption algorithms [15]. Each participant has a unique set of public and private cryptographic keys, where the public key is used for message encryption and the private key is used for decryption.

3. Channel model

When blockchain transactions are processed and data is transmitted from one node (transmitting terminal) to another node (receiving terminal) within a blockchain network, the integrity and sequence of the data packets may undergo significant variations due to the distributed nature and potential delays introduced by the multi-node consensus mechanism [16].

In the context of blockchain, the "envelopes" of data can be likened to the structure and sequence of transactions or blocks within the chain [17]. These "envelopes" may experience notable changes as they traverse the network, primarily due to the following factors related to the multipath effect in a decentralized environment:

Propagation Delays: Given the decentralized nature of blockchain networks, which comprise numerous nodes spread across the globe, the duration for validating and disseminating a transaction or block to all nodes can differ considerably. This delay, akin to refraction in wireless signal transmission, can affect the timing and order of transactions [18]..

Forks and Reorganizations: In some blockchain systems, temporary forks can occur when multiple blocks are proposed simultaneously [19]. These forks are eventually resolved through the consensus mechanism, but during this period, the "envelope" of the blockchain (i.e., the sequence of blocks) may appear to change dramatically as nodes sync up and reorganize around the longest valid chain.

Typically, the modeling of the wireless channel is conducted as outlined below.

$$R(t) = H(t)X(t) + \varepsilon(t) \tag{1}$$

Consider the scenario where R(t), H(t), and X(t) represent the received signal, channel parameters, and transmitted signal respectively, all as functions of time t. Additionally, $\varepsilon(t)$ denotes the added Gaussian white noise (AWGN). Taking into account the multipath effect, the channel parameter h(t) is characterized by modeling the superposition of multiple paths during transmission. This can be mathematically expressed as follows:

$$h(t) = \sum_{\xi=1}^{\Psi} A_\xi e^{j(2\pi f_\xi^t + \partial_\xi)} \tag{2}$$

In this scenario, let ξ denote the serial number of the transmission paths, with Ψbeing the total count of these paths. Then A_ξ, $f_\xi^t$and ϑ_ξ are the amplitude, the Doppler frequency shift and the phase angle of the ξ -th transmitting path. As for the Doppler frequency shift f_ξ , it can be calculated by

$$f_\xi = \frac{V}{C} f_d \cos\phi \tag{3}$$

Here, V and C represent the velocity of the receiving terminal and the speed of electromagnetic waves in air, respectively. f_d signifies the maximum Doppler frequency shift, while ϕ indicates the angle as illustrated in Fig. 1.

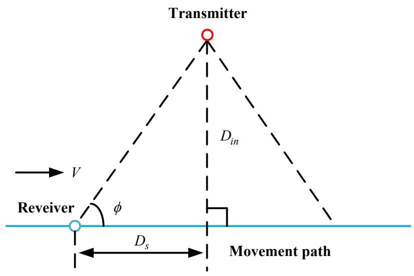


Fig. 1. Figure for calculation of cos ϕ.

For the Rayleigh channel, the probability density function (PDF) is

$$f(z) = \frac{z}{\varpi} e^{-\frac{z^2}{2\varpi^2}}, z \geq 0 \tag{4}$$

and the cumulative distribution function (CDF) can be obtained by

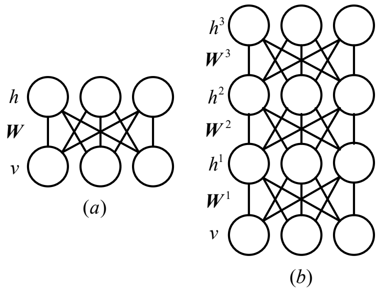
$$F(z) = 1 - e^{-\frac{z^2}{2\varpi^2}} \tag{5}$$

In a Rayleigh fading channel, ϖ^2 represents the power of the scattered components.

4. The fast channel prediction method based on BDBN

4.1 BDBN

In Ref., DBN is mainly designed to image recognition and classification in artificial intelligence (AI). Due to its excellent prediction performance, DBN can also be applied to predict fading channels, together with the characteristics of blockchain [20], therefore, a channel prediction model based on BDBN is proposed in this letter.



(a)The architecture of a Restricted Boltzmann Machine (RBM)
(b) The conventional layout of a Deep Belief Network (DBN).
Fig. 2. Typical structure diagram

The typical instruct of BDBN with inputs and outputs is showed in Fig. 2(b), which is filled with several RBM showed in Fig. 2(a). A Restricted Boltzmann Machine (RBM) comprises an input layer known as the visible layer ($v \in (0,1)^D$) and a single hidden layer denoted as $h \in (0,1)^K$, the W is the connected weight. It is noted that neurons in same layer are not connected, while neurons in different layers are connected. So assuming an energy function $E(v, h)$ for all units in input layer and hidden layer, the joint probability can be calculated by

$$p(v, h) = \frac{e^{-E(v,h)}}{Z} \tag{6}$$

The normalizing factor, also referred to as the partition function, is denoted by Z. Based on the explanations provided, the energy function for the input units v_i and hidden units h_i can be derived as follows

$$\begin{aligned} E(\boldsymbol{v},\boldsymbol{h}) &= -\boldsymbol{h}^T\boldsymbol{W}\boldsymbol{v} - c^T\boldsymbol{v} - b^T\boldsymbol{h} \\ &= -\sum_{i=1}^{D}\sum_{j=1}^{K} W_{ij}v_ih_j - \sum_{i=1}^{D} c_iv_i - \sum_{j=1}^{K} b_jh_j \end{aligned} \tag{7}$$

Here, W represents the weight matrix, while c and b signify the biases for the hidden layer and the visible layer, respectively. In order to reduce computations, the Gibbs sample is employed. Hence, the conditional distributions in visible layer and hidden layer can be obtained by

$$\begin{aligned} p(h_i = 1|v) &= \mathcal{F}(W_iv + c_i) \\ p(v_j = 1|h) &= \mathcal{F}({W^T}_jh + b_j) \end{aligned} \tag{8}$$

where

$$\mathcal{F}(x) = \frac{1}{1+e^{-x}} \tag{9}$$

where $\mathcal{F}(\bullet)$is an active function. Then when to update weights$W$, Employing the contrastive divergence (CD) method, we consider the biases b for the visible layer and c for the hidden layer, which are

$$\begin{gathered} \Delta W_{ij} = \tau(v_i h_j - v_{i\_re} h_{j\_re}) \\ \Delta b_i = \tau(v_i - v_{i\_re}) \\ \Delta c_j = \tau(h_j - h_{i\_re}) \end{gathered} \tag{10}$$

where the τ donates the learning rate in (0,1), the $v_{i_re}$and the $h_{j_re}$are the recovered input of visible layer and hidden layer. By this approach, we can determine the appropriate weight matrix W, as well as the biases for the visible layer b and the hidden layer c.

As explanations above, the BDBN is a stack of many RBMs and a BPNN with two layers. The former can extract data characters with several RBMs, while whole weights of all layers are adjusted by the latter in prediction network. In this way, an effective data predictor based on BDBN is obtained.

4.2 Fast channel prediction based on BDBN

Considering Eq.(2), N channel samples serve as the input data (x) for the Bidirectional Deep Belief Network (BDBN), and the subsequent N+1 sample is the output (y) of the BDBN, which are

$$\begin{cases} x(t) = [h(tT_s), h((t+1)T_s), \ldots, h((t+N)T_s]^T \\ y(t) = h((t+N+1)T_s) \end{cases} \tag{11}$$

With Ts representing the sample interval and $t \in (1,2,3,\ldots,N_T)$ indicating the sequence, NT denotes the total count of channel samples in the aforementioned equation. The gathered channel sample data is then split into two segments: training data N_s and testing data N_c. Notably, to minimize computational complexity, the training phase can be executed offline. Additionally, the Bidirectional Deep Belief Network (BDBN) can be retrained for diverse scenarios as needed.

Hence, the processes of proposed channel prediction method based on BDBN are followed:

Training processes of RBMs:

Step1. Acquire the training dataset N_s and the testing dataset N_c from the channel samples.

Step2. Set up the initial parameters for the Bidirectional Deep Belief Network (BDBN): specify the number of Restricted Boltzmann Machines (RBMs) as N, determine the neuron count in each hidden layer labeled $L = (L_1, L_2, \ldots, L_{\mathcal{N}})$, assign a learning rate τ for all RBMs, establish a reference convergence accuracyϵ_{DBN}, and define the maximum number of training iterations for BDBN as NBDBN.

Step3. Configure the initial parameters for the Restricted Boltzmann Machine (RBM): assign weights $W^{L_{\mu-1}\times L_{\mu}}$, set biases for the visible layer $b^{1\times L_{\mu}}$ and hidden layer $c^{1\times L_{\mu-1}}$, and define the maximum number of training iterations as NRBM.

Step4. Adjust the weights W_{μ}, visible layer bias b_{μ}, and hidden layer bias c_{μ} according to Algorithm 1.

Step5. Evaluate if the current number of training iterations for the RBM μhas reached the maximum allowed (N). If not, increment U by 1 and revert to Step 3.

Step6. Determine if the current iteration count κ for the RBM has met the specified maximum (NRBM). If it hasn't, increment κ by 1 and return to Step 2.

Step7. Obtain weight W_{μ}, the bias c_{μ}, $\mu = 1,2,3,\ldots,\mathcal{N}$.

Training processes of BDBN:

Step8. Build a Backpropagation Neural Network (BPNN) comprising an input layer, a hidden layer $\mathcal{N}$, and an output layer.

Step9. Set the initial weights for the output layer in the BPNN, denoted as $W_{out}^{1\times\mathcal{N}}$

Step10. Based on gradient descent, update W_{out},W_{μ} and biasc_{μ}. Due to space limit, the updated processes are not explained in detail.

Step11. Calculate the cost function:

$$\mathrm{F} = \frac{1}{2}\sum_{i=1}^{m}(N_{si} - \widehat{N}_{si})^2 \tag{12}$$

where the m is the length of training channel.

Step12. Evaluate if the current value of the cost function F matches the predefined convergence precision ϵ_{DBN}. If it does not, increment the current training iteration counter l by 1 and revert to Step 9.

Prediction process:

Step13. Obtain optimal output weightsW_{out_opt}, the hidden weight W_{μ_opt} and hidden bias c_{μ_opt}.

Step14. Test prediction performances using testing channel samplesN_c.

4.3 The complexity analysis

The computational complexity is a crucial aspect for assessing the efficiency of algorithms. In order to reduce complexity, the training processes are completed offline. Hence, the input channel samples are processed by weights in hidden layers and finally, the prediction channel value is obtained from the output layer [21]. Based on explanations above, so when to obtain the $h_1^{o\times L_1}$,the complexity is $\mathcal{O}(oNL_1)$. Here, the N is the input data and o donates the dimension of input data. In this way, when to obtain $h_1^{o\times L_{\mathcal{N}}}$, the complexity is $\mathcal{O}(oL_{\mathcal{N}-1}L_{\mathcal{N}})$. Hence, the complexity of our proposed way is $\mathcal{O}(max(\,oNL_1, oL_1L_2,\ldots,oL_{\mathcal{N}-1}L_{\mathcal{N}})$. Then according to Ref., the complexity of SVM is $\mathcal{O}(o^3)$. For AR, if the order is N_{AR}, its complexity is $\mathcal{O}(N_{AR})$. Due to $o \gg N$, so we can obtain the conclusion that the proposed channel prediction method based on BDBN has comparable complexity than AR, and has less computation complexity than SVM.

Some strategies to further reduce the complexity of such algorithms are detailed below, focusing on hardware acceleration, pruning techniques, and optimized training strategies.

(1) Hardware Acceleration

GPUs and TPUs: Utilizing Graphics Processing Units (GPUs) or Tensor Processing Units (TPUs) can significantly speed up the computation process. These specialized processors are designed to handle parallel computations efficiently, which is crucial for deep learning algorithms.

FPGAs: Field-Programmable Gate Arrays (FPGAs) offer another hardware acceleration option. Unlike GPUs, FPGAs can be programmed to perform specific tasks more efficiently, potentially leading to lower power consumption and higher performance for certain workloads.

ASICs: Application-Specific Integrated Circuits (ASICs) are custom-designed chips tailored for specific algorithms. While they are more expensive and less flexible than GPUs or FPGAs, they can offer the highest performance and efficiency for specific tasks.

(2) Pruning Techniques

Weight Pruning: In deep learning models, many weights may be close to zero and contribute little to the model's performance. Pruning involves removing these weights, which can reduce the model's size and computational requirements without significantly affecting accuracy.

Layer Pruning: Besides individual weights, entire layers or sub-networks can sometimes be pruned if they do not contribute significantly to the model's output. This can lead to more substantial reductions in complexity.

Structured Pruning: Unlike unstructured pruning, which removes weights individually, structured pruning removes entire groups of weights (e.g., filters in convolutional layers). This can make the pruned model more compatible with existing hardware and libraries.

(3) Optimized Training Strategies

During training, monitoring the model's performance on the validation set helps identify when the model begins to overfit. Early stopping involves terminating training when performance on the validation set is no longer improving, thus saving computational time and resources. Batch Normalization This technique normalizes the inputs of each layer, resulting in a more stable training process and higher learning rates. It also reduces the need for careful parameter tuning. Finding the best combination of hyperparameters using techniques such as grid search, stochastic search, or Bayesian optimization can improve training efficiency and model performance.

By combining hardware acceleration, pruning techniques, and optimized training strategies, the computational complexity of your proposed channel prediction method based on BDBN can be further mitigated. This will not only improve the algorithm's efficiency but also make it more suitable for real-time applications and resource-constrained environments. Each of these strategies has its own set of advantages and trade-offs, so it's important to carefully evaluate them based on your specific needs and constraints.

When considering its performance in real-time, low-latency scenarios, especially in mobile or IoT environments, several key aspects need to be discussed.

(1) Real-Time and Low-Latency Performance

Blockchain technology effectively reduces transaction latency through optimized consensus algorithms (e.g., proof of equity instead of proof of work) and off-chain expansion schemes (state channel, side chain), while combining model quantization and pruning technology to alleviate the computational burden brought by deep learning integration. For the demand of high concurrency scenarios, blockchain design needs to support horizontal scaling architecture, in which sharding technology significantly improves throughput by splitting the network load, and advanced frameworks such as BDBN further realize the balance between low latency and scalability through resource elasticity deployment to meet the demand of real-time transaction processing.

(2) Practical Deployment in Mobile or IoT Environments

For the blockchain deployment challenges of mobile and IoT devices, BDBN needs to adopt lightweight blockchain architectures (e.g., low-complexity private chains) and optimize the power consumption of deep learning models to adapt to the limited computation, storage, and battery resources of the devices. At the communication level, the high latency and low bandwidth limitations of wireless networks are effectively overcome through data compression, adaptive protocols and edge computing synergy. In terms of security, relying on blockchain tamper-proof features, combined with encrypted access control and anomaly detection mechanism, while integrating differential privacy and joint learning technology, value mining is realized under the premise of safeguarding the sovereignty of data, forming a three-dimensional solution of resource efficiency, communication performance, and security and privacy.

(3) Conclusion

BDBN, as a blockchain-based deep trust network, has the potential to perform well in real-time, low-latency scenarios, especially in mobile or IoT environments. However, its success depends on careful consideration of resource constraints, connectivity and communication challenges, and security and privacy concerns. By optimizing the blockchain implementation, leveraging efficient deep learning models, and integrating additional security measures, BDBN can be deployed effectively in these environments to provide a secure, trustworthy, and real-time data processing solution.

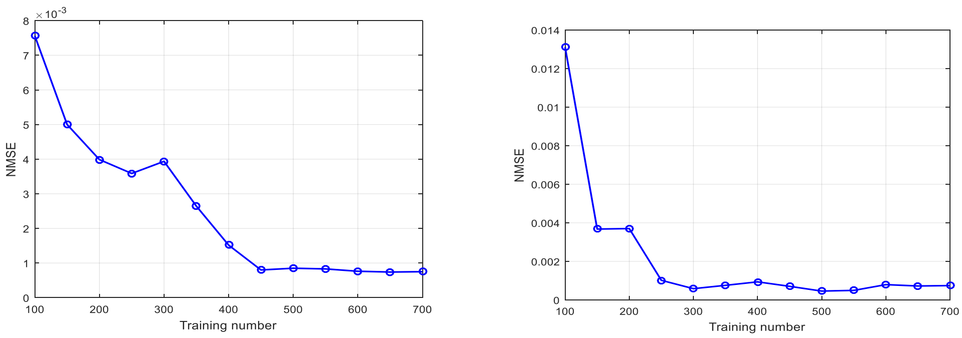
5. Simulations and discussions

To evaluate the performance of our proposed method, we conduct several tests and select relevant parameters for analysis: $f_d = 500Hz$, $T_s = 5 * 10^{-5}$, $f_s = 20kHz$, Ds=500m, Dmin=50m, $N = 6$, $N_s = 600$, $N_c = 400$, $\mathcal{N} = 2$, $L_1 = 30$,$L_2 = 20$, $\tau = 1$, $\epsilon_{DBN} = 10^{-3}$, NBDBN = 500, NRBM = 10.

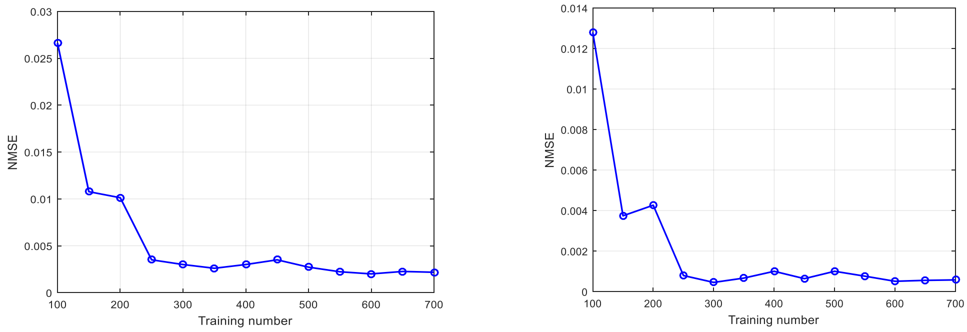
We employ the normalized mean square error (NMSE) as a metric to assess prediction accuracy, formulated as follows:

$$NMSE = \frac{\sum_{n=N_s+1}^{N_c} \left| h((n+N)T_s) - \hat{h}((n+N)T_s) \right|^2}{\sum_{n=N_s+1}^{N_c} |h((n+N)T_s)|^2} \tag{13}$$

where the $\hat{h}((n+N)T_s)$ is the predicted CSI and the $h((n+N)T_s)$ is the actual CSI. Hence, the NMSE for training samples in various RBMs are showed in Fig. 3. In Fig. 3(a), we observe that the NMSE (Normalized Mean Square Error) gradually decreases and stabilizes as the amount of training data increases. (1) The effect of data volume: more training data means that the model is able to learn richer features and patterns. This helps the model to be more accurate in its predictions because it has “seen” more situations, and thus is better able to generalize to unseen data. (2) Stability: As the volume of training data reaches a certain threshold, the reduction in NMSE diminishes, signaling that the model's performance has reached a state of stability. Because the model has already learned the main features in the data, the performance improvement becomes limited by increasing the amount of data. This finding has important implications for practical applications. It suggests that we need to reasonably determine the upper limit of the training data volume when constructing deep learning models to avoid unnecessary consumption of computational resources. In this example, when the training data volume reaches about 600, the NMSE has stabilized, so 600 can be considered a reasonable upper limit of the training data volume.



(a)one RBM (b) two RBMs



(c)RBMs (d) four RBMs

Fig. 3. Curves of NMSEs with training samples with various RBM.

In Fig. 3(b), (c) and (d), we observe that as the number of RBMs (Restricted Boltzmann Machines) in the BDBN increases, the NMSE value does not change significantly. (1) Limitation of fitting capacity: Upon reaching a specific number of RBMs, the model's fitting capability approaches its maximum limit. It means that further increase in the number of RBMs has very limited performance enhancement because the model has been able to capture the features in the data very well. (2) Increase in computational complexity: although increasing the number of RBMs may improve the model's fitting ability to a certain extent, it will also significantly increase the computational complexity and training time. This is detrimental to real-world applications because the consumption of computational resources will increase as a result. In this example, when the number of RBMs is increased to 2, the NMSE value has already stabilized, and the performance enhancement by increasing the number of RBMs further is very limited. Therefore, 2 can be considered as a reasonable upper limit for the number of RBMs, which ensures the performance of the model and controls the consumption of computational resources.

Impact of changing the number of RBMs on accuracy and computational load:

Impact on Accuracy: Increasing the number of RBMs improves the representation of the model, allowing it to capture more complex data features. It helps to improve the performance of the model on the training data and may improve the prediction accuracy to some extent. However, when the number of RBMs is too high, the model may fall into an overfitting state, i.e., it performs well on the training data but has reduced generalization ability on the test data. Therefore, a balance between accuracy and overfitting needs to be found by choosing an appropriate number of RBMs.

Impact on computational load: increasing the number of RBMs significantly increases the computational load of the model. Each RBM requires parameter updates and gradient calculations, so more RBMs means more computational resources and time consumption. In practical applications, the appropriate number of RBMs needs to be selected based on hardware resources and time constraints.

Too many RBMs may lead to too long training time, even beyond the limit of hardware resources, thus affecting the practicality and efficiency of the model.

Furthermore, the comparisons with AR, SVM and our proposed channel prediction method are showed in Fig.4(a). The order NR of AR is set as 6, and the input channel data$o = 400$, the kernel function is polynomial and the prediction model is set as regression fitting. The simulation was conducted using MATLAB 2016a on a Windows 7 system, powered by an Intel(R) CPU E5-1620 operating at 3.6 GHz and equipped with 8.0 GB of RAM.

In Fig. 4, we show the effect of noise power on NMSE (Normalized Mean Square Error). By comparing the NMSE values of our proposed method (BDBN-based channel prediction method) with those of AR (Autoregressive Model) and SVM (Support Vector Machine), we can draw the following conclusions:(1) Noise Robustness Comparison: as the noise power in the wireless signals increase, the NMSE values of our proposed method are lower as compared to AR and SVM. This indicates that our method is more robust in noisy environments and is able to cope with noise interference better, thus maintaining a low prediction error. (2) Performance advantage: In wireless communication systems, noise is an unavoidable interference factor. Hence, robust channel prediction techniques are crucial for enhancing the stability and dependability of communication systems. Our method performs well in this regard, showing its potential advantages in practical applications. This discovery holds significant implications for the design and optimization of wireless communication systems. It underscores the necessity to prioritize noise robustness in developing channel prediction methods, ensuring accurate predictions even in noisy conditions.

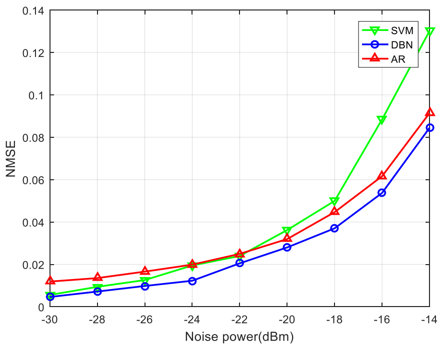


Fig. 4. Comparisons with AR, SVM and proposed method

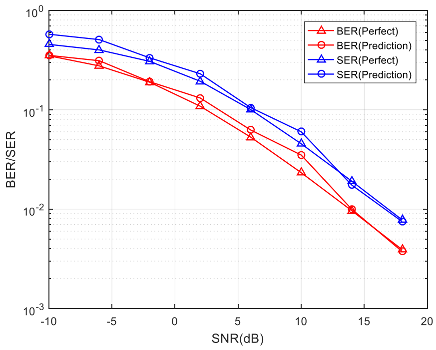


Fig. 5. BER/SER with perfect CSIs and prediction CSI

Fig. 5 presents the Bit Error Rate (BER) and Symbol Error Rate (SER) comparisons between perfect Channel State Information (CSI) and BDBN-predicted CSI across various Signal-to-Noise Ratios (SNRs) when using Quadrature Phase Shift Keying (QPSK) modulation. By comparing and analyzing the effects of SNR on communication quality, we can conclude the following: As the SNR rises, both the BER and SER decrease progressively for both perfect CSI and BDBN-predicted CSI. This aligns with the fundamental principle of communication systems, where an elevated SNR corresponds to improved communication quality. An increase in SNR signifies a greater ratio of signal to noise, thereby minimizing the disruptive impact of noise on signal transmission.

Effectiveness of the prediction method: the BER and SER curves of CSI predicted based on BDBN are very close to those of perfect CSI. This indicates that our channel prediction method has high accuracy and can accurately predict the characteristics of wireless channels, thus providing strong support for the optimization and performance improvement of wireless communication systems. This finding further validates the effectiveness and feasibility of our BDBN method. In practical applications, we can use the method to predict the characteristics of the wireless channel so as to optimize the parameter settings of the communication system and improve the communication quality and efficiency.

This study introduces a wireless channel prediction approach leveraging blockchain and deep belief network (BDBN) technologies, with its effectiveness and viability experimentally confirmed. The experimental results show that BDBN excels in prediction accuracy, robustness and computational efficiency, especially in dealing with noise interference and complex nonlinear problems.

Shortcomings: although BDBN achieves good performance in experiments, its computational complexity is still high, especially when dealing with large-scale datasets. This study mainly focuses on short-term channel prediction, and the performance and applicability for long-term channel prediction need to be further verified.

In addition, blockchain integration has an impact on prediction accuracy:

Increased security and credibility: blockchain technology ensures data security and integrity through distributed ledgers and encryption algorithms. In BDBN, blockchain integration prevents data from being tampered with or leaked, which improves the credibility of the prediction results. It helps maintain the stability and reliability of the model, and although it does not directly improve prediction accuracy, it provides a more solid foundation for prediction.

Data Transparency and Traceability: the transparency of blockchain makes the process of data use and change traceable, which helps to identify problems in the data and make timely corrections. In BDBN, this transparency helps to improve the accuracy and reliability of the data, which in turn has a positive impact on forecast accuracy.

Indirect impact on prediction accuracy: While blockchain itself does not directly improve prediction accuracy, it provides higher quality data inputs to the model by ensuring the security and integrity of the data. High-quality data inputs help models learn and generalize better, which indirectly improves prediction accuracy.

Impact of removing the blockchain: if the blockchain is removed, BDBN will lose the guarantee of data security, transparency and trustworthiness. It may lead to data tampering or leakage, which in turn affects the stability and reliability of the model. In extreme cases, the insecurity of the data may result in the model not being able to learn and generalize correctly, thus severely reducing the prediction accuracy.

In addition, the performance of BDBN is compared with traditional deep learning methods (e.g., LSTM and CNN) on specific tasks to evaluate the effectiveness of BDBN, as shown in Table 1. By selecting representative and challenging datasets, such as the MNIST handwritten digit dataset, the CIFAR-10 image dataset, and publicly available datasets containing multiple types of data (e.g., images, text, etc.). After preprocessing, the quality and consistency of the input data are ensured. Fairness and comparability of the experiments are ensured by setting the same training parameters such as learning rate, batch size and number of iterations. Methods such as cross-validation are used to reduce the risk of overfitting and underfitting and improve the generalization ability of the model.

Therefore, BDBN, LSTM and CNN show different advantages and limitations on different types of data and tasks. BDBN models record the model training process through blockchain technology. The key information and parameters in the model are recorded through blockchain technology, which enhances the transparency and traceability of the model and helps to build deep trust.

Table 1

Comparison of experimental results

Model	Data type	Accuracy	Precision	Recall	F1 Score	Loss Function (Training Set)	Loss Function (Test Set)
BDBN	Images	92%	90%	91%	90.5%	0.15	0.16
LSTMs	Text	88%	85%	87%	86%	0.30	0.32
CNNs	Images	90%	88%	89%	88.5%	0.18	0.20
BDBN	text	86%	84%	85%	84.5%	0.28	0.30

Image data: BDBN outperforms CNN on image data with higher accuracy, precision, recall and F1 score. This is due to the fact that BDBN incorporates blockchain technology, which increases the trustworthiness and security of the data, thus improving model performance.

Text data: LSTM outperforms BDBN and CNN on text data, especially when dealing with long text and complex linguistic structures. This is because LSTMs are good at capturing long-term dependencies in sequential data.

Loss Function: BDBN and CNNs have relatively low loss function values on the training and test sets, which indicates that they have better generalization ability.

Future Development Prospects and Recommendations:

(1) Explore and develop advanced deep learning algorithms to decrease the computational burden of BDBN and enhance its real-time capabilities.

(2) Explore the possibility of applying BDBN to long-term channel prediction and verify its performance in real communication systems.

(3) Verify the performance of BDBN in different communication scenarios and environments to assess its generalization ability and adaptability.

(4) Investigate further the implementation of blockchain technology for data security and privacy safeguards, ensuring the robustness of wireless communication systems.

6. Discussion

The performance of BDBN in different wireless environments is affected by a variety of factors, including the stability of the network environment, data transmission rate, latency, and signal strength.

Urban environment: cities usually have a more stable and high-speed wireless network environment, which is conducive to the data transmission and synchronization of BDBN. In urban environments, due to the high density of base stations and wide signal coverage, the data transmission rate is usually faster, and BDBN is able to process and verify the data faster and improve the efficiency of the whole network. Meanwhile, with distributed ledgers and encryption algorithms, BDBN can prevent data from being tampered with or leaked.

Rural environment: the wireless network environment in rural areas may be relatively unstable and signal coverage may not be as extensive as in cities. This may cause BDBN to experience delays or interruptions during data transmission.

High-speed mobile scenarios (e.g., high-speed rail): In high-speed mobile scenarios, devices may switch base stations frequently, leading to network instability and increased latency. This poses a higher challenge to the performance of the BDBN, as data synchronization and validation require fast and reliable network connectivity. Bandwidth constraints may also affect the data processing capability of the BDBN. Optimizing the design of BDBN for more efficient data synchronization algorithms and verification mechanisms can improve its performance in high-speed mobile scenarios.

In summary, the performance of BDBN in different wireless environments is affected by a variety of factors, such as the stability of the network environment, data transmission rate, delay, and signal strength. In urban environments, BDBNs are usually able to exhibit high performance and reliability; in rural areas, it may be necessary to improve its performance by improving the network environment and optimizing the BDBN design; in high-speed mobile scenarios, adopting technologies such as mobile edge computing and optimizing the BDBN design is an effective strategy to cope with the challenges.

Meanwhile, the distributed trust mechanism of BDBN can improve the reliability of 6G network in terms of 6G network application, ensuring the security and integrity of data in the transmission process. By optimizing blockchain technology, BDBN can support efficient data processing in 6G networks to meet the demand for high throughput. The distributed architecture of BDBN can well support large-scale device connections in 6G networks and improve the scalability and flexibility of the network. In order to cope with the high throughput and low latency demands that 6G networks also face, BDBN needs to continuously optimize its network performance, including improving the efficiency of consensus algorithms, optimizing data transmission and storage, and so on. Interoperability with other blockchain networks is needed to enable data sharing and interaction. This can be achieved by introducing cross-chain communication protocols and technologies.

As a result, BDBNs have significant advantages in supporting the scalability of 6G networks. However, it also faces challenges such as data security and privacy protection, network performance optimization, and cross-chain interoperability. Through continuous technological innovation and optimization, BDBN is expected to become an important part of 6G networks in the future, providing strong support for building a more secure, reliable and efficient communication network.

Additionally, blockchain consensus mechanisms, including Proof of Work (PoW) and Proof of Stake (PoS), can have a significant impact on prediction latency and can indeed be a bottleneck in some cases.PoW relies on solving complex mathematical problems, which requires a significant amount of computational power. This computational intensity can lead to longer transaction processing times, which can increase predictive latency. While PoS transactions are typically faster than PoW, network congestion can still occur during periods of high activity, which can lead to increased transaction confirmation latency and prediction latency. both PoW and PoS systems must balance security and transaction speed. Adding security measures, such as adding an authentication layer or increasing the number of authenticators, can lead to an increase in prediction delay. Therefore, the consensus mechanism of the blockchain does affect the prediction latency and can be a bottleneck depending on the specific mechanism and network conditions.PoW tends to require longer transaction processing time and higher prediction latency due to its high computational requirements and energy consumption. PoS, on the other hand, offers faster transactions and lower predictive latency due to its lower computational and energy costs and potential scalability. However, both mechanisms can suffer from network congestion and validator performance issues, which can lead to increased prediction latency. Therefore, when designing blockchain applications, the trade-offs between different consensus mechanisms must be carefully evaluated to ensure optimal performance and user experience.

7. Conclusions

In this paper, we introduce a swift channel prediction technique utilizing a block diagonal deep confidence network (BDBN). When compared to existing methods such as the autoregressive model (AR) and support vector machine (SVM), our proposed method demonstrates acceptable computational complexity and exhibits superior noise robustness compared to AR and SVM. By utilizing the DBN's feature extraction Using the advantages of DBN in feature extraction, this paper innovatively applies it to the channel prediction problem in wireless communication, especially further optimizing the block diagonal structure to adapt to the characteristics of wireless communication channels. By comparing with other prediction methods (e.g., AR and SVM), the advantages of the BDBN method in improving the channel prediction accuracy and anti-noise interference are demonstrated. In addition, by using blockchain technology for data encryption, the method ensures the security and availability of data during transmission and storage, protects user privacy, and prevents data leakage. Meanwhile, BDBNs with LSTMs and CNNs show different advantages and limitations on different types of data and tasks. In practical applications, appropriate models need to be selected according to the characteristics of specific tasks and datasets.

By conducting comparative experiments with perfect Channel State Information (CSI), this paper validates the efficacy and practicality of the proposed channel prediction method. It establishes a robust theoretical framework and experimental evidence for future research endeavors and applications. The fast channel prediction method based on BDBN has a broad application prospect in the field of wireless communication, which not only improves the prediction accuracy and robustness, but also ensures the security and availability of data. This holds immense significance in advancing the evolution and progression of wireless communication technology.

R E F E R E N C E S

[1] Vajs I, Brkić S, Ivaniš P, et al. Neural Network SNR Prediction for Improved Spectral Efficiency in Land Mobile Satellite Networks. Electronics, 2024, 13(18):3659.

[2] Nguyen L N, Le T A, Nguyen T L P, et al. Performance Prediction of Power Beacon-Aided Wireless Sensor-Powered Non-Orthogonal Multiple-Access Internet-of-Things Networks under Imperfect Channel State Information. Applied Sciences, 2024,14(11).

[3] Zhao X, Hou C, and Wang Q. A new SVM-based modeling method of cabin path loss prediction. Int. J. Antennas and Propagation. Apr 2013:1–7.

[4] Zhao Y, Gao H, Beaulieu N C, Chen Z, and Ji H. Echo State Network for Fast Channel Prediction in Ricean Fading Scenarios, IEEE Communications Letters. March 2017, 21(3):672-675.

[5] Hinton G, Osindero S and Teh Y. A fast-learning algorithm for deep belief nets, Neural Computation, 2006, 18(7):1527-1554.

[6] Zhong P, Gong Z, Li S, and Schönlieb C B. Learning to Diversify Deep Belief Networks for Hyperspectral Image Classification. IEEE Transactions on Geoscience and Remote Sensing, June 2017, 55(6):3516-3530.

[7] Wu F, Wang Z, Lu W, Li X, and Yang Y, et al. Regularized Deep Belief Network for Image Attribute Detection. IEEE Transactions on Circuits and Systems for Video Technology, July 2017, 27(7):1464-1477.

[8] Shao H, Jiang H, Zhang H and Liang T. Electric Locomotive Bearing Fault Diagnosis Using a Novel Convolutional Deep Belief Network. IEEE Transactions on Industrial Electronics, March 2018, 65(3):2727-2736.

[9] Danielle A M, De R B, Bang A H. A decision support system for safer airplane landings: Predicting runway conditions using XGBoost and explainable AI. Cold Regions Science and Technology, 2022, 199.

[10] Nie L, Jiang D, Yu S and Song H. Network Traffic Prediction Based on Deep Belief Network in Wireless Mesh Backbone Networks. 2017 IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, 2017:1-5.

[11] Chen Y F. Shortwave channel state feature extraction and state prediction. Harbin Engineering University, 2021.DOI:10.27060/d.cnki.ghbcu.2021.000116.

[12] Hajji E M, Saady E Y, Addi A M, et al. Optimization of agrifood supply chains using Hyperledger Fabric blockchain technology. Computers and Electronics in Agriculture, 2024, 227 (P1): 109503-109503.

[13] Srokosz W, Lenio P, Sobiecki G. Blockchain Technology in Project Finance: A Legal and Practical Model for Financing Mega-Investments. Taylor & Francis: 2024-10-12. DOI:10.4324/9781003568797.

[14] Rai M H, Shukla K K, Tightiz L, et al. Enhancing data security and privacy in energy applications: Integrating IoT and blockchain technologies. Heliyon, 2024, 10(19):e38917-e38917.

[15] Mubarakali A, Basha A A. Blockchain technology for secure electronic healthcare record in cloud based cryptographic hash generator. Peer-to-Peer Networking and Applications, 2024, (prepublish): 1-17.

[16] Hou Y F, Webber J L, Yano K, et al. Modeling and Predictability Analysis on Channel Spectrum Status Over Heavy Wireless LAN Traffic Environment. IEEE ACCESS,2021,9: 85795-85812.

[17] Sui Y B, He Y G, Cheng T T, et al. Broad Echo State Network for Channel Prediction in MIMO-OFDM Systems. IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY,2020, 69(11):13383-13399.

[18] Gogoi J A, Baishnab L K. Dynamic Threshold Based Throughput Enhancement in Cognitive Radio Network Using Hidden Markov Model with State Prediction. Wireless Personal Communications,2020, 115(3):1-19.

[19] Telecommunications; Findings from Case Western Reserve University Reveals New Findings on Telecommunications (Channel State Information Prediction for 5g Wireless Communications: a Deep Learning Approach). Telecommunications Weekly, 2020,

[20] Luo C, Ji J, Wang Q, et al. Channel State Information Prediction for 5G Wireless Communications: A Deep Learning Approach. IEEE Transactions on Network Science and Engineering, 2020, 7(1):227-236.

[21] Sui Y B, He Y G, Cheng T T, et al. Adaptive Elastic Echo State Network for Channel Prediction in IEEE802.11ah Standard-Based OFDM System. IEEE Access, 2020, 810169-10185.