mathematics

MDPI

Article

Research on Signal Detection of OFDM Systems Based on the LSTM Network Optimized by the Improved Chameleon Swarm Algorithm

Yunshan Sun, Yuetong Cheng, Ting Liu *, Qian Huang, Jianing Guo and Weiling Jin

School of Information Engineering, Tianjin University of Commerce, Tianjin 300134, China; sunyunshan@tjcu.edu.cn (Y.S.); chengyuetong@stu.tjcu.edu.cn (Y.C.); huang_qian@stu.tjcu.edu.cn (Q.H.)
* Correspondence: liuting@tjcu.edu.cn

Abstract: In order to improve the signal detection capability of orthogonal frequency-division multiplexing systems, a signal detection method based on an improved LSTM network for OFDM systems is proposed. The LSTM network is optimized by the Chameleon Swarm Algorithm (CLCSA) with the coupling variance and lens-imaging learning. The signal detection method based on the traditional LSTM network has the problem of a complex manual tuning process and insufficient stability. To solve the above problem, the improved Chameleon Swarm Algorithm is used to optimize the initial hyperparameters of the LSTM network and obtain the optimal hyperparameters. The optimal hyperparameters initialize the CLCSA-LSTM network model and the CLCSA-LSTM network model is trained. Finally, the trained CLCSA-LSTM network model is used for signal detection in the OFDM system. The simulation results show that the signal detection performance of the OFDM receiver has been significantly improved, and the dependence on CP and pilot overhead can be reduced. Under the same channel environment, the proposed method in this paper has better performance than other signal detection methods, and is close to the performance of the MMSE method, but it does not need prior statistical characteristics of the channel, so it is easy to implement.

Keywords: OFDM systems; signal detection; channel estimation; LSTM network; chameleon swarm algorithm

MSC: 37M10

Citation: Sun, Y.; Cheng, Y.; Liu, T.; Huang, Q.; Guo, J.; Jin, W. Research on Signal Detection of OFDM Systems Based on the LSTM Network Optimized by the Improved Chameleon Swarm Algorithm. *Mathematics* **2023**, *11*, 1989. https://doi.org/10.3390/math11091989

Academic Editors: Andrey Koucheryavy, Ahmed A. Abd El-Latif and Ammar Muthanna

Received: 28 March 2023
Revised: 18 April 2023
Accepted: 21 April 2023
Published: 23 April 2023

1. Introduction

Due to the complex wireless communication environment, the signal from the transmitter is affected by noise and other factors, which makes it difficult to recover the signal from the receiver. In order to recover the transmitted signal accurately and improve the communication quality, it is often necessary to carry out the corresponding channel estimation and signal detection.

Orthogonal Frequency Division Multiplexing (OFDM) is an efficient multi-carrier modulation technique that plays an important role in wireless communication systems [1]. OFDM has the advantages of high channel utilization, strong resistance to frequency selective fading, and simple implementation. The OFDM system can estimate Channel Status Information (CSI) through the pilot before detecting the transmitted signal, and then recover the transmitted signal at the receiving end by using the estimated CSI [2]. The problem of channel estimation in OFDM systems has been extensively studied. Traditional estimation algorithms, such as Least Square (LS) or Minimum Mean Square Error (MMSE), have been widely used. The LS method does not require prior statistical information about the channel, but has the problem of low performance [3]. By using second-order statistics for the channel, the MMSE method can usually bring a better detection performance, but it requires the prior statistics of the channel and requires a large amount of calculation [4].

Moreover, the characteristics of the wireless channel are complex, and the signal will experience the influence of multipath fading and noise in the wireless environment, resulting in the low bit-error-rate performance of the whole communication system. Therefore, in order to ensure the effectiveness and reliability of communication, it is of great significance to seek more effective signal detection methods.

Deep learning is a kind of neural network developed on the basis of a shallow neural network, which has a deeper network structure. Deep learning can better fit the input and output characteristics to obtain better system-performance characterization.

As a deep learning network, the Long Short-Term Memory (LSTM) network can effectively cope with gradient vanishing, gradient explosion, and low estimation accuracy of traditional OFDM systems, due to the influence of pilot frequency and cyclic prefix length. Meanwhile, the LSTM network can solve the problem of long sequence dependence. The LSTM network can be applied to OFDM systems for channel estimation and signal detection. However, the prediction accuracy of the LSTM network model largely depends on the selection of hyperparameters, which usually need to be manually selected. This method not only requires a large amount of calculation, but also cannot obtain the optimal solution in most cases, which is time-consuming and laborious. Therefore, using the swarm intelligence algorithm to optimize deep neural network hyperparameters is a solution.

The swarm intelligence method optimizes the parameter setting of the deep neural network, which mainly uses the heuristic principle of the swarm intelligence algorithm to provide an appropriate solution close to the global optimal in a reasonable time. Therefore, the Whale Optimization Algorithm (WOA) was used in the literature [5] to optimize the hyperparameters of the deep neural network in order to achieve an optimal control strategy for the autonomous-driving control problem. The literature [6] uses Grey Wolf Optimization (GWO) and Genetic Algorithm (GA) meta-heuristics to tune the hyperparameters of machine learning algorithms. In addition, the literature [7] uses Particle Swarm Optimization (PSO) combined with the fastest gradient descent algorithm, to determine the optimal network configuration for the deep neural network. In the parameter setting of the swarm intelligent optimization network, there are still problems of slow convergence speed and low convergence accuracy. In order to better solve these problems, we constantly explore and improve the intelligent optimization algorithm or put forward a new intelligent optimization algorithm to solve the problems of convergence speed and accuracy.

The Chameleon Swarm Algorithm (CSA) is a simple and efficient metaheuristic swarm intelligence optimization algorithm proposed by Malik Shehadeh Braik [8] in 2021. The algorithm mathematically models and implements the behavioral steps of chameleon searching for food in trees, deserts, swamps and other places. It has the advantages of simple operation and few adjustment parameters, but it has poor performance in solving high- and multi-mode problems [9]. The traditional CSA algorithm also has problems such as poor population diversity, slow convergence and low accuracy. To overcome the shortcomings of traditional Chameleon Swarm Algorithms, a new Chameleon Swarm Algorithm (coupling variance and lens-imaging learning Chameleon Swarm Algorithm, CLCSA) is proposed. Finally, an improved chameleon swarm algorithm is used to solve the initialization parameter-setting problem.

In this paper, aiming at the problems of poor communication reliability caused by the fading phenomenon in signal transmission of the OFDM system and the problems of the complex manual parameter-adjustment process and the insufficient stability of traditional LSTM network signal-detection methods, in order to better recover outgoing signals and improve the accuracy of signal detection, a signal detection method for OFDM systems is proposed, based on coupling variance and lens-imaging learning Chameleon Swarm Algorithm (CLCSA)-optimized LSTM network. The signal detection network designed in this paper can replace the channel estimation and signal detection module of the traditional algorithm. Firstly, the traditional chameleon swarm algorithm is improved, and the improved Chameleon Swarm Algorithm (CLCSA) is proposed. Then, the CLCSA algorithm is used to optimize the hyperparameters of the LSTM network. The optimal

hyperparameter configuration obtained through iteration is taken as the initial parameters of the CLCSA-LSTM network model and it is trained. Finally, the trained CLCSA-LSTM network model is applied to the OFDM system for signal detection, so as to realize the end-to-end recovery of the transmitted data.

The rest of this paper is organized as follows. Section 2 introduces the deep learning method, and the principle of the OFDM system and its channel estimation method. Section 3 introduces the principles of the LSTM network. Section 4 introduces the basic principle of the Chameleon Swarm Algorithm. Section 5 introduces the improved Chameleon Swarm Algorithm and the signal detection method of the OFDM system, based on the improved Chameleon Swarm Algorithm, to optimize the LSTM network. Section 6 analyzes the experimental results. Section 7 gives the conclusion of this paper.

2. Basic Theory

2.1. Deep Learning Method

Deep learning is a kind of neural network developed on the basis of shallow neural network, which has a deeper network structure. Deep learning can better fit the input and output characteristics to obtain a better system performance characterization. At present, deep learning has been widely applied in different fields, such as image processing [10–12], speech recognition [13], emotion recognition [14], path planning [15], etc.

In wireless communication, deep learning is widely used in signal detection [16], pilot design [17], channel state information feedback [18,19], channel balancing [20–22] and other aspects. For channel estimation and signal detection, on the one hand, the Deep Neural Network (DNN) is used to replace the channel estimation and balancing module in the whole wireless communication system, so as to restore the sending signal in an end-to-end way. For example, in the literature [23], the off-line trained DNN model is used to directly recover the signals transmitted online, which has stronger robustness compared with the traditional receiver. In addition, the literature [24] proposes a receiver for underwater acoustic OFDM systems based on DNN, which only uses a single neural network to realize the whole signal processing, further improving the practicability of the system. Aiming at the low recognition accuracy of existing signal modulation recognition algorithms, literature [25] proposes an improved spatiotemporal multi-channel network model, which effectively improves the recognition accuracy. Unlike traditional schemes designed for fixed channels, the literature [26] can perform multiple-input and multiple-output (MIMO)-detection under Rayleigh fading channels, extending the application of DNNs to time-varying channels. The literature [27] proposes that the basic DNN structure is applied to the signal detection of the single-tap MIMO channel, and the convolutional neural network and recursive neural network structure are applied to the signal detection of MIMO systems under the multi-path fading channel.

On the other hand, deep neural networks can be used to optimize channel estimation and improve the performance of channel estimation and signal detection. For example, a deep-learning-based channel estimation scheme was proposed in the literature [28–31], and the channel estimation accuracy was significantly improved. In orthogonal frequency-division multiplexing systems for wireless communications with high mobility, channel estimation becomes challenging, due to the dual selectivity of the channel. To this end, a deep-learning-based channel estimation network has been designed in the literature [32,33] to improve the estimation performance. On this basis, a novel inter-carrier interference-aware OFDM channel estimation network is proposed in the literature [34] for fast time-varying channels. For specialized hydroacoustic channel characteristics, a DNN model is proposed in the literature [35], which solves the channel estimation problem over multipath channels in underwater acoustic OFDM systems. Since DNN models are widely used in the image domain, many scholars have proposed research ideas to solve the channel estimation problem using deep learning in image processing; for example, the literature studies in [36,37] used a deep learning approach based on image super-resolution to produce high-quality channel estimates. The literature study in [38] treats the channel matrix as

a two-dimensional image and proposes a denoising-based approximate message-passing neural network for channel estimation.

2.2. OFDM Systems Principle

Based on the pilot in OFDM systems, the setup of each OFDM symbol contains N subcarriers, the input sequence of bits in the OFDM systems is modulated and also requires serial parallel conversion and insertion of the pilot information, and after performing the Inverse Fast Fourier Transform (IFFT), the time domain OFDM signal is expressed as:

$$s(n) = \frac{1}{N}\sum_{k=0}^{N-1} S(k)e^{j(\frac{2\pi}{N}kn)}, 0 \le n \le N-1 \tag{1}$$

where $S(k)$ is the symbol sent on the kth subcarrier, $k = 0, 1, 2, \ldots, N-1$.

After the IFFT transform, a cyclic prefix (CP) is inserted into the signal, which is usually longer than the maximum channel delay, to prevent inter-symbol interference (ISI) and to send the newly generated signal $s\prime(n)$ through a noisy multipath channel after the parallel–serial transform. From this, it is possible to obtain the received signal $y\prime(n)$,

$$y\prime(n) = s\prime(n) \otimes h(n) + w(n) \tag{2}$$

where $w(n)$ is the additive Gaussian white noise and $h(n)$ is the impulse response of the channel. After removing the CP to obtain the received signal $y(n)$, Equation (2) is discrete Fourier transformed to obtain the frequency domain signal,

$$Y(k) = S(k)H(k) + W(k), 0 \le k \le N-1 \tag{3}$$

where $S(k)$ is the transmitted signal, $Y(k)$ is the received signal, $H(k)$ is the frequency response of the multipath channel, and $W(k)$ is the noise response in the frequency domain.

2.3. Traditional Pilot-Based OFDM Systems Channel Estimation Technology

It is assumed that all subcarriers of OFDM symbols are orthogonal to each other, so that the pilot frequency symbols of N subcarriers can be represented by matrix $\mathbf{X}$.

$$\mathbf{X} = \begin{bmatrix} X(0) & 0 & \cdots & 0 \\ 0 & X(1) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & X(N-1) \end{bmatrix} \tag{4}$$

where $X(k)$ is the guide frequency value at the kth subcarrier, $\mathrm{E}[X(k)] = 0$ and $\mathrm{Var}(X(k)) = \sigma_x^2$.

The signal $Y(k)$ received at the receiving end can be expressed as:

$$\mathbf{Y} = \begin{bmatrix} Y(0) \\ Y(1) \\ \vdots \\ Y(N-1) \end{bmatrix} = \begin{bmatrix} X(0) & 0 & \cdots & 0 \\ 0 & X(1) & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & X(N-1) \end{bmatrix} \begin{bmatrix} H(0) \\ H(1) \\ \vdots \\ H(N-1) \end{bmatrix} + \begin{bmatrix} W(0) \\ W(1) \\ \vdots \\ W(N-1) \end{bmatrix} = \mathbf{XH} + \mathbf{W} \tag{5}$$

where $\mathbf{H}$ is the frequency domain response vector of the channel, given that $\mathbf{H} = [H(0), H(1), \ldots, H(N-1)]^T$, $\mathbf{W}$ is an additive noise vector and satisfies $\mathbf{W} = [W(0), W(1), \ldots, W(N-1)]^T$ as well as $\mathrm{E}[W(k)] = 0$, $\mathrm{Var}(W(k)) = \sigma_w^2$. The general noise is Gaussian white noise.

2.3.1. LS Method

The channel response value estimated by the LS method is determined by the received signal and the transmitted signal. To obtain the channel estimate $\hat{\mathbf{H}}_{\mathrm{LS}}$, the expression for the cost function of the LS method is shown in Equation (6).

$$\begin{aligned} J(\hat{\mathbf{H}}_{\mathrm{LS}}) &= \|\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}}\|^2 = (\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}})^{\mathrm{H}}(\mathbf{Y}-\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}}) \\ &= \mathbf{Y}^{\mathrm{H}}\mathbf{Y}-\mathbf{Y}^{\mathrm{H}}\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}}-\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\mathbf{X}^{\mathrm{H}}\mathbf{Y}+\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\mathbf{X}^{\mathrm{H}}\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}} \end{aligned} \tag{6}$$

Let the partial derivative of the cost function to $\hat{\mathbf{H}}_{\mathrm{LS}}$ be 0, and we can obtain Equation (7).

$$\frac{\partial J(\hat{\mathbf{H}}_{\mathrm{LS}})}{\partial \hat{\mathbf{H}}_{\mathrm{LS}}} = -2\left(\mathbf{X}^{\mathrm{H}}\mathbf{Y}\right)^{*} + 2\left(\mathbf{X}^{\mathrm{H}}\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}}\right)^{*} = 0 \tag{7}$$

Then, it can be concluded that $\mathbf{X}^{\mathrm{H}}\mathbf{X}\hat{\mathbf{H}}_{\mathrm{LS}} = \mathbf{X}^{\mathrm{H}}\mathbf{Y}$, and the solution of the LS method can be obtained as Equation (8):

$$\hat{\mathbf{H}}_{\mathrm{LS}} = \left(\mathbf{X}^{\mathrm{H}}\mathbf{X}\right)^{-1}\mathbf{X}^{\mathrm{H}}\mathbf{Y} = \mathbf{X}^{-1}\mathbf{Y} \tag{8}$$

The elements in the $\hat{\mathbf{H}}_{\mathrm{LS}}$ vector can be represented by $\hat{H}_{\mathrm{LS}}(k)$. It can be seen from the above assumptions that $\mathbf{X}$ is a diagonal matrix, so the LS channel estimation results on each subcarrier can be expressed as Formula (9).

$$\hat{H}_{\mathrm{LS}}(k) = \frac{Y(k)}{X(k)} \tag{9}$$

The mean squared error (MSE) of the LS method is shown in Equation (10).

$$\begin{aligned} &\mathrm{MSE}_{\mathrm{LS}} = \mathrm{E}\left[(\mathbf{H}-\hat{\mathbf{H}}_{\mathrm{LS}})^{\mathrm{H}}(\mathbf{H}-\mathbf{H}_{\mathrm{LS}})\right] \\ &= \mathrm{E}\left[\left(\mathbf{H}-\mathbf{X}^{-1}\mathbf{Y}\right)^{\mathrm{H}}\left(\mathbf{H}-\mathbf{X}^{-1}\mathbf{Y}\right)\right] \\ &= \mathrm{E}\left[(\mathbf{H}-\mathbf{W})^{\mathrm{H}}(\mathbf{H}-\mathbf{W})\right] \\ &= \mathrm{E}\left[\mathbf{W}^{\mathrm{H}}\left(\mathbf{X}\mathbf{X}^{\mathrm{H}}\right)^{-1}\mathbf{W}\right] \\ &= \frac{\sigma_w^2}{\sigma_{\mathrm{x}}^2} \end{aligned} \tag{10}$$

It can be seen from Equation (10) that the mean square error of the LS method is inversely related to the signal-to-noise ratio, which is greatly affected by noise.

2.3.2. MMSE Method

The MMSE method utilizes the second-order statistical prior knowledge of the channel response, and takes the influence of error into account comprehensively. It is a channel estimation method which reduces the influence of noise on the basis of the LS method and aims to minimize the mean square error between the estimated value and the real value of channel response in a frequency domain. Here, we define $\hat{\mathbf{H}}_{\mathrm{MMSE}} = \mathbf{Z}\hat{\mathbf{H}}_{\mathrm{LS}}$, and the objective function of MMSE estimation can be expressed as Equation (11).

$$J(\hat{\mathbf{H}}_{\mathrm{MMSE}}) = \mathrm{E}\left[\|\mathbf{e}\|^2\right] = E\left[\|\mathbf{H}-\hat{\mathbf{H}}_{\mathrm{MMSE}}\|^2\right] \tag{11}$$

In MMSE channel estimation, choosing a suitable weighting value **Z** minimizes the objective function above, and, according to the principle of orthogonality, the error vector **e** is obtained to be orthogonal to $\hat{\mathbf{H}}_{\mathrm{LS}}$, thus satisfying Equation (12).

$$\begin{aligned} &\mathrm{E}\left[\mathbf{e}\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] = \mathrm{E}\left[\left(\mathbf{H}-\hat{\mathbf{H}}_{\mathrm{MMSE}}\right)\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\left(\mathbf{H}-\mathbf{Z}\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right)\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] - \mathbf{Z}\mathrm{E}\left[\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] \\ &= \mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}} - \mathbf{Z}\mathbf{R}_{\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}} = 0 \end{aligned} \tag{12}$$

where $\mathbf{R}_{\mathrm{AB}}$ is the reciprocal correlation matrix of matrices **A** and **B**, i.e., $\mathbf{R}_{\mathrm{AB}} = \mathrm{E}\left[\mathbf{A}\mathbf{B}^{\mathrm{H}}\right]$. Solving Equation (12) yields Equation (13).

$$\mathbf{Z} = \frac{\mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}}}{\mathbf{R}_{\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}}} \tag{13}$$

where $\mathbf{R}_{\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}}$ is the autocorrelation matrix of the LS channel estimate, i.e.,

$$\begin{aligned} &\mathbf{R}_{\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}} = \mathrm{E}\left[\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\mathbf{X}^{-1}\mathbf{Y}\left(\mathbf{X}^{-1}\mathbf{Y}\right)^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\left(\mathbf{H}+\mathbf{X}^{-1}\mathbf{W}\right)\left(\mathbf{H}+\mathbf{X}^{-1}\mathbf{W}\right)^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\mathbf{H}\mathbf{H}^{\mathrm{H}}\right]+\mathrm{E}\left[\mathbf{X}^{-1}\mathbf{W}\mathbf{W}^{\mathrm{H}}\left(\mathbf{X}^{-1}\right)^{\mathrm{H}}\right] \\ &= \mathrm{E}\left[\mathbf{H}\mathbf{H}^{\mathrm{H}}\right] + \frac{\sigma_w^2}{\sigma_x^2}\mathbf{I} \end{aligned} \tag{14}$$

where **I** denotes the unit matrix. The expression for MMSE channel estimation is Equation (15):

$$\hat{\mathbf{H}}_{\mathrm{MMSE}} = \frac{\mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}}}{\mathbf{R}_{\hat{\mathbf{H}}_{\mathrm{LS}}\hat{\mathbf{H}}_{\mathrm{LS}}}}\hat{\mathbf{H}}_{\mathrm{LS}} = \mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}}\left(\mathbf{R}_{\mathbf{H}\mathbf{H}}^{\mathrm{H}} + \frac{\sigma_w^2}{\sigma_x^2}\mathbf{I}\right)^{-1}\hat{\mathbf{H}}_{\mathrm{LS}} \tag{15}$$

where $\mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathrm{LS}}}$ represents the correlation matrix between the true channel frequency-domain impulse response and the estimated channel frequency-domain response.

From Equation (15), it can be seen that, although the MMSE method solves the problem of the LS method being affected by noise, the MMSE method requires the calculation of the inter-correlation matrix of the channel.

According to the above analysis, it can be found that the LS method is simple to implement but has poor performance. The MMSE method has better performance than the LS method, but it requires more computation and prior knowledge of channel statistics. The signal detection performance of these two methods is poor when CPs and pilots are removed in small numbers, while the deep learning method has a superior data processing ability. Therefore, deep learning can be introduced into the communication system, and the deep learning method can be used to make up for the existing shortcomings of traditional methods, so as to further improve the accuracy of signal detection.

3. LSTM Network

LSTM is a chain structure composed of several neurons connected from the beginning to the end. Within each neural unit is a gated structure and cellular memory unit, enabling it to be used to predict time series data with long-term dependencies. Therefore, the LSTM network can be applied to OFDM systems for channel estimation and signal detection. The literature [39] proposes a deep-learning-based signal detection method that uses the LSTM network to alleviate the interdependence of adjacent symbols and improve the accuracy of

signal recovery. The literature [40] proposes the use of a deep neural network for signal detection in uplink OFDM systems over time-varying channels.

LSTM regulates information transmission at every moment, by introducing mechanisms such as a forgetting gate, input gate, output gate and memory unit. The network structure of LSTM is shown in Figure 1.

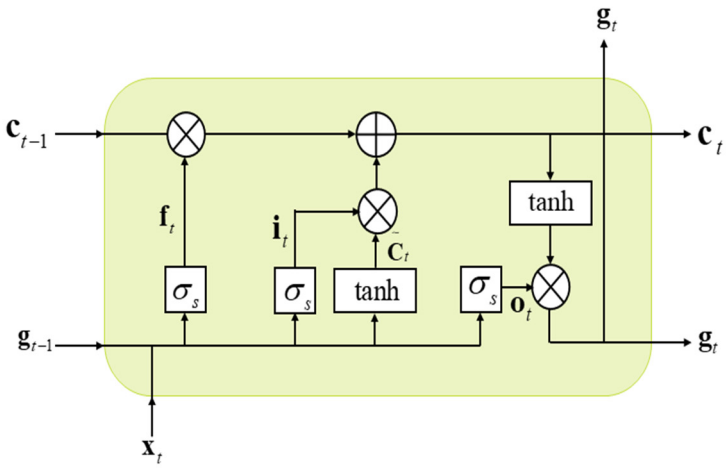

Figure 1. LSTM structure diagram.

Let $\mathbf{g}_{t-1}$ be the state of the hidden layer at the previous moment and $\mathbf{x}_t$ represent the input vector. Then, the LSTM network works as follows:

The forgetting gate $\mathbf{f}_t$ is responsible for deciding the amount of information to be discarded from the state of the previous moment.

$$\mathbf{f}_t = \sigma_s\left(\mathbf{w}_f\left[\mathbf{g}_{t-1}, \mathbf{x}_t\right] + \mathbf{b}_f\right) \tag{16}$$

where $\mathbf{w}_f$ and $\mathbf{b}_f$ are the corresponding weight matrices and bias weights of the forgetting gates, respectively, and σ_s is the sigmoid activation function, expressed as shown in Equation (17):

$$\sigma_s(\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{x}}} \tag{17}$$

The input gate can determine the amount of information to be updated into the cell at the current moment. $\mathbf{i}_t$ will determine how much of the newly acquired information is selected to update the status, and $\tilde{\mathbf{c}}_t$ represents the current input to the unit status.

$$\mathbf{i}_t = \sigma_s\left(\mathbf{w}_i\left[\mathbf{g}_{t-1}, \mathbf{x}_t\right] + \mathbf{b}_i\right) \tag{18}$$

$$\tilde{\mathbf{c}}_t = \tanh\left(\mathbf{w}_c\left[\mathbf{g}_{t-1}, \mathbf{x}_t\right] + \mathbf{b}_c\right) \tag{19}$$

where $\mathbf{w}_i$ and $\mathbf{w}_c$ denote the matrix of weights corresponding to the input gate, respectively, and $\mathbf{b}_i$ $\mathbf{b}_c$ denote the bias weights, respectively. tanh is the activation function, as shown in Equation (20):

$$\tanh(\mathbf{x}) = \frac{e^{\mathbf{x}} - e^{-\mathbf{x}}}{e^{\mathbf{x}} + e^{-\mathbf{x}}} \tag{20}$$

The cell memory unit $\mathbf{c}_t$ represents the unit state vector of the hidden layer at moment t.

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \tag{21}$$

where $\mathbf{c}_{t-1}$ is the previous cell state information and $\odot$ denotes the Hadamard product by element.

The output gate $\mathbf{o}_t$ is used to determine the amount of information that needs to be output by the output gate at the current moment.

$$\mathbf{o}_t = \sigma_s\left(\mathbf{w}_o\left[\mathbf{g}_{t-1}, \mathbf{x}_t\right] + \mathbf{b}_o\right) \tag{22}$$

$$\mathbf{g}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \tag{23}$$

where $\mathbf{w}_o$ and $\mathbf{b}_o$ are the weight matrix and bias parameter of the output gate, respectively, and $\mathbf{g}_t$ is the state of the hidden layer at the current moment, which is also the output state of the LSTM model at the current moment.

4. Chameleon Swarm Algorithm

The CSA models the behavior of chameleons as they search for food in trees, deserts and swamps, and includes the three stages of searching for prey, rotating the eyes to locate the prey, and firing the tongue at high speed to capture the prey.

In an D-dimensional search space, the position of chameleon i at iteration t is:

$$\mathbf{y}_t^i = [y_t^{i,1}, y_t^{i,2}, ..., y_t^{i,D}] \tag{24}$$

where $i = 1, 2, \ldots, SN$, and SN denotes population size.

At the beginning of the algorithm, the initial population is randomly generated, denoted as:

$$y^{i,j} = l^j + r \times (u^j - l^j) \tag{25}$$

where u^j and l^j are the upper and lower bounds of the jth dimensional search space, respectively, and $j = 1, 2, \ldots, D$, r is a uniformly generated random number within [0, 1].

4.1. Search for Prey

The chameleon's search for prey is updated with the following equation:

$$y_{t+1}^{i,j} = \begin{cases} y_t^{i,j} + p_1(P_t^{i,j} - G_t^j)r_2 + p_2(G_t^j - y_t^{i,j})r_1 & r_i \geq p_p \\ y_t^{i,j} + \mu((u^j - l^j)r_3 + l^j)\mathrm{sgn}(rand - 0.5) & r_i < p_p \end{cases} \tag{26}$$

where $P_t^{i,j}$ denotes the best position of chameleon i so far in dimension j at iteration t and G_t^j denotes the global optimum position. p_1 and p_2 are the two parameters that control exploration capacity. r_1, r_2, r_3 and r_i are all uniformly generated random numbers within [0, 1]. p_p is the probability of the chameleon sensing its prey, and is generally taken as 0.1. $sgn\ (rand - 0.5)$ is the direction of rotation, and takes the value of 1 or -1.

μ is a parameter with the following expression:

$$\mu = \gamma \mathrm{e}^{(-\alpha t/T)^\beta} \tag{27}$$

where t and T are the current and maximum number of iterations, respectively. The parameters γ, α and β are three constants, generally taken as 1.0, 3.5 and 3.0, respectively.

4.2. Rotate the Eyes to Locate the Prey

The eyes of chameleons move independently, which allows them to skillfully explore space to locate their prey, in the sense that chameleons will rotate and move to update their position depending on the location of their prey. Thus, the new position vector of the chameleon after rotation is:

$$\mathbf{y}_{t+1}^i = \mathbf{yr}_t^i + \overline{\mathbf{y}}_t^i \tag{28}$$

where $\overline{\mathbf{y}}_t^i$ is the central position vector of the chameleon before rotation. $\mathbf{yr}_t^i$ denotes the center of the rotation coordinates, as follows:

$$\mathbf{yr}_t^i = \mathbf{m} \times \mathbf{yc}_t^i \tag{29}$$

where $\mathbf{m}$ denotes the rotation matrix. $\mathbf{yc}_t^i$ is the centering coordinate, defined as follows:

$$\mathbf{yc}_t^i = \mathbf{y}_t^i - \overline{\mathbf{y}}_t^i \tag{30}$$

$$\mathbf{m} = \mathbf{R}(\theta, \mathbf{z_1}, \mathbf{z_2}) \tag{31}$$

where $\mathbf{z_1}$ and $\mathbf{z_2}$ are two orthogonal vectors in the search space and $\mathbf{R}$ is the rotation matrix on the respective axis defined below. θ denotes the angle of rotation of the chameleon, as defined below:

$$\theta = r\mathrm{sgn}(rand - 0.5) \times 180^{\circ} \tag{32}$$

where r is a uniformly generated random number within [0, 1].

Based on the three dimensions, the rotation matrices for the x and y axes are:

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{33}$$

where ϕ denotes the angle of rotation about the x axis.

$$\mathbf{R}_y = \begin{bmatrix} \cos\eta & 0 & \sin\eta \\ 0 & 1 & 0 \\ -\sin\eta & 0 & \cos\eta \end{bmatrix} \tag{34}$$

where η denotes the angle of rotation about the y axis.

4.3. Capture Prey

Chameleons that are close to their prey are considered to be the best chameleons. When the chameleon is very close to its prey, it sticks out its tongue to catch it, so that its position is updated accordingly. This mechanism helps the chameleon make the most of its searching space and effectively hunt its prey.

The speed of the chameleon's tongue as it falls towards its prey is:

$$v_{t+1}^{i,j} = \omega v_t^{i,j} + c_1(G_t^j - y_t^{i,j})r_1 + c_2(P_t^{i,j} - y_t^{i,j})r_2 \tag{35}$$

where $c_1 = c_2 = 1.75$ and ω is the inertia weight, which is updated as shown in the following equation.

$$\omega = (1 - t/T)^{(\rho\sqrt{(t/T)})} \tag{36}$$

where ρ takes the value of 1. When a chameleon sticks out its tongue to catch its prey, the position of the chameleon is updated according to the following equation:

$$y_{t+1}^{i.j} = y_t^{i,j} + \left((v_t^{i,j})^2 - (v_{t-1}^{i,j})^2 \right) / (2a) \tag{37}$$

where a is the acceleration of the chameleon's tongue, defined as follows:

$$a = 2590 \times \left(1 - e^{-\log(t)}\right) \tag{38}$$

5. Channel Estimation Method for OFDM Systems Based on Improved Chameleon Swarm Algorithm for Optimized LSTM Network

5.1. Chameleon Swarm Algorithm Based on Coupled Variation and Lens-Imaging Learning (CLCSA)

Compared with other swarm intelligence algorithms, CSA has the advantages of simple operation and fewer parameters to adjust. However, CSA still has some shortcomings, such as poor initial population quality, low convergence accuracy and slow convergence speed. This paper proposes the following improvement methods to solve the above problems.

5.1.1. Somersault Foraging Strategy

One of the main shortcomings of CSA is the imbalance between global exploration and local development capabilities. The stronger the global exploration ability, the better the global optimization result, but the convergence rate will be slowed down. The stronger the local development ability, the better the local optimization effect and the faster the convergence speed. However, it is easy to cause precocious phenomena. Only by balancing these two abilities can the algorithm performance be improved. To address this shortcoming, the somersault foraging strategy of the Manta Ray Foraging Optimization algorithm (MRFO) [41] was introduced in this paper to improve the chameleon's prey searching stage.

In the prey search phase, when $p > 0.5$, the global search is carried out according to the original strategy, and when $p \leq 0.5$, the somersault foraging strategy is introduced. In this behavior, each chameleon tends to move around its prey and then move to a new position by performing a somersault foraging strategy. Thus, they always update their position around the best position they have found so far. The search strategy is defined by the following equation.

$$y_{t+1}^{i,j} = \begin{cases} y_t^{i,j} + p_1(P_t^{i,j} - G_t^j)r_2 + p_2(G_t^j - y_t^{i,j})r_1 & r_i \geq p_p, p > 0.5 \\ y_t^{i,j} + \mu((u^j - l^j)r_3 + l^j)\mathrm{sgn}(rand - 0.5) & r_i < p_p, p > 0.5 \\ y_t^{i,j} + S(r_4 P_t^{i,j} - r_5 y_t^{i,j}) & p \leq 0.5 \end{cases} \tag{39}$$

where, S is the somersault factor, which is used to determine the range of the somersault, taking $S = 2$. r_4, r_5 and p are random numbers generated uniformly within [0, 1]. As can be seen from the specific formula of the somersault strategy, with the definition of the somersault range, each individual can move to a position near the current optimal position and a new position between its symmetric position, to search. The range of somersault foraging decreases with the increase in the number of iteration times, which is conducive to a better local search and can effectively improve the convergence rate.

5.1.2. Coupling Center-Drift Initialization and Coupled Boundary Neighborhood Update Strategy

(1) Initialization strategy for coupling center-drifting.

The initialization mode of CSA is random generation, and the individual quality of the population is often low, which will affect the solving speed of the algorithm. Opposition learning strategy has been proved to be an important strategy to improve the quality of the initial solution of the stochastic optimization algorithm. It selects the best solution by generating opposition points. When the search space is small, the processing effect is usually better. However, when the search space is large, the function of this method will be reduced, and a large range of search space points will be found.

In the initialization stage, this paper performs a random offset of the initial solution by means of a central wandering formula, so as to segregate and locate the merit-seeking space and achieve a multi-point simultaneous search for the current optimal position of chameleon individuals, thereby enhancing the diversity of the population. This initialization process is as follows: generate the initial solution $y^{i,j}$ according to Equation (25);

calculate its opposition points according to Equation (40); and perform a central wander on it according to Equation (41):

$$y^{io,j} = u^j + l^j - y^{i,j} \tag{40}$$

$$y^{ic,j} = y^{i,j} + \delta \times y^{io,j} \tag{41}$$

where $y^{io,j}$ and $y^{ic,j}$ denote the position of the opposing point and the central wander of individual i in dimension j, respectively, and δ denotes the coefficient by which the central wander was performed, which was taken empirically and took the value of 0.518.

(2) Correction strategies for coupled boundary neighborhood updates.

During the iterative process, the CSA usually uses the adjacent boundary value instead of the out-of-bounds chameleon. This method has little effect on the population when the number of transboundary chameleons is small. However, if a large number of transboundary chameleons exist, the above treatment methods will reduce the population diversity and optimization efficiency. Therefore, this paper proposes to correct the position of the transboundary chameleon using the boundary neighborhood updating method, which generates random and uniform points in a neighborhood within the boundary to maintain the diversity of the population. The specific correction formula is as follows:

$$y_t^{i,j} = \begin{cases} u^j \times U(0.8, 1) & y_t^{i,j} > u^j \\ l^j + (u^j - l^j) \times U(0, 0.1) & y_t^{i,j} < l^j \end{cases} \tag{42}$$

where U denotes a uniform distribution over the interval.

5.1.3. Stochastic Update Strategy with Coupled Double-Weighting Factors

As can be seen from Equation (35), the larger the inertia weight, the stronger the global search ability, and the smaller the inertia weight, the stronger the local development ability. In order to improve the global search capability of the algorithm in the early iteration process, relatively large inertia weights can be used. In the late iteration period, in order to enhance the local development ability of the algorithm, a relatively small inertia weight can be used to achieve a more efficient optimization ability for the algorithm.

Therefore, this paper proposes a double-weighting factor stochastic perturbation strategy in the prey capture phase. The strategy updates the chameleon by two inversely varying weighting factors ω_1 and ω_2, where ω_1 is non-linearly decreasing and ω_2 is non-linearly increasing. At the beginning of the iteration, the optimal chameleon direction is less influenced, and the weights should be larger at this time to increase the population diversity to a specific extent. As the iterative process proceeds, the chameleon is dynamically updated in the direction of the optimal chameleon under the effect of the inertia weight ω_2, and the convergence speed is effectively improved.

The formula for the double-weighting factor is as follows.

$$\omega_1 = 1 - \cos(\pi t/(2/T)) \tag{43}$$

$$\omega_2 = 1 + \sin(\pi t/(2/T)) \tag{44}$$

The improved prey capture formula has been updated as follows:

$$v_{t+1}^{i,j} = \omega_1 v_t^{i,j} + c_1\omega_2(G_t^j - y_t^{i,j})r_1 + c_2\omega_2(P_t^{i,j} - y_t^{i,j})r_2 \tag{45}$$

$$y_{t+1}^{i,j} = \omega_1 y_t^{i,j} + \omega_2\left((v_t^{i,j})^2 - (v_{t-1}^{i,j})^2\right)/(2a) \tag{46}$$

5.1.4. Lens-Imaging Learning Strategy

The lack of diversity in solving complex optimization problems in CSA leads to precocity. Therefore, the lens-imaging learning strategy is introduced in this paper after the exploration stage to further improve the diversity of the population and promote the algorithm to jump out of the local space restriction and continue searching.

Reverse learning is an improved strategy to expand the search range by calculating the reverse solution of the current position. The use of reverse learning strategy in swarm intelligence algorithms can effectively improve the algorithm's search performance, but there are certain disadvantages of reverse learning; for example, in the late iteration reverse learning cannot make the algorithm effectively jump out of the local optimum, resulting in insufficient convergence accuracy of the algorithm. The reverse learning of lens imaging can effectively solve the above problems. The law of convex lens imaging is a law in the field of optics. Figure 2 shows a diagram of the imaging of a lens. As shown in Figure 2, suppose the projection of object E on the transverse axis is x and the height is h_g. An image E^* can be obtained through a convex lens, and let the projection of image E^* on the transverse axis be x^*. Set the projections of E and E^* on the transverse axis to lie within the interval $[A, B]$. The relationship satisfied by x and the reverse learning individual x^* is shown in Equation (47).

$$\frac{(A+B)/2 - x}{x^* - (A+B)/2} = \frac{h_g}{h_g{}^*} \tag{47}$$

Let $n = h_g/h_g{}^*$. Equation (47) is converted to give x^*, i.e.,

$$x^* = \frac{A+B}{2} + \frac{A+B}{2n} - \frac{x}{n} \tag{48}$$

When $n = 1$, the above equation can be simplified to the following form:

$$x^* = A + B - x \tag{49}$$

Obviously, the inverse solution obtained using general inverse learning is fixed, whereas in lenticular-imaging inverse learning, by adjusting the size of n, dynamically changing inverse solutions can be obtained, further enhancing the diversity of the population and improving the algorithm's ability to find the best.

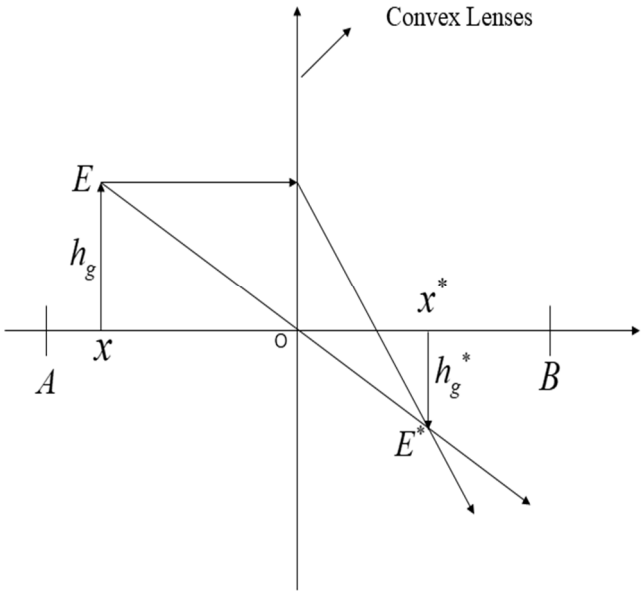

Figure 2. Schematic diagram of lens imaging.

In this paper, by adopting an adaptive scaling factor n, the algorithm is able to better adapt to the entire search process, as follows:

$$n = \left(1 + (t/T)^{1/2}\right)^{q} \tag{50}$$

where q is a constant and takes the value 10.

In order to further enhance the search diversity of solutions and improve the algorithm's search capability, this paper introduces dynamic switching probability, which alternates the execution of the general reverse learning strategy and the lens-imaging reverse learning strategy according to a certain probability, so that the candidate solutions can be dynamically updated with the parameters, retaining the advantages of the general reverse learning strategy while giving full play to the advantages of the lens-imaging reverse learning. The expression is as follows:

$$y_{t+1}^{i.j} = \begin{cases} u^j + l^j - y_t^{i,j} & rand \geq J_t \\ \frac{u^j + l^j}{2} + \frac{u^j + l^j}{2n} - \frac{y_t^{i.j}}{n} & rand < J_t \end{cases} \tag{51}$$

where J_t is the dynamic switching probability, defined as follows:

$$J_t = 0.3(t/T - 1) + 0.8 \tag{52}$$

In view of the shortcomings of the traditional chameleon swarm algorithm, a new Chameleon Swarm Algorithm (CLCSA) based on coupling variation and lens-imaging learning was proposed. Firstly, the coupling center wandering-initialization strategy was adopted to initialize the population and improve the initial population quality. Secondly, the flip foraging strategy is adopted to achieve the balance between global exploration and local exploitation. Then, the lens-imaging learning strategy and the random updating strategy coupled with the double weight factor were used to further improve the diversity of the population and help the population jump out of the local optimal solution. Finally, the location of the transboundary chameleon was corrected using the coupled boundary neighborhood updating strategy, which maintained the diversity of the population by generating random and uniform points in a neighborhood within the boundary. Compared with the traditional Chameleon Swarm Algorithm, the improved Chameleon Swarm Algorithm has stronger searching ability, better balance between search and development ability, and superior performance in convergence speed and convergence accuracy.

5.2. CLCSA Algorithm to Optimize LSTM Network

The complexity of the tuning process of LSTM network models leads to a lack of stability, which can have a large impact on the estimation results. To address this problem, this paper uses the Chameleon Swarm Algorithm (CLCSA) based on coupled variance and lens-imaging learning to determine a better set of initial parameters for the LSTM network model; i.e., the position of the best chameleon can be used as the initial parameters of the network model. The trained network model is used in OFDM systems for signal detection to improve detection performance.

The flow chart of optimizing the LSTM network by using the CLCSA algorithm is shown in Figure 3.

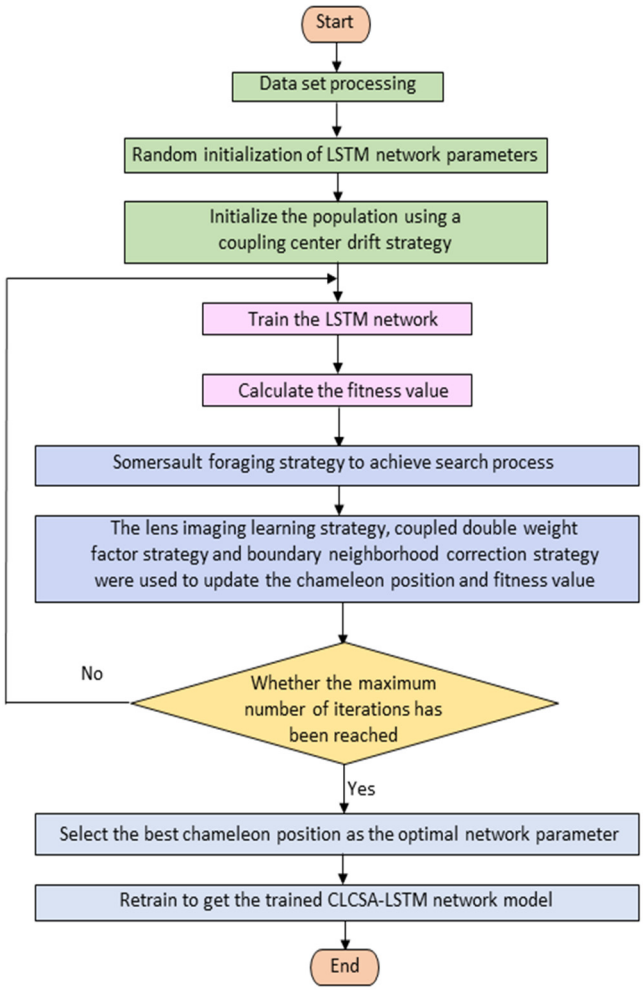

Figure 3. Algorithm flow for optimizing LSTM neural network based on improved Chameleon Swarm Algorithm.

The specific execution steps of optimizing the LSTM network by using the CLCSA algorithm are as follows:

Step 1: Data collection is carried out, 11111111111 and the collected data set is divided into a training set and a validation set. The binary bitstream is first generated and then passed into the OFDM systems, where the modulated data is labelled as the label of the training set and the verification set. The data in the training and validation sets are the pre-processed received signals of the OFDM systems. The pre-processing is as follows: the real and imaginary parts of the received signals at the receiver end are connected in series after serial–parallel conversion and the removal of CP and FFT. Four-fifths of the acquired sample data and labels are taken as the training data set and on-fifth as the validation data set.

Step 2: Randomly initialize the LSTM network parameters. The model parameters to be optimized are selected as the initial learning rate, the maximum number of iterations, and the minimum batch size.

Step 3: The population is initialized, according to the coupling center-drift initialization strategy. The position of each chameleon represents the parameters of the LSTM network, which are the initial learning rate, the maximum number of iterations and the minimum batch size. Set the fitness function of CLCSA algorithm. The fitness function is expressed as the square of the difference between the predicted data and the transmitted data of the neural network.

Step 4: Train the LSTM network.

Step 5: Calculate the fitness value of the chameleon and determine the best chameleon, based on the fitness value.

Step 6: During the prey search phase, global and local searches are implemented based on the somersault foraging strategy; the optimal fitness value in the solution space is searched for and information on the location of the best chameleon is updated.

Step 7: The chameleon position and fitness values are updated, according to the lens-imaging learning strategy.

Step 8: The new position of the chameleon after rotation is generated during the rotating-eye positioning phase.

Step 9: In the prey capture stage, the speed and position formulas of the chameleon are updated, according to the random update strategy coupled with double weight factors, and the position of the transgression chameleon is updated to obtain the global optimal position and its fitness value.

Step 10: Determine whether the maximum number of iterations is reached, and, if it is satisfied, use the current best chameleon position as the initial learning rate and the maximum number of iterations and minimum batch size of the LSTM network; otherwise, return to Step 4.

Step 11: After retraining, a trained CLCSA-LSTM network model is obtained.

5.3. Signal Detection Model for OFDM Systems Based on CLCSA-LSTM

An end-to-end signal detection network is designed based on the LSTM neural network. Firstly, the optimal model is obtained through offline training of OFDM systems simulation data, and then the signal is detected online. This network replaces the channel estimation and signal detection of traditional methods. Compared with the traditional signal detection method, the signal detection method designed in this paper has better performance when the number of leads is small and there is no CP, which improves the signal detection capability. The signal detection model of the CLCSA-LSTM-based OFDM systems is shown in Figure 4, which includes two stages: offline training and online signal detection.

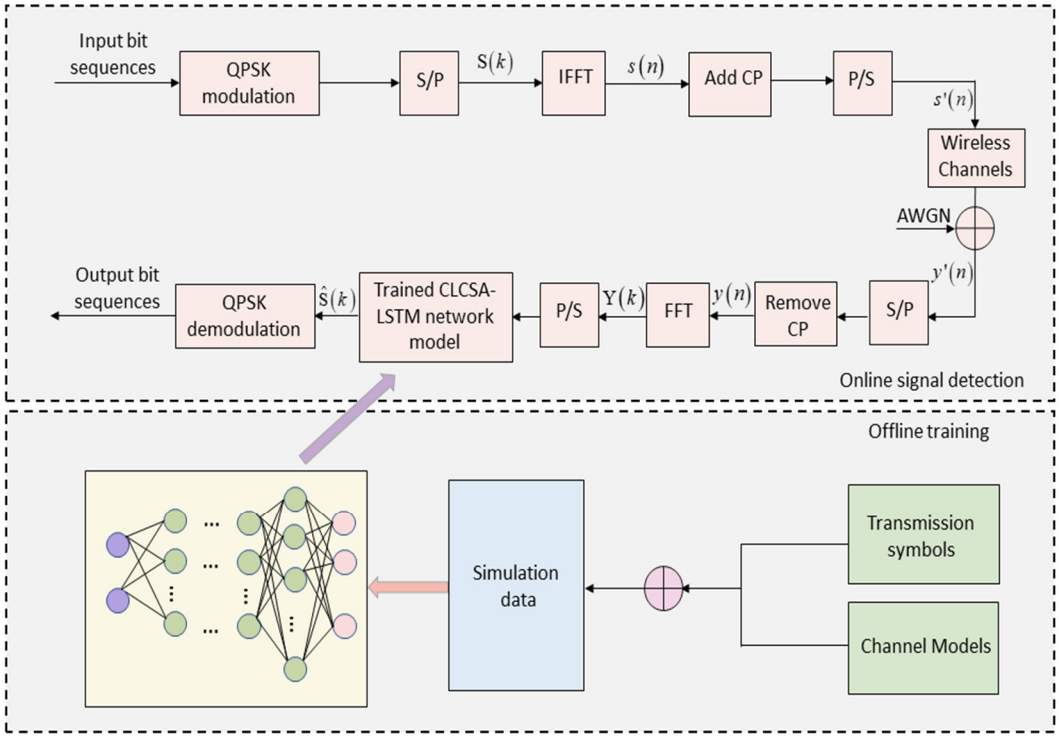

Figure 4. Overall flow structure of the model.

In the offline training stage, the best chameleon position obtained by the CLCSA algorithm is used as the parameters of the LSTM network, and the received OFDM samples are used to train the network model, so that it can better learn the channel state characteristics. These samples are generated by using various information sequences under different channel conditions. Then the trained CLCSA-LSTM network model is obtained.

In the stage of online signal detection, the offline-trained CLCSA-LSTM network model is placed into the OFDM system model for online signal detection. In each simulation, the received data is taken as input, and it passes through the offline-trained CLCSA-LSTM network model, so as to realize the end-to-end recovery of the transmitted data. The process is as follows: the input bit sequence at the transmitter of the OFDM system is modulated, and serial–parallel conversion is also required; pilot information is inserted to obtain the signal $S(k)$, and the time-domain OFDM signal $s(n)$ is obtained after the implementation of the inverse fast Fourier transform. A cyclic prefix is inserted into the signal, and the newly generated signal $s\prime(n)$ is sent out through a multipath channel with noise after the parallel–series transformation. The received signal can be obtained after serial–parallel transformation and removal of the cyclic prefix. The received signal in the frequency domain can be obtained after discrete Fourier transform, and the received signal $y\prime(n)$ can be obtained after serial–parallel transformation. The received signal is taken as input to make it pass the offline-trained CLCSA-LSTM network model, so as to recover the transmitted data in an end-to-end manner and obtain the data $\hat{S}(k)$. After signal demodulation, the bit sequence is recovered.

6. Numerical Result Analysis

The CLCSA-LSTM model was trained with bit error rates (BER) as the evaluation criterion and compared with traditional LS and MMSE methods and other deep learning methods optimized by swarm intelligence algorithms at different signal-to-noise ratios. The simulations were programmed in Matlab and the key parameters of the simulation system used in the OFDM systems are shown in Table 1. Other parameter settings of different swarm intelligent algorithms in the comparative analysis experiment in this paper are shown in Table 2.

Table 1. Simulation parameters.

Parameter	Numerical Value
Number of subcarriers	64
Modulation method	QPSK
Carrier frequency	2.6
Channel type	Multipath channel
Number of paths	24
CP length	16
Channel noise type	AWGN
Data set size	10,000

Table 2. Parameter setting of different swarm intelligence algorithms.

Algorithm	Parameter	Value
GWO	a	[2.0, 0.0]
SCA	a	2
CSA	p1, p2, ρ, c1, c2	0.25, 1.50, 1.0, 1.75, 1.75
CLCSA	p1, p2, ρ, c1, c2	0.25, 1.50, 1.0, 1.75, 1.75

6.1. Performance Analysis of Different Signal Detection Methods

Table 3 and Figure 5 analyze and compare the BER performance of the LSTM signal detection method with parameters set to empirical optima and the LSTM1 signal detection method with non-empirical optima, the CLCSA-LSTM signal detection method proposed

in this paper, and the conventional LS method and the MMSE method in the same channel environment under the condition of the pilot number of 64 and the CP number of 16. From the simulation results in Figure 5, the CLCSA-LSTM signal detection method is significantly better in performance than the LS method and the LSTM1 signal detection method where the parameters are set to non-empirical optimal values. To achieve the same BER value as the MMSE method when the signal-to-noise ratio is greater than 18 dB, there is a difference of approximately 1 to 2 dB between the CLCSA-LSTM signal detection method and the MMSE method, which indicates that the signal detection method is similar in performance to the MMSE method. The CLCSA-LSTM signal detection method is comparable to the LSTM signal detection method where the parameters are set to empirical optima and can compensate for channel distortion to a certain extent.

Table 3. BER performance analysis of different signal detection methods.

Method \ SNR	0	2	4	6	8	10	12	14	16	18	20
LSTM1	0.5538	0.5052	0.4313	0.3520	0.2569	0.1642	0.0928	0.0379	0.0125	0.0033	0.0005
LSTM	0.5866	0.5265	0.4566	0.3734	0.2818	0.1776	0.0963	0.0401	0.0123	0.0026	0.0003
LS	0.6769	0.6339	0.5842	0.5104	0.4171	0.2948	0.1841	0.0881	0.0286	0.0052	0.0005
MMSE	0.6083	0.5490	0.4658	0.3793	0.2781	0.1746	0.0913	0.0310	0.0053	0.0007	0.0000
CLCSA-LSTM	0.5876	0.5260	0.4529	0.3698	0.2718	0.1791	0.0968	0.0435	0.0131	0.0041	0.0002

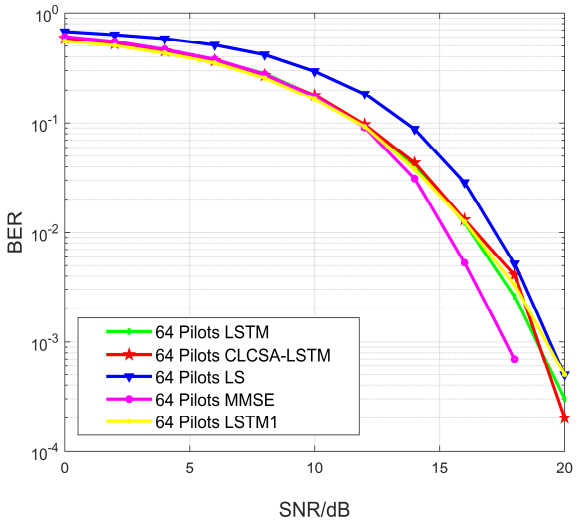

Figure 5. BER performance curves for LS, MMSE, LSTM, LSTM1 and CLCSA-LSTM.

The analysis of Table 3 shows that when the signal-to-noise ratio is greater than 18 dB the CLCSA-LSTM signal detection method proposed in this paper can still effectively reduce the performance of bit error rate, and has better performance than the LSTM signal detection method with the parameter set to the empirical optimal value, the LSTM1 signal detection method with the non-empirical optimal value and the traditional LS method.

Compared with the LSTM signal detection method, the CLCSA-LSTM signal detection method does not require manual adjustment of parameters, which effectively solves the problem of a complicated manual tuning process and the unstable optimization of the LSTM network during training.

Figure 6 compares the BER of the signal detection method based on the conventional Chameleon Swarm Algorithm-optimized LSTM network (CSA-LSTM), the signal detection method based on the LSTM network optimized by the Grey Wolf Optimization Algorithm [42] (GWO-LSTM), the signal detection method based on the LSTM network optimized by the Sine Cosine Algorithm [43] (SCA-LSTM), and the CLCSA-LSTM signal detection method proposed in this paper in the same channel environment, under the

condition that the number of pilots is 64 and the number of CPs is 16. From the simulation results in Figure 6, the BER of all four of the above signal detection methods decreases as the signal-to-noise ratio continues to increase. When the signal-to-noise ratio is greater than 18dB, the BER of the CLCSA-LSTM signal detection method decreases the fastest, and is significantly better than the other three signal detection methods. This indicates that the CLCSA-LSTM signal detection method can learn the statistical characteristics of the channel well, after several iterations. This is due to the superior performance of the CLCSA algorithm itself, which can effectively avoid local optimal solutions and find the global optimal solution in a continuous iterative operation. As a result, the CLCSA-LSTM signal detection method presents better performance compared to the three signal detection methods, namely CSA-LSTM, GWO-LSTM and SCA-LSTM.

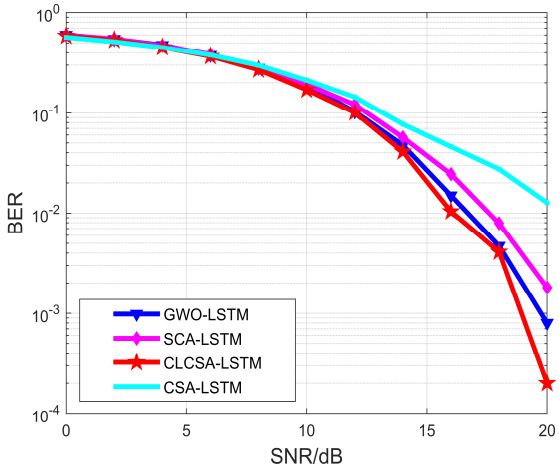

Figure 6. BER performance curves for four different signal detection methods.

6.2. Influence of Pilot Numbers on the Performance of Signal Detection Methods

The number of pilots affects the channel estimation performance, to a certain extent. From a theoretical point of view, the higher the number of pilots, the more accurate the estimation and the higher the accuracy of signal detection, but the higher the number of pilots, the more band resources are occupied and the lower the band utilization. Figure 7 compares the BER performance of the LS method, the MMSE method, the CSA-LSTM signal detection method and the CLCSA-LSTM signal detection method, under conditions where the number of CPs is always 16 and the number of leads is different (64 pilots, 8 pilots). It can be found that the performance of signal detection is improved with an increase in the number of pilots. As can be seen from Figure 7a, the CSA-LSTM signal detection method has the worst performance when 64 pilots are used per frame for signal detection, and the CLCSA-LSTM signal detection method has better performance compared to the CSA-LSTM signal detection method and the LS method. The MMSE method performs best, because the second-order statistics of the channel are assumed to be known. However, from the simulation results in Figure 7b, the BER curves of the conventional LS and MMSE methods decrease slowly when the signal-to-noise ratio is higher than 17 dB, while the CSA-LSTM signal detection method and the proposed CLCSA-LSTM signal detection method are still able to reduce their BER as the signal-to-noise ratio increases, indicating that fewer pilots can be used for channel parameter estimation, thus further improving the spectrum utilization and signal detection accuracy. The BER of the CLCSA-LSTM signal detection method is lower than that of the CSA-LSTM signal detection method, indicating that the improvement in the Chameleon Swarm Algorithm in this paper is effective. Under the condition that other conditions remain unchanged and the number of pilots is small, the CLCSA-LSTM signal detection method proposed in this paper has better detection performance.

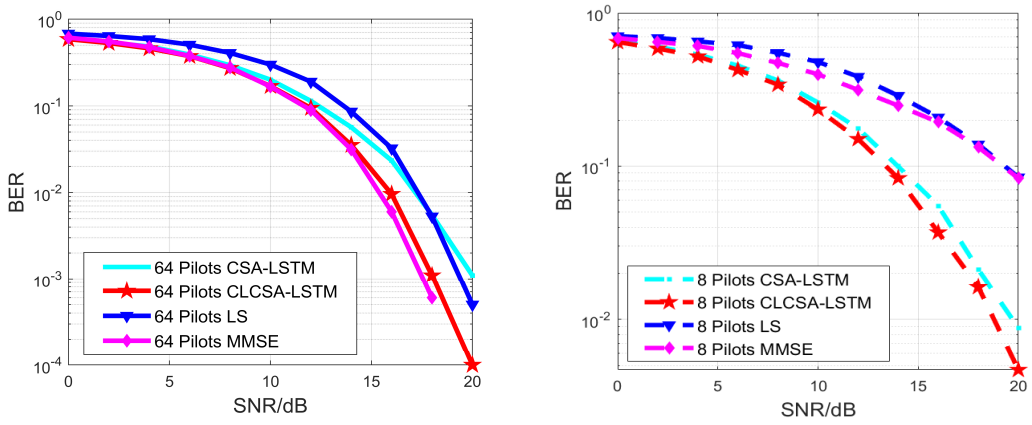

Figure 7. BER performance curves of the LS method, the MMSE method, the CSA-LSTM signal detection method and the CLCSA-LSTM signal detection method under conditions where the number of CPs is always 16 and the number of leads is different (64 pilots, 8 pilots). (**a**) 64 pilots and 16 CPs. (**b**) 8 pilots and 16 CPs.

6.3. Influence of Cyclic Prefixes on the Performance of Signal Detection Methods

CP can solve the ISI and ICI problems caused by delay scaling, but transmission requires time and effort. In this experiment, we studied the performance of removing CP.

Figure 8 shows the BER curves of the OFDM systems with and without CP under the premise of 64 pilots. As can be seen from Figure 8a, the CSA-LSTM signal detection method has the worst performance in the presence of CP, and the CLCSA-LSTM signal detection method has better performance compared to the CSA-LSTM signal detection method and the LS method. The MMSE method has the best performance. As can be seen from Figure 8b, the BER of all four signal detection methods gradually decreases as the signal-to-noise ratio increases in the absence of CP, and when the signal-to-noise ratio is greater than 15 dB the BER curves of the conventional LS method and the MMSE method gradually become slower compared to the CSA-LSTM signal detection method combined with the deep learning method and the CLCSA-LSTM signal detection method. The BER of the CLCSA-LSTM signal detection method decreases the fastest; therefore, the CLCSA-LSTM signal detection method has better performance, proving that the CLCSA-LSTM signal detection method can learn the characteristics of the wireless channel better, based on the training data generated by the model, thus improving the accuracy of signal detection.

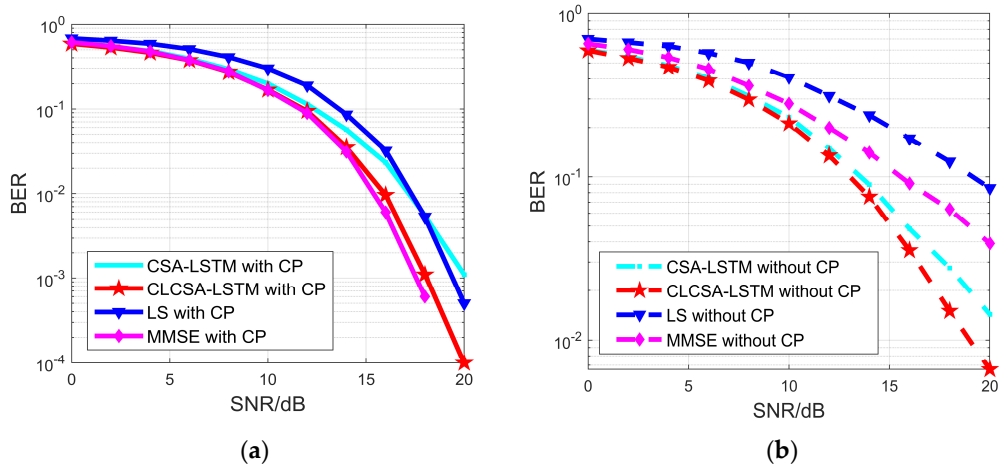

Figure 8. BER performance curves of OFDM systems with and without CP, respectively, under the premise of 64 pilots. (**a**) With CP, 64 pilots. (**b**) Without CP, 64 pilots.

6.4. Influence of Subcarrier Numbers on the Performance of Signal Detection Methods

Figure 9 shows the bit-error-rate curves of the OFDM systems with 16 and 64 subcarriers, respectively, under the conditions of pilot frequency of 64 and CP number of 16. By analyzing Figure 9a,b, it can be found that, with the increase in the number of subcarriers, the bit-error-rate performance curves of the four signal detection methods all decline, which indicates that the number of subcarriers has a certain impact on the system performance. As can be seen from Figure 9a, when the number of subcarriers is 16, the bit error rate of the four signal detection methods can gradually decrease with the increase in the signal-to-noise ratio. The performance of the LS signal detection method is the worst, while that of the CLCSA-LSTM signal detection method is the best. As can be seen from Figure 9b, when the number of subcarriers is 64 the performance of the CSA-LSTM signal detection method is the worst, and the CLCSA-LSTM signal detection method has better performance than the CSA-LSTM signal detection method and the LS method. Therefore, the proposed CLCSA-LSTM signal detection method has good detection performance.

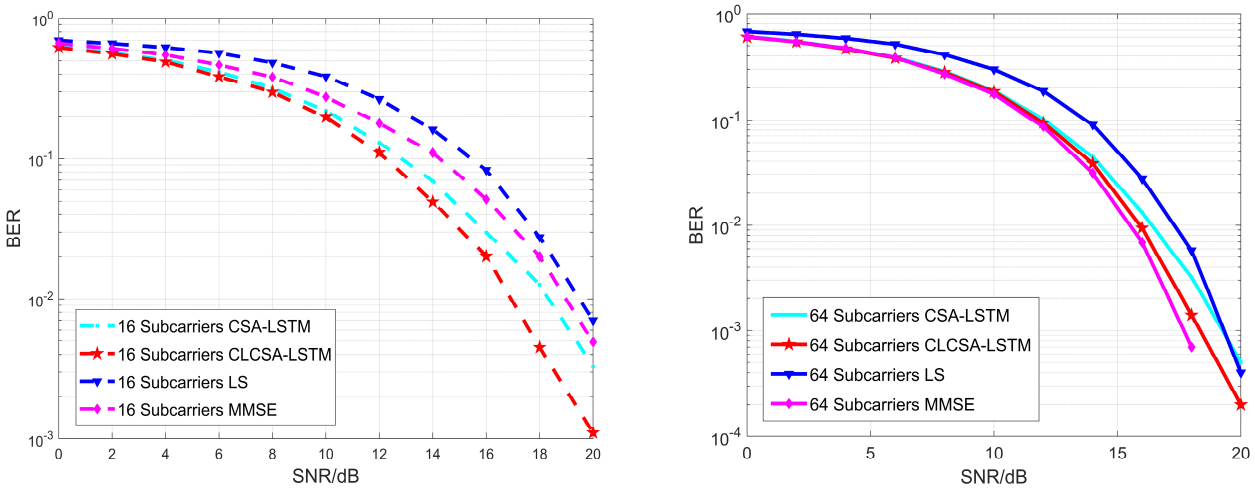

Figure 9. Bit-error-rate performance curve of OFDM system with 16 and 64 subcarriers under the premise of 64 pilots and 16 CPs, respectively. (**a**) 16 subcarriers. (**b**) 64 subcarriers.

6.5. Computing Complexity Analysis

In the basic CSA algorithm, the chameleon population size in the algorithm is set as SN, the maximum number of iterations is T, and the dimension of space is D. As can be seen from the above, the time complexity of basic CSA is $O(SN * T * D)$. The CLCSA algorithm proposed in this paper is based on the basic CSA algorithm, by adopting five improved strategies, such as the coupling center-drift initialization strategy, the somersault foraging strategy, the lens-imaging learning strategy, the coupled double-weight-factor random-update strategy and the coupled boundary neighborhood update strategy. The initialization strategy of the coupled center-drift, the correction strategy of the coupled boundary neighborhood update and the random-update strategy of the coupled double-weight factor do not increase the extra time complexity. According to the execution steps of the CLCSA algorithm, the introduction of the somersault foraging strategy increases the operational complexity by $O(SN * T * D)$. By introducing the lens-imaging learning strategy, the reverse solution with dynamic changes can be obtained, which increases the amount of computation for $O(SN * T * D)$. Therefore, the time complexity of CLCSA is $O(3 * SN * T * D)$. The space complexity of standard CSA is $O(SN * D)$. Compared with CSA, CLCSA can store a matrix with the shape of individual chameleon fitness as $1 * N$. Therefore, the space complexity of CLCSA is $O(SN * D + SN)$. To sum up, the computational complexity of CLCSA is $O(3 * SN * T * D + SN * D + SN)$.

Table 4 compares the computational complexity of the LS method, MMSE method, LSTM signal detection method and CLCSA-LSTM signal detection method.

Table 4. Computational complexity analysis.

Algorithm	Computational Complexity
LS	$O(k)$
MMSE	$O(k^3)$
LSTM	$O(M_1(\mathrm{n_1m_1} + n_1^2 + n_1))$
CLCSA-LSTM	$O(M_1(\mathrm{n_1m_1} + n_1^2 + n_1) + 3 * SN * T * D + SN * D + SN)$

It is assumed that k subcarriers are activated within an OFDM symbol. The computational complexity of the LS method and MMSE method is $O(k)$ and $O(k^3)$, respectively. The computational complexity of the LSTM signal detection method mainly comes from the multiplicative number required for the activation of all neurons in the network layer. The parameters in the network are simplified into two matrices, input U_1 and output V_1. The network training requires M_1 iterations, and n_1 and m_1, respectively, represent the number of neurons in the hidden layer and input layer. The dimension of U_1 is $n_1 \times m_1$, and the dimension of V_1 is $n_1 \times n_1$, so the total multiplicative number required is $M_1(\mathrm{n_1m_1} + n_1^2 + n_1)$, and the computational complexity of the LSTM signal detection method is $O(M_1(\mathrm{n_1m_1} + n_1^2 + n_1))$. The computational complexity of the CLCSA-LSTM signal detection method is $O(M_1(\mathrm{n_1m_1} + n_1^2 + n_1) + 3 * SN * T * D + SN * D + SN)$.

7. Conclusions

In order to improve the accuracy of signal detection in OFDM systems, aiming at the problems of poor communication reliability caused by the fading phenomenon during signal transmission, a complex manual parameter-adjustment process and insufficient optimization stability of the LSTM network signal detection method, this paper proposes a signal detection method (CLCSA-LSTM) based on the coupled variation and lens-imaging learning Chameleon Swarm Algorithm optimizing the long and short-term memory neural network. An improved Chameleon Swarm Algorithm (CLCSA) was proposed by introducing the coupling center-drift initialization strategy, somersault foraging strategy, lens-imaging learning strategy, coupled double-weight-factor random updating strategy and coupled boundary neighborhood updating strategy into the classical Chameleon Swarm Algorithm. The LSTM network model can handle the timing problem better. The CLCSA algorithm is used to find the optimal hyperparameters of the LSTM network, which can avoid the complexity and uncertainty of manual parameter tuning and improve the accuracy of detection. Using the trained network model for signal detection in OFDM systems, the results show that the method not only achieves better signal detection accuracy when the radio channel is affected by noise or other factors, but also reduces the dependence on the CP and the required pilot overhead. When the number of subcarriers changes, the advantages of the CLCSA-LSTM signal detection method can also be retained. In addition, the CLCSA-LSTM signal detection method shows better detection accuracy and universality, compared to the LSTM network signal detection methods, signal detection methods using the traditional Chameleon Swarm Algorithm and other swarm intelligence algorithms for optimizing the LSTM network. Moreover, the estimation results are comparable when compared to the MMSE method, but the method proposed in this paper does not require channel statistics or noise-related information as well as a large number of matrix inversion operations, and therefore has a lower complexity and is more conducive to practical signal transmission.

Author Contributions: Conceptualization, Y.S. and Y.C.; methodology, Y.S. and Y.C.; software, Y.S. and Y.C.; validation, Y.C. and T.L.; formal analysis, Y.S., Y.C. and T.L.; investigation, W.J. and J.G.; resources, T.L. and Q.H.; data curation, Y.C. and J.G.; writing—original draft preparation, Y.S. and Y.C.; writing—review and editing, Y.S. and T.L.; visualization, Y.S. and T.L.; supervision, Y.C. and T.L.; project administration, Y.S. and T.L.; funding acquisition, Y.S. and T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Innovation and Entrepreneurship Training Program for College Students, project No.202210069337 and Tianjin Graduate Research Innovation Project, project No.2022SKYZ390).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Li, L.; Chen, H.; Chang, H.-H.; Liu, L. Deep Residual Learning Meets OFDM Channel Estimation. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 615–618. [CrossRef]
2. Ye, H.; Li, G.Y.; Juang, B.-H. Power of Deep Learning for Channel Estimation and Signal Detection in OFDM Systems. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 114–117. [CrossRef]
3. Mthethwa, B.; Xu, H. Deep Learning-Based Wireless Channel Estimation for MIMO Uncoded Space-Time Labeling Diversity. *IEEE Access* **2020**, *8*, 224608–224620. [CrossRef]
4. Khan, I.; Zafar, M.H.; Ashraf, M.; Kim, S. Computationally Efficient Channel Estimation in 5G Massive Multiple-Input Multiple-output Systems. *Electronics* **2018**, *7*, 382. [CrossRef]
5. Ashraf, N.M.; Mostafa, R.R.; Sakr, R.H.; Rashad, M.Z. Optimizing Hyperparameters of Deep Reinforcement Learning for Autonomous Driving Based on Whale Optimization Algorithm. *PLoS ONE* **2021**, *16*, e0252754. [CrossRef]
6. Nematzadeh, S.; Kiani, F.; Torkamanian-Afshar, M.; Aydin, N. Tuning Hyperparameters of Machine Learning Algorithms and Deep Neural Networks Using Metaheuristics: A Bioinformatics Study on Biomedical and Biological Cases. *Comput. Biol. Chem.* **2022**, *97*, 107619. [CrossRef]
7. Ye, F. Particle Swarm Optimization-Based Automatic Parameter Selection for Deep Neural Networks and Its Applications in Large-Scale and High-Dimensional Data. *PLoS ONE* **2017**, *12*, e0188746. [CrossRef]
8. Braik, M.S. Chameleon Swarm Algorithm: A Bio-Inspired Optimizer for Solving Engineering Design Problems. *Expert Syst. Appl.* **2021**, *174*, 114685. [CrossRef]
9. Rizk-Allah, R.M.; El-Hameed, M.A.; El-Fergany, A.A. Model Parameters Extraction of Solid Oxide Fuel Cells Based on Semi-empirical and Memory-based Chameleon Swarm Algorithm. *Int. J. Energy Res.* **2021**, *45*, 21435–21450. [CrossRef]
10. Yamada, Y.; Uchiyama, M.; Jobashi, M.; Koizumi, T.; Tamai, T.; Sato, N.; Tanabe, J.; Kimura, K.; Ojima, Y.; Murakami, R.; et al. A 20.5 TOPS Multicore SoC with DNN Accelerator and Image Signal Processor for Automotive Applications. *IEEE J. Solid-State Circuits* **2020**, *55*, 120–132. [CrossRef]
11. Yang, X.; Li, F.; Liu, H. A Survey of DNN Methods for Blind Image Quality Assessment. *IEEE Access* **2019**, *7*, 123788–123806. [CrossRef]
12. Nakahara, M.; Nishimura, M.; Ushiku, Y.; Nishio, T.; Maruta, K.; Nakayama, Y.; Hisano, D. Edge Computing-Assisted DNN Image Recognition System with Progressive Image Retransmission. *IEEE Access* **2022**, *10*, 91253–91262. [CrossRef]
13. Oh, Y.R.; Park, K.; Park, J.G. Online Speech Recognition Using Multichannel Parallel Acoustic Score Computation and Deep Neural Network (DNN)- Based Voice-Activity Detector. *Appl. Sci.* **2020**, *10*, 4091. [CrossRef]
14. Punuri, S.B.; Kuanar, S.K.; Kolhar, M.; Mishra, T.K.; Alameen, A.; Mohapatra, H.; Mishra, S.R. Efficient Net-XGBoost: An Implementation for Facial Emotion Recognition Using Transfer Learning. *Mathematics* **2023**, *11*, 776. [CrossRef]
15. Xu, Y.; Wei, Y.; Jiang, K.; Wang, D.; Deng, H. Multiple UAVs Path Planning Based on Deep Reinforcement Learning in Communication Denial Environment. *Mathematics* **2023**, *11*, 405. [CrossRef]
16. Shamasundar, B.; Chockalingam, A. A DNN Architecture for the Detection of Generalized Spatial Modulation Signals. *IEEE Commun. Lett.* **2020**, *24*, 2770–2774. [CrossRef]
17. Wen, C.-K.; Shih, W.-T.; Jin, S. Deep Learning for Massive MIMO CSI Feedback. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 748–751. [CrossRef]
18. Gao, X.; Jin, S.; Wen, C.-K.; Li, G.Y. ComNet: Combination of Deep Learning and Expert Knowledge in OFDM Receivers. *IEEE Commun. Lett.* **2018**, *22*, 2627–2630. [CrossRef]
19. Mei, K.; Liu, J.; Zhang, X.; Cao, K.; Rajatheva, N.; Wei, J. A Low Complexity Learning-Based Channel Estimation for OFDM Systems with Online Training. *IEEE Trans. Commun.* **2021**, *69*, 6722–6733. [CrossRef]
20. Logins, A.; He, J.; Paramonov, K. Block-Structured Deep Learning-Based OFDM Channel Equalization. *IEEE Commun. Lett.* **2022**, *26*, 321–324. [CrossRef]

21. Chi, N.; Zhao, Y.; Shi, M.; Zou, P.; Lu, X. Gaussian Kernel-Aided Deep Neural Network Equalizer Utilized in Underwater PAM8 Visible Light Communication System. *Opt. Express* **2018**, *26*, 26700–26712. [CrossRef]
22. Miao, P.; Chen, G.; Cumanan, K.; Yao, Y.; Chambers, J.A. Deep Hybrid Neural Network-Based Channel Equalization in Visible Light Communication. *IEEE Commun. Lett.* **2022**, *26*, 1593–1597. [CrossRef]
23. Zhang, Y.; Li, J.; Zakharov, Y.; Li, X.; Li, J. Deep Learning Based Underwater Acoustic OFDM Communications. *Appl. Acoust.* **2019**, *154*, 53–58. [CrossRef]
24. Zhang, J.; Cao, Y.; Han, G.; Fu, X. Deep Neural Network-based Underwater OFDM Receiver. *IET Commun.* **2019**, *13*, 1998–2002. [CrossRef]
25. Hou, S.; Fan, Y.; Han, B.; Li, Y.; Fang, S. Signal Modulation Recognition Algorithm Based on Improved Spatiotemporal Multi-Channel Network. *Electronics* **2023**, *12*, 422. [CrossRef]
26. He, H.; Wen, C.-K.; Jin, S.; Li, G.Y. A Model-Driven Deep Learning Network for MIMO Detection. In Proceedings of the 2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Anaheim, CA, USA, 26–29 November 2018.
27. Baek, M.-S.; Kwak, S.; Jung, J.-Y.; Kim, H.M.; Choi, D.-J. Implementation Methodologies of Deep Learning-Based Signal Detection for Conventional MIMO Transmitters. *IEEE Trans. Broadcast.* **2019**, *65*, 636–642. [CrossRef]
28. Soltani, M.; Pourahmadi, V.; Sheikhzadeh, H. Pilot Pattern Design for Deep Learning-Based Channel Estimation in OFDM Systems. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 2173–2176. [CrossRef]
29. Tsai, W.-C.; Chen, C.-W.; Teng, C.-F.; Wu, A.-Y. Low-Complexity Compressive Channel Estimation for IRS-Aided mmWave Systems with Hypernetwork-Assisted LAMP Network. *IEEE Commun. Lett.* **2022**, *26*, 1883–1887. [CrossRef]
30. Balevi, E.; Doshi, A.; Andrews, J.G. Massive MIMO Channel Estimation With an Untrained Deep Neural Network. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 2079–2090. [CrossRef]
31. Moon, S.; Kim, H.; Hwang, I. Deep Learning-Based Channel Estimation and Tracking for Millimeter-Wave Vehicular Communications. *J. Commun. Netw.* **2020**, *22*, 177–184. [CrossRef]
32. Mattu, S.R.; Chockalingam, A. Learning-Based Channel Estimation and Phase Noise Compensation in Doubly-Selective Channels. *IEEE Commun. Lett.* **2022**, *26*, 1052–1056. [CrossRef]
33. Peng, Q.; Li, J.; Shi, H. Deep Learning Based Channel Estimation for OFDM Systems with Doubly Selective Channel. *IEEE Commun. Lett.* **2022**, *26*, 2067–2071. [CrossRef]
34. Sun, Y.; Shen, H.; Du, Z.; Peng, L.; Zhao, C. ICINet: ICI-Aware Neural Network Based Channel Estimation for Rapidly Time-Varying OFDM Systems. *IEEE Commun. Lett.* **2021**, *25*, 2973–2977. [CrossRef]
35. Jiang, R.; Wang, X.; Cao, S.; Zhao, J.; Li, X. Deep Neural Networks for Channel Estimation in Underwater Acoustic OFDM Systems. *IEEE Access* **2019**, *7*, 23579–23594. [CrossRef]
36. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 295–307. [CrossRef]
37. Soltani, M.; Pourahmadi, V.; Mirzaei, A.; Sheikhzadeh, H. Deep Learning-Based Channel Estimation. *IEEE Commun. Lett.* **2019**, *23*, 652–655. [CrossRef]
38. He, H.; Wen, C.-K.; Jin, S.; Li, G.Y. Deep Learning-Based Channel Estimation for Beamspace mmWave Massive MIMO Systems. *IEEE Wirel. Commun. Lett.* **2018**, *7*, 852–855. [CrossRef]
39. Cao, M.; Yao, R.; Xia, J.; Jia, K.; Wang, H. LSTM Attention Neural-Network-Based Signal Detection for Hybrid Modulated Faster-Than-Nyquist Optical Wireless Communications. *Sensors* **2022**, *22*, 8992. [CrossRef]
40. Wang, S.; Yao, R.; Tsiftsis, T.A.; Miridakis, N.I.; Qi, N. Signal Detection in Uplink Time-Varying OFDM Systems Using RNN With Bidirectional LSTM. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 1947–1951. [CrossRef]
41. Zhao, W.; Zhang, Z.; Wang, L. Manta Ray Foraging Optimization: An Effective Bio-Inspired Optimizer for Engineering Applications. *Eng. Appl. Artif. Intell.* **2020**, *87*, 103300. [CrossRef]
42. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
43. Mirjalili, S. SCA: A Sine Cosine Algorithm for Solving Optimization Problems. *Knowl. Based Syst.* **2016**, *96*, 120–133. [CrossRef]