

Article

A Novel Hybrid Artificial Bee Colony-Based Deep Convolutional Neural Network to Improve the Detection Performance of Backscatter Communication Systems

Sina Aghakhani ¹, Ata Larijani ², Fatemeh Sadeghi ³, Diego Martín ^{3,*} and Ali Ahmadi Shahrakht ³

¹ Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, IA 50011, USA; sina75@iastate.edu

² Department of Management Information Systems, Oklahoma State University, Stillwater, OK 74074, USA; ata.larijani@okstate.edu

³ ETSI de Telecomunicación, Universidad Politécnica de Madrid, Av. Complutense 30, 28040 Madrid, Spain; fatemehsadeghi3587@gmail.com (F.S.); aliahmadish@gmail.com (A.A.S.)

* Correspondence: diego.martin.de.andres@upm.es

Abstract: Backscatter communication (BC) is a promising technology for low-power and low-data-rate applications, though the signal detection performance is limited since the backscattered signal is usually much weaker than the original signal. When the detection performance is poor, the backscatter device (BD) may not be able to accurately detect and interpret the incoming signal, leading to errors and degraded communication quality. This can result in data loss, slow data transfer rates, and reduced reliability of the communication link. This paper proposes a novel approach to improve the detection performance of backscatter communication systems using evolutionary deep learning. In particular, we focus on training deep convolutional neural networks (DCNNs) to improve the detection performance of BC. We first develop a novel hybrid algorithm based on artificial bee colony (ABC), biogeography-based optimization (BBO), and particle swarm optimization (PSO) to optimize the architecture of the DCNN, followed by training using a large set of benchmark datasets. To develop the hybrid ABC, the migration operator of the BBO is used to improve the exploitation. Moving towards the global best of PSO is also proposed to improve the exploration of the ABC. Then, we take advantage of the proposed deep architecture to improve the bit-error rate (BER) performance of the studied BC system. The simulation results demonstrate that the proposed algorithm has the best performance in training the benchmark datasets. The results also show that the proposed approach significantly improves the detection performance of backscattered signals compared to existing works.

Keywords: backscatter communication; detection performance; bit-error rate; deep convolutional neural network; hybrid artificial bee colony



Citation: Aghakhani, S.; Larijani, A.; Sadeghi, F.; Martín, D.; Shahrakht, A.A. A Novel Hybrid Artificial Bee Colony-Based Deep Convolutional Neural Network to Improve the Detection Performance of Backscatter Communication Systems. *Electronics* **2023**, *12*, 2263. <https://doi.org/10.3390/electronics12102263>

Academic Editors: Bouziane Brik, Junaid Ahmed Khan and Guangjie Han

Received: 23 March 2023

Revised: 2 May 2023

Accepted: 11 May 2023

Published: 16 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Wireless communication has undergone tremendous advancements in recent years, with backscatter communication (BC) emerging as a promising technique for low-power and low-cost wireless communication. In BC, the transmitter sends a signal to the receiver, which is reflected by a semi-passive device such as a backscatter device (BD), modulating the signal to convey information. In fact, BC allows BDs to transmit data to legitimate receivers by reflecting and modulating an existing radio frequency (RF) signal, instead of generating their own signal [1–5]. The BC technique uses a low-power, battery-free BD as a tag or sensor, which is equipped with a small antenna and a microcontroller. BC eliminates the need for a dedicated transmitter in the BD, which saves power and reduces the size and cost of the device, and this makes it suitable for the Internet of Things (IoT) [6–9].

However, the performance of BC systems is limited by the low signal-to-noise ratio (SNR), interference, and other factors, making it challenging to detect the backscattered signals accurately. The backscatter signal is typically weaker than the original RF signal, due to losses in the BD antenna, reflection, and transmission. As a result, the SNR at the receiver can be low, which can affect the detection performance. The low SNR can also cause interference from other sources, such as thermal noise, multipath fading, or other signals in the same frequency band. BC typically uses simple modulation schemes, such as on-off keying (OOK) or amplitude shift keying (ASK), which have a lower data rate and a higher susceptibility to noise and interference. The modulation scheme can also affect the bit-error rate (BER) performance, as some schemes are more sensitive to changes in the signal phase or frequency. BC performance can be affected by the distance between the BD and the reader. As the distance increases, the backscatter signal strength decreases, and the detection performance becomes worse [10–14]. This distance-dependent performance can be mitigated using a higher-power reader, a higher-gain antenna, or a larger BD antenna. BC can be susceptible to interference from other devices operating in the same frequency band. For example, if there are multiple BDs or readers in the same area, they can interfere with each other's signals, causing errors or collisions [15,16].

When utilizing deep learning (DL)-based approaches for signal detection in BC systems, there are two primary obstacles to overcome. Firstly, due to the low signal power received from the backscatter link in comparison to the direct link, the two hypotheses (whether the BD backscatters ambient RF source signals or not) have a minor difference, making it challenging for DL to complete the classification task. Secondly, unlike the typical classification tasks in computer vision, where a 10% error rate is acceptable, the BD signal detection task in BC systems has a lower BER requirement, usually less than 0.01. To address this challenge, our paper proposes a novel approach to improving the detection performance of BC systems by leveraging the advantage of evolutionary DL. Specifically, we focus on training deep convolutional neural networks (DCNNs) to enhance the detection accuracy of BC.

1.1. Paper Motivation

A review of the literature shows that various machine learning (ML) algorithms have been suggested to improve the detection performance of backscatter communication systems, such as DL and neural networks. However, there is no clear indication of which model is the most effective. Since 2006, DL has been a popular topic in the ML field due to the development of system processing power and data availability. DL models have been shown to be better than traditional ML models. These methods have become more popular due to their ability to reduce computation time and increase the convergence curve. Over the past few years, DCNNs have exhibited notable proficiency in several applications, such as estimation, classification, and detection, particularly for complex and high-dimensional data. DCNNs are well-suited to detecting patterns in complex and noisy signals, making them an ideal candidate for improving the detection performance of BC systems [17–19]. DCNNs are inspired by the way neurons transmit information in biological processes. Unlike traditional networks that only comprise input and output layers, DCNNs are based on multilayer perceptron with a fully connected architecture.

DCNN is a well-known DL approach that has been proven to be effective because of its ability to train the hierarchical layers successfully. It requires minimal pre-processing and fine-tunes all the layers of the network. DCNN reduces the number of parameters by utilizing spatial relationships between features. DCNNs have the ability to automatically acquire progressively more intricate features from unprocessed pixel data, enabling them to capture and depict high-level concepts and abstractions. Fine-tuning a DCNN trained on one task for a related task can enhance its performance and decrease the amount of training data required. DCNNs are also highly parallelizable, which makes them suitable for processing vast amounts of image data simultaneously using multiple GPUs. As a

result, this study employs a DCNN to improve the detection performance of backscatter communication systems.

1.2. Paper Contributions

To optimize the architecture of the DCNN, we develop a novel hybrid algorithm that combines artificial bee colony (ABC) [20–22], biogeography-based optimization (BBO) [23–28], and particle swarm optimization (PSO) techniques. This work proposes a novel approach for improving the detection performance of BC. The proposed algorithm is capable of efficiently searching the vast and complex space of DCNN architectures, identifying the most effective architecture for improving the detection performance of BC. We then train the optimized DCNN using a set of benchmark datasets, and demonstrate the efficacy of our approach by showing that it outperforms existing methods in training benchmark datasets and significantly improves the BER performance of the studied BC system model. Our paper's contributions include proposing a novel approach to improving the detection performance of BC systems, developing a hybrid algorithm for optimizing DCNN architectures, and demonstrating the efficacy of our approach in improving the BER performance of BC systems.

The major contributions of this paper can be summarized as:

- We introduce a novel hybrid optimization algorithm named HABC based on ABC, BBO, and PSO, in which new agents such as neighborhood search, the migration of bees, and movement towards the global best are introduced to improve the exploitation and exploration abilities of the ABC algorithm. In the proposed HABC, the migration operator of the BBO is used to improve the exploitation of the algorithm. Moving towards the global best of PSO is also proposed to improve the exploration of the ABC;
- We use HABC to adjust the optimization parameters of the DCNN and significantly enhance the detection accuracy of the DCNN as the BD signal detector;
- Extensive experiments are carried out using both benchmark functions and BC sample data [29]. The results demonstrate that the suggested HABC-DCNN signal detection technique can attain exceptional BER efficiency in comparison with the related works. The simulation results for well-known real-world benchmark datasets show that the proposed HABC algorithm achieves a better performance compared with some other more well-known algorithms.

2. Literature Review

Several schemes have recently been developed to efficiently detect the BD signals in BC systems. For instance, Liu et al. [30] implemented practical BDs and proposed an energy detector that uses a differential coding scheme to decode BD signals, which is a step towards achieving BC systems. Lu et al. [31] created an improved energy detection method that requires perfect channel state information (CSI) knowledge. To address this issue, Qian et al. [32] designed a semi-coherent detection method that needs only a few pilots and unknown data symbols, substantially reducing signaling overhead and decoding complexity. To avoid channel estimation, Wang et al. [33] applied a differential encoding scheme to BD signals and proposed a minimum BER detector with an optimal detection threshold.

Qian et al. in [34] conducted fundamental studies of BER performance for non-coherent detectors on top of [33]. However, these methods either explicitly require channel estimation or lead to unsatisfactory detection performance. As a solution, ML-based methods have been developed recently to directly recover BD signals using only a few training pilots, without explicitly estimating relevant channel parameters. For example, Zhang et al. [35] proposed a clustering method to extract constellation symbol features, and developed two constellation learning-based signal detection methods that obtain better BER performance. Nevertheless, these schemes were designed for systems where the signal from the RF source is a constellation-modulated signal.

Hu et al. [36] changed the task of BD signal detection to a classification problem, and developed the support vector machine (SVM)-based energy detection scheme to enhance the BER performance. However, this ML technique requires a large number of training samples, and there is a significant gap between the proposed scheme and the optimal one. On the other hand, DL techniques, which utilize a neural network to explore features optimally in a data-driven manner, have demonstrated better performance in various areas [37–42]. However, wireless communication channels in real-world scenarios undergo significant fluctuations over time, and the range of these changes can be very large. Additionally, wireless transmission is often intermittent and can have a variable transmission pattern. The authors of [29] proposed a method based on deep transfer learning (DTL), which utilizes a DNN to extract time-varying features using a small amount of online training data. They showed that their proposed method achieves a BER performance that is quite similar to that of the optimal detection method when perfect CSI is available.

Today, in much of the research, meta-heuristic algorithms are used for learning deep learning architectures [17,18,43–46]. Yang et al. proposed an intelligent identification approach using variational mode decomposition (VMD), composite multi-scale dispersion entropy (CMDE), and a particle swarm optimization deep belief network (PSO-DBN)—namely, VMD-CMDE-PSO-DBN—for bearing faults. Since the accuracy of identifying faults cannot be maximized by manually setting the parameters of DBN nodes, PSO is utilized to optimize the parameters of the DBN model. Based on the experimental comparison and analysis, the VMD-CMDE-PSO-DBN approach has practical application in intelligent fault diagnosis [45].

Training DL architectures is considered to be one of the most difficult problems in ML. Gradient-based methods have significant drawbacks, such as being stuck in local minima in multi-objective cost functions, requiring expensive execution times, and relying on continuous cost functions. Training DLs is an NP-hard problem, and optimizing their parameters using meta-heuristics has become increasingly popular. In recent decades, many meta-heuristic algorithms have been introduced as potential solutions to various engineering problems. It is challenging to adapt exploration and exploitation to solve complicated optimization problems. To overcome these challenges, this paper introduces a novel HABC to train DCNN. Using meta-heuristic algorithms to train DLs enhances the learning process, improves algorithm accuracy, and reduces execution time [17].

3. System Model

This paper examines a common BC system, which includes a semi-passive BD, a reader as a legitimate receiver, and an RF source, as illustrated in Figure 1. The RF source and the passive BD both have one antenna, while the reader has an M-element antenna array for detecting the backscattered signals. As the RF source operates through broadcasting, its signal is received by both the reader and the BD simultaneously. Even though the BD is considered a passive low-power device, it can transmit its binary BD symbols by deciding whether to reflect the RF signals back to the reader. In this scenario, the reader can interpret the BD symbols by detecting the variations in the received signals.

Assuming the baseband discrete-time signal model, the BD's received signal can be expressed as follows:

$$y_B = \sqrt{P_S}h_{SB} + n_B \quad (1)$$

where y_B , P_S , h_{SB} , and n_B denote the received signal at BD, the source's power, the channel coefficient between the RF source and BD, and the additive white Gaussian noise (AWGN) at BD with zero mean and variance δ_B^2 , respectively.

Now, according to Equation (1) and Figure 1, the reader's received signal can be expressed as follows:

$$y_R = \sqrt{P_S}S(t)h_{SB}h_{BR} + \sqrt{P_S}h_{SR} + n_R \quad (2)$$

where y_R , $S(t)$, h_{BR} , h_{SR} , and n_B denote the received signal at the reader, the information signal backscattered from BD (assuming that BD is emitting with a unit power), the channel

coefficient between BD and the reader, the channel coefficient between the RF source and the reader, and the AWGN at the reader with zero mean and variance δ_R^2 , respectively. Here, $t \in \{1, \dots, T\}$ is the t -th BD symbol period. It is worth noting that the noise power produced by the BD's antenna is significantly smaller than the signal from the RF source, thus it is disregarded in Equation (2) according to [42]. According to Equation (2), the received SNR at the reader can be obtained as follows:

$$\Gamma_R = \frac{|\sqrt{P_S}(h_{SB}h_{BR} + h_{SR})|^2}{\delta_R^2} \tag{3}$$

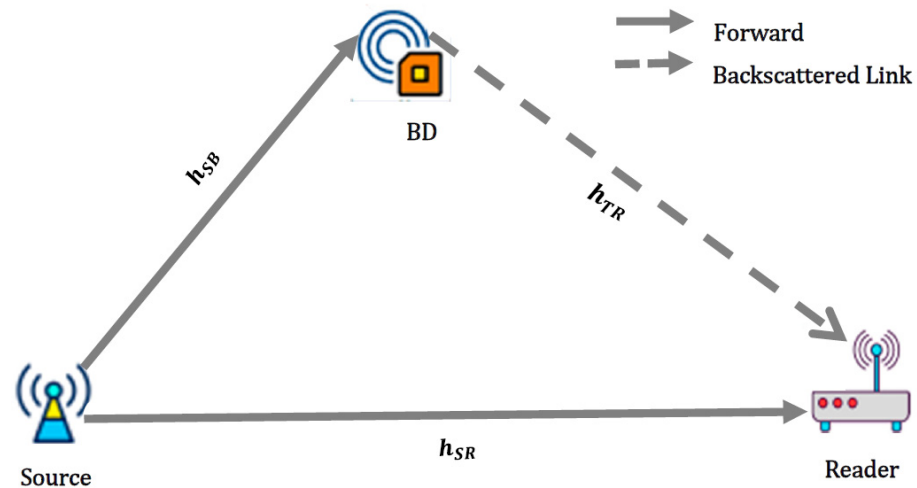


Figure 1. The studied wireless BC system model with a single BD and reader case.

By assuming that the channel coefficient, h , follows Rayleigh distribution, then $g = |h|^2$, as the channel gains will follow the exponential distribution. Then, the SNR at the reader can be rewritten as follows:

$$\Gamma_R = \frac{P_S g_{SB} g_{BR}}{\delta_R^2 d_{SB}^\chi d_{BR}^\chi} + \frac{P_S g_{SR}}{\delta_R^2 d_{SR}^\chi} \tag{4}$$

where d_{SB}^χ , d_{BR}^χ , d_{SR}^χ , and χ denote the distance between the RF source and BD, the distance between the BD and reader, the distance between the RF source and reader, and the channel path-loss exponent, respectively. Additionally, the ratio of the average channel gains between the direct link and the backscattered link in the studied BC system model is determined as their relative coefficient, which is expressed as follows:

$$\Psi_R = \frac{E(g_{SB} g_{BR})}{E(g_{SR})} \tag{5}$$

where $E(\cdot)$ denotes the probability of statistical expectation. Now, let us define the source-to-BD ratio (SBR), which represents the number of RF source symbols in a single BD symbol period, as equal to N . This means that each BD data symbol in the t -th BD symbol period ($S(t)$) remains constant over N periods of RF source symbols. Therefore, for $\forall t$, Equation (2) can be rewritten as follows:

$$y_R = \mu_t \sqrt{P_S} S(t) h_{SB} h_{BR} + \sqrt{P_S} h_{SR} + n_R \tag{6}$$

where $\mu_t \in M = \{0, 1\}$ is used to represent the data symbol for the t -th BD in the context of binary on-off keying modulation. In other words, when $\mu_t = 0$, it means that BD is not backscattering any data symbol, and when $\mu_t = 1$, it means that BD chooses to reflect the RF source together with its information. The objective of the BD signal detector is to retrieve

μ_t according to the received N RF source symbols. Therefore, the process of detecting the BD signal can be expressed as a binary hypothesis testing problem, as follows:

$$\begin{cases} H_1 : y_R = \sqrt{P_S}S(t)h_{SB}h_{BR} + \sqrt{P_S}h_{SR} + n_R \\ H_0 : y_R = \sqrt{P_S}h_{SR} + n_R \end{cases} \quad (7)$$

where H_1 and H_0 signify the hypotheses that $\mu_t = 1$ and $\mu_t = 0$, respectively.

4. The Proposed HABC

ABC, a well-known swarm-based meta-heuristic algorithm, was initially presented by Karaboga in 2005 [20]. The algorithm is modeled after the intelligent search behavior of honey bees. The ABC algorithm has been effectively utilized to address a diverse range of optimization problems, such as function optimization, parameter tuning, and feature selection. It is reputed for its uncomplicated nature, effectiveness, and ability to perform well on both unimodal and multimodal problems. The ABC algorithm utilizes a group of artificial bees to find the best possible solution to a problem. These bees are categorized into three types: employed bees, onlooker bees, and scout bees, all of which play a crucial role in the search process.

The scout bees conduct random searches in the exploration area to identify fresh territories that have yet to be investigated. Exploring space is the responsibility of the scout bees. Some scout bees who possess a strong fitness function are transformed into employed bees. They evaluate the quality of the solutions using a fitness function, and communicate the information about their best solutions to the onlooker bees. Based on the input provided by the employed bees, the onlooker bees choose the most favorable solutions and enhance them by investigating the exploration space near those selected solutions. These bees have the dual role of carrying out both the exploration and exploitation phases of the algorithm.

The ABC algorithm employs two main approaches to arrive at the best possible solution: scout and onlooker bees utilize random search, while employed and onlooker bees engage in neighbor search. Employed bees focus on exploring the solution positions and onlooker bees are created to select new food positions and explore the surrounding areas. In addition, the algorithm generates random scout bees to discover new food positions in the search space. The ABC algorithm can be represented mathematically using Equations (8)–(10) [20].

$$P_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \quad (8)$$

$$V_{ij} = X_{ij} + \varphi_{ij}(X_{ij} - X_{kj}) \quad (9)$$

$$X_L^j = X_{min}^j + rand(0,1)(X_{max}^j - X_{min}^j) \quad (10)$$

where fit_i = the fitness function of the i -th solution, X_{min}^j = the low limit of search space, P_i = the probability of selecting employed bees by onlooker bees, V_{ij} = onlooker bee, X_L^j = scout bees, SN = number of employed bees, X_{max}^j = high limit of search space, $i \in \{1, 2, \dots, SN\}$, j = dimension $\in \{1, 2, \dots, D\}$, k = onlooker bee number, φ_{ij} is the random number $\in [0, 1]$, and L = scout bee number.

The standard ABC has limitations, such as low exploitation and the possibility of getting trapped in local minima. The standard ABC focuses on conducting localized searches around employed bees, while onlooker bees explore new food sources and conduct searches in their vicinity. The main goal of onlooker bees is to enhance the fitness function of solutions through targeted neighborhood searches around the employed bees. Suppose the onlooker bees venture beyond conducting neighborhood searches around employed bees, and also explore several of the best positions. In that case, this will result in the discovery of diverse optimal solutions and the exploration of previously unexplored regions of the search space. Consequently, the ABC executes a more effective local and global search

while simultaneously promoting diversity in the solutions. This leads to an improvement in the algorithm’s convergence rate.

In this paper, two other operators are proposed to improve the exploitation and exploration of the standard ABC. To develop a hybrid ABC (HABC), the migration operator of the BBO is used to improve the exploitation of the algorithm. Moving towards the global best of PSO is also proposed to improve the exploration of the ABC. The HABC algorithm utilizes the onlooker bees to search around the employed bee’s neighborhood, and they also move towards the population’s best solution. This movement promotes diverse solutions and explores different regions of the search space. Essentially, this transfer of information from the best solution to other solutions increases the algorithm’s convergence speed. To achieve this objective, the onlooker bee is designed to make a small movement towards the best solution of the population. This movement is mathematically represented as Equation (11).

$$VG_{ij} = V_{ij} + \gamma(XG_{ij} - V_{ij}) \tag{11}$$

where VG_{ij} = movement towards the global best, XG_{ij} = the best global experience of bees, γ = calibration rate, and V_{ij} = onlooker bee. The process of migration happens when a guest habitat sends its species to a host habitat. In the proposed HABC, the generalized sinusoidal migration model is utilized. The emigration and immigration rates are defined in Equations (12) and (13), respectively, as functions of the number of their species.

$$\mu_{k(j)} = \frac{E}{2} \times \left(-\cos\left(\frac{k(j)\pi}{N} + \varphi\right) + 1 \right) \tag{12}$$

$$\lambda_{k(j)} = \frac{I}{2} \times \left(\cos\left(\frac{k(j)\pi}{N} + \varphi\right) + 1 \right) \tag{13}$$

where $k(j)$ represents the rank of the species living in the j -th habitat, N represents population size, and E and I display the highest rates of emigration and immigration, respectively. Figure 2 shows the flowchart of the proposed HABC.

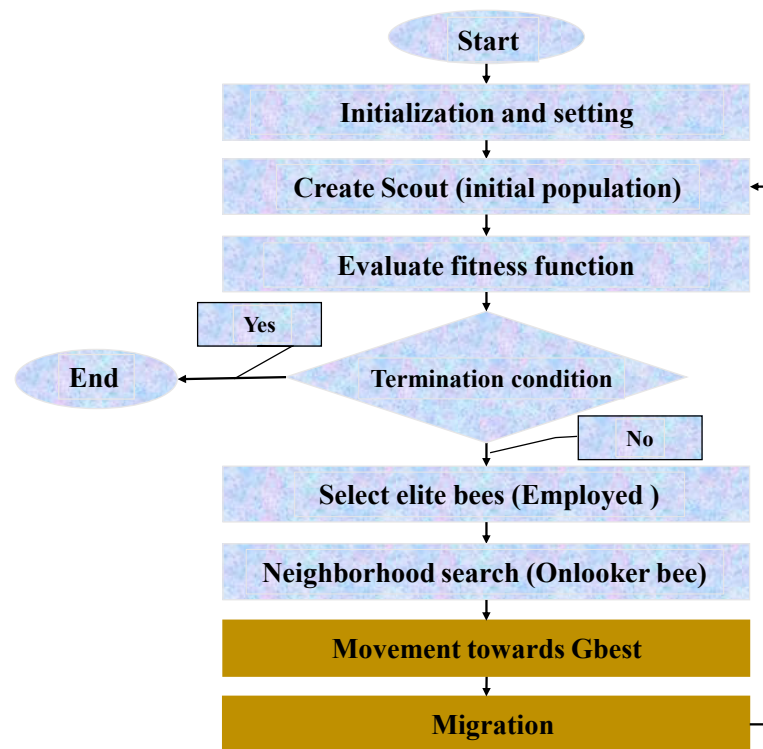


Figure 2. The flowchart of the proposed HABC.

As can be seen, the proposed HABC aims to find a balance between exploration and exploitation by using both local and global search methods. The local search is performed by employed and onlooker bees, as well as a migration model, while scouts, onlookers, and movement towards the global best are used for global search. Figure 3 indicates the proposed migration model, movement towards the global best, and neighborhood search of the HABC algorithm. The fitness function is equivalent to the mean square error (MSE), and a lower MSE corresponds to a better solution.

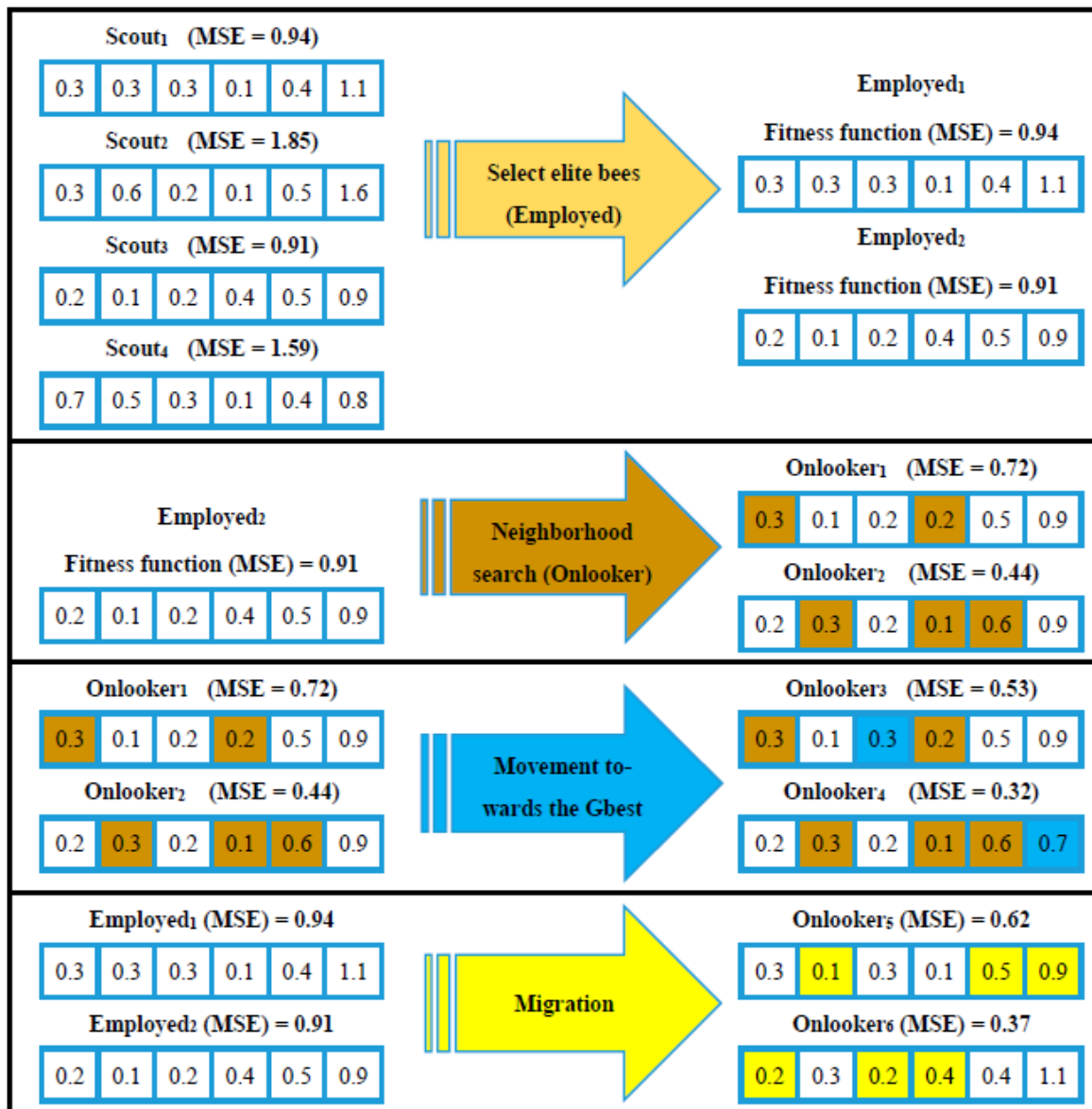


Figure 3. An example of the migration model, movement towards the global best, and neighborhood search of the proposed HABC.

5. HABC-DCNN Design

In recent years, DCNNs have been extensively applied in various fields as one of the most potentially effective DL techniques. The layering capability of DCNNs is a significant advantage that contributes to their success. Figure 4 reveals that the main structure of a DCNN is composed of four types of layers: convolution layers, pooling layers, activation functions, and fully connected neural networks. The role of the convolution layer is to identify characteristics in the input data by employing a convolution filter. This involves taking a set of weights and multiplying them with the input data. The outcome

of this convolution process is then passed through a Rectified Linear Unit (ReLU) as a non-linear activation function. The purpose of the pooling layer is to gradually decrease the dimensions of the input data while preserving only the essential information. Following numerous iterations of convolution and pooling processes, the DCNN culminates in a fully connected neural network [17].

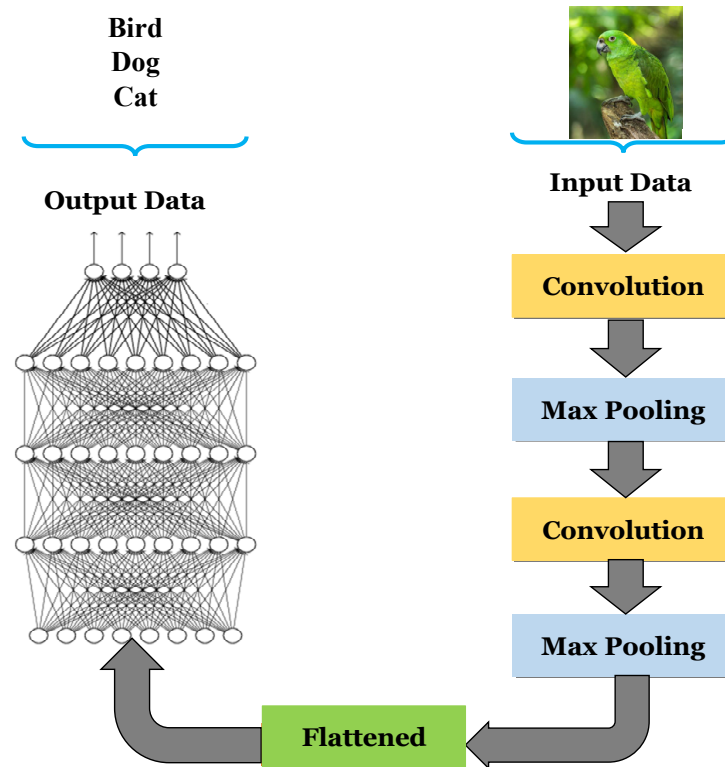


Figure 4. Different layers of a standard DCNN.

The deep architecture used in our experiment consists of four 1D convolutional layers. The number of filters in each layer increases sequentially (32, 64, 128, and 256) and each layer uses a kernel size of 5. To reduce the dimensionality of the feature maps, max-pooling layers with pool sizes of 2 are applied after each convolutional layer. To prevent overfitting, each convolutional layer is followed by a dropout layer with a rate of 0.25. The output of the fourth convolutional layer is flattened and passed through two fully connected layers with 512 and 256 units, respectively. Each of these layers is also followed by a dropout layer with a rate of 0.5.

This paper's significant contribution is training the DCNN using the HABC algorithm, which has been mentioned multiple times. In this approach, we treat the weights and biases of the DCNN as optimization parameters. The HABC algorithm optimally updates the vector of weights and biases to meet the problem conditions, replacing the traditional BP algorithm. Figure 5 illustrates the structure of a bee in the proposed HABC algorithm. Each bee is a vector that contains the weights and biases of the DCNN. The cost function is defined by Equation (14).

$$\text{Mean Square Error (MSE)} = \frac{1}{k} \sum_{i=1}^k (O_i - D_i)^2 \quad (14)$$

where k = the total number of samples, O_i = system output, and D_i = desire.


Bees definition	Weights				Biases			
	W_1	W_2	...	W_n	B_1	B_2	...	B_m

Figure 5. An example of bee definition for DCNN training.

6. Simulation Results

This section is divided into three subsections. In the first and second subsections, the performance of the proposed HABC in real-world engineering problems is evaluated. In the third subsection, the performance of the proposed HABC-DCNN is evaluated. To evaluate the performance of HABC, five well-known and state-of-the-art competitive algorithms called PSO, BBO, ABC, orchard algorithm (OA) [22], and grey wolf optimizer (GWO) [43] are used. To evaluate the performance of HABC-DCNN, two DL architectures called long short-term memory (LSTM) and recurrent neural network (RNN), and three ML models from previous studies are used. All algorithms were coded in MATLAB.

Meta-heuristic algorithms require calibration of various parameters. Thus, it is crucial to identify the optimal parameter combination prior to evaluating the algorithm’s performance. The paper employs a trial-and-error approach to adjust parameter calibration, where each parameter is tested with different values while keeping other variables constant. The fitness function is the primary criterion used to measure and calibrate the algorithm’s parameters. Although a large number of values were tested for each calibration parameter, only a limited number of instances were selected and are presented in Table 1.

Table 1. Parameter setting by trial-and-error method.

Algorithm	Parameter	Value
HABC	The global movement rate (γ)	0.14
	The probability range for migrating into each gene	[0, 1]
	Maximum emigration (I) and immigration (E) coefficient	1
	Number of onlooker bees	80
	Number of employed bees	120
	Number of scout bees (population size)	150
	Iteration	300
OA	N_high	50
	N_low	40
	N_trans	60
	α	0.7
	β	0.3
	Population size	150
	Iteration	300
GWO	C	0.7
	A	0.3
	α	[0, 2]
	Population size	150
	Iteration	300

Table 1. Cont.

Algorithm	Parameter	Value
ABC	Number of onlooker bees	80
	Number of employed bees	120
	Number of scout bees (population size)	150
	Iteration	300
BBO	The probability range for migrating	[0, 1]
	Maximum emigration (I) and immigration (E) rates	1
	Elitism percent	10%
	Mutation rate	0.12
	Population size	150
PSO	Iteration	300
	The inertial movement rate (α)	0.11
	The movement toward the best personal experience rate	0.65
	The movement toward the best global experience rate	0.93
	Population size	150
	Iteration	300

Table 1 displays the optimal parameter values for HABC and other algorithms. The results indicate that a global movement rate of 0.14 yielded the best outcome, as larger values resulted in reduced convergence. The migration probability range of [0, 1] produced the most favorable results. Additionally, the ideal numbers of onlooker and employed bees were determined to be 80 and 120, respectively. The algorithm initially failed to identify an appropriate solution with a population of 40 scout bees, but increasing the population to 110 bees improved the fitness functions of the algorithm. Ultimately, with a population of 150 bees, the algorithm was able to identify superior solutions.

6.1. Pressure Vessel Design

There are countless real-world engineering problems that engineers are working to solve every day [47–49]. This section analyzes how well the suggested algorithms perform when it comes to minimizing the cost of producing a pressure vessel. A schematic view of the problem can be seen in Figure 6, and its formulation is expressed in Equations (15)–(20) [22].

$$f(X) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \quad (15)$$

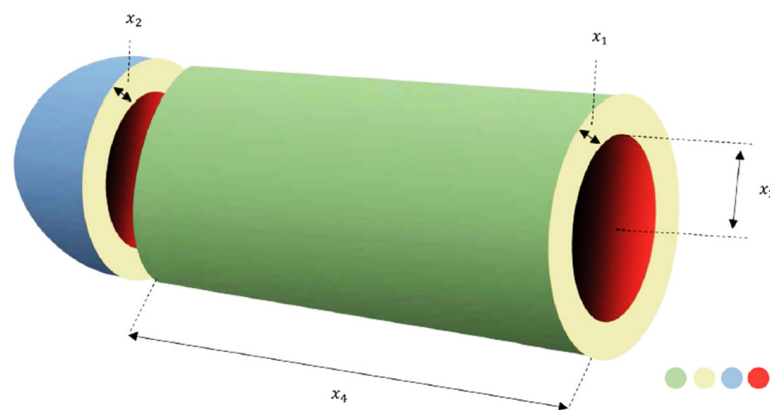


Figure 6. Schematic view of the pressure vessel design.

Subjected to

$$g_1(X) = -x_1 + 0.0193x_3 \leq 0 \tag{16}$$

$$g_2(X) = -x_2 + 0.00954x_3 \leq 0 \tag{17}$$

$$g_3(X) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 129,600 \leq 0 \tag{18}$$

$$g_4(X) = x_4 - 240 \leq 0 \tag{19}$$

$$0 \leq x_1, \quad x_2 \leq 100, \quad 10 \leq x_3, \quad x_4 \leq 200 \tag{20}$$

where x_1 = the shell thickness (T_s), x_2 = the head thickness (T_h), x_3 = the radius of the cylindrical shell (R), and x_4 = the length of the shell (L).

Table 2 presents the outcomes of the meta-heuristic algorithms applied to the pressure vessel system. Among them, the HABC achieved the best outcome for the cost function, with a value of USD 5910/8253. This result was superior to that obtained by other algorithms. Additionally, HABC outperformed the others in the Mean F(x) and SD F(x) measures. Figure 7 illustrates the convergence trend of algorithms for the pressure vessel system. HABC exhibited a faster convergence rate than the other algorithms.

Table 2. The outcomes of the algorithms proposed for the pressure vessel system.

Algorithm	Best F(x)	Mean F(x)	SD F(x)
HABC	5910.8253	5922.8652	0.0041590
OA	5948.1589	5975.9625	0.0098562
GWO	6072.3698	6220.1946	0.0856987
ABC	6143.8965	6486.1785	2.1896527
BBO	6128.1289	6759.7563	4.8965232
PSO	6286.3214	7169.7248	8.7452328

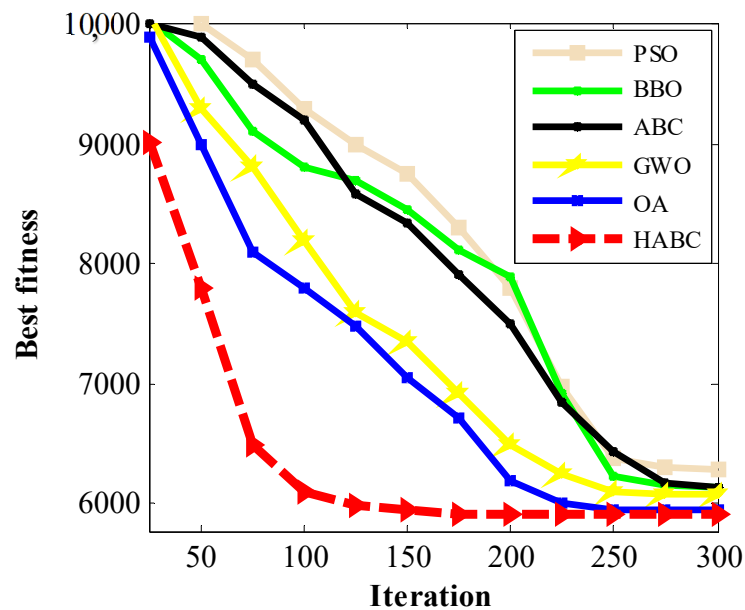


Figure 7. The convergence trend of algorithms for the pressure vessel system.

6.2. Tension Springs Problem

In this section, the effectiveness of the HABC in designing tension springs is evaluated. The schematic view of this problem is depicted in Figure 8 and can be expressed using Equations (21)–(26) [22].

$$f(X) = (x_3 + 2)x_2x_1^2 \tag{21}$$

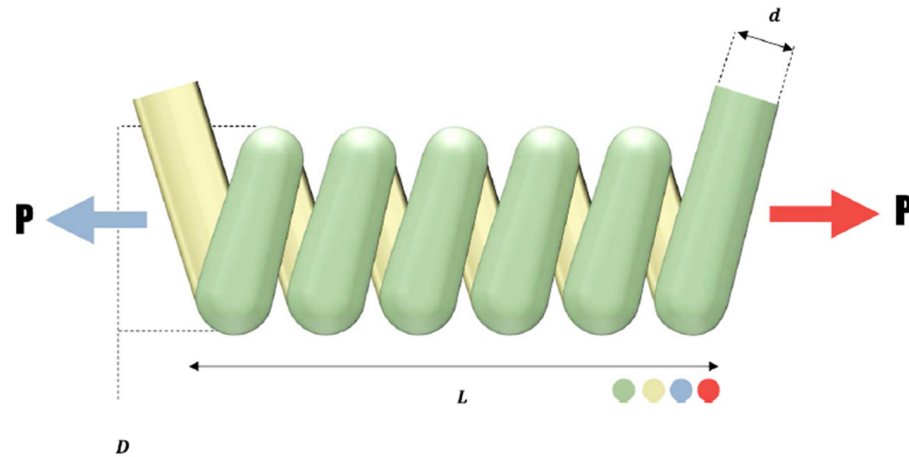


Figure 8. Schematic view of the tension spring problem.

Subjected to

$$g_1(X) = 1 - \frac{x_3x_2^3}{71,785x_1^4} \leq 0 \tag{22}$$

$$g_2(X) = \frac{4x_2^2 - x_1x_2}{12,566(x_2x_1^3 - x_1^4)} + \frac{1}{5108x_1^2} - 1 \leq 0 \tag{23}$$

$$g_3(X) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0 \tag{24}$$

$$g_4(X) = \frac{x_1+x_2}{1.5} - 1 \leq 0 \tag{25}$$

$$0.05 \leq x_1 \leq 2.00, \quad 0.25 \leq x_2 \leq 1.30, \quad 2 \leq x_3 \leq 15 \tag{26}$$

where x_1 = wire diameter (d), x_2 = mean coil diameter (D), and x_3 = the number of active coils (N).

The outcomes of the algorithms applied to the tension spring problem are presented in Table 3. As demonstrated, HABC obtained the optimal value of the objective function, which was 0.012653. The convergence curve of the algorithms for this problem is depicted in Figure 9, where it can be observed that HABC converges more rapidly than other algorithms.

Table 3. The outcomes of the proposed algorithms for the tension spring problem.

Algorithm	Best F(x)	Mean F(x)	SD F(x)
HABC	1.26653×10^{-2}	1.28845×10^{-2}	0.0000274
OA	1.26678×10^{-2}	1.29508×10^{-2}	0.0001456
GWO	1.27253×10^{-2}	1.38786×10^{-2}	0.0189652
ABC	1.27662×10^{-2}	1.41256×10^{-2}	1.2451960
BBO	1.27723×10^{-2}	1.55412×10^{-2}	2.9125355
PSO	1.27945×10^{-2}	1.96352×10^{-2}	4.0189632

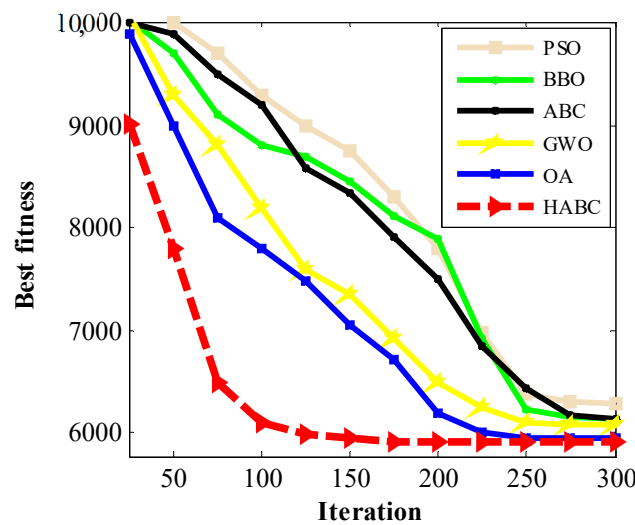


Figure 9. The convergence trend of algorithms for the tension spring problem.

6.3. Detecting Backscatter Signals

In this section, we conduct numerous simulations to assess how well the proposed HABC-DCNN performs in detecting backscattered signals. Our focus is on a BC system containing a single-antenna RF source and BD, and a multi-antenna reader, as described in Section 3. We use 50,000 examples as training and 2000 examples as test datasets, produced by using the data augmentation techniques in [29,44]. Furthermore, the SNR is expressed in Equation (4), and the relative coefficient (as expressed in Equation (5)) is initialized as $\Psi = -20$ dB.

Figure 10 shows the BER performance versus SNR of the proposed signal detection scheme in this paper, compared with the other schemes. According to this figure, our HABC-DCNN method has demonstrated exceptional performance in backscattered signal detection, surpassing other state-of-the-art techniques. Our simulations show that HABC-DCNN achieves the best BER performance in the detection of backscattered signals. Moreover, our analysis indicates that increasing the SNR leads to a significant reduction in BER. This implies that HABC-DCNN is highly reliable in detecting backscattered signals even in the presence of noise.

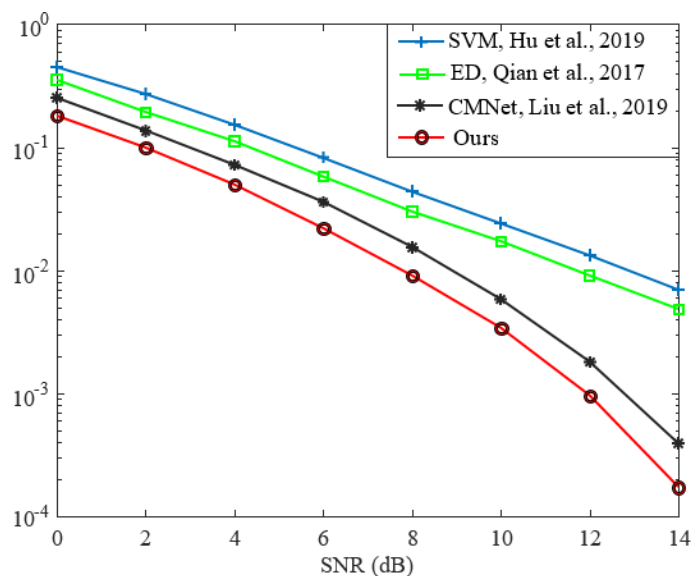


Figure 10. BER versus SNR for $N = 20$ and $\Psi_R = -20$ dB [29,32,36].

According to Figure 11, it can be observed that the BER decreases as the number of RF source symbols, N , increases. This signifies that our method has a high sensitivity in detecting backscattered signals, and its performance can be further enhanced by increasing the number of RF source symbols used in the detection process. The results obtained from our simulations also confirm the effectiveness of HABC-DCNN compared with other works.

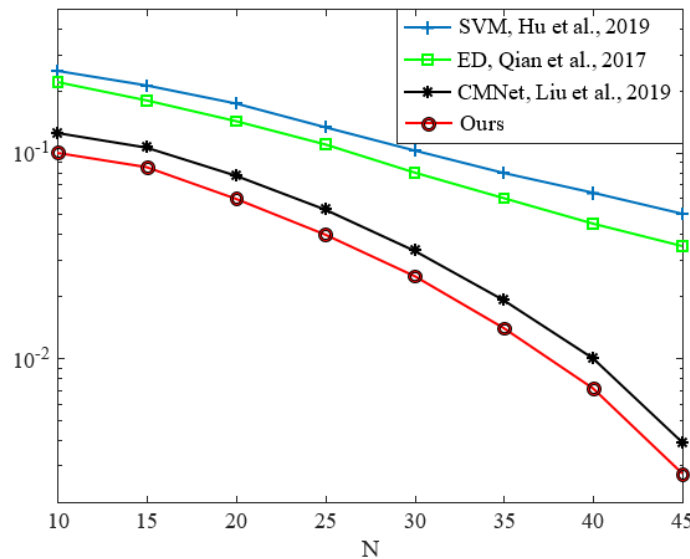


Figure 11. BER versus N for SNR = 5 dB and $\Psi_R = -20$ dB [29,32,36].

In addition, Figure 12 reveals that increasing the ratio of the average channel gains between the direct link and the backscattered link, Ψ_R , results in a decrease in BER. This highlights the ability of HABC-DCNN to effectively distinguish between the direct and backscattered links, and its sensitivity to changes in the channel gain ratios. This figure also shows that HABC-DCNN can provide more reliable and accurate backscattered signal detection compared to the other works, particularly in scenarios where the direct and backscattered links exhibit significant differences in channel gain.

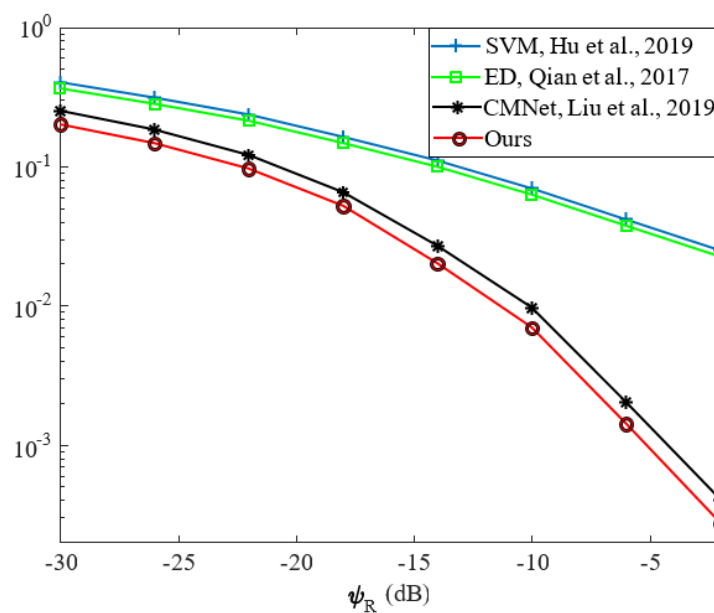


Figure 12. BER versus Ψ_R for $N = 20$ and SNR = 5 dB [29,32,36].

The simulation results in Figure 13 indicate that the BER increases as the distance between the BD and the reader, d_{BR} , increases. This analysis reveals the limitations of conventional backscattering methods in long-range sensing applications, as the signal power decreases with increasing distance, resulting in a degraded BER. However, HABC-DCNN shows promise in addressing this limitation, as it can effectively learn and capture the underlying features of the backscattered signal, enabling it to provide more accurate and reliable detection even at greater distances.

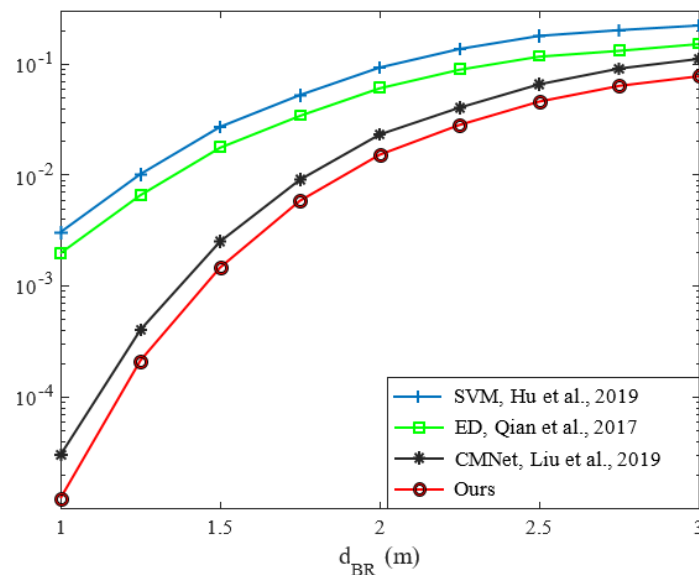


Figure 13. BER versus d_{BR} for $N = 20$ and $\text{SNR}_{BD} = 15$ dB [29,32,36].

This section also assesses the performance of the HABC-DCNN and other architectures for detecting backscatter signals. The evaluation involves conducting sensitivity, accuracy, and specificity analyses, which are based on the confusion matrix and can be computed using Equations (27)–(29).

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (27)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (28)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} \quad (29)$$

where TP = true positive, FN = false negative, TN = true negative, and FP = false positive. Table 4 displays the sensitivity, specificity, and accuracy of various evolutionary architectures designed for detecting backscatter signals. It is evident from the table that the HABC-DCNN architecture outperforms the others in terms of sensitivity, specificity, and accuracy in both the training and validation datasets. The HABC-DCNN architecture attained accuracy values of 98.32% and 99.26% in the test and training datasets, respectively. In addition, the HABC-DCNN obtained sensitivity values of 98.89% and 99.74% in the test and train datasets, respectively.

Figures 14 and 15 depict a comparison of architectures in the training and validation datasets, respectively. Based on the figures, the ranking of the architectures from highest to lowest is as follows: HABC-DCNN, OA-DCNN, GWO-DCNN, BBO-DCNN, ABC-DCNN, PSO-DCNN, RNN, standard DCNN, and LSTM. The outcomes show that the suggested architectures have been effectively trained using meta-heuristic algorithms, as evidenced by the stability of accuracy across different hybrid DL architectures in both the test and train datasets.

Table 4. The results of architectures in the test and train datasets.

DL Models	Train			Test		
	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
HABC-DCNN	99.74%	96.48%	99.26%	98.89%	95.24%	98.32%
OA-DCNN	98.85%	95.91%	98.79%	98.32%	94.92%	98.16%
GWO-DCNN	98.34%	95.21%	98.27%	98.03%	94.36%	97.34%
ABC-DCNN	97.76%	94.82%	97.67%	96.74%	93.88%	96.61%
BBO-DCNN	97.82%	94.72%	97.73%	96.84%	93.76%	96.72%
PSO-DCNN	97.42%	94.31%	97.23%	96.51%	93.21%	95.96%
RNN	96.14%	93.29%	96.52%	95.27%	92.35%	94.86%
Standard DCNN	95.91%	92.88%	96.43%	95.39%	92.09%	94.51%
LSTM	95.18%	92.43%	96.19%	94.95%	92.28%	94.29%

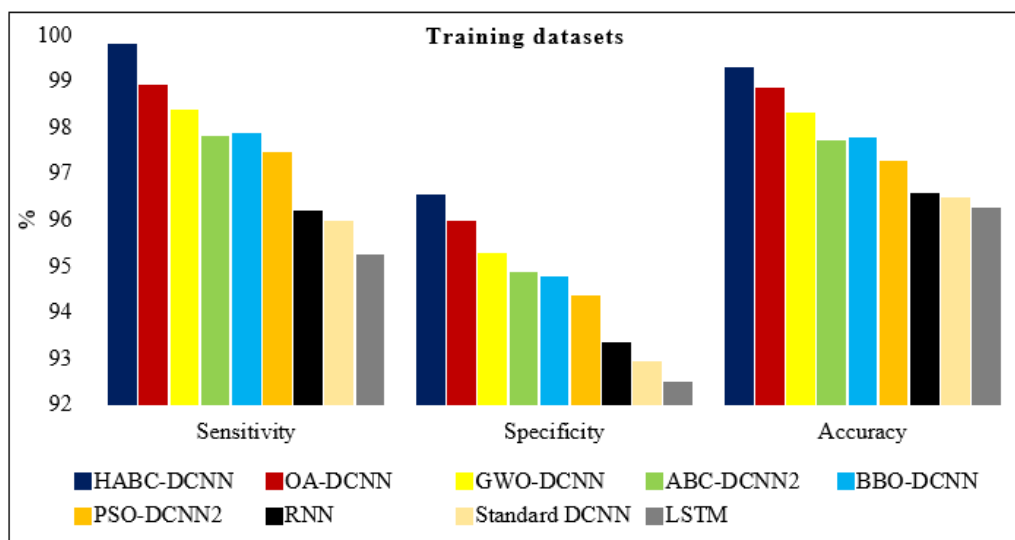


Figure 14. Comparison of DL models in training datasets.

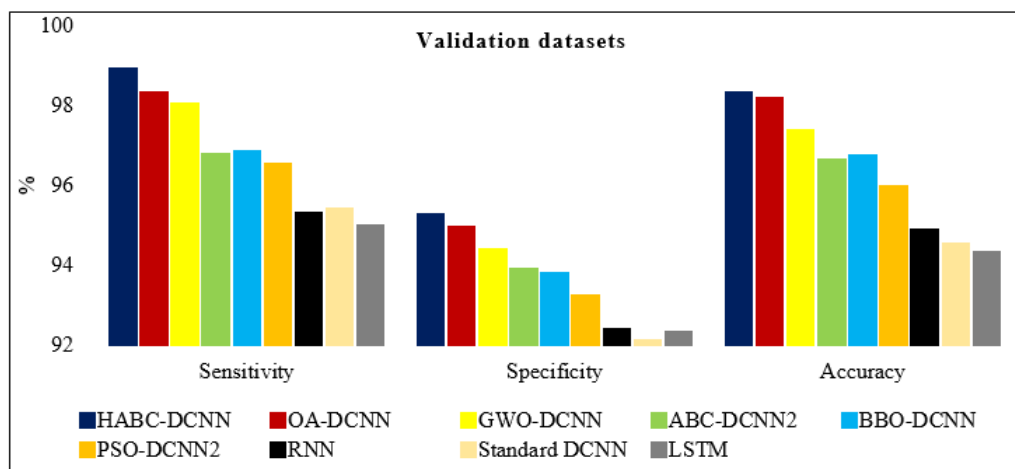


Figure 15. Comparison of DL models in validation datasets.

According to Figure 16, the ROC curves of various architectures are depicted, representing a useful tool for evaluating model performance and comparing different classifiers. The area under the curve (AUC) of HABC-DCNN outperforms other architectures, which is clearly visible in the graph.

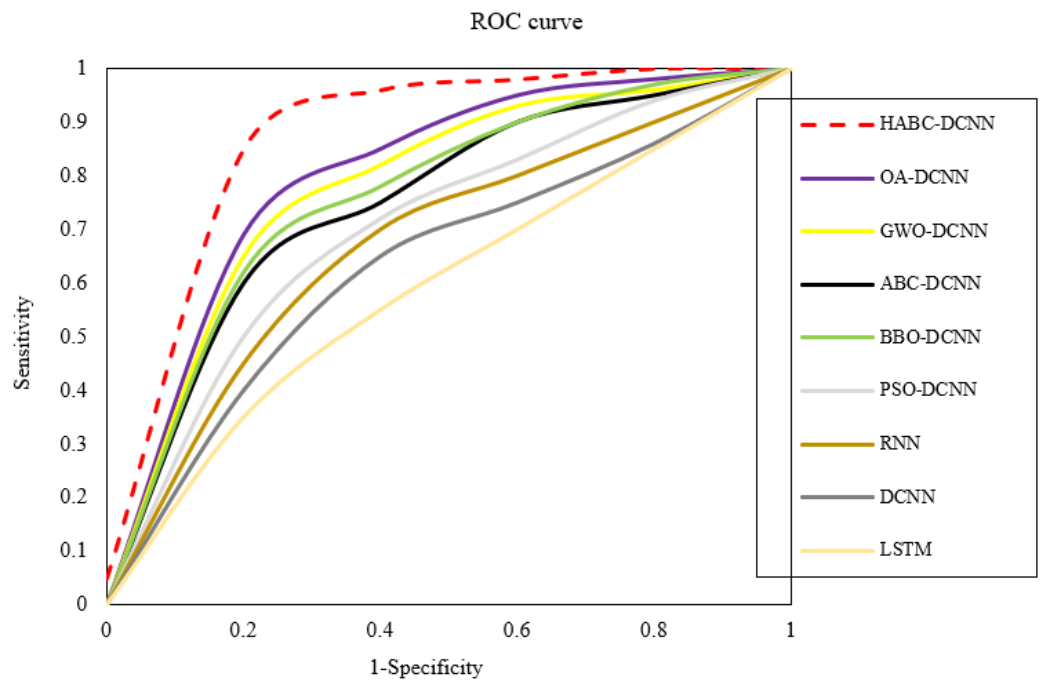


Figure 16. Comparison of the ROC curves of DL architectures.

Mean square error (MSE) criteria are also used to compare the proposed models in Table 5. The HABC-DCNN architecture has a lower MSE than the other architectures, indicating that the proposed approach is effective for use in this problem. In the proposed HABC, the migration operator of the BBO is used to improve the exploitation of the algorithm. Moving towards the global best of PSO is also proposed to improve the exploration of the ABC. As a result, the algorithm is aided in steering clear of getting stuck in local minimal. Figure 17 illustrates that the HABC-DCNN architecture converges faster than the other architectures. At epoch = 100, the HABC-DCNN architecture has achieved nearly the lowest MSE, while other architectures have higher MSE. However, the HABC-DCNN architecture exhibits high stability and rapid convergence as the epochs proceed.

Table 5. The MSE values of the different architectures.

Algorithm	MSE	
	Training Datasets	Validation Datasets
HABC-DCNN	0.00008	0.00096
OA-DCNN	0.00086	0.00896
GWO-DCNN	0.02186	0.19652
ABC-DCNN	0.12452	0.47562
BBO-DCNN	0.10592	0.35896
PSO-DCNN	0.21745	0.59856
RNN	0.42691	0.69853
Standard DCNN	0.55239	0.75263
LSTM	0.59852	0.88745

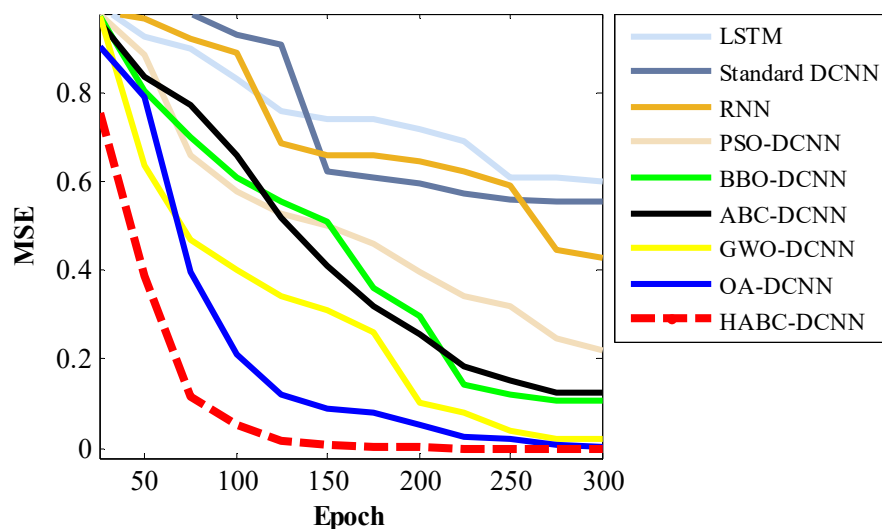


Figure 17. The convergence trend of DL architectures.

7. Conclusions

This paper has presented a new evolutionary deep learning method to enhance the detection performance of BC systems. The proposed approach trains the DCNN as a signal detector using a hybrid optimization algorithm based on ABC, BBO, and PSO to optimize the DCNN architecture. The research leverages the benefits of the proposed deep architecture to improve the BER performance of the BC system. The simulation results indicate that the proposed HABC-DCNN achieves superior performance in training the benchmark datasets and significantly enhances the detection performance of backscattered signals compared to previous works.

Overall, optimizing DL models using meta-heuristic algorithms remains a difficult task, requiring further investigation. HABC, like many other meta-heuristics, employs a range of operators that designers must accurately model to effectively apply to real problems. Additionally, the computational time required for HABC in real-world scenarios poses a significant challenge, including the computation of fitness functions for all possible solutions and the selection of the optimal solution.

There are several possible research directions for future studies. Fine-tuning selective parameters and thresholds in the equations of HABC is an area that requires further work. Additionally, applying HABC to various fields, such as image processing, smart homes, data mining, big data, and industry, could be a valuable contribution. Since acquiring labeled data is often expensive, the next generation of DL models will be more focused on semi-supervised and unsupervised learning. In this context, clustering algorithms could be employed to improve the performance of DLs.

Author Contributions: Conceptualization, S.A., A.L., F.S., D.M. and A.A.S.; methodology, A.L., D.M. and F.S.; software, S.A. and A.L.; validation, A.L. and F.S.; investigation, S.A.; writing—original draft preparation, S.A. and A.A.S.; writing—review and editing, S.A., A.L., F.S. and D.M.; supervision, D.M.; funding acquisition, D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cao, Y.; Xu, S.; Liu, J.; Kato, N. IRS Backscatter Enhancing Against Jamming and Eavesdropping Attacks. *IEEE Internet Things J.* **2023**, *2023*, 1–12. [\[CrossRef\]](#)
2. Miri, S.; Kaveh, M.; Shahhoseini, H.S.; Mosavi, M.R.; Aghapour, S. On the security of ‘an ultra-lightweight and secure scheme for communications of smart meters and neighborhood gateways by utilization of an ARM Cortex-M microcontroller’. *IET Inf. Secur.* **2023**, *17*, 544–551. [\[CrossRef\]](#)
3. Kaveh, M.; Martín, D.; Mosavi, M.R. A lightweight authentication scheme for V2G communications: A PUF-based approach ensuring cyber/physical security and identity/location privacy. *Electronics* **2020**, *9*, 1479. [\[CrossRef\]](#)
4. Aghapour, S.; Kaveh, M.; Martín, D.; Mosavi, M.R. An ultra-lightweight and provably secure broadcast authentication protocol for smart grid communications. *IEEE Access* **2020**, *8*, 125477–125487. [\[CrossRef\]](#)
5. Basharat, S.; Hassan, S.A.; Mahmood, A.; Ding, Z.; Gidlund, M. Reconfigurable intelligent surface-assisted backscatter communication: A new frontier for enabling 6G IoT networks. *IEEE Wirel. Commun.* **2022**, *29*, 96–103. [\[CrossRef\]](#)
6. Aghapour, S.; Kaveh, M.; Mosavi, M.R.; Martín, D. An ultra-lightweight mutual authentication scheme for smart grid two-way communications. *IEEE Access* **2021**, *9*, 74562–74573. [\[CrossRef\]](#)
7. Kaveh, M.; Mosavi, M.R. A lightweight mutual authentication for smart grid neighborhood area network communications based on physically unclonable function. *IEEE Syst. J.* **2020**, *14*, 4535–4544. [\[CrossRef\]](#)
8. Khan, W.U.; Jameel, F.; Ihsan, A.; Waqar, O.; Ahmed, M. Joint optimization for secure ambient backscatter communication in NOMA-enabled IoT networks. *Digit. Commun. Netw.* **2023**, *9*, 264–269. [\[CrossRef\]](#)
9. Kaveh, M.; Aghapour, S.; Martín, D.; Mosavi, M.R. A secure lightweight signcryption scheme for smart grid communications using reliable physically unclonable function. In Proceedings of the International Conference on Environment and Electrical Engineering and Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), Madrid, Spain, 9–12 June 2020; pp. 1–6.
10. Wu, W.; Wang, X.; Hawbani, A.; Yuan, L.; Gong, W. A survey on ambient backscatter communications: Principles, systems, applications, and challenges. *Comput. Netw.* **2022**, *216*, 109235. [\[CrossRef\]](#)
11. Lotfy, A.; Kaveh, M.; Martín, D.; Mosavi, M.R. An efficient design of Anderson PUF by utilization of the Xilinx primitives in the SLICEM. *IEEE Access* **2021**, *9*, 23025–23034. [\[CrossRef\]](#)
12. Liang, Y.C.; Zhang, Q.; Wang, J.; Long, R.; Zhou, H.; Yang, G. Backscatter communication assisted by reconfigurable intelligent surfaces. *Proc. IEEE* **2022**, *110*, 1339–1357. [\[CrossRef\]](#)
13. Najafi, F.; Kaveh, M.; Martín, D.; Reza Mosavi, M. Deep PUF: A highly reliable DRAM PUF-based authentication for IoT networks using deep convolutional neural networks. *Sensors* **2021**, *21*, 2009. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Khan, W.U.; Ihsan, A.; Nguyen, T.N.; Ali, Z.; Javed, M.A. NOMA-enabled backscatter communications for green transportation in automotive-industry 5.0. *IEEE Trans. Ind. Inform.* **2022**, *18*, 7862–7874. [\[CrossRef\]](#)
15. Fard, S.S.; Kaveh, M.; Mosavi, M.R.; Ko, S.B. An efficient modeling attack for breaking the security of XOR-Arbiters PUFs by using the fully connected and long-short term memory. *Microprocess. Microsyst.* **2022**, *94*, 104667. [\[CrossRef\]](#)
16. Liu, W.; Huang, K.; Zhou, X.; Durrani, S. Next generation backscatter communication: Systems, techniques, and applications. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 69. [\[CrossRef\]](#)
17. Kaveh, M.; Mesgari, M.S. Application of meta-heuristic algorithms for training neural networks and deep learning architectures: A comprehensive review. *Neural Process. Lett.* **2022**, 1–104. [\[CrossRef\]](#)
18. Baniasadi, S.; Rostami, O.; Martín, D.; Kaveh, M. A novel deep supervised learning-based approach for intrusion detection in IoT systems. *Sensors* **2022**, *22*, 4459. [\[CrossRef\]](#)
19. Toro, U.S.; ElHalawany, B.M.; Wong, A.B.; Wang, L.; Wu, K. Machine-learning-assisted signal detection in ambient backscatter communication networks. *IEEE Netw.* **2021**, *35*, 120–125. [\[CrossRef\]](#)
20. Karaboga, D. *Technical Report-tr06: An Idea Based on Honey Bee Swarm for Numerical Optimization*; Erciyes University, Engineering Faculty, Computer Engineering Department: Kayseri, Turkey, 2005; Volume 200, pp. 1–10.
21. Kaveh, M.; Mesgari, M.S. Hospital site selection using hybrid PSO algorithm-Case study: District 2 of Tehran. *Sci.-Res. Q. Geogr. Data (SEPEHR)* **2019**, *28*, 7–22.
22. Kaveh, M.; Mesgari, M.S.; Saedian, B. Orchard Algorithm (OA): A new meta-heuristic algorithm for solving discrete and continuous optimization problems. *Math. Comput. Simul.* **2023**, *208*, 19–35. [\[CrossRef\]](#)
23. Kaveh, M.; Mesgari, M.S.; Martín, D.; Kaveh, M. TDMBBO: A novel three-dimensional migration model of biogeography-based optimization (case study: Facility planning and benchmark problems). *J. Supercomput.* **2023**, *79*, 9715–9770. [\[CrossRef\]](#)
24. Li, X.D.; Wang, J.S.; Hao, W.K.; Wang, M.; Zhang, M. Multi-layer perceptron classification method of medical data based on biogeography-based optimization algorithm with probability distributions. *Appl. Soft Comput.* **2022**, *121*, 108766. [\[CrossRef\]](#)
25. Mosavi, M.R.; Kaveh, M.; Khishe, M.; Aghababaie, M. Design and implementation a sonar data set classifier using multi-layer perceptron neural network trained by elephant herding optimization. *Iran. J. Mar. Technol.* **2018**, *5*, 1–12.
26. Zhang, Z.; Gao, Y.; Zuo, W. A Dual Biogeography-Based Optimization Algorithm for Solving High-Dimensional Global Optimization Problems and Engineering Design Problems. *IEEE Access* **2022**, *10*, 55988–56016. [\[CrossRef\]](#)
27. Rabiei, H.; Kaveh, M.; Mosavi, M.R.; Martín, D. MCRO-PUF: A Novel Modified Crossover RO-PUF with an Ultra-Expanded CRP Space. *Comput. Mater. Contin.* **2023**, *74*, 4831–4845. [\[CrossRef\]](#)

28. Li, X.; Chen, J.; Zhou, D.; Gu, Q. A modified biogeography-based optimization algorithm based on cloud theory for optimizing a fuzzy PID controller. *Optim. Control Appl. Methods* **2022**, *43*, 722–739. [[CrossRef](#)]
29. Liu, C.; Wei, Z.; Ng, D.W.K.; Yuan, J.; Liang, Y.C. Deep transfer learning for signal detection in ambient backscatter communications. *IEEE Trans. Wirel. Commun.* **2020**, *20*, 1624–1638. [[CrossRef](#)]
30. Liu, V.; Parks, A.; Talla, V.; Gollakota, S.; Wetherall, D.; Smith, J.R. Ambient backscatter: Wireless communication out of thin air. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 39–50. [[CrossRef](#)]
31. Lu, K.; Wang, G.; Qu, F.; Zhong, Z. Signal detection and BER analysis for RF-powered devices utilizing ambient backscatter. In Proceedings of the International Conference on Wireless Communications & Signal Processing (WCSP), Nanjing, China, 15–17 October 2015; pp. 1–5.
32. Qian, J.; Gao, F.; Wang, G.; Jin, S.; Zhu, H. Semi-coherent detection and performance analysis for ambient backscatter system. *IEEE Trans. Commun.* **2017**, *65*, 5266–5279. [[CrossRef](#)]
33. Wang, G.; Gao, F.; Fan, R.; Tellambura, C. Ambient backscatter communication systems: Detection and performance analysis. *IEEE Trans. Commun.* **2016**, *64*, 4836–4846. [[CrossRef](#)]
34. Qian, J.; Gao, F.; Wang, G.; Jin, S.; Zhu, H. Noncoherent detections for ambient backscatter system. *IEEE Trans. Wirel. Commun.* **2016**, *16*, 1412–1422. [[CrossRef](#)]
35. Zhang, Q.; Guo, H.; Liang, Y.C.; Yuan, X. Constellation learning-based signal detection for ambient backscatter communication systems. *IEEE J. Sel. Areas Commun.* **2018**, *37*, 452–463. [[CrossRef](#)]
36. Hu, Y.; Wang, P.; Lin, Z.; Ding, M.; Liang, Y.C. Machine learning based signal detection for ambient backscatter communications. In Proceedings of the IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.
37. Sadeghi, F.; Rostami, O.; Yi, M.K.; Hwang, S.O. A deep learning approach for detecting COVID-19 using the chest X-ray images. *CMC-Comput. Mater. Contin.* **2023**, *74*, 751–768.
38. Sadeghi, F.; Larijani, A.; Rostami, O.; Martín, D.; Hajirahimi, P. A Novel Multi-Objective Binary Chimp Optimization Algorithm for Optimal Feature Selection: Application of Deep-Learning-Based Approaches for SAR Image Classification. *Sensors* **2023**, *23*, 1180. [[CrossRef](#)] [[PubMed](#)]
39. Kaveh, M.; Mesgari, M.S.; Khosravi, A. Solving the local positioning problem using a four-layer artificial neural network. *Eng. J. Geospat. Inf. Technol.* **2020**, *7*, 21–40.
40. de Oliveira, R.A.; Bollen, M.H. Deep learning for power quality. *Electr. Power Syst. Res.* **2023**, *214*, 108887. [[CrossRef](#)]
41. Aslani, S.; Jacob, J. Utilisation of deep learning for COVID-19 diagnosis. *Clin. Radiol.* **2023**, *78*, 150–157. [[CrossRef](#)]
42. Liu, Y.; Ye, Y.; Hu, R.Q. Secrecy outage probability in backscatter communication systems with tag selection. *IEEE Wirel. Commun. Lett.* **2021**, *10*, 2190–2194. [[CrossRef](#)]
43. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [[CrossRef](#)]
44. Shorten, C.; Khoshgoftaar, T.M. A survey on image data augmentation for deep learning. *J. Big Data* **2019**, *6*, 60. [[CrossRef](#)]
45. Yang, E.; Wang, Y.; Wang, P.; Guan, Z.; Deng, W. An intelligent identification approach using VMD-CMDE and PSO-DBN for bearing faults. *Electronics* **2022**, *11*, 2582. [[CrossRef](#)]
46. Li, W.; Zhang, L.; Chen, X.; Wu, C.; Cui, Z.; Niu, C. Predicting the evolution of sheet metal surface scratching by the technique of artificial intelligence. *Int. J. Adv. Manuf. Technol.* **2021**, *112*, 853–865. [[CrossRef](#)]
47. Rajabi, M.S.; Beigi, P.; Aghakhani, S. Drone Delivery Systems and Energy Management: A Review and Future Trends. *arXiv* **2022**, *2206*, 10765.
48. Aghakhani, S.; Rajabi, M.S. A new hybrid multi-objective scheduling model for hierarchical hub and flexible flow shop problems. *AppliedMath* **2022**, *2*, 721–737. [[CrossRef](#)]
49. Rajabi, M.; Habibpour, M.; Bakhtiari, S.; Rad, F.; Aghakhani, S. The development of BPR models in smart cities using loop detectors and license plate recognition technologies: A case study. *J. Future Sustain.* **2023**, *3*, 75–84. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.