IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 30 April 2023, accepted 15 May 2023, date of publication 18 May 2023, date of current version 1 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3277625

RESEARCH ARTICLE

A Metaheuristics-Based Hyperparameter Optimization Approach to Beamforming Design

KIEU-XUAN THUC, (Member, IEEE), HOANG MANH KHA, (Member, IEEE), NGUYEN VAN CUONG, AND TONG VAN LUYEN, (Member, IEEE)
Faculty of Electronics, Hanoi University of Industry, Hanoi 100000, Vietnam
Corresponding author: Tong Van Luyen (luyentv@haui.edu.vn)

ABSTRACT The paradigm shift from "connected things" to "connected intelligence" is anticipated to be made possible by the sixth-generation wireless systems, which typically use millimeter wave beamforming to mitigate the significant propagation loss. However, beamforming design in millimeter wave communications poses many different challenges owing to the large antenna arrays with the limitation of radio frequency chains and analog beamforming architectures. To circumvent this problem, deep learning models have recently been utilized as a disruptive method for solving difficult optimization problems in sixth-generation mobile systems, such as maximizing spectral efficiency. However, it is still unclear how to produce high-performance deep learning models which require considering appropriate hyperparameters. This study proposes a metaheuristics-based approach for optimizing hyperparameters that are used to build optimized deep learning models to maximize spectral efficiency. The research results demonstrate that the proposed approach-based models establish higher spectral efficiency than the state-of-the-art approach-based models and the reference model whose hyperparameters are based on empirical trials.

INDEX TERMS Hyperparameter optimization, beamforming, metaheuristics, millimeter wave, large-scale antenna arrays.

I. INTRODUCTION

Since the first generation of mobile telecommunications was introduced in the 1970s, wireless communication technology has advanced incredibly quickly. By 2030, newly developed data-hungry applications and a greatly expanded wireless network will have required the use of the sixth generation (6G) communication, which is a significant improvement over other network generations and might cover nearly the entire surface of the earth as well as the vicinity of space. In addition, as the number of wireless consumer devices and the Internet of Things grows rapidly, the amount of mobile data transfer nearly doubles every year, surpassing that of cable communication. Therefore, in the future 6G network, millimeter wave (mmWave) technology will play a significant part in attaining the anticipated network performance and communication responsibilities with greater speed and reliability than previous generations of networks [1].

The associate editor coordinating the review of this manuscript and approving it for publication was Chengpeng Hao.

Millimeter-wave communication with gigahertz or tens of gigahertz bandwidths is also viewed as a possible technology for 6G wireless systems. Communication in these bandwidths will ease the spectrum deficiency and capacity constraints of existing wireless systems [2].

Large-phased arrays are typically used in mmWave communication to mitigate the significant propagation loss using mmWave beamforming, which includes hybrid or analog beamforming when one or several radio frequency (RF) chains are present. Analog and hybrid beamforming are bound by the constraint of constant modulus since only phase shifters are used to adjust excited antenna weights [2]. Fully digital beamforming systems are impractical for mmWave/Sub-THz frequency because each antenna element requires a specialized RF transceiver chain, which is neither cost-effective nor energy-effective to construct for large-scale arrays and bandwidths [3]. In addition, analog beamforming, in which phase shifting is accomplished in the analog domain, has been frequently used owing to its affordable price and ease of implementation [1].

The needs for the 6G systems have necessitated the granular optimization of radio resources and the efficient acquisition of network-related data [4]. Due to the huge size, high density, the varied quality of services, and integrated multi-functional cross-layer architecture, 6G optimization problems might be exceedingly complex and time-sensitive, posing many challenges to the development of effective optimization algorithms. Deep learning (DL) has recently been utilized as a disruptive method to solve difficult optimization problems in 6G and to support a number of artificial intelligence services and the Internet of Everything applications [4]. It has also been proven to be a useful tool for dealing with difficult non-convex problems and high-computability concerns owing to its excellent recognition and representation capabilities [5]. To enable a paradigm shift from traditional optimization theory-based approaches for employing more promising DL architectures, DL-based optimization algorithm design aims to achieve near-optimal performance with excellent computing efficiency for challenging large-scale optimization problems in 6G systems [4]. In particular, superior performance, scalability and generalizability, computational efficiency, and robustness are some benefits of using DL for large-scale optimization.

Hyperparameters, however, allow the performance of the DL approach to be greatly tuned. The values of these parameters must be carefully chosen in order to get the best performance because they typically have a significant impact on the learner's complexity, behavior, speed, and other aspects. Human trial-and-error selection of these values is time-consuming, prone to error, frequently biased, and computationally impossible to reproduce unreproducible. As the mathematical formulation of hyperparameter optimization (HPO) is basically black-box optimization with higher-dimensional spaces, it is preferable to transfer this task to suitable algorithms in order to improve efficiency and guarantee reproduction [6]. Over the past 20 to 30 years, numerous HPO strategies have been developed to facilitate and automate the search for hyperparameter combinations with optimal performance. However, more advanced HPO techniques are not utilized as frequently as they could (or should) be. This may be due to a combination of the following reasons [6]: (i) a lack of understanding of HPO techniques by prospective users, who could consider them as complicated ''black boxes''; (ii) low belief among prospective users in the superiority of HPO processes over rudimentary methods, resulting in doubt over the anticipated return on investment (time); (iii) the absence of guidance on the selection and configuration of pertinent HPO approaches to the issue at hand; (iv) difficulty accurately defining the search space of HPO approaches. The primary objective of HPO is to automate the process of searching hyperparameters and enable users to utilize optimized DL models for real-world problems. A DL model's optimal model architecture is expected to be attained using an HPO procedure. To effectively utilize HPO approaches, it is essential to choose an appropriate optimization strategy to identify optimal hyperparameters. Numerous HPO problems are non-convex or non-differentiable optimization problems. Therefore, traditional optimization approaches dealing with these HPO problems might lead to a local solution rather than a global solution [7].

Though traditional optimization algorithms can be effective for the local search, metaheuristic algorithms, also known as metaheuristics, have significant advantages for global optimization due to the fact that they typically treat the problem as a black box and are therefore flexible and easy to implement. In addition, these optimizers have no stringent mathematical criteria (e.g., differentiability, smoothness), making them acceptable for problems with various features, such as discontinuities and nonlinearity [8]. A metaheuristic is considered a potential solution to optimization problems if it can strike a tradeoff between exploration (diversification) and exploitation (intensification). Exploitation is required to find regions of the search space that contain solutions of high quality. Exploration is necessary to intensify the search in some prospective regions based on gathered search knowledge [9], [10]. Metaheuristics are aimed at obtaining acceptable solutions in a realistic running time and providing practical solutions to a variety of problems [11], [12]. Metaheuristics have also gained appeal over exact methods for addressing optimization issues due to the ease and robustness of the solutions they give in a range of sectors, including engineering, business, transportation, and even the social sciences. The metaheuristic community has also conducted substantial research, which includes the development of novel methods, applications, and performance evaluations [13], [14].

It can be seen that DL and metaheuristics both provide their own distinct advantages, but what is missing from the past studies is a comprehensive approach to utilizing DL in the context of beamforming design. Our study contributes to finding solutions for beamforming design based on the combination of metaheuristics and DL in a manner that facilitates synergy between the two approaches. Specifically, we propose an HPO approach utilizing metaheuristics for designing beamforming in mmWave communication systems. By applying this approach, obtained hyperparameters can be used to build DL models with high performance. The proposed approach-based model has proved to outperform the state-of-art approach-based model [15] and the reference model in [16] with respect to spectral efficiency, convergence characteristics, and computational time.

The structure of this study is as follows. Section II discusses related studies on HPO and DL-based beamforming design for mmWave systems. Section III sheds light on DL-based beamforming design in mmWave communication systems. Section IV formulates the HPO problem based on metaheuristics and introduces an algorithm for optimizing hyperparameters. Results and comparative analysis are shown in Section V, and the discussion is presented in Section VI in Section VII.

II. RELATED WORK

Recently, some HPO techniques have been developed with their own merits and demerits. Grid search (GS) is a straightforward approach, but it suffers from the dimensionality curse and takes a long time [17], [18]. In comparison to GS, random search (RS) is more effective and supports all kinds of hyperparameters. In real-world applications, RS evaluation of the hyperparameter values chosen at random enables analysts to search a wide area. However, as RS does not take the outcomes of earlier tests into account, it may include numerous pointless evaluations, which reduces its effectiveness [7], [18]. The iterative Bayesian optimization (BO) algorithm is a well-liked solution to HPO problems. In contrast to GS and RS, BO bases the next hyperparameter value on the outcomes of prior evaluations in order to cut down on pointless assessments and increase efficiency. As a result, BO needs fewer iterations to find the ideal set of hyperparameters than GS and RS. However, it is challenging to parallelize BO models since they operate sequentially to balance the search for unexplored areas and the utilization of currently tested regions [7]. Although GS, RS, and BO are frequently used to configure hyperparameters, they are unworkable when the complexity of the problem and the number of parameters are high. Both Hyperband and RS offer simultaneous executions, but Hyperband can be considered an enhanced form of RS. Hyperband is more effective than RS, especially when time and resources are at a premium. It balances model performance with resource utilization. GS, RS, BO, and Hyperband treat each hyperparameter independently and do not take into account hyperparameter correlations. This is a significant limitation for any of these approaches. They will therefore be ineffective in logistic regression, support vector machines, and density-based spatial clustering of noisy applications, which are all DL algorithms [7].

In addition, to automate the search for DLs' designs and settings, researchers have also presented new studies based on metaheuristic optimization techniques. The differential evolution approach was used in the work [17] to give a framework for automating the search for long short-term memory hyperparameters, such as the number of hidden neurons and batch size. The experimental findings demonstrated that the system's average accuracy, which was based on an optimized long short-term memory network using differential evolution and particle swarm optimization algorithms, improved dramatically over time. Besides, the work [19] trained DL by adjusting its parameters for the vehicle logo recognition system. The learning rate, the number of filters, and the size of the filters, in each convolutional layer, were all optimized hyperparameters. They claimed that when compared to existing manual feature extraction techniques, the DL framework optimized by particle swarm optimization obtained more accuracy. A hyper-heuristic parameter optimization approach was proposed in the work [20] for configuring deep belief network parameters. On the MNIST, CalTech 101 Silhouettes, and Semeion datasets, this approach was contrasted with various metaheuristic algorithms such as particle swarm optimization. In almost all datasets, the hyper-heuristic parameter optimization had the lowest test mean square error.

In the context of the mmWave communication systems, the implementation of DL research advancements has also enhanced solutions for these systems [21]. There are several productive applications namely designing beamforming for weighted sum-rate maximization [22], predicting the optimal transmit/receive beam pairs by utilizing DL models as the role of hybrid precoding [23], using an autoencoder DL model to improve hybrid precoding [5], leveraging deep reinforcement learning for beamforming [24]. A technique based on convolutional neural networks for joint antenna selection and beamforming is proposed [25]. Works [16], [26] have shown that in comparison to conventional approaches, DL approaches are computationally more efficient in their search for optimum beamformers and tolerant of imperfect channel inputs. Based on compressive channel data learned by deep auto-encoders, the work [23] has designed beamformer vectors. BSs that collect the mobile user's omni-beampatterns for codebook-based beamforming have been taken into account for the DL-based wideband beamforming in [27]. Moreover, in the case of assuming perfect channel covariance matrix knowledge at the transmitter, DL-based statistical hybrid beamforming is studied in [28]. mmWave multiple-input multiple-output systems can considerably benefit from the application of DL approaches to their essential components, as evidenced by these works. However, hyperparameters in these DL models are all determined experimentally or not based on any principles at all. Therefore, HPO approaches for DL models in mmWave communication problems are imperative.

III. DL-BASED BEAMFORMING

A. SYSTEM MODEL

The downlink of narrowband multiple-input single-output mmWave systems using analog beamforming architectures in Fig. 1 is studied, in which base stations with a single RF chain and N_t antennas transmit one data stream to a user equipped with a single antenna [16]. Let s represent the symbol with normalized average symbol energy throughout transmission. The symbol is multiplied by a scalar digital precoder υ_{D} (υ_{D} is a scalar because there is only one RF chain) before being multiplied by an $N_t \times 1$ analog precoder vector ($\mathbf{v}_{\mathrm{RF}}$) that is used by phase shifters. The final signal after precoding is $\mathrm{x} = \mathbf{v}_{\mathrm{RF}}\upsilon_{\mathrm{D}}s$.

The received signal through the mmWave channel is denoted as $r = \mathbf{h}^{\dagger}_{\mathrm{channel}}\mathbf{v}_{\mathrm{RF}}\upsilon_{\mathrm{D}}s + n$, where n is the

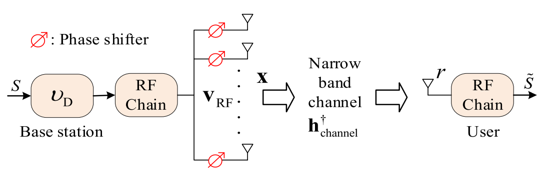

FIGURE 1. The diagram of a multiple-input single-output mmWave system using one RF chain [16].

additive white gaussian noise satisfying the circularly-symmetric complex normal with zero mean and covariance σ^2, $\mathbf{h}^{\dagger}_{\text{channel}}$ is mmWave channel vector between the base station and the user, and † denotes Hermitian transpose. With one line-of-sight path and $L-1$ non-line-of-sight paths, the widely employed Saleh-Valenzuela mmWave channel is expressed as [29]:

$$\mathbf{h}^{\dagger}_{\text{channel}} = \sqrt{\frac{N_t}{L}} \sum_{\ell=1}^{L} \alpha_\ell \mathrm{a}_t^{\dagger}\left(\phi_t^\ell\right), \tag{1}$$

where α_ℓ represents the complex gain of the ℓth path, ϕ_t^ℓ is the azimuth angle of departure of the ℓth path, and $\mathrm{a}_t^{\dagger}\left(\phi_t^\ell\right)$ is the antenna array response vector at the base station. The term with $\ell = 1$ means the line-of-sight path in $\mathbf{h}^{\dagger}_{\text{channel}}$.

The optimization objective function of the problem is considered the spectral efficiency that is widely utilized in current beamforming design works. This function is given as [16]:

$$R = \log_2\left(1 + \frac{\gamma}{N_t}\left\|\mathbf{h}^{\dagger}_{\text{channel}}\mathbf{v}_{\text{RF}}\right\|^2\right), \tag{2}$$

where γ represents the Signal-to-Noise Ratio (SNR). The beamformer aims to generate the optimized analog beamforming vectors $\mathbf{v}_{\text{RF}}$ so that the spectral efficiency is maximized. Then, the beamforming optimization problem with the constant modulus constraint of $\mathbf{v}_{\text{RF}}$ can be given by [16]:

$$\begin{aligned} \underset{\mathbf{v}_{\text{RF}}}{\text{minimize}} \quad & -\log_2\left(1 + \frac{\gamma}{N_t}\left\|\mathbf{h}^{\dagger}_{\text{channel}}\mathbf{v}_{\text{RF}}\right\|^2\right) \\ \text{subject to} \quad & |[\mathbf{v}_{\text{RF}}]_{nt}|^2 = 1, \quad \text{for } nt = 1, \ldots, N_t. \end{aligned} \tag{3}$$

As the SNR is often regarded as being more correctly measured than the channel, the SNR γ and the estimated SNR γ_{est} are assumed to be equal, i.e., $\gamma_{est} = \gamma$.

B. DL-BASED BEAMFORMING DESIGN

In this study, we take the DL-based beamformer designed in [16] as the reference one to verify our proposed approach. This beamformer consists of two stages, which are illustrated in Fig. 2, directly output $\mathbf{v}_{\text{RF}}$ to solve (3). During the offline training stage, random channel samples are generated using via simulation on the system model. The base station then applies a practical channel estimator to achieve partial channel state information. The mmWave channel estimator in [29] is adopted, where the mmWave channels are estimated by sending pilot symbols in a hierarchical codebook and then receiving the user's decision feedback based on the signal received r_p. The estimated channel $\mathbf{h}^{\dagger}_{\text{channel\_est}}$ and the estimated SNR γ_{est} are inputs for the DL-based beamformer with $\gamma_{est} = \gamma$. By minimizing a loss function, the beamformer then can generate optimized beamforming vectors $\mathbf{v}_{\text{RF}}$. As the SNR values and channel samples are produced randomly by the simulation (called generated channels in this study), they can be used directly in the loss computation. By utilizing the estimated channels as the input and generated channels in the loss function, the beamformer can be trained to figure out how to obtain as close to the ideal spectral efficiency with the estimated channels as possible and become robust to channel estimation errors. During the online deployment phase, the base station uses the same mmWave channel estimator. The estimated channels are then inputted to the trained beamformer to obtain optimized beamforming vectors for maximizing spectral efficiency. It is important to note that generated channels are only necessary during the offline training stage to compute the loss. This is because all the parameters of the trained beamformer have already been fixed, and the trained beamformer is ready to accept practical mmWave channels as inputs to directly output beamforming vectors. Multiple offline training samples are necessary to ensure the generalizability of DL models, so $1e^5$ samples to train and $5e^3$ samples to test are used in this study.

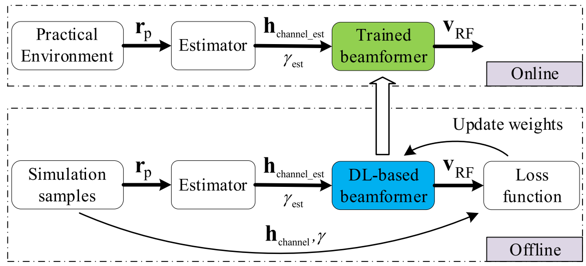

FIGURE 2. The illustration of offline and online stages for DL-based beamformer [16] .

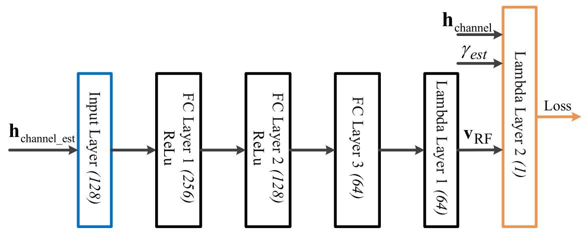

FIGURE 3. The architecture of the reference DL model.

The architecture of the reference model for the beamformer in [16] in the offline stage consists of six main layers which are demonstrated in Fig. 3. The inputs are the generated channels $\mathbf{h}_{\text{channel}}$, the estimated SNRs γ_{est} (random in the training stage), and the estimated channels $\mathbf{h}_{\text{channel\_est}}$, where complex-valued $\mathbf{h}_{\text{channel\_est}}$ with the size of $N_t = 64$ is separated into real and imaginary parts, and then these parts are concatenated into a vector with the size of 128. The output is the optimized analog beamforming vectors $\mathbf{v}_{\text{RF}}$ that are applied to analog phase shifters. Besides, Lambda layer 1 is added to compute complex-valued vectors $\mathbf{v}_{\text{RF}}$ based on N_t real-valued phases so that the constant modulus constraint is satisfied. With the input of $\mathbf{v}_{\text{RF}}$, γ_{est} and $\mathbf{h}_{\text{channel}}$, the Lambda layer 2 is used to compute the loss function which is defined as:

$$\text{Loss} = -\frac{1}{N_{sam}} \sum_{ns=1}^{N_{sam}} \log_2\left(1 + \frac{\gamma_{ns}}{N_t}\left\|\mathbf{h}^{\dagger}_{\text{channel},ns}\mathbf{v}_{\text{RF},ns}\right\|^2\right). \tag{4}$$

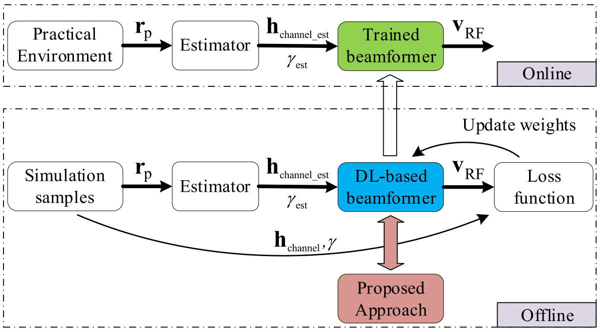

FIGURE 4. The utilization of the proposed approach for hyperparameter optimization.

The loss function has a direct relationship with the objective function in (3) with N_{sam} training samples, where $\mathbf{v}_{\mathrm{RF},ns}$, γ_{ns} and $\mathbf{h}^{\dagger}_{\mathrm{channel},ns}$ represent the optimized analog beamforming vectors, SNRs, and the generated channels associated with the nsth sample. Note that the reduction in loss correlates precisely with the increase in the average spectral efficiency. The fully connected (FC) layers 1, 2, and 3 include 256, 128, and 64 neurons, respectively, with corresponding activation functions in Fig. 3 and batch normalization layers preceded these FC layers. The Adam optimizer is adopted with a learning rate of 0.001, and the channel samples are related to different random SNRs between −20dB and 20dB.

IV. PROPOSED APPROACH

HPO approaches aim to improve DL architectures by identifying the best combinations of hyperparameters [7]. As shown in Fig. 4, our proposed approach is adopted to optimize hyperparameters for the DL model described in the previous section. The key ideas of the HPO problem and the algorithm are described in this section.

A. FORMULATION OF HPO PROBLEM

The process of searching hyperparameter combinations involves four main parts [7]: an estimator (a classifier or regressor) with its fitness functions, a search space (configuration space), an optimization or a search method, and an evaluation function to evaluate how well various hyperparameter configurations perform. A hyperparameter's domain can be categorical (e.g., type of optimizer), binary (e.g., whether to apply early stopping), discrete (e.g., number of clusters), or continuous (e.g., learning rate). For an HPO problem, in general, the aim is to obtain:

$$\mathbf{h}^{*} = \arg\min_{\mathbf{h}\in\mathbf{H}} f\left(\mathbf{h}\right), \tag{5}$$

where $f\left(\mathbf{h}\right)$ is the fitness function to be minimized, $\mathbf{h}^{*}$ is a hyperparameter vector that yields the optimum value of $f\left(\mathbf{h}\right)$ while a hyperparameter vector $\mathbf{h}$ can take any value in the search space $\mathbf{H}$. The goal of HPO is to tune hyperparameters within allowed budgets to produce optimal or nearly optimal model performance. Some metrics, including accuracy or loss such as root mean square error can be used to evaluate the performance of the model. DL models are retrained if a new hyperparameter set is evaluated, and the validation set should be processed to produce a score that measures the model's performance [7].

For DL models, the search space $\mathbf{H}$ can include the number of filters, the size of the filters in convolutional layers, the dimensionality of the output in long short-term memory layers, the number of neurons in fully connected layers, activation functions, optimizers, and the learning rate. Assuming that a DL model requires optimizing m different hyperparameters and that the domain of these hyperparameters are categorical and discrete, each hyperparameter has n_i choices in the i-th search space H_i for $i = 1, 2, \ldots, m$. Hence, the search space can be expressed as:

$$\mathbf{H} = \begin{matrix} H_{1,1} & H_{1,2} & \ldots & H_{1,n_1} \\ H_{2,1} & H_{2,2} & \ldots & H_{2,n_2} \\ \ldots & \ldots & \ldots & \ldots \\ H_{m,1} & H_{m,2} & \ldots & H_{m,n_m} \end{matrix}. \tag{6}$$

The vector $\mathbf{h}^{*} = [h_1, h_2, \ldots, h_m]^T$ consists of m optimal hyperparameters. To determine this vector, the index vector $\mathbf{k} = [k_1, k_2, \ldots, k_m]^T$, which includes m values mapping to $\mathbf{H}$, should be optimized. The values in the index vector are less than or equal to the choices in H_i. For example, if the first search space H_1 has n_1 choices, k_1 is less than or equal to n_1, and the first optimal hyperparameter h_1 is H_{1,k_1}. Therefore, it is necessary to apply the proper optimization methods to HPO problems to determine the index vector and then identify optimal hyperparameter configurations for DL models.

B. PROPOSED ALGORITHM

The proposed algorithm is developed based on Binary Bat Algorithm (BBA) [30], which is one of the best metaheuristics for solving problems with discrete binary search spaces, to identify the optimal hyperparameter vector $\mathbf{h}^{*}$. However, other metaheuristics can still be applied based on the proposed algorithm instead of BBA. The pseudocode is demonstrated in Algorithm 1. It can be briefly described as follows:

1) INITIALIZATION

First, the type of learning (supervised versus unsupervised) and datasets should be determined. Next, the search space $\mathbf{H}$, such as the number of neurons in FC layers, activation functions, the number of choices or upper and lower limits for each hyperparameter, and whether to apply early stopping or not, should be defined. Because the goal of the optimization problem is to minimize the fitness function, this function is determined according to performance metrics such as the spectral efficiency on test datasets. After that, the number of populations and iterations are initialized, and the dimension of solutions of BBA (d) is calculated based on $\mathbf{H}$ as follows:

$$d = \sum_{i=1}^{m} \lceil \log_2 n_i \rceil, \tag{7}$$

where $\lceil \bullet \rceil$ denotes rounding up to the nearest number. The bats' solutions, which are binary numbers, are initialized

Algorithm 1 The Proposed Algorithm for HPO

1: **Determine**: Datasets; the search space $\mathbf{H}$; performance metrics; fitness function *Fitness*; number of populations (*numPop*); number of iterations, and dimension of solutions.
2: **Initialize** populations and then obtain $\mathbf{h}$ from solutions in initialized populations; train and test DL models with $\mathbf{h}$; and find the current best solution (the current $\mathbf{h}^*$).
3: **Repeat**
4: Adjust frequency and update velocities; compute transfer function; and then update positions.
5: **if** *rand* $>$ *pulse rate* **then**
6: Select randomly binary values among the best solutions (G_{best}).
7: Change binary values in **sol** with the selected binary values in G_{best}.
8: **end if**
9: Obtain new $\mathbf{h}$ from current solutions; train and test DL models with new $\mathbf{h}$.
10: Compute the fitness function; rank the bats and determine the current G_{best}.
11: **Until** Termination conditions are satisfied.
12: Obtain $\mathbf{h}^*$ from the final G_{best}.
13: Train DL models with $\mathbf{h}^*$ and then use trained models to directly output beamforming vectors.

randomly. The solutions or bats' positions, **sol**, are a binary number vector, so they should be converted to a decimal number vector that is $\mathbf{k}$. For $i = 1, 2, \ldots, m$, the element k_i in $\mathbf{k}$ is determined as follows:

$$k_i = \left[\frac{n_i}{2^{\lceil \log_2 n_i \rceil} - 1} \mathrm{int}\,(\mathbf{sol}) \right], \tag{8}$$

where $[\bullet]$ and $\mathrm{int}\,(\bullet)$ denotes rounding to the nearest number and converting to integer numbers, respectively. Next, the hyperparameter vector $\mathbf{h}$ can be obtained by mapping $\mathbf{k}$ into $\mathbf{H}$. At this point, it can build DL models with $\mathbf{h}$, then train models and test models to find the current best hyperparameter vector based on performance metrics.

2) FINDING THE BEST HYPERPARAMETERS

The search operation of BBA is implemented. For the p-th bat with $p = 1, 2, \ldots, numPop$, the frequency Q_p and the velocity V_p^{iter} at the *iter*-th iteration are updated as follows:

$$Q_p = Q_{\min} + (Q_{\max} - Q_{\min})\, rand, \tag{9}$$

$$V_p^{iter} = V_p^{iter-1} + \left(\mathbf{sol}_p^{iter-1} - G_{best}\right) Q_p, \tag{10}$$

where $Q_{\min}$, $Q_{\max}$, G_{best}, and *rand* are the minimum frequency, the maximum frequency, the current best solutions, and random values drawn from the uniform distribution in (0, 1), respectively. To map velocity values to binary values for updating the positions or forcing bats to move in a binary space, the following V-shaped transfer function is used to update the position of the p-th bat:

$$F_{tranfer}\left(V_p^{iter}\right) = \left| \frac{2}{\pi} \arctan\left(\frac{2}{\pi} V_p^{iter}\right) \right|, \tag{11}$$

$$\mathbf{sol}_p^{iter} = \begin{cases} \left(\mathbf{sol}_p^{iter-1}\right)^{-1} & \text{if } rand < F_{tranfer}\left(V_p^{iter}\right) \\ \mathbf{sol}_p^{iter-1} & \text{if } rand \geq F_{tranfer}\left(V_p^{iter}\right), \end{cases} \tag{12}$$

where $(\bullet)^{-1}$ indicates the complement of binary numbers. If *rand* is greater than *pulse rate*, change binary numbers in **sol** with the randomly selected binary values in G_{best} so that the local solution, **sol**, moves towards the current best solution G_{best}, where *pulse rate* represents the pulse emission rate of bats. At the step of obtaining new $\mathbf{h}$ from current solutions, $\mathbf{h}$ is derived from the same manner as explained above. The operation is finished when the termination conditions are satisfied. In this paper, the optimization process is terminated after running 15 iterations, which is chosen based on experiments.

3) BUILDING, TRAINING, AND EMPLOYING DL MODELS WITH OPTIMIZED HYPERPARAMETERS

From the best solution (binary numbers), the best hyperparameter vector $\mathbf{h}^*$ can be obtained. Next, the optimal DL model is built and trained. Finally, the trained model is used to output beamforming vectors.

V. RESULTS AND COMPARATIVE ANALYSIS

The efficiency of the proposed approach will be evaluated in this section. Firstly, the reference model's parameters, BBA's parameters, and $\mathbf{H}$ are described. Next, the convergence ability is demonstrated. Finally, the proposed approach-based model is compared to the reference model and the Hyperband approach-based model in terms of maximizing spectral efficiency. In all figures, reference, Hyperband, and proposal refer to the reference model, the Hyperband approach-based model, and the proposed approach-based model, respectively.

A. PARAMETER SETUP

This study focuses on verifying the proposed approach, so we use the same datasets as used for the reference model. Datasets, source code, and trained weights for the reference model are publicly provided by authors in [16]. The number of total paths (L) is 3 and the estimation of channel samples with the pilot-to-noise power ratio is 20dB.

BBA belongs to one type of metaheuristics; in addition, the maximum number of iterations and the population size are two factors that have a close relationship with the metaheuristics' performance [8]. Based on experiments, we have determined that the population size and the maximum number of iterations should be 20 and 15, respectively, for this problem. Termination conditions are that all iterations have been completed. Other parameters are set as suggested by [30]:

pulse rate $= 0.5$; $Q_{\min} = 0$; $Q_{\max} = 2$. The illustrated results are the average value of 20 independent runs.

This study verifies the proposed approach by optimizing hyperparameters of DL models that have six main layers same as the model of the reference beamformer. The search space **H**, which is expressed in (13), includes the number of neurons in the first two FC layers (corresponding to the first two rows), activation functions after the first two FC layers (the fourth row), optimizers (the fifth row), and the initial learning rate (the last row). The order of hyperparameters in the search space is not required to be in the order of each layer in the reference DL model. These hyperparameters are determined by the empirical trials in [16], so they will be optimized by our proposed approach for achieving the ideal spectral efficiency. Assume that each hyperparameter has 4 choices, the dimension of one solution d is 12, calculated by (7).

$$\mathbf{H} = \begin{matrix} 128 & 192 & 256 & 320 \\ 64 & 96 & 128 & 192 \\ \text{ELU} & \text{ReLU} & \text{Sigmoid} & \text{Tanh} \\ \text{AdaMax} & \text{Adam} & \text{RMSprop} & \text{Nadam} \\ 1e^{-4} & 5e^{-4} & 1e^{-3} & 5e^{-3} \end{matrix}. \tag{13}$$

The network complexity of DL models increases proportionally with the number of neurons, so the search space for the number of neurons in the first two FC layers is set to values in a range that includes the number of neurons set in the reference model. Exponential Linear Unit (ELU), Sigmoid, Rectified Linear Unit (ReLU), and Tanh are the most prevalent and widespread non-linearity layers and are proven to be effective solutions to non-zero mean and zero gradient problems, as well as the accuracy versus training time tradeoff [31]. AdaMax, Adam, RMSProp, and Nadam are the most efficient and widely used optimization algorithms in DL [32], [33], [34]. A large learning rate helps the model to learn quicker at the expense of arriving at a suboptimal final set of weights. A smaller learning rate may enable the model to acquire a more optimum or even globally optimal set of weights, but it may require much more time to train [35]. The learning rate range to be taken into consideration is from $1e^{-4}$ to $5e^{-3}$, including the learning rate which is set in the reference model.

The fitness function for the proposed algorithm is built based on the spectral efficiency function in (2) as follows:

$$Fitness = \frac{1}{\sum_{snr=-20}^{20} |R_{snr}|}. \tag{14}$$

The spectral efficiency is evaluated on test datasets with SNRs from -20dB to 20dB with the step of 5. Note that the spectral efficiency increases as the fitness function decreases.

B. CONVERGENCE CHARACTERISTICS

In this subsection, the convergence ability and the training loss produced by DL models on test datasets are evaluated. The values of the fitness function in Fig. 5 indicate that the proposed approach nearly converges after the 6th iteration

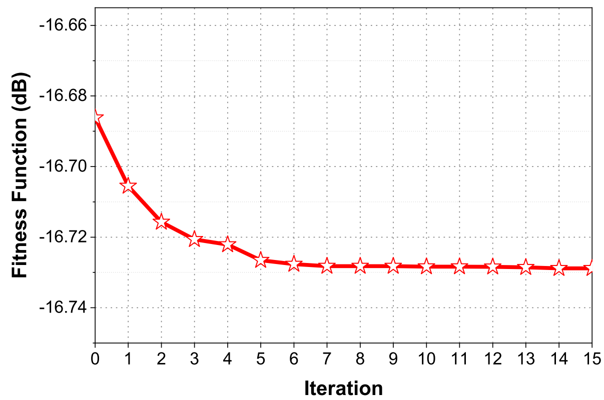

FIGURE 5. **The fitness function over 15 iterations.**

with the value of -16.728dB and insignificantly decreases from the 7th iteration onwards. This means that at the 6th iteration, the proposed approach can figure out optimized hyperparameters that are listed in Table 1. Fig. 6 compares the training loss between the reference model, the proposed approach-based model, and the Hyperband approach-based model, where optimized hyperparameters of these DL models are in Table 1. Both HBO approach-based models achieve lower loss values and converge faster than the reference model even though both have more trainable parameters. However, the proposed approach-based model achieves -5.302 while the Hyperband approach-based model is -5.252, and the reference model is -5.136.

C. SPECTRAL EFFICIENCY CHARACTERISTICS

This subsection compares the achievable spectral efficiency between the reference model, the Hyperband approach-based model, and the proposed approach-based model. The spectral efficiency versus SNR performance in Fig. 7 shows that the proposed approach-based model produces higher spectral efficiency than the reference model. To obtain 9.72 bits/s/Hz, for example, the optimized model achieves around 1dB in SNR over the reference model. Besides, the proposed approach-based model is also slightly better than the Hyperband approach-based model for spectral efficiency.

There are estimation errors in estimating L in practical systems. Owing to the estimation complexity and the sparsity of mmWave channels, the estimated number of channel paths should be set to a small value [29]. Moreover, L in practice often differs from those in training, so the consideration of the mismatch between training and deployment plays an important role. Assuming that the online deployment stage's channel model has three paths ($L = 3$), but the DL-based models are trained with L_{Tr} paths. The impact of the channel model's mismatch between training and deployment stages is depicted in Fig. 8. This figure demonstrates the achievable spectral efficiency with the output of the DL-based models which have been trained with $L_{Tr} = 2, 3$, respectively. Even though there is a model mismatch when $L_{Tr} = 2, 3$, the losses between the training and deployment stages are limited, which indicates the robustness and generalizability

TABLE 1. Hyperparameters for three DL models.

Hyperparameter	Reference model	Hyperband-based model	Proposed-based model
Number of neurons in 1st FC layer	256	320	320
Number of neurons in 2nd FC layer	128	192	192
Activation function	ReLu	Sigmoid	Sigmoid
Optimizer	Adam	RMSprop	Nadam
Initial learning rate	$1e^{-3}$	$1e^{-3}$	$5e^{-3}$

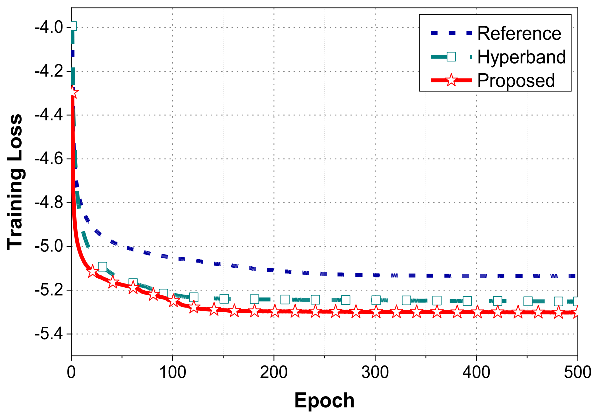

FIGURE 6. The training loss versus epochs.

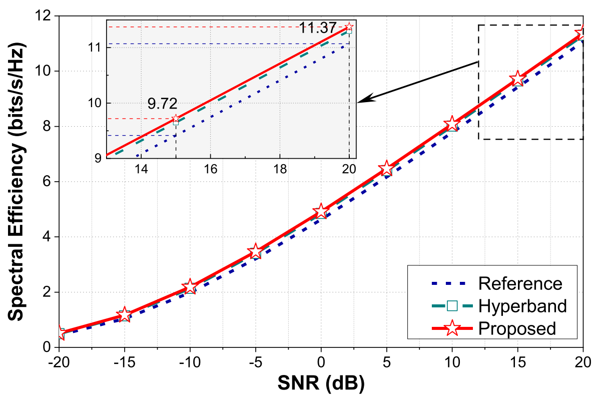

FIGURE 7. The spectral efficiency versus SNR.

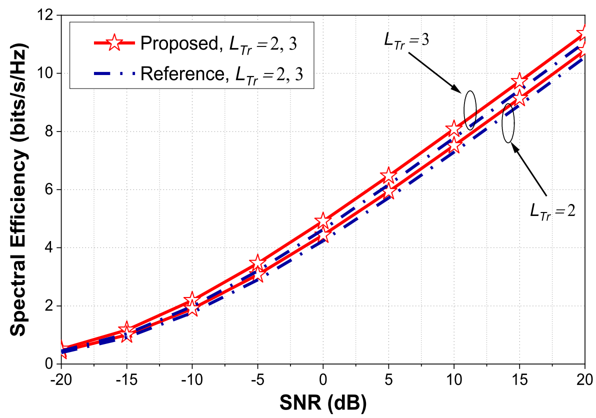

FIGURE 8. The impact of the channel model's mismatch.

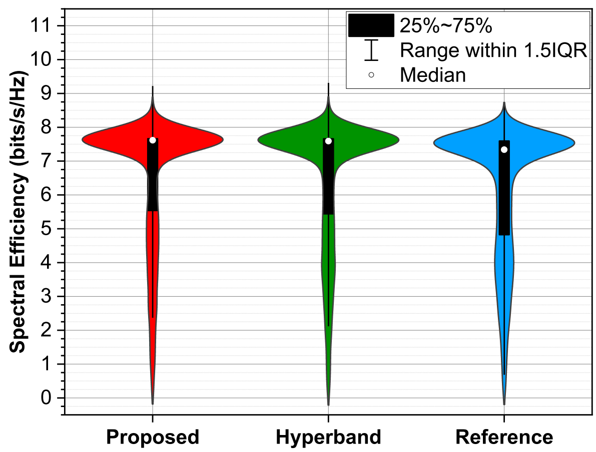

FIGURE 9. The distribution of the spectral efficiency.

of DL-based models to the model mismatch issue. In these models, the proposed approach-based model produces higher spectral efficiency than the reference model by about 0.041 to 0.304 bits/s/Hz.

Fig. 9 shows violet plots, and Table 2 shows the median, the first and the third quartiles of the distribution of the spectral efficiency with SNR $=$ 5dB of three DL models. The median, the first and third quartiles based on the proposed approach-based model are 7.616, 5.538, and 7.663, respectively, which are higher than both those based on the reference model and the Hyperband-based model. Moreover, the shape of the distribution from the minimum value to the first quartile in Fig. 9 indicates that the spectral efficiency of the proposed approach-based model is thinner than those of the other two models and is highly concentrated around the median compared to the reference model.

Typically, considering the resolution of practical phase shifters is limited. When beamforming coefficients or the output of DL models are quantized with q bits, the spectral efficiency performance versus as a function of these bits is considered, which is shown in Fig. 10. As q increases, the performance loss lessens, and it is negligible when $q > 4$. For SNR $=$ 5dB, moreover, the proposed approach-based model is better than both the Hyperband approach-based model and the reference model in respect of spectral efficiency. With $q = 3$, for example, the proposed approach-based model achieves 6.412 bits/s/Hz while the Hyperband approach-based model and the reference model only achieve 6.365 and 6.104 bits/s/Hz, respectively.

Once the DL model is trained in the offline stage, this model will be adopted to output beamforming vectors. Therefore, the computational time for yielding these vectors should be carefully considered in the online stage. Table 3 shows the

TABLE 2. The median, the first and the third quartiles produced by three DL models.

Parameter	Reference model	Hyperband-based model	Proposed-based model
Median	7.340	7.593	7.616
The first quartile	4.826	5.442	5.538
The third quartile	7.596	7.661	7.663

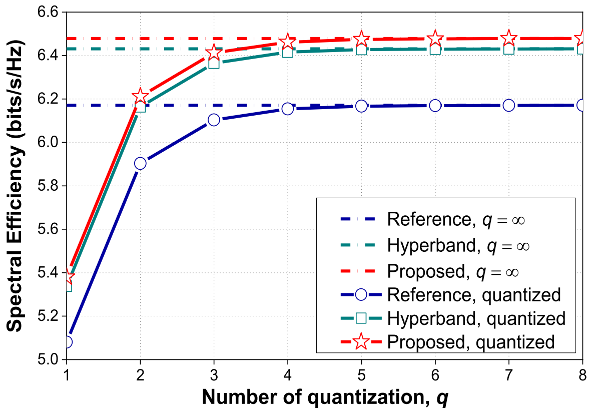

FIGURE 10. The spectral efficiency performance versus the resolution of phase shifters.

TABLE 3. Computational time (in milliseconds) to output beamforming vectors.

	Reference model	Hyperband-based model	Proposed-based model
Time	631.187	628.080	626.956

average computational time in milliseconds to output beamforming vectors on 5000 test samples over 1000 independent runs in computers equipped with an NVIDIA T4 Tensor Core GPU. The proposed approach-based model not only takes less time than the other two models but also achieves higher spectral efficiency.

VI. DISCUSSION

This study has combined metaheuristics and DL in a manner that facilitates synergy between these two approaches to propose an HPO approach. This combination solves not only the HPO problem but also the following problems [36]: training DL models, architecture optimization (architecture search), and optimization at feature representation levels. Interestingly, these types of optimization problems are amenable to solutions via metaheuristic algorithms. Based on the knowledge of solutions, the selection operators of metaheuristic algorithms direct the search for promising regions in the search space, making them efficient approaches for solving challenging problems.

Besides, in recent years, there is considerable interest in DL due to its ability to develop intelligent systems that can make effective decisions and accurate predictions. DL approaches help significantly enhance efficiency compared to conventional communication systems [27]. Therefore, the proposed approach can be considered a premise for optimizing hyperparameters for various DL-based problems in general, not just problems for mmWave communication systems.

Although the proposed approach is specifically verified by optimizing main hyperparameters such as the number of neurons in FC layers, and activation functions in this study, it can have good generality for more complex models and problems. For instance, the proposed approach can be used to optimize hyperparameters in convolutional and long short-term memory neural networks. In [37], for example, the following hyperparameters can be optimized: the number of filters, the size of pooling windows in convolutional neural network modules, and the output size of long short-term memory modules. Eventually, DL models built with optimized hyperparameters will output predictive beamforming matrices. These matrices are utilized to approach achievable sum rates of the upper bound method for vehicular networks with the integration of sensing and communication.

VII. CONCLUSION

This study has proposed an HPO approach based on metaheuristics for DL models. The proposed approach was applied to optimizing hyperparameters in DL models that aim to output optimized beamforming coefficients to approach the ideal spectral efficiency in mmWave communication systems with large-scale antenna arrays. Results have shown the ability to optimize hyperparameters and provided an insightful solution to forthcoming HPO problems. Comparative analysis has also indicated that the proposed approach-based models can produce higher spectral efficiency than the Hyperband approach-based models and the reference model. As for future work, it would be interesting to apply the proposed approach to more complex DL models and beamforming problems using hybrid beamforming architectures for reconfigurable intelligent surfaces, and integrated sensing and communication in 6G wireless communication systems.

REFERENCES

[1] W. Hong, Z. H. Jiang, C. Yu, D. Hou, H. Wang, C. Guo, Y. Hu, L. Kuai, Y. Yu, Z. Jiang, Z. Chen, J. Chen, Z. Yu, J. Zhai, N. Zhang, L. Tian, F. Wu, G. Yang, Z. Hao, and J. Y. Zhou, "The role of millimeter-wave technologies in 5G/6G wireless communications," *IEEE J. Microw.*, vol. 1, no. 1, pp. 101–122, Jan. 2021.

[2] L. Zhu, Z. Xiao, X. Xia, and D. Oliver Wu, "Millimeter-wave communications with non-orthogonal multiple access for B5G/6G," *IEEE Access*, vol. 7, pp. 116123–116132, 2019.

[3] M. Y. Javed, N. Tervo, M. E. Leinonen, and A. Pärssinen, "Wideband inter-beam interference cancellation for mmW/sub-THz phased arrays with squint," *IEEE Trans. Veh. Technol.*, early access, Feb. 3, 2023, doi: 10.1109/TVT.2023.3242133.

[4] Y. Shi, L. Lian, Y. Shi, Z. Wang, Y. Zhou, L. Fu, L. Bai, J. Zhang, and W. Zhang, "Deep learning for large-scale optimization in 6G wireless networks," 2023, *arXiv:2301.03377*.

[5] H. Huang, Y. Song, J. Yang, G. Gui, and F. Adachi, "Deep-learning-based millimeter-wave massive MIMO for hybrid precoding," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3027–3032, Mar. 2019.

[6] B. Bischl, M. Binder, M. Lang, T. Pielok, J. Richter, S. Coors, J. Thomas, T. Ullmann, M. Becker, A. Boulesteix, D. Deng, and M. Lindauer, "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," *WIREs Data Mining Knowl. Discovery*, vol. 13, no. 2, pp. 1–10, Mar. 2023.

[7] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, Nov. 2020.

[8] Q. Li, S.-Y. Liu, and X.-S. Yang, "Influence of initialization on the performance of metaheuristic optimizers," *Appl. Soft Comput.*, vol. 91, Jun. 2020, Art. no. 106193.

[9] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf. Sci.*, vol. 237, pp. 82–117, Jul. 2013, doi: 10.1016/J.INS.2013.02.041.

[10] L. V. Tong, K. M. Kha, C. V. Nguyen, T. L. Nguyen, and G. B. V. Truong, "Null-steering beamformers for suppressing unknown direction interferences in sidelobes," *J. Commun.*, vol. 10, pp. 600–607, Jan. 2022.

[11] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Comput. Ind. Eng.*, vol. 137, Nov. 2019, Art. no. 106040.

[12] T. V. Luyen and N. V. Cuong, "An effective beamformer for interference suppression without knowing the direction," *J. Electr. Comput. Eng.*, vol. 13, pp. 601–610, Jan. 2022.

[13] K. Hussain, M. N. M. Salleh, S. Cheng, and Y. Shi, "Metaheuristic research: A comprehensive survey," *Artif. Intell. Rev.*, vol. 52, no. 4, pp. 2191–2233, Dec. 2019.

[14] K. M. Hoang, L. Van Tong, and C. Van Nguyen, "A null synthesis technique-based beamformer for uniform rectangular arrays," in *Proc. Int. Conf. Adv. Technol. Commun. (ATC)*, Oct. 2022, pp. 13–17.

[15] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," *J. Mach. Learn. Res.*, vol. 18, no. 185, pp. 1–52, 2018.

[16] T. Lin and Y. Zhu, "Beamforming design for large-scale antenna arrays using deep learning," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 103–107, Jan. 2020.

[17] B. Nakisa, M. N. Rastgoo, A. Rakotonirainy, F. Maire, and V. Chandran, "Long short term memory hyperparameter optimization for a neural network based emotion recognition framework," *IEEE Access*, vol. 6, pp. 49325–49338, 2018.

[18] N. Bacanin, T. Bezdan, E. Tuba, I. Strumberger, and M. Tuba, "Optimizing convolutional neural network hyperparameters by enhanced swarm intelligence metaheuristics," *Algorithms*, vol. 13, no. 3, p. 67, Mar. 2020.

[19] F. C. Soon, H. Y. Khaw, J. H. Chuah, and J. Kanesan, "Hyper-parameters optimisation of deep CNN architecture for vehicle logo recognition," *IET Intell. Transp. Syst.*, vol. 12, no. 8, pp. 939–946, Oct. 2018.

[20] N. R. Sabar, A. Turky, A. Song, and A. Sattar, "An evolutionary hyper-heuristic to optimise deep belief networks for image reconstruction," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 105510.

[21] V. Raj, N. Nayak, and S. Kalyani, "Deep reinforcement learning based blind mmWave MIMO beam alignment," *IEEE Trans. Wireless Commun.*, vol. 21, no. 10, pp. 8772–8785, Oct. 2022.

[22] H. Huang, W. Xia, J. Xiong, J. Yang, G. Zheng, and X. Zhu, "Unsupervised learning-based fast beamforming design for downlink MIMO," *IEEE Access*, vol. 7, pp. 7599–7605, 2019.

[23] X. Li and A. Alkhateeb, "Deep learning for direct hybrid precoding in millimeter wave massive MIMO systems," in *Proc. 53rd Asilomar Conf. Signals, Syst., Comput.*, Nov. 2019, pp. 800–805.

[24] Q. Wang, K. Feng, X. Li, and S. Jin, "PrecoderNet: Hybrid beamforming for millimeter wave systems with deep reinforcement learning," *IEEE Wireless Commun. Lett.*, vol. 9, no. 10, pp. 1677–1681, Oct. 2020.

[25] A. M. Elbir and K. V. Mishra, "Joint antenna selection and hybrid beamformer design using unquantized and quantized deep learning networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1677–1688, Mar. 2020.

[26] P. Dong, H. Zhang, G. Y. Li, I. S. Gaspar, and N. NaderiAlizadeh, "Deep CNN-based channel estimation for mmWave massive MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 5, pp. 989–1000, Sep. 2019.

[27] A. Alkhateeb, S. Alex, P. Varkey, Y. Li, Q. Qu, and D. Tujkovic, "Deep learning coordinated beamforming for highly-mobile millimeter wave systems," *IEEE Access*, vol. 6, pp. 37328–37348, 2018.

[28] A. M. Elbir, "A deep learning framework for hybrid beamforming without instantaneous CSI feedback," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 11743–11755, Oct. 2020.

[29] A. Alkhateeb, O. E. Ayach, G. Leus, and R. W. Heath Jr., "Channel estimation and hybrid precoding for millimeter wave cellular systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 831–846, Oct. 2014.

[30] S. Mirjalili, S. M. Mirjalili, and X.-S. Yang, "Binary bat algorithm," *Neural Comput. Appl.*, vol. 25, nos. 3–4, pp. 663–681, Sep. 2014.

[31] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark," *Neurocomputing*, vol. 503, pp. 92–108, Sep. 2022.

[32] T. Dozat, "Incorporating Nesterov momentum into Adam," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–12.

[33] D. Soydaner, "A comparison of optimization algorithms for deep learning," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 34, no. 13, Dec. 2020, Art. no. 2052013.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[35] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*.

[36] B. Akay, D. Karaboga, and R. Akay, "A comprehensive survey on optimizing deep learning models by metaheuristics," *Artif. Intell. Rev.*, vol. 55, no. 2, pp. 829–894, Feb. 2022.

[37] C. Liu, W. Yuan, S. Li, X. Liu, D. W. K. Ng, and Y. Li, "Predictive beamforming for integrated sensing and communication in vehicular networks: A deep learning approach," in *Proc. IEEE Int. Conf. Commun.*, May 2022, pp. 1948–1954.

KIEU-XUAN THUC (Member, IEEE) received the B.E. and M.S. degrees from the Hanoi University of Technology, Vietnam, in 1999 and 2004, respectively, and the Ph.D. degree from the University of Ulsan, in February 2012. In 1999, he joined the Hanoi University of Industry. His research interests include intelligent signal processing algorithms and next-generation wireless communication systems.

HOANG MANH KHA (Member, IEEE) received the B.E. and M.E. degrees in electronics and telecommunications engineering from the Hanoi University of Science and Technology, in 2002 and 2004, respectively, and the Ph.D. degree in communications engineering from the University of Paderborn, Germany, in 2016. His research interests include digital signal processing, wireless communication, positioning engineering, deep learning, pattern classification, and metaheuristics.

NGUYEN VAN CUONG received the engineering and master's degrees from the Hanoi University of Industry, in 2020 and 2022, respectively. His research interests include beamforming for antenna arrays, smart antennas, nature-inspired optimization algorithms, deep learning, and convex optimization.

TONG VAN LUYEN (Member, IEEE) received the B.S. and M.S. degrees from the Hanoi University of Science and Technology, in 2002 and 2004, respectively, and the Ph.D. degree from the VNU University of Engineering and Technology, in 2019. His research interests include beamforming and beam-steering for antenna arrays, smart antennas, optimum array processing, nature-inspired optimization algorithms, and artificial intelligence.