Received 7 August 2023; revised 23 August 2023; accepted 4 September 2023. Date of publication 7 September 2023; date of current version 27 September 2023.
Digital Object Identifier 10.1109/OJCOMS.2023.3312670

Efficient Radar-Target Assignment in Low Probability of Intercept Radar Networks: A Machine-Learning Approach

HAMID AMIRIARA [1], SEYED MEHDI HOSSEINI ANDARGOLI [1], AND VAHID MEGHDADI [2]

[1] Department of Electrical and Computer Engineering, Noshirvani University of Technology, Babol 4714873113, Iran
[2] XLIM, UMR CNRS 7252 Laboratory, University of Limoges, 87068 Limoges, France
CORRESPONDING AUTHOR: S. M. HOSSEINI ANDARGOLI (e-mail: smh_andargoli@nit.ac.ir)

ABSTRACT To achieve low probability of intercept (LPI) in radar networks for multiple target detection, it is necessary to find the optimal assignment of distributed radars to targets. The multi-radar to multi-target assignment (MRMTA) problem aims to find the best radar combination, but its brute-force (BF)-based approach over all possible sensor combinations has exponential complexity, making it challenging to implement in networks with a large number of radars or targets. This limits the implementation of the BF approach in networks that prioritize low latency and complexity. To address this challenge, we propose a supervised machine-learning (ML)-based solution for the MRMTA problem. Our proposed implementation scheme performs the training procedure offline, leading to a significant reduction in assignment complexity and processing latency. We conducted extensive numerical simulations to design an ML structure with high accuracy, convergence speed, and scalability. Simulation results demonstrate the efficiency and effectiveness of our proposed ML-based MRMTA solution, which achieves near-optimal LPI performance with considerably lower computation time than benchmark schemes. Our proposed solution has the potential to optimize the assignment of distributed radars to targets in LPI radar networks and improve the performance of complex networks with low latency and complexity requirements.

INDEX TERMS Multi-radar to multi-target assignment (MRMTA), supervised machine-learning (ML), low probability intercept radars, feed-forward neural network (FNN), radar network, information fusion.

I. INTRODUCTION

MULTI-RADAR to multi-target assignment (MRMTA) is crucial for achieving low probability of intercept (LPI) support and better information retrieval in distributed radar networks [1], [2], and it has recently attracted considerable attention from radar engineers [3], [4], [5], [6], [7], [8]. LPI radars are designed to search or track targets while remaining hidden from the enemy's equipment, and this property can be adapted to distributed radar networks that are netted together [9], [10], [11], [12], [13], [14]. In such networks, fusing radar decisions and applying location diversity can improve the LPI property [15], [16], [17], [18]. Efficient radar-target assignment (RTA) is crucial for multi-radar multi-target networks to maintain the LPI property while preserving constraints on detection performance, complexity, and power budgets [19], [20].

A. RELATED WORKS

In this subsection, we provide an overview of the existing works on RTA. Yang et al. [9] proposed a method to reduce the number of required radar switches in the assignment scheme. They analyzed the MRMTA problem for detecting ballistic missiles and developed an objective function that integrates tracking accuracy and radar switch rate. Reference [10] proposed a resource allocation scheme that not only assigns multiple radars to each target separately but also minimizes the resource consumption of each radar group. This resource allocation method is limited to preserving tracking accuracies required to detect ballistic missile targets and aviation targets. Reference [11], similar to [10], proposed a cooperative MRMTA in addition to dwelling time allocation for a phased array radar network. Reference [12] transformed the task allocation problem of inverse synthetic aperture radar imaging into a time resource optimization problem for a radar network. The proposed model utilizes radar resources to meet the required imaging resolutions. In [13], [14], [15], genetic-based algorithms were proposed to find a solution for the complex combinatorial MRMTA problem of radar networks. Reference [16] proposed a heuristic algorithm to solve the MRMTA problem considering the LPI as the objective function and detection probability as the quality-of-service constraint. Reference [17] proposed a binary linear programming method to solve a linearized RTA problem. The proposed method had obvious performance advantages in terms of the number of RTAs and radar switches. References [18], [19] used non-linear goal programming and mixed-integer programming to solve weapon-target assignment schemes and used numerical experiments to demonstrate the effectiveness of the proposed methods. Reference [20] studied the power allocation and RTA problem of distributed multi-input multi-output (MIMO) radar networks and designed a joint allocation strategy based on a two-step spectral projected gradient-based solution.

B. MOTIVATIONS OF THIS WORK

In this subsection, we discuss the motivations behind our work. Although MRMTA has many advantages, finding an optimal solution requires exhaustive search, i.e., a brute-force (BF) approach, which has exponential complexity and is not practical for networks with a large number of targets and/or radars. This means that an optimal combination of radars must be selected for each target by sequentially or exhaustively searching through all possible combinations, resulting in unacceptable latency and increased system costs [16]. Therefore, an MRMTA scheme that can support latency-critical and complexity-critical networks is still needed.

Previous research in the field of edge computing has proposed various algorithms, such as MOERA [21] and an A3C-based scheduler [22], to optimize resource allocation in dynamic and stochastic environments. However, these algorithms were developed for different problem domains and may not be directly applicable to the MRMTA problem addressed in our work. Although some preliminary work has attempted to address this aspect [23], most studies have designed machine-learning (ML) models with fixed inputs, which may not be suitable for varying environmental factors such as changing numbers of radars or targets. Notably, Meng et al. [23] have shown that MRMTA based on deep reinforcement learning outperforms random assignment methods and heuristic algorithms in terms of cumulative detection duration. However, their approach may not be viable for scenarios with limited data due to the extensive training data required for deep reinforcement learning. Moreover, their incremental search method for reducing the action space may not be effective in highly dynamic environments where the number of targets and radar nodes can rapidly fluctuate. Consequently, alternative approaches are still necessary to achieve efficient and effective MRMTA in various scenarios.

C. OUR CONTRIBUTIONS

This paper proposes a supervised ML approach for MRMTA using a feed-forward neural network (FNN) with preprocessing and postprocessing techniques. To prepare the dataset, a convex optimization (CO)-based algorithm for power allocation (PA) is proposed, and MRMTA labels are added through a BF approach. During the online phase, the *trained* FNN is applied to the input data to assign radars to the targets. The proposed ML-based MRMTA eliminates the need for a complex BF approach, which significantly reduces computation time. An adaptive FNN structure was designed using various numerical simulations, and the effectiveness of its adaptive designation was demonstrated for different environmental factors. Succinctly, the contributions and novelties of this paper are highlighted as follows:

- We develop a formulation for joint power allocation and radar-target assignment (JPARTA) in multi-radar multi-target scenarios to achieve low probability of intercept (LPI) radar networks.
- We propose an algorithm that jointly optimizes power allocation and radar-target assignment. Our approach involves a convex optimization (CO)-based algorithm for the power allocation sub-problem and a BF-based algorithm for the radar-target assignment sub-problem. We also introduce an alternating algorithm with two steps for solving the JPARTA problem.
- We convert the JPARTA problem into a regression problem to enable it to be solved using machine learning (ML) approaches.
- We have finally incorporated ML into the JPARTA problem to enhance its computational efficiency, making it a more attractive option for practical implementation. Additionally, we have designed an ML structure that ensures high accuracy, fast convergence speed, and scalability.

D. ORGANIZATION

The organization of the paper is as follows. Section II presents the multi-radar multi-target network model and formulates the JPARTA problem to improve LPI performance. Section III focuses on the CO approach to the PA problem. Section IV discusses the challenges of using the BF approach for MRMTA and presents the supervised ML framework for MRMTA. Section V presents the implementation of the ML framework, followed by the numerical simulations and discussions in Section VI. Finally, Section VII concludes the paper and highlights a future direction.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. SYSTEM CONFIGURATION

In this paper, we consider a radar network architecture with M radars and N targets, as shown in Fig. 1. We assume

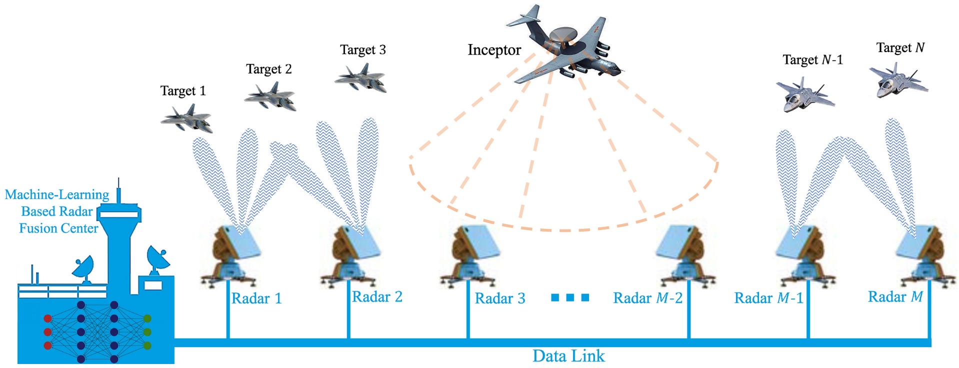

FIGURE 1. The proposed Machine-learning based LPI radar network with M radars and N targets.

that multiple radars can be assigned to each target in a multi-target environment, meaning that a single target can be detected by multiple radars. The radar nodes share their detections with a fusion center through data links.[1] A single fusion center intends to combine information from the M radars for the N targets. Therefore, there are M^N possible radar-to-target assignment schemes.

The variable $p^t_{m,n}$ is defined as the transmit power of the m-th radar when it is assigned to the n-th target where $m \in \mathcal{M} \triangleq \{1, 2, \ldots, M\}$ denoting the radar index, and $n \in \mathcal{N} \triangleq \{1, 2, \ldots, N\}$ denoting the target index. According to the well-known radar equation, the signal-to-noise ratio (SNR) of the received signal at the m-th radar from the n-th target can be expressed as [16]

$$\mathrm{SNR}_{m,n} = \frac{p^t_{m,n} G_{t_m} G_{r_m} \sigma_{m,n} \lambda^2_m}{(4\pi)^3 r^4_{m,n} \delta_m}, \tag{1}$$

where G_{t_m} and G_{r_m} are the transmitting antenna and receiving antenna gain of the m-th radar, respectively. The average radar cross-section (RCS) of the n-th target with respect to the m-th radar is represented by $\sigma_{m,n}$. λ_m denotes the wavelength of the transmitted wave, and $r_{m,n}$ is the distance between the m-th radar and the n-th target.[2] Additionally, δ_m represents the thermal noise power in the radar receiver, which can be defined as $\delta_m = KT_{0m}B_{rm}F_{rm}L_{rm}$, where K is Boltzmann's constant (1.3×10^{-23} JK^{-1}), T_{0m} is the ambient temperature (290°K), B_{rm} and F_{rm} represent the instantaneous receiver bandwidth in Hz and the noise figure of the receiver subsystem, respectively. Finally, $L_{r,m}$ denotes the system loss.[3]

The detection probability of target n in the m-th radar, $P^{\mathrm{D}}_{m,n}$, is assumed to follow a Swerling model of type I or II for the target RCS in this work. According to [26], it can be expressed as

$$P^{\mathrm{D}}_{m,n} = \left(P^{\mathrm{FA}}\right)^{\left(1+\mathrm{SNR}_{m,n}\right)^{-1}}, \tag{2}$$

where P^{FA} denotes the desired false alarm probability.

B. LPI PROBLEM FORMULATION

In scenarios with multiple targets, optimizing the usage of network resources is crucial for minimizing the intercept probability of the network. This involves identifying the best subset of radars to assign to each target and optimizing their transmit powers. It's worth noting that the power of radar signals received by the interceptor is directly proportional to the transmit powers assigned to the selected radars. Hence, higher transmit power leads to a higher intercept probability [24].

The interceptor operating as a passive sensor, remains undetectable while it senses radar signals. Furthermore, if it intercepts only one of the radars within a network, all radars in the network become compromised, resulting in the entire radar network being considered intercepted. Therefore, the probability of interception is directly influenced by the radar with the highest power. To effectively reduce the likelihood of interception, it is advised to decrease the maximum radiated power of active radars in the network.

In this regard, our objective in this paper is to formulate the LPI optimization problem for JPARTA. This involves considering the collective detection performance of all radars in the network and aiming to minimize the maximum power radiated by the radars while meeting the required detection probability for the entire network. The problem can be stated as

$$(\mathrm{P}): \min_{x_{m,n}, p^t_{m,n}} \max_{m\in\mathcal{M}, n\in\mathcal{N}} \; x_{m,n} p^t_{m,n} \tag{3a}$$

$$\text{subject to } 1 - \prod_{m=1}^{M}\left(1 - x_{m,n}P^{\mathrm{D}}_{m,n}\right) \geq P^{\mathrm{D}}_{\mathrm{Th}}, \forall n \in \mathcal{N}, \tag{3b}$$

$$x_{m,n}p^t_{m,n} \leq p_{\mathrm{Max}}, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{N}, \tag{3c}$$

$$x_{m,n} \in \{0, 1\}, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{N}, \tag{3d}$$

where $x_{m,n}$ is the assignment index, $P^{\mathrm{D}}_{\mathrm{Th}}$ represents the desired detection probability, and p_{Max} indicates the peak

1. Wireless communication technologies such as microwave, satellite, and cellular networks can serve as data links.
2. It should be noted that the information regarding the number or RCS and distance between the radar and target can be effectively shared among all radars through a data link, enabling the radar network to handle varying target characteristics.
3. The simulations presented in this paper were carried out under the assumption that the radar can distinguish targets from clutter.

power of each radar. The constraints (3b) and (3c) consider the global detection probability of each target, which is the performance metric, and the limited power budget of each radar, respectively.[4]

The joint optimization problem (P) is very complicated due to the integer nature of the assignment index $x_{m,n}$ and the mutual dependency of PA and MRMTA. Therefore, solving it with rational complexity is not feasible [25]. To tackle this issue, we solve the PA problem in the first sub-problem for a given radar-target assignment scheme, and then, in the second sub-problem, we select the best assignment scheme based on the obtained powers.

Specifically, we solve the optimization problem (P) with the following two related steps:

Step 1: Sub-problem 1- Optimizing PA for a given RTA and repeating for all considered RTA schemes.

Step 2: Sub-problem 2- selecting the best RTA scheme based on obtained PA schemes. The PA related to the best RTA scheme is the optimum JPARTA scheme.

III. SUB-PROBLEM 1: POWER ALLOCATION

In the first sub-problem, we aim to solve the PA problem for a given radar-target assignment, such as the c-th assignment scheme. Specifically, we address this problem for the n-th target using the following optimization problem formulation:

$$(\text{P1}): \min_{p^t_{m,n}(c)} \max_{m \in \mathcal{M}_n(c)} \quad x_{m,n} p^t_{m,n}(c) \tag{4a}$$

$$\text{subject to} \quad 1 - \prod_{m \in \mathcal{M}_n(c)} \left(1 - x_{m,n} P^{\text{D}}_{m,n}\right) \geq P^{\text{D}}_{\text{Th}}, \tag{4b}$$

$$p^t_{m,n}(c) \leq p_{\text{Max}}, \qquad \forall m \in \mathcal{M}_n(c), \tag{4c}$$

where $\mathcal{M}_n(c)$ represents a subset of radars assigned to the n-th target in the scheme c. It is worth noting that the optimization problem (4) is convex and can be solved using a classic CO-based framework.

The cost function (4a) is designed to prioritize lower power levels, while (2) reveals that detection probability increases as the assigned powers grow. Consequently, it becomes clear that achieving equality in (4b) is essential to attain the optimal lower power level, as

$$1 - \prod_{m \in \mathcal{M}_n(c)} \left(1 - x_{m,n} P^{\text{D}}_{m,n}\right) = P^{\text{D}}_{\text{Th}}. \tag{5}$$

It is important to note that in the min-max problem, the optimal solution involves allocating the same power to all assigned radars [16]. This allocation strategy ensures that the power distribution achieves a balance that minimizes the maximum performance across all radars. By enforcing equal

4. We assume that the control center uses the OR rule for simplicity, although there are other decision rules such as AND, or M-out-of-N. It can be noted that this work can be similarly extended to other decision rules as well.

Algorithm 1 Proposed Power Allocation Algorithm

1: **Initialize** $a = 0,\ b = 2p_{\text{Max}}$
2: **while** $|a - b| > \epsilon$ **do**
3: $\quad p^t_n(c) = \frac{a+b}{2}$
4: $\quad$ **if** (7) holds true **then**
5: $\qquad b = p^t_n(c)$
6: $\quad$ **else**
7: $\qquad a = p^t_n(c)$
8: $\quad$ **end if**
9: **end while**
10: $p^t_n(c) = \frac{a+b}{2}$

power allocation, the same power must be allocated to all radars assigned to the n-th target, i.e.,

$$p^t_{m,n}(c) = \begin{cases} p^t_n(c), \text{ if m} \in \mathcal{M}_\text{n}(\text{c}), \\ 0, \quad \text{otherwise.} \end{cases} \tag{6}$$

By substituting (1), (2), and (6) into (5), we can compute $p^t_n(c)$ by solving the following non-linear equation:

$$\left(1 - \prod_{m \in \mathcal{M}_n(c)} \left[1 - \left(P^{\text{FA}}\right)^{\left(1 + \frac{p^t_n(c) G_{t_m} G_{r_m} \sigma_{m,n} \lambda_m^2}{(4\pi)^3 r^4_{m,n} \delta_m}\right)^{-1}}\right]\right) - P^{\text{D}}_{\text{Th}} = 0. \tag{7}$$

The left-hand side of the equation is an increasing function of $p^t_n(c)$, so the unique root of the equation can be obtained by a well-known numerical bisection-based search method. The proposed PA algorithm is presented in Algorithm 1. This algorithm guarantees convergence to either the root of Equation (7) within the interval $[0, p_{\text{Max}}]$ or to the value of p_{Max}. To achieve an error smaller than ϵ, a minimum number of iterations is required, given by:

$$n \geq \frac{\ln(2p_{\text{Max}}) - \ln(\epsilon))}{\ln(2)}. \tag{8}$$

IV. SUB-PROBLEM 2: MULTI-RADAR TO MULTI-TARGET ASSIGNMENT

The second sub-problem aims to find the optimal MRMTA scheme, which requires an exhaustive BF search over all possible schemes. However, this approach has high computational demands, and its complexity grows exponentially with the number of targets and radars, resulting in a complexity of $\mathcal{O}(2^{MN})$, where $\mathcal{O}(\cdot)$ describes the order of complexity. Consequently, the BF approach is not practical for real-time implementation, particularly when dealing with a large number of targets and/or radars. Although some sub-optimal heuristic techniques, such as those introduced in [16] and [29], have been proposed to solve the RTA problem with reduced complexity, their computational complexity remains high, making them unattractive for practical implementation. It is worth noting that optimizing the RTA in the general case, where multiple radars can be assigned to each target in a multi-target environment, is fundamental and essential, as highlighted in previous studies [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [23].

In this subsection, we aim to bridge this gap by introducing supervised ML to enable low-complexity MRMTA. This can be achieved by training a FNN with a BF approach database that returns accurate subsets of selected radars. The proposed algorithm is summarized in Algorithm 2, which operates in two phases: the learning phase and the prediction phase.

During the learning phase, we employ an offline BF approach to find the optimum RTA as labeled data, while simultaneously collecting the dataset necessary for training the ML-based model. Once the dataset is fully acquired, the FNN is trained using the training dataset, which is then utilized in the prediction phase. More details regarding the proposed algorithm will be explained in the following subsections.

Algorithm 2 ML-Based Algorithm for JPARTA Problem

Initialize $\mathbf{P}^t = 0 \in \mathbb{R}^{C\times N}$, $\mathbf{R} = 0 \in \mathbb{R}^{M\times N}$, $\mathbf{X}^* = 0 \in \mathbb{R}^{M\times N}$
Phase I: Data Generation
for $s = 1$ **to** S **do**
  A. Power Allocation Sub-Problem
  for $n = 1$ **to** N **do**
    for $c = 1$ **to** C **do**
      Obtain $p_n^t(c)$ using Algorithm 1
    end for
  end for
  B. Radar-Target Assignment Sub-Problem
  for $c = 1$ **to** C **do**
    $c^* = \min_c \max_{n\in\mathcal{N}} p_n^t(c)$
    for $m = 1$ **to** M **do**
      for $n = 1$ **to** N **do**
        if $m \in \mathcal{M}(c^*)$ **and** $n \in \mathcal{N}$ **then**
          $x_{m,n}^* = 1$
        else
          $x_{m,n}^* = 0$
        end if
      end for
    end for
    The optimal assignment matrix $\mathbf{X}^*$ is converted into a real-valued vector $\mathbf{x}$.
  end for
  $\mathbf{D}(s) \leftarrow \{\mathbf{R}, \mathbf{x}\}$, store the new entry in the learning dataset.
end for
Train the ML model using the learning dataset $\mathbf{D}$.
Phase II: Prediction
while True **do**
  The fusion control calculates $\mathbf{R}$.
  Predict the assignment matrix $\hat{\mathbf{X}}$ using the trained FNN.
  Power allocation can be obtained by utilizing the predicted assignment matrix through Algorithm 1
end while

A. DATASET INFORMATION

1) INPUT DATA GENERATION

In the surveillance region, which is a square area with dimensions of $L \times L$ km^2, we deploy N targets uniformly. To adequately cover the region, M radars are distributed throughout the network. In each individual run, the distances of the targets to the radars are determined, resulting in the generation of an input matrix R. This matrix $\mathbf{R}$ has dimensions of $M \times N$ and contains the distances ($r_{m,n}$) between the m-th radar and the n-th target. A total of 100, 000 samples are generated in this process.[5]

2) OUTPUT DATA GENERATION

Fusing decisions from all radars can result in excessive data traffic over the data links connected to the fusion center, which can increase the global false alarm probability as more radars are fused. To account for this, we limit the network complexity by assuming that at most l radars can be combined to detect each target. This means that each target can be detected based on the fusion of one, two, or at most l radars' decisions.

To generate the optimal RTA matrix $\mathbf{X}^*$, we apply a brute force approach over all possible assignment schemes. The number of possible assignment schemes is equal to $C = \binom{M}{l}^N = (\frac{M!}{l!(M-l)!})^N$. For each assignment scheme, we obtain $p_n^t(c)$ by using a bisection-based search over equation (7) for each n. We then determine the optimum scheme c^* by finding the scheme with the minimum required effective power, which is the one that assigns a subset of radars to the n-th target. We set $x_{m,n}^* = \begin{cases} 1, & \text{if } m \in \mathcal{M}(c^*), n \in \mathcal{N} \\ 0, & \text{otherwise} \end{cases}$ in the RTA matrix.

To convert the target matrix $\mathbf{X}^*$ into a regression problem format, each column of $\mathbf{X}^*$, i.e., $[x_{1,n}^*, x_{2,n}^*, \ldots, x_{M,n}^*]^T$ for $n \in \mathcal{N}$, is transformed into an integer value. This is done by representing each column as a binary string, where 1 indicates radar assignment and 0 indicates no assignment. For example, if the column for target n is given as $[0, 1, 1, 0]$, it represents that the second and third radars are assigned to target n, while the first and fourth radars are not. The binary string is then converted to its corresponding integer value $\tilde{x}_n \in [1, (1-2^{-l})2^M]$. The transformation ensures that each column's binary representation uniquely maps to an integer value. The resulting transformation yields an $N \times 1$ vector $\mathbf{x}$, which represents the radar assignments for each target in a compact form suitable for regression analysis.

5. It is noteworthy that while increasing the number of training samples generally improves the performance of ML model, there is a diminishing return effect. In our experiments, we observed that beyond a certain point, adding more training samples (beyond 100, 000 samples) did not significantly enhance the accuracy.

3) DATASET GENERATION

Let $\{\mathbf{R}, \mathbf{x}\}$ denote a labeled dataset. We assume that $\Lambda_s = \{\mathbf{R}_s, \mathbf{x}_s\}, \forall s \in \mathcal{S} = \{1, 2, \ldots, S\}$ represents S different labeled datasets. The goal of supervised learning is to learn a general rule for mapping $\mathbf{R}_s$ to $\mathbf{x}_s$ using the labeled dataset, resulting in the estimated subset of selected radars $\hat{\mathbf{x}}_s$.

4) DATASET SPLITTING

The data generated is divided into training and test datasets, with 90% of the data allocated for training purposes, while the remaining 10% is reserved for the test set. The test set evaluates the model's generalization by containing unseen scenarios not encountered during training. This assessment ensures accurate predictions for new data and real-world applicability.

B. PREPROCESSING

Preprocessing is crucial in enhancing the training efficiency of ML algorithms. The radar-target distances, denoted by $\mathbf{R}s = [r_{m,n}(s)]_{m \in \mathcal{M}, n \in \mathcal{N}}$, have a large input span in the range of $(0, \infty)$, which creates significant challenges in ML convergence. To address this issue, we propose a two-step preprocessing procedure.

In the first step, we perform dimensionality reduction by representing the rearranged radar-target distance matrix, i.e., $\mathbf{R}_s$, as an $MN \times 1$ vector $\mathbf{c}_s$. In the second step, we normalize the vector $\mathbf{c}_s$ using a simple per-dataset scaling method, resulting in the input vector $\mathbf{v}_s$. This normalization method ensures a stable and meaningful learning process. Specifically, all samples are normalized using one constant value over the entire input data, and we obtain $\mathbf{v}_s = \frac{\mathbf{c}_s}{\max\limits_s ||\mathbf{c}_s||_\infty}$ for $s = 1, \ldots, S$. After applying these two preprocessing steps, the input span is restricted to $(0, 1]$.

To improve the training efficiency further, we also normalize the labels. We use a similar per-dataset normalization method, where every optimal radar-target assignment vector, denoted as $\mathbf{x}_s$, is normalized using the maximum value of the dataset, i.e., $(1-2^{-l})2^M$. The output of the normalization process over the label of the s-th sample of the dataset is denoted by $\mathbf{q}_s$. By normalizing both the input and output, we ensure that the target-radar distance information is preserved while addressing the convergence issue.

C. ML STRUCTURE AND TRAINING PROCEDURE

We assume a basic fully-connected FNN architecture with D layers consisting of an input layer, an output layer, and $D-2$ hidden layers, as shown in Fig. 2. Each layer has N_d neurons, where $d \in 1, 2, \ldots, D$. We train the proposed FNN-based approach using the backpropagation algorithm with the Mean Squared Error (MSE) loss function [30], [31].

The loss function $L(\mathbf{x}, \hat{\mathbf{x}})$ is defined as:

$$L\big(\mathbf{x}, \hat{\mathbf{x}}\big) = \frac{1}{N} \sum_{n=1}^{M} \big(\hat{x}_n - x_n\big)^2, \tag{9}$$

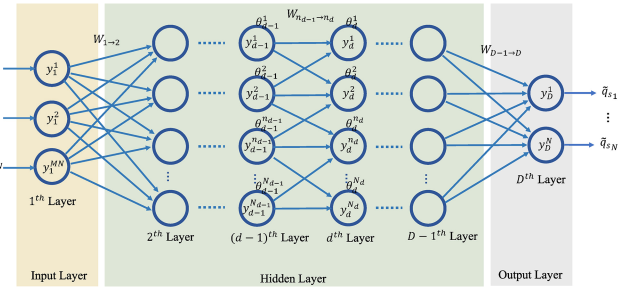

FIGURE 2. Structure of the fully-connected feedforward neural network (FNN) used in this paper.

where $\hat{\mathbf{x}}$ denotes the label or ground truth, and $\mathbf{x}$ denotes the output of the neural network.

The optimization problem can be formulated as:

$$\arg\min_{\mathbf{W}, \Theta} \sum_{s=1}^{S} L(\mathbf{x}_s, \mathbf{q}_s), \tag{10}$$

where $\mathbf{W} = \{w_{n_d \to n_{d+1}}\}_{d=1}^{D}$ and $\Theta = \{\theta_d^{n_d}\}_{d=1}^{D}$ denote the weights and biases, respectively. We apply the stochastic gradient descent (SGD) method [32] for the optimization.

We randomly partition the input data into mini-batches of size B and update the weights and biases of the neural network based on each mini-batch. The update rule can be formulated as:

$$\begin{aligned} w_{n_d \to n_{d+1}}(e+1) &= w_{n_d \to n_{d+1}}(e) - \alpha \frac{\partial L}{\partial w_{n_d \to n_{d+1}}}(e), \\ \theta_d^{n_d}(e+1) &= \theta_d^{n_d}(e) - \alpha \frac{\partial L}{\partial \theta_d^{n_d}}(e), \end{aligned} \tag{11}$$

where α is the learning rate, and $\frac{\partial L}{\partial w_{n_d \to n_{d+1}}}(e)$ and $\frac{\partial L}{\partial \theta_d^{n_d}}(e)$ are the partial derivatives of the loss function with respect to the weights and biases at the e-th epoch, respectively.

In this paper, all hidden layers use Rectified Linear Units (ReLU) as the activation function [32]. Specifically, the activity of the n_d-th neuron can be expressed as:

$$y_d^{n_d}(e) = \mathrm{ReLU}\left(\sum_{m=1}^{N_{d-1}} w_{m \to n_d}(e) y_{d-1}^{m}(e) + \theta_d^{n_d}(e)\right), \tag{12}$$

where $\mathrm{ReLU}(.) = \max(0, .)$.

The partial derivative of the loss function with respect to the weights and biases can be calculated using the backpropagation algorithm. At each epoch, the neural network updates its weights and biases based on the mini-batches of the training data set, and the training process continues until the convergence criterion is met. In this paper, we stop the training procedure when the validation loss does not decrease for a certain number of epochs.[6]

6. It's worth mentioning that even though retraining is unnecessary for changes in the target's position, updating the trained ML model is necessary when the number of radars and/or targets is changed.

D. POSTPROCESSING

The FNN output is an $N \times 1$ vector $\widetilde{\mathbf{q}}_s$. To reconstruct the RTA matrix from this vector, we propose a three-step inverse postprocessing procedure:

- In the first step, the entries in $\widetilde{\mathbf{q}}_s$ are multiplied by $(1-2^{-l})2^M$ to construct an intermediate $N \times 1$ vector $\widetilde{\mathbf{p}}_s$.
- In the second step, we round the entries of $\widetilde{\mathbf{p}}_s$ to the nearest integers within a bound[7] of $(1-2^{-l})2^M$, and define this as the $N \times 1$ vector $\widetilde{\mathbf{f}}_s$.[8]
- In the final step, the integer values in $\widetilde{\mathbf{f}}_s$ are converted to binary strings, forming the estimated RTA matrix $\widetilde{\mathbf{X}}_s$, which represents the subset of selected radars for each target in the radar network.

This procedure allows for the reconstruction of the estimated RTA matrix $\widetilde{\mathbf{X}}_s$ from the FNN output $\widetilde{\mathbf{q}}_s$, providing the radar assignments for each target in the radar network.

E. PERFORMANCE METRICS

To evaluate the FNN training performance, we construct a vector of optimal indexes of the selected radars, denoted by $\mathbf{f}_s$, from $\mathbf{X}_s$. The MSE over epochs is then defined as $\text{MSE} = \mathbb{E}\{||\mathbf{f}_s - \hat{\mathbf{f}}_s||_2^2\}$ to measure the training performance, where $\|\cdot\|_n$ represents the n-norm operator, and $\mathbb{E}\{\cdot\}$ denotes statistical expectation. We also define the network accuracy as the correct detection rate, given by ACC (%) $= \frac{N_c}{N \times S} \times 100$, where N_c denotes the number of correctly predicted $\hat{\mathbf{f}}_s$.

V. ML-BASED APPROACH IMPLEMENTATION

The implementation of FNNs codes were carried out in Python 3.9.7 using TensorFlow 2.8.0. The batch size of 432 and a learning rate of 0.01 were selected to ensure the lowest possible MSE with a relatively high convergence speed. To compare the performance of back-propagation, three classic optimizers, namely adaptive moment (Adam), stochastic gradient descent (SGD), and root mean squared propagation (RMSProp) were adopted.[9] For simplicity, all neural biases were initialized with zero, while all link weights were initialized randomly. In our experiments, each training round is configured with a maximum of 100 epochs, i.e., $E = 100$.

In this study, we aimed to design an ML structure that ensures high accuracy, fast convergence speed, and scalability. To achieve this goal and investigate the impacts of different structures on performance, we created a set of FNN structures and compared them. The FNN layers were specified using the same names and properties as presented in Table 1, including the input (labeled as I), output (labeled as O), and six hidden layers (labeled as H_n). To address the bottleneck effect in the structures, we applied a dual-funnel shape and a slice-reconstruct manner to the layers, as suggested by [28].

7. The n-th entry in $\widetilde{\mathbf{f}}_s$ represents the approximate indexes of the selected radars for the n-th target.

8. Steps 1 and 2 ensure the strict adherence of FNN outputs to the constraint of assigning at most l radars to a single target.

9. It is important to note that various optimizers use different optimization algorithms to adjust the ML parameters.

TABLE 1. Specifications of the FNN layers used in this paper.

Layer ID	Layer type	Layer connection	Neuron number	Activation function
I	Input	FC	MN	ReLU
H_1	Hidden	FC	MN	ReLU
H_2	Hidden	FC	$2MN$	ReLU
H_4	Hidden	FC	$4MN$	ReLU
H_8	Hidden	FC	$8MN$	ReLU
H_{16}	Hidden	FC	$16MN$	ReLU
H_{32}	Hidden	FC	$32MN$	ReLU
O	Output	FC	N	Linear

In this regard, we slice the FNN Structure 6 into 13 layers and then select some layers to reconstruct new FNN structures. We study six structures of the FNN in the experiments, which are designed in the above-mentioned manner and numbered as follows:

- Structure $6 = I, H_1, H_2, H_4, H_8, H_{16}, H_{32}, H_{16}, H_8, H_4, H_2, H_1, O$
- Structure $5 = I, H_1, H_2, H_4, H_8, H_{16}, H_8, H_4, H_2, H_1, O$
- Structure $4 = I, H_1, H_2, H_4, H_8, H_4, H_2, H_1, O$
- Structure $3 = I, H_1, H_2, H_4, H_2, H_1, O$
- Structure $2 = I, H_1, H_2, H_1, O$
- Structure $1 = I, H_1, O$.

VI. SIMULATION RESULTS

To assess the effectiveness of the proposed ML-based JPARTA algorithm (Algorithm 2), we conducted numerical simulations under various scenarios.

The surveillance region considered in our simulations has dimensions of 180×180 km^2. We set $M = 4$ and $N = 2$ in Figures 3-7. However, to comprehensively evaluate the performance of our proposed method, we extended the simulation of this study to more complex scenarios involving $N = M = 2, 4, 8$ as shown in Table 3. The RCS of all targets is assumed to be 1 m^2.[10]

We generated the training dataset by solving subproblem 2 for different instantaneous target positions, which were uniformly and randomly placed in the surveillance region. All radars had the same parameters and were assumed to be of airport surveillance radar (ASR)-9 type [27], with specifications given in Table 2. We set P_{Th}^{D} to 0.9 for all radars.

To provide a comprehensive evaluation of our proposed ML-based assignment method, we have included several simulations in the following subsections. These simulations encompass a wide range of performance metrics, including MSE, LPI, computational complexity, computation time, and accuracy.

10. The MSE values are averaged over multiple runs with different random initializations.

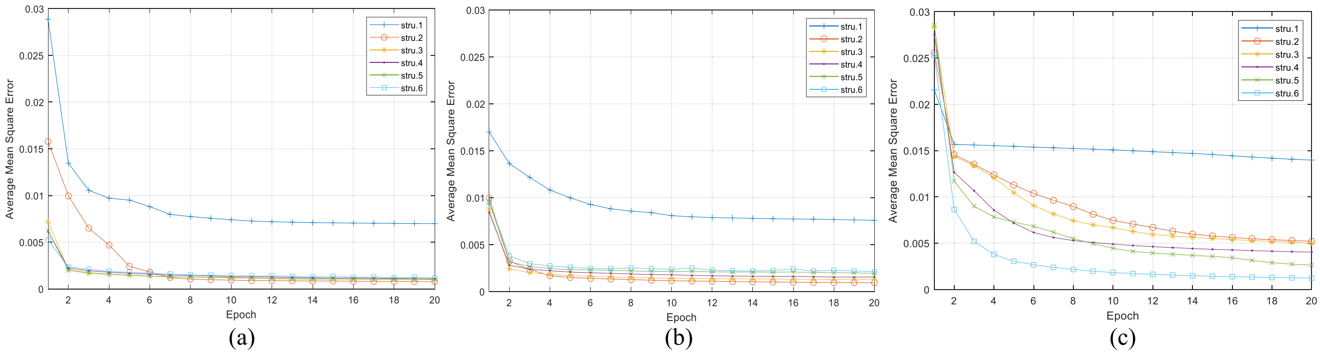

FIGURE 3. MSE for different FNN structures when (a) Adam, (b) RMSProp, and (c) SGD are adopted as the optimizer.

TABLE 2. Radar parameters for airport surveillance radar (ASR)-9 type.

Description	Symbol	Value
Peak Power	p_{Max}	1.3 MWatt
Transmitted Antenna Gain	G_{t_m}	30 dB
Received Antenna Gain	G_{r_m}	30 dB
System Loss	L_m	21.1 dB
Bandwidth	B_{r_m}	1 MHz
wavelength	λ_m	0.1 m

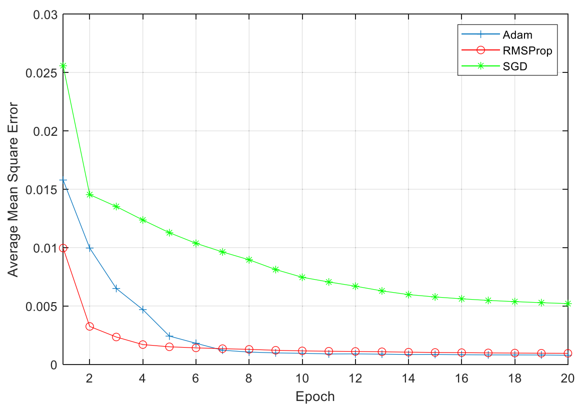

FIGURE 4. MSE for the fully connected feedforward neural network with Structure 2, trained using different optimizers.

A. SELECTION OF FNN STRUCTURE AND OPTIMIZER FOR ML

To gain insights into the performance of different FNN structures and optimizers, we compare their MSE on the training set, which is a key performance metric, in Fig. 3 and Fig. 4.

Fig. 3 reveals that all FNN structures improve the MSE to varying degrees, but the oversimplified Structure 1 performs relatively poorly. Interestingly, it is observed that more complex FNN structures, e.g., Structures 3-6, do not necessarily lead to the best performance. Instead, Structure 2 with five layers achieves the best performance, albeit with a longer convergence time than Structures 3-6.

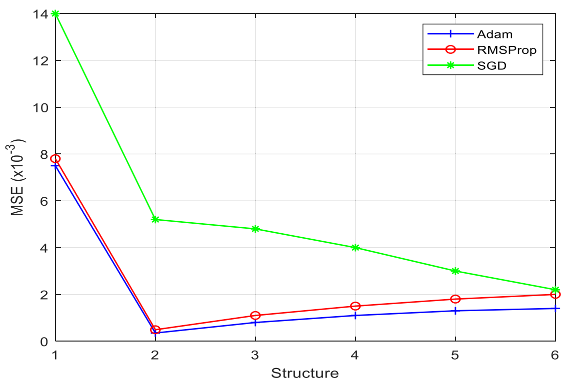

FIGURE 5. MSE of the test data for different optimizers using different FNN structures.

Fig. 4 shows that, for the MRMTA problem, the Adam optimizer outperforms the others with the minimum MSE. However, the RMSProp optimizer demonstrates the fastest convergence, and the SGD optimizer is the least suitable for our model, resulting in significantly higher MSE.

Fig. 5 presents the MSE for different training algorithms with respect to different FNN structures. A model with the lowest MSE is considered a remarkable fit model. The results in Fig. 5 confirm that Structure 2 with the Adam optimizer yields the best model with minimum error.

B. ANALYSIS OF ML TRAINING PROCESS

Fig. 6 depicts the MSE of the training and test data sets during the training phase at different epoch levels to analyze the training process. We used Structure 2 as the FNN structure and Adam as the optimizer to ensure sufficient expressive power and the best optimizer. As shown in Fig. 6, the performance of both data sets keeps improving as the number of epochs increases, and the MSE of both the training and test data sets decreases, indicating that overfitting has not occurred. These results demonstrate that the *trained* FNN Structure 2 can perfectly explain a training data set and generalize well, making it the best performer among all structures for solving the MRMTA problem.

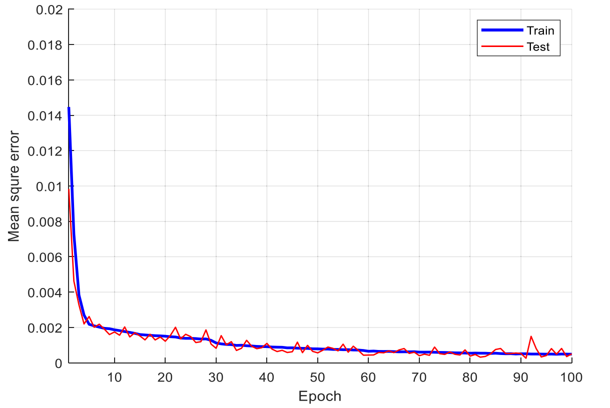

FIGURE 6. MSE of training and test datasets versus epoch level during the training phase.

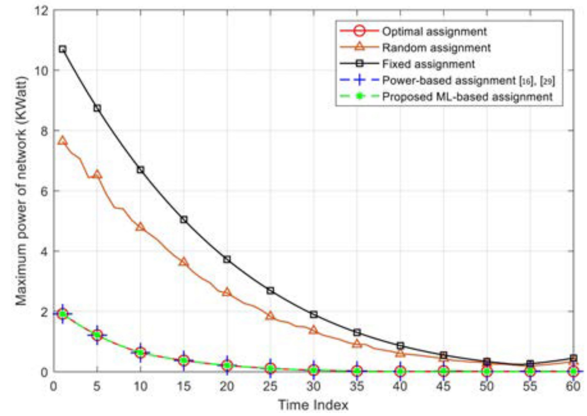

FIGURE 7. LPI performance versus time for different RTA schemes.

TABLE 3. Computational complexity and time consumption for 100, 000 samples using different assignment methods [a].

Method	Optimal assignment	Power-based assignment	Proposed ML-based assignment	
	Time (sec)	Time (sec)	Time (sec)	Accuracy (%)
$M = N = 2$	10.02	9.95	0.227	98.97
$M = N = 4$	2.03×10^2	71.22	0.232	93.52
$M = N = 8$	45.9×10^3	610.5	0.308	90.28

C. COMPUTATION AND SCALABILITY ANALYSIS

In this sub-section, we analyze the computational advantage of supervised ML and compare it with optimal assignment and power-based assignment. Table 3 shows the computational complexity of each method, where the optimal assignment needs to traverse all possibilities and select the best choice using the BF approach, and the power-based assignment assigns targets to radars by a heuristic algorithm based on the power matrix obtained from sub-problem 1 [16], [29]. To ensure efficiency, we adopt Structure 2 with Adam optimizer as the ML configuration. All measurements were carried out on the same computer (i5-2450M CPU, Nvidia Geforce GT 620M).

From Table 3, it can be observed that the computation time of the *trained* FNN is much smaller than that of optimal assignment or power-based assignment. Notably, when $M = N = 2$, even with less than 2% loss in accuracy,[11] the ML-based method achieves a more than 43 times increase in time efficiency, which validates the trade-off between accuracy and computation times of our proposed ML-based method. While the optimal assignment has a computational complexity of $\mathcal{O}\left(\binom{N}{l}^N\right)$ and the power-based assignment has a computational complexity of $\mathcal{O}\left(\binom{N}{l} \times N\right)$ [16], the computational complexity of ML-based assignment reduces to $\mathcal{O}(N^2)$ [30], which is practical for many applications.

To evaluate the scalability of our proposed method, we present the performance for different problem sizes in Table 3, including $N = M = 2, 4, 8$. As shown in the table, the computation time increases with the problem size due to its increased complexity. However, for all cases, the accuracy of the ML-based model remains almost constant and entirely satisfactory. This is due to the adaptive design of the FNN structure with the input size, as proposed in this paper. Therefore, our method is scalable and can handle larger problem sizes without sacrificing accuracy.

D. LPI PERFORMANCE ANALYSIS

Table 3 also shows that the proposed ML-based algorithm's radar set allocation is not matched with the labels in less than 10% of realizations. However, it is worth noting that these different allocations do not significantly affect the power of active radars.

In this context, aside from examining the training performance, it is important to evaluate the impact of employing supervised ML on the LPI performance (i.e., maximum transmitted power) of the netted radars. Therefore, we evaluated the LPI performance of PA and MRMTA in the proposed ML-based algorithm. For this purpose, we considered a scenario in which four radars in a surveillance region with a given arrangement detect two targets with different azimuths and ranges. In Fig. 7, we compared the LPI performance of the proposed ML-based assignment with the optimal assignment, random assignment, fixed assignment, and power-based assignment. In the random assignment, each radar is assumed to have an equal chance of being selected for each target, while in the fixed assignment, radars are randomly assigned to the target and remain unchanged until the end of the scenario.

11. We also conducted k-fold cross-validation [33] (with $k = 10$) to further scrutinize the performance of our proposed FNN. Remarkably, the results obtained through k-fold cross-validation align closely with our findings, confirming the robustness and reliability of our proposed ML-based assignment.

The results depicted in Fig. 7 indicate that the LPI performance achieved by the proposed ML-based assignment is higher than that of random and fixed RTA schemes and approximately matches that of optimal and power-based ones. In fact, this indicates that the proposed algorithm can enable the fusion center to identify a near-optimal set of radars for targets, even when the allocation does not match the labeled data set. This ensures that the LPI performance is not significantly degraded.

VII. CONCLUSION AND FURTHER RESEARCH

In this study, we addressed the issue of improving the LPI performance of a radar network system in a multi-radar multi-target environment by minimizing the maximum radiated power of the network through joint power allocation and radar-target assignment (JPARTA). To solve the JPARTA problem, we used variable separation to transform it into two sub-problems: power allocation (PA) and multi-radar to multi-target assignment (MRMTA). The PA problem was solved using convex optimization (CO), while the MRMTA problem was tackled using a supervised ML-based approach, which was developed to overcome the high computation time associated with the non-linear and non-convex nature of the problem. Extensive numerical simulations were conducted to evaluate the effectiveness of the proposed ML-based scheme, which showed that our algorithm was computationally efficient and achieved equivalent performance compared to the BF-based and power-based assignment methods. Our proposed algorithm also reduced computation time by up to 43 times compared to the benchmark. Therefore, we have provided a new low computational complexity solution for complex MRMTA problems using supervised ML. By employing our algorithm, the application of MRMTA can be extended to latency-critical and complexity-critical networks.

Future research directions can explore the possibility of incorporating additional performance metrics into the optimization process. For example, in addition to minimizing the maximum radiated power, one could aim to minimize the total power consumption of the radar network or maximize the detection probability of the targets. Another avenue for future research could involve the use of unsupervised machine learning methods for MRMTA problems. For instance, it may be worth investigating the use of clustering algorithms, reinforcement learning or deep-Q neural (DQN) techniques.

REFERENCES

[1] F. Hejazikookamari, Y. Norouzi, E. S. Kashani, and M. M. Nayebi, "A novel method to detect and localize LPI radars," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 5, pp. 2327–2336, Oct. 2019, doi: 10.1109/TAES.2018.2885109.

[2] T. Ravi Kishore and K. D. Rao, "Automatic intrapulse modulation classification of advanced LPI radar waveforms," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 53, no. 2, pp. 901–914, Apr. 2017, doi: 10.1109/TAES.2017.2667142.

[3] C. Xue, L. Wang, and D. Zhu, "Joint aperture and power allocation strategy for a radar network localization system based on low probability of interception optimization," *Sensors*, vol. 23, no. 5, p. 2613, 2023, doi: 10.3390/s23052613.

[4] X. Wang, G. Xu, H. Yan, D. Zhu, Y. Wen, and Z. Luo, "LPI radar signals modulation recognition based on ACDCA-ResNeXt," *IEEE Access*, vol. 11, pp. 45168–45180, 2023, doi: 10.1109/ACCESS.2023.3270231.

[5] P. Gong, K. Xu, Y. Wu, J. Zhang, and H. C. So, "Optimization of LPI-FDA-MIMO radar and MIMO communication for spectrum coexistence," *IEEE Wireless Commun. Lett.*, vol. 12, no. 6, pp. 1076–1080, Jun. 2023, doi: 10.1109/LWC.2023.3261419.

[6] K. Chen, L. Wang, J. Zhang, S. Chen, and S. Zhang, "Semantic learning for analysis of overlapping LPI radar signals," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–15, 2023, doi: 10.1109/TIM.2023.3242013.

[7] Y. Su, T. Cheng, and Z. He, "LPI-constrained collaborative transmit beampattern optimization and resource allocation for maneuvering targets tracking in colocated MIMO radar network," *Signal Process.*, vol. 207, Jun. 2023, Art. no. 108935, doi: 10.1016/j.sigpro.2023.108935.

[8] J.-H. Bang, D.-H. Park, W. Lee, D. Kim, and H.-N. Kim, "Accurate estimation of LPI radar pulse train parameters via change point detection," *IEEE Access*, vol. 11, pp. 12796–12807, 2023, doi: 10.1109/ACCESS.2023.3242684.

[9] S. Yang, K. Tian, W. Wu, W. Liu, and G. Zhou, "Target assignment method for phased array radar network in anti-missile early warning," *J. Beijing Univ. Aeronaut. Astronaut.*, vol. 46, no. 7, pp. 1325–1334, Jul. 2020, doi: 10.13700/j.bh.1001-5965.2019.0473.

[10] J. Yan, J. Dai, W. Pu, S. Zhou, H. Liu, and Z. Bao, "Quality of service constrained-resource allocation scheme for multiple target tracking in radar sensor network," *IEEE Syst. J.*, vol. 15, no. 1, pp. 771–779, Mar. 2021, doi: 10.1109/jsyst.2020.2990409.

[11] J. Yan, W. Pu, H. Liu, S. Zhou, and Z. Bao, "Cooperative target assignment and dwell allocation for multiple target tracking in phased array radar network," *Signal Process.*, vol. 141, pp. 74–83, Dec. 2017, doi: 10.1016/j.sigpro.2017.05.014.

[12] X. Liu, Q. Zhang, Y. Chen, L. Su, and Y. Chen, "Task allocation optimization for multi-target ISAR imaging in radar network," *IEEE Sensors J.*, vol. 18, no. 1, pp. 122–132, Jan. 2018, doi: 10.1109/jsen.2017.2771804.

[13] X. Xu, L. Dou, and B. Xin, "Generalized target-radar assignment using genetic algorithm," in *Proc. Chinese Automat. Congress (CAC)*, 2017, pp. 5287–5292, doi: 10.1109/CAC.2017.8243720.

[14] X. Li, D. Zhou, Z. Yang, Q. Pan, and J. Huang, "A Novel genetic algorithm for the synthetical sensor-weapon-target assignment problem," *Appl. Sci.*, vol. 9, no. 18, p. 3803, 2019, doi: 10.3390/app9183803.

[15] Y. Zhao, Y. Chen, Z. Zhen, and J. Jiang, "Multi-weapon multi-target assignment based on hybrid genetic algorithm in uncertain environment," *Int. J. Adv. Robot. Syst.*, vol. 17, no. 2, 2020, Art. no. 172988142090592, doi: 10.1177/1729881420905922.

[16] S. Hosseini Andargoli and J. Malekzadeh, "LPI radar network optimization based on geometrical measurement fusion," *Optim. Eng.*, vol. 20, no. 1, pp. 119–150, 2018, doi: 10.1007/s11081-018-9401-x.

[17] J. Kim, D. Cho, W. Lee, S. Park, and H. Choi, "Optimal target assignment with seamless handovers for networked radars," *Sensors*, vol. 19, no. 20, p. 4555, 2019, doi: 10.3390/s19204555.

[18] M. Hocaoglu, "Weapon target assignment optimization for land based multi-air defense systems: A goal programming approach," *Comput. Ind. Eng.*, vol. 128, pp. 681–689, Feb. 2019, doi: 10.1016/j.cie.2019.01.015.

[19] M. Shin, D. Lee, and H. Choi, "Weapon-target assignment problem with interference constraints using mixed-integer linear programming," 2019, *arXiv:1911.12567*, doi: 10.48550/arXiv.1911.12567.

[20] H. Zhang, W. Liu, Z. Zhang, W. Lu, and J. Xie, "Joint target assignment and power allocation in multiple distributed MIMO radar networks," *IEEE Syst. J.*, vol. 5, no. 1, pp. 694–704, Mar. 2021, doi: 10.1109/JSYST.2020.2986020.

[21] L. Wang, L. Jiao, J. Li, J. Gedeon, and M. Mühlhäuser, "MOERA: Mobility-agnostic online resource allocation for edge computing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 8, pp. 1843–1856, Aug. 2019, doi: 10.1109/TMC.2018.2867520.

[22] S. Tuli, S. Ilager, K. Ramamohanarao, and R. Buyya, "Dynamic scheduling for stochastic edge-cloud computing environments using A3C learning and residual recurrent neural networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 3, pp. 940–954, Mar. 2022, doi: 10.1109/TMC.2020.3017079.
[23] F. Meng, K. Tian, and C. Wu, "Deep reinforcement learning-based radar network target assignment," *IEEE Sensors J.*, vol. 21, no. 14, pp. 16315–16327, Jul. 2021, doi: 10.1109/JSEN.2021.3074826.
[24] P. Pace, *Detecting and Classifying Low Probability of Intercept Radar*. New York, NY, USa: Artech House, 2009.
[25] H. Amiriara, F. Ashtiani, M. Mirmohseni, and M. Nasiri-Kenari, "IRS-user association in IRS-aided MISO wireless networks: Convex optimization and machine learning approaches," *IEEE Trans. Veh. Technol.*, early access, Jun. 2, 2023, doi: 10.1109/TVT.2023.3282272.
[26] M. A. Richards, J. A. Scheer, and W. A. Holm, *Principles of Modern Radar-Volume I: Basic Principles*. Raleigh, NC, USA: SciTech Publ., 2010.
[27] C. Wolff and G. Neubrandenburg. "Signal-to-noise ratio–radar tutorial." Radartutorial.eu. 2022. Accessed: Jul. 4, 2022. [Online]. Available: https://www.radartutorial.eu/18.explanations/ex09.en.html
[28] S. Dang, J. Tang, J. Li, M. Wen, S. Abdullah, and C. Li, "Combined relay selection enabled by supervised machine learning," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3938–3943, Apr. 2021, doi: 10.1109/TVT.2021.3065074.
[29] S. Hosseini Andargoli and J. Malekzadeh, "LPI optimization framework for search radar network based on information fusion," *Aerosp. Sci. Technol.*, vol. 67, pp. 206–214, Aug. 2017, doi: 10.1016/j.ast.2017.04.004.
[30] M. Lee, Y. Xiong, G. Yu, and G. Li, "Deep neural networks for linear sum assignment problems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 6, pp. 962–965, Dec. 2018, doi: 10.1109/lwc.2018.2843359.
[31] H. Amiriara, M. R. Zahabi, and V. Meghdadi, "Power-location optimization for cooperative nomadic relay systems using machine learning approach," *IEEE Access*, vol. 9, pp. 74246–74257, 2021, doi: 10.1109/ACCESS.2021.3079171.
[32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge (EE. UU.), MA, USA: MIT Press, 2016.
[33] C. Schaffer, "Selecting a classification method by cross-validation," *Mach. learn.*, vol. 13, pp. 135–143, Oct. 1993.