

Adaptive Resource Optimization for IoT-Enabled Disaster-Resilient Non-Terrestrial Networks using Deep Reinforcement Learning

Fathe JERIBI^{1,2}, R. John MARTIN¹

¹Dept. of Computer Science, College of Engineering and Computer Science, Jazan University, Jazan 45142, Saudi Arabia

²Engineering and Technology Research Center, Jazan University, P.O. Box 114, Jazan 82817, Saudi Arabia

fjeribi@jazanu.edu.sa, jmartin@jazanu.edu.sa

Submitted January 17, 2025 / Accepted April 2, 2025 / Online first May 6, 2025

Abstract. *The increasing deployment of IoT devices across sectors such as agriculture, transportation, and infrastructure has intensified the need for connectivity in remote and non-terrestrial regions. Non-terrestrial networks (NTNs), which include maritime and space platforms, face unique challenges for IoT connectivity, including mobility and weather conditions, which are critical for maintaining quality of service (QoS), especially in disaster management scenarios. The dynamic nature of NTNs makes static resource allocation insufficient, necessitating adaptive strategies to address varying demands and environmental conditions during disaster management. In this paper, we propose an adaptive resource optimization approach for disaster-resilient IoT connectivity in non-terrestrial environments using deep reinforcement learning. Initially, we design the chaotic plum tree (CPT) algorithm for clustering IoT nodes to maximize the number of satisfactory connections, ensuring all nodes meet sustainability requirements in terms of delay and QoS. Additionally, unmanned aerial vehicles (UAVs) are used to provide optimal coverage for IoT nodes in disaster areas, with coverage optimization achieved through the non-linear smooth optimization (NLSO) algorithm. Furthermore, we develop the multi-variable double deep reinforcement learning (MVD-DRL) framework for resource management, which addresses congestion and transmission power of IoT nodes to enhance network performance by maximize successful connections. Simulation results demonstrate that our MVD-DRL approach reduces the average end-to-end delay by 50.24% compared to existing approaches. It also achieves a throughput improvement of 13.01%, an energy consumption efficiency of 68.71%, and an efficiency in the number of successful connections of 17.51% compared to current approaches.*

Keywords

Internet of Things (IoT), disaster management, resource optimization, deep reinforcement learning, non-terrestrial network

1. Introduction

The upcoming sixth-generation (6G) communication technologies are poised to deliver high-speed connectivity worldwide, boasting ultra-high data rates, minimal latency, and robust information security [1]. Integrated terrestrial and non-terrestrial networks combine ground-based infrastructures like fiber-optic cables and cell towers with satellite and drone systems, ensuring seamless and reliable connectivity [2]. This fusion enhances data transmission efficiency by using both terrestrial and non-terrestrial technologies for optimized performance [3], [4]. Non-terrestrial networks (NTN) have gained significance due to technique advancements and incorporation into 3GPP standards [5]. However, with the rising demand for machine-type communication, including internet of things (IoT) and machine-type communication (mMTC) strategies, wireless cellular services now cater to broader range of applications beyond traditional smartphone users [6]. Unmanned autonomous intelligent systems (UAISs) like self-driving automobiles, multi-robot schemes, unmanned aerial vehicles (UAVs), and unmanned vehicles require ubiquitous wireless [7], [8]. With optimal solution, airborne communications [9], [10] are key component of the 6G and revolutionize wireless connectivity, especially in disaster relief efforts. During major calamities like earthquakes and floods, traditional terrestrial networks frequently prove insufficient, mostly because of infrastructure damage, increased communication demand overload and physical impediments [11–14]. In non-terrestrial environments, ensuring disaster-resilient IoT connectivity is complex task due to stringent quality of service (QoS) requirements such as energy efficiency, low latency, and reliability, alongside the challenges of managing resource optimization for disaster relief efforts [15]. However, machine learning (ML) [16], particularly deep reinforcement learning (DRL) [17], [18], has emerged as a critical enabler for managing disaster relief operations in real-time, especially in highly complex scenarios. DRL adapts to changing environmental conditions and QoS demands by learning optimal decision-making policies through continuous interaction with the

environment [19]. In disaster relief contexts, DRL plays a pivotal role in optimizing IoT connectivity by enabling adaptive resource allocation, dynamic network configuration, fault tolerance, and resilience [20].

1.1 Related Works

The failure mode and effect analysis (FMEA) model using multi-objective optimization by ratio analysis (MULTIMOORA) approach is utilized to systematically assess and prioritize potential failure modes in industrial systems, enhancing decision-making for risk mitigation [21]. For dependable data-driven real-time industrial IoT application, AI-driven network and processing framework (AIDA) is employed [22]. The MEC server is developed using a digital twin (DT)-enabled multi-access edge computing architecture, which facilitates real-time monitoring and optimization of edge computing resources to improve system performance [23].

Block chain-based encoded IoT data is the foundation of the privacy-preserving support vector machine (PP-SVM) [24], [25], enabling secure data sharing while addressing real-world constraints and idealized assumptions through enhanced privacy mechanisms. Spatial spectrum reuse limits are widely followed while carefully allocating bandwidth to individual links according to their particular demands using linear programming framework [26]. The multi-agent dueling double deep Q network (MA3DQN) with centralized training and distributed execution employing centralized training and distributed execution, offers scalable and effective learning-based solutions for complex multi-agent environments [27].

Reconfigurable intelligent surfaces (RIS) on UAVs are incorporated into paradigm for hybrid optical/RF-based HAP-enabled ISATRNs [28] to optimize access in densely

populated locations. For scenarios involving high user mobility, an adaptive 5G New Radio (5G-NR) solution is proposed, enabling seamless connectivity for User Equipment (UE) in dynamic environments [29]. The front-loaded DM-RS-based channel estimation method can maximize spectral efficiency up to 5.05 bps/Hz under 64-QAM and enhance link reliability [30]. Table 1 summarizes the key aspects of the benchmarked works and highlights the identified research gaps.

1.2 Problem Description

The rapid proliferation of IoT devices across sectors such as agriculture, transportation, and infrastructure has significantly increased the demand for reliable connectivity, particularly in remote and non-terrestrial environments [31–40]. Traditional terrestrial networks often fail to deliver uninterrupted coverage in these regions due to structural limitations and geographical constraints. Non-terrestrial networks (NTNs), including maritime, aerial, and space-based platforms, have emerged as promising alternatives to address this challenge. However, NTNs present unique difficulties such as platform mobility (e.g., satellites and UAVs), dynamic environmental conditions, and limited energy resources, all of which hinder network reliability and the ability to meet Quality of Service (QoS) requirements—especially in critical disaster response scenarios. Conventional static or convex optimization methods are inadequate in addressing the dynamic and multi-dimensional nature of resource allocation in such environments. Therefore, there is a need for intelligent and adaptive optimization strategies that can dynamically manage IoT clustering, energy efficiency, transmission power, and network congestion to ensure robust and resilient connectivity. In response to this, the present study formulates the problem as a multi-objective resource optimization task

	Ref.	Methodology	Technique used	Findings	Research gaps
Review on Industrial IoT	Aboshosha et al. 2023	Data-driven predictive maintenance relying	Fuzzy and artificial neural network	They achieved 78% detection rate of faults in their design.	Difficult to cover maximum coverage range which limits the QoS
	Alnakhli 2024	Holistic AI-driven networking	Support vector machine (SVM)	The network utilization is improved as 89%	The complex network model affects the performance of in-resource management
	Birabwa et al. 2023	NOMA-enabled industrial IoT network	Digital twin and DRL	They optimize offloading and resource allocation	Failed to achieve optimal data forwarding process which limits the QoS requirements
	Chahed et al. 2023	Privacy preserving framework	Grasshopper–Black Hole Optimization	Ensure the security issues by strong key generation	Falling into local minimum of objective function particularly in complex problems
	Jeremiah et al. 2024	Privacy preserved data sharing	Blockchain and SVM	Maximizes the data security in the complex environment	Random function generation is maximum problem of blockchain model
Review on 6G NTN	Kumar et al. 2023	Optimal spectrum efficiency	Non-linear matrix function	Reduces delay tolerant traffic close to 100%	The matrix function is complex which creates the redundancy problem
	Li et al. 2023	User association and resource allocation	Multi-agent dueling double deep Q network	Maintains handoff probability of zero	Long propagation delay because of long training process of DDQN
	Pandey et al. 2023	Optimized UAV trajectory	LSTM-DDQN framework	It provides high scalability and optimal runtime	LSTM model is limited by control value selection, still this problem is not solved
	Pawase & Chang [29]	DM-RS based channel estimation	DM-RS symbol pattern, TDL channel model	Achieves the maximum spectral efficiency 5.05 bps/Hz	Non-modified design is not suitable for dynamical environment because of mobility
	Yin et al. [30]	Dynamic routing and resource allocation	Cross-segment optimization	Solve the multi objective optimization problem.	Dynamical channel estimation model limits the reduction in number of successful connections

Tab. 1. Research gap summary from existing works on industrial IoT and 6G NTN.

in disaster-resilient IoT-enabled NTN. The proposed solution integrates a CPT algorithm for optimal clustering of IoT nodes to meet delay and QoS constraints, a Non-Linear Smooth Optimization (NLSO) algorithm for energy-efficient UAV-based coverage, and a Multi-Variable Double Deep Reinforcement Learning (MVD-DRL) framework for adaptive resource allocation. This approach ensures seamless IoT communication in dynamic and challenging environments, significantly improving network performance, energy efficiency, and the reliability of connections during disaster scenarios.

1.3 Research Contributions

To effectively address the IoT connectivity challenges in non-terrestrial network (NTN) environments, the proposed methodology adopts an adaptive resource optimization framework based on deep reinforcement learning (DRL). The approach holistically integrates three interconnected components to enhance network performance in disaster-resilient scenarios.

1. Firstly, the CPT algorithm is employed for clustering IoT nodes, aiming to maximize the number of satisfactory connections while ensuring sustainability in terms of delay and quality of service (QoS). This clustering mechanism enables efficient grouping of nodes, reducing communication overhead and improving connectivity.
2. Secondly, unmanned aerial vehicles (UAVs) are strategically deployed to ensure optimal coverage for IoT nodes in disaster-affected regions. The Non-Linear Smooth Optimization (NLSO) algorithm is utilized to optimize UAV positioning, ensuring reliable communication support in dynamic and challenging environments.
3. A multi-variable double deep reinforcement learning (MVD-DRL) framework is developed for resource management. By dynamically adjusting resource allocation based on network conditions, the MVD-DRL model effectively mitigates congestion, controls transmission power, and maximizes the number of successful connections. This integrated methodology ensures seamless, energy-efficient, and resilient IoT communication in NTN environments, especially during disaster recovery operations.

The remaining sections of the paper are organized as follows. The methodology of the proposed work is explained in Sec. 2 with the cluster formation using CPT algorithm in Sec. 2.2, UAV deployment optimization using NLSO algorithm in Sec. 2.3 and resource management using MVD-DRL model in Sec. 2.4. In Sec. 3, the simulation setup and their scenarios are explained. The results and comparative analysis are given in Sec. 4. Finally, the paper concludes in Sec. 5.

2. Materials and Methods

2.1 System Model of the Proposed Approach

The rapid proliferation of IoT devices across various sectors underscores the urgent need for reliable connectivity, especially in remote and non-terrestrial environments. These regions, characterized by unique challenges such as mobility, weather variability, and infrastructure limitations, often struggle to maintain QoS, particularly during disaster scenarios. Traditional static resource allocation methods fall short in addressing the dynamic and unpredictable nature of Non-Terrestrial Networks (NTNs). This research is motivated by the pressing requirement to develop adaptive resource optimization strategies that can effectively manage IoT connectivity in disaster-resilient applications. By leveraging advanced techniques such as deep reinforcement learning and novel algorithms like the CPT and non-linear smooth optimization (NLSO), this study aims to enhance network performance, improve connection reliability, and optimize resource allocation to ensure robust communication in critical situations.

Figure 1 shows the network model for disaster-resilient NTN with IoT sensor nodes, using clustering, optimal channel state information (CSI) and resource management. In this model, the NTN supports IoT nodes dispersed throughout the disaster area in an uplink scenario, involving UAVs and monitoring IoT nodes as users. NTN comprises two types of components: aerial base stations (ABS), which provide links and mobile edge computing for the nodes, and high altitude platform stations (HAPS), which relay messages between ABSs in the disaster area and external entities. Let $\mathbf{Z} = \{1, \dots, z, \dots, Z\}$ denote the set of IoT nodes where Z shows the total number of nodes. Furthermore, $\mathbf{X} = \{1, \dots, x, \dots, X\}$ is considered as the set of ABSs where X represents the total number of ABSs and denotes HAPS. In particular, the total operating time is 100 s, which is divided into 2000 slots. The time slot is represented by t , and its length is 50 ms, denoted by ζ . It is noteworthy that ζ is the length of the time slot, and δ is the length of the traveling part. For CSI gathering, IoT nodes send known pilot signals to UAVs, which use these signals to estimate the channel response and calculate CSI. IoT nodes collect the environmental information, request connection to the ABS, and send their data. In the proposed approach, we collect information from IoT and UAV nodes. Utilizing the data from IoT sensor nodes, the CPT is used for clustering to maximize satisfactory connections. UAVs are deployed to provide optimal coverage for IoT nodes in disaster areas, optimized using the NLSO algorithm.

2.2 Cluster Formation

Cluster formation involves organizing IoT networks into groups called clusters, each managed by a leader. To

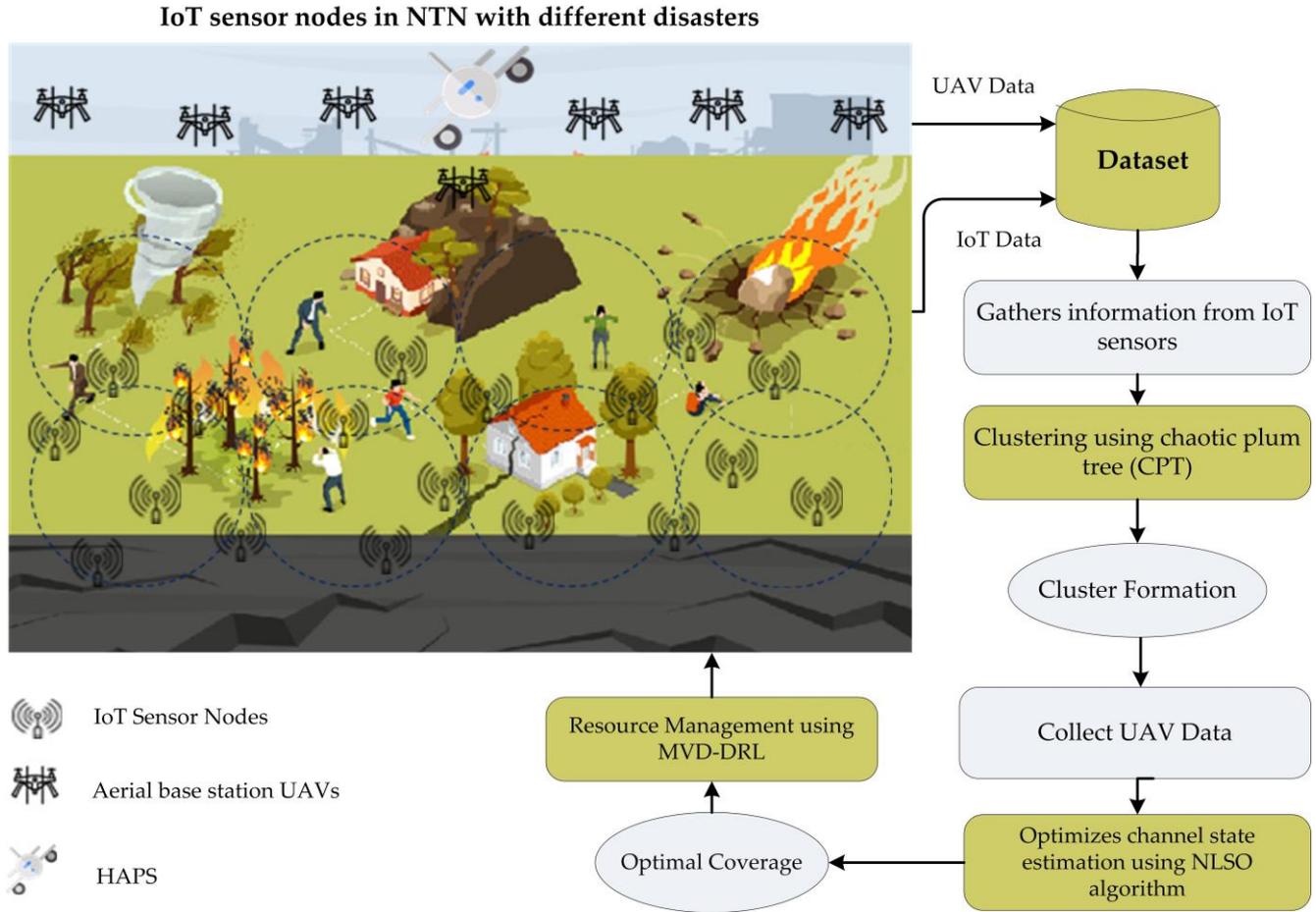


Fig. 1. The conceptual framework of the proposed work.

create these clusters, the CPT algorithm is used, which helps find the best connections between IoT nodes. This algorithm ensures that all nodes operate within the required time frame and meet the expected quality of service. Since the CPT algorithm uses chaos-based principles, it can thoroughly explore all possible clustering options, increasing the chances of finding the most efficient arrangement.

The CPT algorithm is an innovative clustering technique inspired by the natural growth patterns and characteristics of plum trees, which effectively structures IoT networks into clusters with designated leaders. By using chaotic dynamics, the CPT algorithm enhances the clustering process, making it particularly suitable for complex and dynamic environments like the Internet of Things (IoT). The incorporation of chaotic systems allows for the generation of pseudo-random sequences, enabling a thorough exploration of potential clustering configurations. The iterative nature of CPT facilitates dynamic cluster formation, where IoT nodes are grouped based on specific criteria such as proximity or connectivity. Each iteration refines the clusters, accommodating the addition or removal of nodes in the network. Furthermore, each cluster features a leader node responsible for coordinating activities within the cluster, optimizing resource management, data aggregation, and communication among nodes, thereby enhancing overall network efficiency. The CPT algorithm

is also designed to ensure that all clustered IoT nodes meet specific sustainability time frames and QoS requirements, which is crucial for maintaining consistent performance and longevity in IoT applications. The CPT algorithm is a chaotic process used to optimize flower positions by selecting random integers between the maximum and minimum solutions. It identifies the globally optimal plum location. Each flower, B , is initialized in a C -dimensional search space, with values randomly selected within the given solution range $[Xr_{Min}, Xr_{Max}]$.

$$\mathbf{Flowers} = \begin{bmatrix} f_{1,1}^0 & \dots & f_{1,C}^0 \\ \dots & \dots & \dots \\ f_{B,1}^0 & \dots & f_{B,C}^0 \end{bmatrix} \quad (1)$$

where B plums are primed with the price of flower:

$$\mathbf{Plums} = \begin{bmatrix} X_{1,1}^0 & \dots & X_{1,C}^0 \\ \dots & \dots & \dots \\ X_{B,1}^0 & \dots & X_{B,C}^0 \end{bmatrix} = \mathbf{Flowers}. \quad (2)$$

The fitness value of each IoT sensor node is computed using the objective function (OF), and the global best IoT node X_h^{iter-1} is updated based on the best fitness value at its location. The top-ranked IoT node location is further refined based on the best fitness value, f_h^{iter-1} , while the

second-best fitness value guides the update of another optimal location. A random value R is selected from the range $[0, 1]$. If R is greater than or equal to the fitness selection (FS) threshold, the position of the IoT node is updated as follows:

$$f_h^{iter} = f_h^{iter-1} + 2R_1(X_{Ripe} - f_h^{iter-1}) + 2R_2(X_{unRipe} - f_h^{iter-1}) \quad (3)$$

where R_1 and R_2 represent a random number within $[0, 1]$, and X_{Ripe} and X_{unRipe} show the ripe and the unripe position, similarly. The chaotic rule is based on deviation σ^2 and mean as follows:

$$\sigma^2 = \begin{cases} 1, & \text{if } Of(X_h^{iter-1}) < Of(X_{Ripe}) \\ \frac{OX(X_{Ripe}) - OX(X_h^{iter-1})}{|Of(X_h^{iter-1})| + E}, & \text{otherwise} \end{cases} \quad (4)$$

where E acts as a safeguard to prevent division by zero. $OX(\cdot)$ represents the optimization function that evaluates the signal strength of the IoT node, while $Of(\cdot)$ denotes the fitness function used to assess the performance of a given IoT node in the network. The selected IoT node is altered to the $[X_{rMin}, X_{rMax}]$ intermission so that when $F_{hg}^{iter} = X_{rMin}$, then $F_{hg}^{iter} > X_{rMin}$, and if $F_{hg}^{iter} > X_{rMin}$, then $F_{hg}^{iter} = X_{rMax}$, where $g = 1$.

$$X_h^{iter} = \begin{cases} f_h^{iter}, & \text{if } Of(f_h^{iter}) < Of(f_h^{iter-1}) \\ X_h^{iter-1}, & \text{otherwise} \end{cases} \quad (5)$$

The global best IoT node X_h^{iter} value is updated based on the location of the IoT node with the best fitness value, according to the objective function. Finally, return the value:

$$S_{All} = S_1 + S_2 + S_3 \quad (6)$$

where S_1 denotes to the optimal best location for the IoT nodes and the HAPS. This represents the delay of IoT in the data center (C) in relation to the energy efficiency of the NTN network. The objective function is determined by the regular visit (h) of the IoT nodes and is represented as follows:

$$\text{MinF}(l) = \sum_{h=1}^{b-1} c_{h,h+1} \quad (7)$$

where T signifies the sequence of protuberances. CPT algorithm in cluster formation leverages chaotic maps to ensure efficient clustering of IoT nodes in NTNs, thereby enhancing overall network performance and resilience, especially in disaster-prone areas. The working process of cluster formation using CPT is given in Algorithm 1.

Input : Amount of IoT nodes, maximum iteration and threshold condition
Output : Cluster formation
1. Begin
2. Initialize the step sizes of the population
3. Define C -dimensional search space with values ranges within $[X_{rMin}, X_{rMax}]$

4.	For $L = 1, 2, 3, \dots, b-1$ do
5.	Calculate the plum's fitness and update the fitness X_h^{iter} To compute the optimal position of flower:
6.	$f_h^{iter} = f_h^{iter-1} + \text{Random}(X_{rMin}, X_{rMax}) \times (X_h^{iter-1} - f_h^{iter-1})$ Define optimal location of flower using chaotic map:
7.	$f_h^{iter} = X_h^{iter-1} \times (1 + B(0, \sigma^2))$
8.	Fix the search region of plum tree: $S_{All} = S_1 + S_2 + S_3$ Compute the optimal fitness for maximum search range:
7.	$S_{All} = S_1 + \sum_1^b (S_D + S_T + S_{hg}) + S_3$
	Define maximum power consumption of IoT nodes:
8.	$\text{MinF}(l) = \sum_{h=1}^{b-1} c_{h,h+1}$
9.	End for
11.	Return

Algorithm 1. Cluster formation using CPT.

2.3 Optimizing CSI Estimation

Channel state information (CSI) estimation is critical aspect of wireless communication systems, providing detailed insights into the properties of the communication channel, such as channel gain, phase shift, delay spread, and Doppler shift. Accurate CSI estimation is essential because it allows the system to adapt to changing channel conditions. The non-linear smooth optimization (NLSO) algorithm is used to optimize CSI estimation of disaster-resilient IoT connectivity. It determines the optimal positions for UAVs to provide maximum coverage for IoT nodes within a disaster area. IoT nodes send known pilot signals to UAVs, which use these signals to estimate the channel response and compute the CSI. NLSO operates by formulating the optimization problem in terms of a smooth objective function that captures the essential characteristics of the communication system. This function typically represents metrics such as signal-to-noise ratio (SNR), throughput, or coverage area, and it is designed to be differentiable, utilize gradient-based optimization techniques. A key advantage of the NLSO algorithm is its ability to handle non-linear constraints, which are often present in real-world scenarios. For instance, in the optimization of unmanned aerial vehicles (UAVs) for disaster-resilient IoT connectivity, the algorithm can account for factors such as altitude limits, battery life, and the need for line-of-sight communication between UAVs and IoT nodes. This flexibility enables the algorithm to provide a robust solution that is both practical and effective. In NLSO algorithm, the initial fitness describes the j_k^B instead of multiplying the direction by a positive definite matrix

$$\text{Min } F(\mathbf{p}_k) + F^o(\mathbf{p}_k; \mathbf{c}) + \frac{1}{2} \mathbf{c}^T \mathbf{N}_k \mathbf{c} \quad (8)$$

where \mathbf{N}_k is a confident certain matrix. Let \mathbf{N}_k be constructive definite and assume $\mathbf{N}_k \mathbf{c}$ be the explanation of Problem. The objective function $\phi(\mathbf{c}) = F^o(\mathbf{p}_k; \mathbf{c}) + \frac{1}{2} \mathbf{c}^T \mathbf{N}_k \mathbf{c}$ is

strictly convex. An exclusive minimized \mathbf{c}^* is computed as follows:

$$\partial F^o(\mathbf{p}_K; \mathbf{c}^*) \subseteq \left\{ \xi \in \partial F(\mathbf{p}_K) : \xi^T \mathbf{c}^* = F^o(\mathbf{p}_K; \mathbf{c}^*) \right\}. \quad (9)$$

Now, the maximal optimal fitness $F^o(\mathbf{p}_K; -\mathbf{N}_K^{-1} \mathbf{J}_K^B)$ is computed as follows:

$$F^o(\mathbf{p}_K; -\mathbf{N}_K^{-1} \mathbf{J}_K^B) = \text{Max}_{\xi \in \partial F(\mathbf{p}_K)} \xi^T (-\mathbf{N}_K^{-1} \mathbf{J}_K^B) = -(\mathbf{J}_K^B)^T \mathbf{N}_K^{-1} \mathbf{J}_K^B \quad (10)$$

where \mathbf{p}_K is sufficient for the fitness function $F^o(\mathbf{p})$, utilizing an efficient one-way quadratic sampling function.

$$F^o(\mathbf{p}) = \text{Max}_{g=1, \dots, x} \left\{ y_g(\mathbf{p}) \right\} \quad (11)$$

Then, we compute the multi-objective sampling function as follows:

$$y_g(\mathbf{p}) = F(\mathbf{q}_g) + \mathbf{j}_g^T (\mathbf{p} - \mathbf{q}_g) + \frac{1}{2} (\mathbf{p} - \mathbf{q}_g)^T \mathbf{I}_g (\mathbf{p} - \mathbf{q}_g). \quad (12)$$

We assume that the sample structure of F contains the maximum number of features, i.e., the number of features p is unknown and must be estimated through a trial-and-error process:

$$\partial F^o(\mathbf{p}) = \partial \text{Max}_{g=1, \dots, x} \left\{ y_g(\mathbf{p}) \right\} \subseteq \left\{ \mathbf{j}_g + \mathbf{I}_g (\mathbf{p} - \mathbf{q}_g), g = 1, \dots, x \right\} \quad (13)$$

$$= D(\mathbf{p}).$$

Follow the major rule $F(\mathbf{p}) \approx F^o(\mathbf{p})$ and the objective function is stated as follows:

$$F^o(\mathbf{p}; \mathbf{c}) = \text{Max}_{t \in \partial F(\mathbf{p})} \mathbf{c}^T \mathbf{t} \quad (14)$$

$$\approx \text{Max}_{t \in \partial F^o(\mathbf{p})} \mathbf{c}^T \mathbf{t} \leq \text{Max}_{t \in D(\mathbf{p})} \mathbf{c}^T \mathbf{t} = \mathbf{c}^T (\mathbf{j}_h + \mathbf{I}_h (\mathbf{p} - \mathbf{q}_h))$$

In the real case, $D(\mathbf{p})$ is the convex hull of a given set of generator vectors $\mathbf{V}_g, g = 1, \dots, x$. Let \mathbf{p}_K be the current iteration of the algorithm. The region of maximal objective function is computed by the unit vector $(\mathbf{c}_h = \pm \mathbf{E}_h)$.

$$t_h = \frac{F(\mathbf{p}_K + \alpha_h \mathbf{c}_h) - F(\mathbf{p}_K)}{\alpha_h} \approx F^o(\mathbf{p}_K; \mathbf{c}_h). \quad (15)$$

For $h = 1$, update the objective function as follows:

$$F^o(\mathbf{p}_K; \mathbf{c}_h) \approx \mathbf{c}_h^T \mathbf{V}_{g_h}, \quad \text{for some } g_h \in \{1, 2, \dots, x\}. \quad (16)$$

Using the normative function of t_h , we solve the optimization problem (17) where R denotes the total number of reference directions considered in the optimization:

$$\text{Min}_{\hat{\mathbf{V}}_1, \dots, \hat{\mathbf{V}}_x} \sum_{h=1}^R \text{Min}_{g=1, \dots, x} \left\{ \left(\mathbf{c}_h^T \hat{\mathbf{V}}_g - t_h \right)^2 \right\}. \quad (17)$$

The initial set of x temporal generators is saved. Each pair (\mathbf{c}_h^T, t_h) is assigned to a generator that provides a better approximation t_h . $\partial F(\mathbf{p}_K)$ is the direction estimate can be calculated by solving the problem of where is the approximation $\text{conv}(\hat{\mathbf{V}}_1, \dots, \hat{\mathbf{V}}_x)$. Algorithm 2 describes the process of optimizing CSI estimation using NLSO.

Input : Initial positions, mobility constraints and energy consumption	
Output : Optimizes CSI estimation	
1.	Begin
2.	Initialize the step sizes of the population
3.	Define minimal fitness level with channel model: $\text{Min } F(\mathbf{p}_K) + F^o(\mathbf{p}_K; \mathbf{c}) + \frac{1}{2} \mathbf{c}^T \mathbf{N}_K \mathbf{c}$
4.	Fix the threshold level with conditional model: $0 \in \partial F^o(\mathbf{p}_K; \mathbf{c}^*) + \mathbf{N}_K \mathbf{c}^*$
5.	Compute optimal fitness function of F^o and update: $F^o(\mathbf{p}_K; -\mathbf{N}_K^{-1} \mathbf{J}_K^B) = \text{Max}_{\xi \in \partial F(\mathbf{p}_K)} \xi^T (-\mathbf{N}_K^{-1} \mathbf{J}_K^B) = -(\mathbf{J}_K^B)^T \mathbf{N}_K^{-1} \mathbf{J}_K^B$
6.	Find the quadratic sampling function: $F^o(\mathbf{p}) = \text{Max}_{g=1, \dots, x} \left\{ y_g(\mathbf{p}) \right\}$
7.	If unit vector = $(\mathbf{c}_h = \pm \mathbf{E}_h)$:
8.	Define the predetermined directions for failure as: $t_h = \frac{F(\mathbf{p}_K + \alpha_h \mathbf{c}_h) - F(\mathbf{p}_K)}{\alpha_h} \approx F^o(\mathbf{p}_K; \mathbf{c}_h)$
7.	Else, estimate the best guess of the t_h 's values, and solve:
8.	$\text{Min}_{\hat{\mathbf{V}}_1, \dots, \hat{\mathbf{V}}_x} \sum_{h=1}^R \text{Min}_{g=1, \dots, x} \left\{ \left(\mathbf{c}_h^T \hat{\mathbf{V}}_g - t_h \right)^2 \right\}$
9.	End if
10.	Find and return the best output value
11.	End

Algorithm 2. Optimizes CSI estimation using NLSO.

2.4 Resource Management

Resource management is used for ensuring efficient and reliable communication, especially in NTN during disaster scenarios. Effective resource management involves the strategic allocation and optimization of network resources is bandwidth, power levels, and transmission channels to maintain optimal network performance and meet QoS requirements. The multi-variable double deep reinforcement learning (MVD-DRL) model is used to address these complex challenges. In this context, the DRL framework involves an agent, which is the learning model used for making decisions regarding resource allocation, and the environment, which includes the IoT network, UAVs, ABSs, and HAPSs. Within the MVD-DRL framework, the agent represents the learning model that interacts with the environment to make resource allocation decisions. The environment encompasses all elements of the NTN, including IoT nodes and communication platforms like UAVs and ABSs. The agent receives observations about the current state of the network and takes actions that impact resource allocation. The feedback from the environment, in terms of rewards or penalties based on the performance of its actions, guides the agent's learning process. During the training phase, the MVD-DRL model explores various resource allocation strategies, receiving rewards based on their outcomes (Reward_1, Reward_2 ... Reward_N). The resource management problem is solved using the Q-network function in the MVD-DRL model.

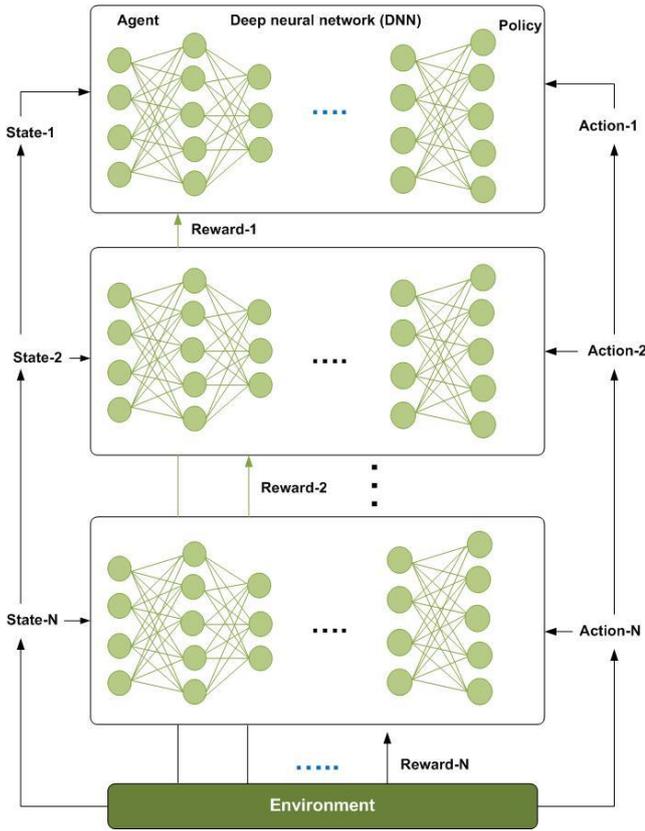


Fig. 2. Structure of MVD-DRL model for resource management in NTN during disaster scenarios.

As shown in Fig. 2, the target function of the MVD-DRL model is computed as follows:

$$\gamma_y^{DDQN} = E_{y+1} + \gamma W(A_{y+1}, \arg \max_s W(A_{y+1}, s; \theta_y^-)) \quad (18)$$

where E_{y+1} is the reward for the next state (State-1, State-2, ... State- N), y is the current W network's parameters, and t is the discount factor γ . The objective of the specialist is to track down an ideal technique to optimize the fitness.

$$H_y = E_{y+1} + \gamma E_{y+2} + K = \sum_{l=0}^{\infty} e^l E_{y+l+1}. \quad (19)$$

The H-index framework \mathbf{O}_{mm} is used, where H_y event represents the situation in which state u cannot transition to state h . The value is shown in the case where the transition from state u to h is possible.

$$\mathbf{O}_{mm} = \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1m} \\ s_{21} & s_{22} & \dots & s_{2m} \\ \dots & \dots & s_{uh} & \dots \\ s_{m1} & s_{m2} & \dots & s_{mm} \end{bmatrix}. \quad (20)$$

In the MVD-DRL model, we utilize the boundaries of brain organization, the ongoing W , the organizational structure, and the objective W network boundaries. We also use the ε -technique to select an action.

We apply the threshold condition to define the objective rule, which operates with a likelihood ε over a period, and claim is achieved with

$$\left\{ \varepsilon_{\min}, 1 - \frac{1 - \varepsilon_{\min} \times \text{step}}{\text{total step}} \right\} \quad (21)$$

as it diminishes throughout the training. The W network boundaries are updated by randomly drawing small batches of samples from the memory pool. The agent then uses DNN gradient descent with backpropagation to update the actions of the Q-network's current parameters (Action-1, Action-2, ..., Action- N).

$$t_h = \begin{cases} e_h & \text{if } a' \text{ is terminal} \\ e_h + \gamma W(a', \max(a', \theta); \theta^-) & \text{otherwise} \end{cases} \quad (22)$$

where t_h denotes the current state's reward. The next state is denoted by θ^- , and the current DNN parameters are represented by W . The loss function is defined as the mean squared error (MSE) of the disparity between the present network value W , and the objective is to minimize it.

$$K(\theta_u) = \text{MSE}(T_y^{DDQN} - W(A_y, S_y; \theta)). \quad (23)$$

The specialist updates the boundaries of the target network W after a predetermined number of steps. Once the model completes all its training phases, the agent enters the end state.

$$W(A_y, S_y, \theta) \leftarrow W(A_y, S_y, \theta) + \alpha \begin{bmatrix} E_{y+1} + \\ \gamma \max_{\alpha} \hat{W}(A_{y+1}, \alpha_y, \theta) - \\ W(A_y, S_y, \theta) \end{bmatrix} \quad (24)$$

where α is the learning rate. The immediate reward $\text{Reward}(a, s)$ obtained by taking action a is negative if the transition of swarm h to state s does not lead to a subtle host. The instantaneous return is used in the event that the state corresponding to s' is a sensitive host.

$$\text{Reward}(a, s) = \text{Reward}_{\text{init}} - \text{Cost}_{\text{vul}} \quad (25)$$

$$\text{Reward}_{\text{Int}} = \text{Score}_{\text{vul}} + \text{Value}_g \quad (26)$$

where $\text{Reward}_{\text{Int}}$ represents the predefined scores related to vulnerability exploitation and reputation acceleration in the exchange grid. The variable h denotes all hosts targeted by the specialist, and Cost_{vul} signifies the activity cost. Algorithm 3 describes the working process of resource management using the MVD-DRL model.

3. The Simulation Environment

The proposed approach is implemented on a network simulator (NS3), with the DRL model coded in Python language. The Network Simulator 3 (NS-3) plays a vital role

Input : Congestion levels, CSI, transmission power and bandwidth
Output : Learning parameter set (Rewards)
<pre> 1 Get mark system topology material. 2 Set the host worth (importance). 3 Apply MVD-DRL to build the boot tree. 4 Perform exploration of chart trails using a complexity-first procedure. Establish the transfer matrix: 5 $\mathbf{O}_{mm} = \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1m} \\ s_{21} & s_{22} & \dots & s_{2m} \\ \dots & \dots & s_{uh} & \dots \\ s_{m1} & s_{m2} & \dots & s_{mm} \end{bmatrix}$ 6 Initialize the neural networks. 7 For each episode = 1 to max_episode do 8 For $t = 1$ to T do Use ϵ-greedy to select action a' based on the w-value: 9 $\left\{ \begin{array}{l} \epsilon_{\min}, 1 - \frac{1 - \epsilon_{\min} \times \text{step}}{\text{total step}} \end{array} \right\}$ 10 Perform action a', observe next state s', reward r, and whether episode is done. 11 Add reward function. 12 Update current state $s = s'$ 13 if $D >$ batch-size then Construct target w-value: 14 $t_h = \begin{cases} e_h & \text{if } a' \text{ is terminal} \\ e_h + \gamma W(a', \max(a', \theta); \theta^-) & \text{otherwise} \end{cases}$ Perform ancestry step using loss function: 15 $K(\theta_u) = \text{MSE}(T_y^{DDQN} - W(A_y, S_y; \theta)).$ Every N steps, update θ using random module for rank computation: 16 $W(A_y, S_y, \theta) \leftarrow W(A_y, S_y, \theta) + \alpha \begin{bmatrix} E_{y+1} + \\ \gamma \max_a \hat{W}(A_{y+1}, \alpha_y, \theta) \\ -W(A_y, S_y, \theta) \end{bmatrix}$ 17 End for (T loop) 18 End for (episode loop) 19 End </pre>

Algorithm 3. Resource management using MVD-DRL model.

in IoT-based research by enabling realistic modeling of large-scale, heterogeneous networks. It supports a wide range of IoT-relevant protocols such as 6LoWPAN, RPL, and CoAP, and allows for detailed evaluation of performance metrics including latency, throughput, energy consumption, and reliability. NS-3 facilitates the simulation of dynamic environments and mobility patterns, making it ideal for applications like smart cities, healthcare monitoring, and vehicular IoT. Additionally, NS-3’s open-source nature and integration with machine learning frameworks like ns3-gym allow for adaptive and intelligent protocol development, ensuring reproducibility and validation of experimental results in a controlled simulation environment.

The NS-3, though primarily written in C++, supports Python bindings for scripting and experiment control, making it suitable for flexible and large-scale IoT simulations. To efficiently implement our methodology using Python, an ideal hardware setup includes a high-performance pro-

cessor such as an Intel Core i7 or AMD Ryzen 7, which ensures fast execution of simulation events. At least 16 GB of RAM is recommended to handle memory-intensive simulations involving numerous IoT nodes and protocol stacks. Solid-state drives (SSD) with a minimum of 100 GB free space are preferred for faster read/write access to logs and simulation data. While a GPU is not required for standard NS-3 operations, an NVIDIA GPU with CUDA support becomes beneficial when integrating machine learning models through frameworks like ns3-gym. Additionally, using a Python version between 3.6 and 3.10 within a virtual environment is recommended for managing dependencies and ensuring compatibility with NS-3 modules.

The results of the proposed MVD-DRL approach are compared with existing state-of-the-art (SOTA) methods, including the greedy radiant model (GRM), distributed soft actor-critic (DSAC), multi-agent deep deterministic policy gradient (MADDPG), and multi-agent recurrent deterministic policy gradient (MARDPG) [41]. We consider three scenarios to demonstrate the effectiveness of our MVD-DRL approach. In the first scenario, we validate the performance of different DRL models by varying the amount of subcarriers, IoT nodes, and end-to-end (E2E) delay. In the second scenario, we conduct a comparative analysis with respect to the objective function. Finally, we assess the fairness of the proposed and existing approaches in relation to the amount of UAVs.

Table 2 shows that the network covers 2000 meters by 2000 meters. This complicates the setting up of IoT devices and UAVs, which are needed to simulate real-world crises. Three ABS-UAVs have been launched to provide critical communication links and processing capacity to IoT sensor nodes in the disaster zone. A scenario can feature 10, 20, 30, 40, or 50 sensors however the number varies every simulation. In addition, the models consider 5, 10, 15, 20, and 25 120 kHz subcarriers to assess the system’s ability to handle varying data traffic and efficiency of the spectrum. Each IoT node requires 0.0004 CPU cycles per second, representing the processing power needed for data computation and transmission tasks. The packet arrival

Description	Value
Network Size	2000 m × 2000 m
Amount of ABS-UAV	3
Amount of IoT sensors	50, 100, 150, 200 and 250
Amount of subcarriers	5, 10, 15, 20 and 25
Bandwidth of subcarriers	120 kHz
Computational resource	0.0004 CPU cycle/sec.
Packet arrival rate	30 pps
Packet length	100 Kbits
Minimum uplink transmission power	100 mW
Maximum uplink transmission power	800 mW
Path loss	0.1 to 20 dB
E2E delay	100 ms
Minimum and maximum flying attitude	20 to 100 m
Maximum velocity of UAVs	4 m/s
Simulation time	240 s

Tab. 2. Description of simulation environment.

rate is set at 30 pps, simulating steady stream of data typical in IoT such as environmental monitoring or disaster response, with each packet having a length of 100 Kbits to provide a realistic data size for transmission. The minimum and maximum uplink transmission powers are set at 100 and 800 mW, respectively. Path loss is modeled between 0.1 to 20 dB, capturing range of signal degradation scenarios due to obstacles, distance, and environmental factors, which is crucial for testing the robustness of connectivity solutions. An E2E delay of 100 milliseconds is considered, serving as a critical metric for assessing the latency performance and the system's ability to meet stringent timing requirements of IoT applications, especially in emergency scenarios. The UAVs operate at altitudes ranging from 20 to 100 meters, allowing the simulation of different operational conditions and the impact of altitude on coverage and connectivity. Furthermore, the UAVs move at maximum speed of 4 m/s, enabling the simulation of dynamic repositioning and adaptive coverage strategies in response to changing network conditions. The total simulation time is set at 240 seconds, divided into 2000 slots, with each slot being 50 ms long.

Table 3 outlines the key parameters of the DRL model used for optimizing resource management in IoT connectivity within non-terrestrial environments. The model processes data in batches of 64 samples and is trained over 800,000 episodes to ensure comprehensive learning. A discount factor of 0.87 balances immediate and future rewards. The actor and critic learning rates are set at 0.00001 and 0.00005, respectively, to maintain stable and controlled parameter updates. ReLU activation functions in the hidden layers help with efficient learning, while the Tanh function in the output layer ensures bounded output values. The parameters collectively enable the DRL model to optimize resource allocation effectively in dynamic and complex IoT scenarios. Channel modeling is crucial for both terrestrial and non-terrestrial networks (NTNs) as it helps understand signal propagation and system performance. Accurate models account for factors like path loss, fading due to multipath propagation, and mobility effects, particularly in dynamic scenarios such as disaster management. Various modeling techniques exist, including empirical models, deterministic models based on geometric optics, and stochastic models using probabilistic approaches. In terrestrial networks, models focus on urban environments, optimizing antenna placement to counteract shadowing and multipath effects. For NTNs, including satellites

Description	Value
Batch size	64
Amount of episodes	800000
Discount factor	0.87
Actor-learning rate	0.00001
Critic-learning rate	0.00005
Activation function of hidden layer	ReLU
Activation function of output layer	Tanh

Tab. 3. Description of DRL model.

and UAVs, channel models must consider long-distance signal travel and atmospheric influences. Detailed channel modeling enhances simulation accuracy, improving system design and performance evaluation, particularly in critical situations where reliable communication is essential.

3.1 Real-World Applicability

The proposed concept of adaptive resource optimization for IoT-enabled disaster-resilient non-terrestrial networks (NTNs) has strong potential for real-world deployment, particularly in critical applications such as post-disaster response, remote monitoring, maritime communication, and agricultural surveillance in underserved regions. In practical scenarios, the integration of UAVs for coverage optimization and the use of deep reinforcement learning for dynamic resource allocation enables the system to adapt to fluctuating traffic loads, node failures, and environmental disruptions. However, the performance of the proposed approach in real-world transmission environments may be influenced by several limiting factors. These include signal attenuation caused by atmospheric conditions (e.g., rain fade or cloud cover in satellite links), latency due to the high mobility of UAVs or satellites, limited battery life of UAVs and IoT nodes, hardware constraints such as low processing power or memory, and spectrum interference in dynamically shared environments. Although these limitations have been theoretically considered in our framework, their real-time impact could vary depending on the deployment conditions. To bridge this gap, our future work includes implementing the proposed model in a real-time testbed involving physical UAVs and IoT platforms, which will allow us to evaluate its robustness, responsiveness, and scalability under operational constraints. This step will be crucial for validating the model's adaptability and effectiveness in ensuring sustainable, QoS-compliant connectivity in real-world NTN deployments.

4. Results and Discussions

In this section, we present the results to evaluate the performance of the proposed and existing resource management approaches. To validate the performance of the MVD-DRL approach, we use various simulation scenarios to jointly optimize resource management and IoT connectivity.

4.1 Comparative Analysis of DRL Models with Number of Successful Connection (NSC)

The results in Fig. 3 illustrate a comparative analysis of the number of successful connections (NSC) achieved by different state-of-art DRL models—GRM, DSAC, MADDPG, MARDPG, and our MVD-DRL approach—across varying amounts of subcarriers. Starting with 5 sub-

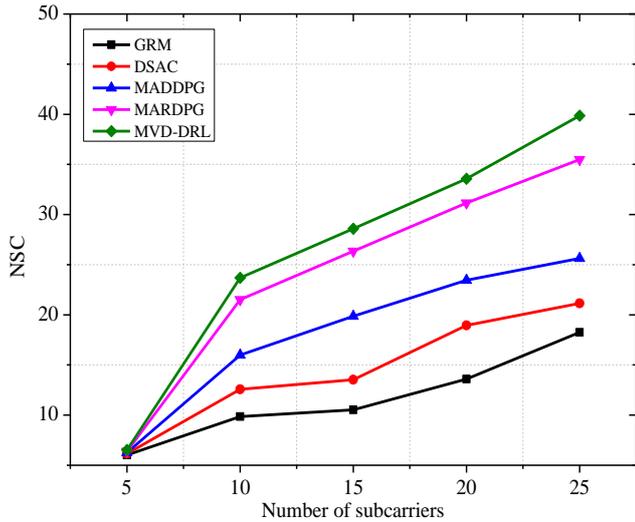


Fig. 3. NSC results comparison with varying amount of subcarriers.

carriers, GRM achieves an NSC of 6.022, which increases modestly to 6.523 with 25 subcarriers, reflecting an overall increase of 8.33%. In contrast, DSAC demonstrates a substantial improvement, starting at 9.856 NSC and reaching 23.685 NSC, marking a 22.312% increase. Similarly, MADDPG begins with NSC of 10.523 and rises to 28.578, resulting in an impressive 13.25% increase. MARDPG shows notable growth, from 13.587 NSC to 33.565, representing a 25.122% increase. The MVD-DRL model, however, stands out with its superior performance. It starts with an NSC of 18.245 at 5 subcarriers and reaches an NSC of 39.859 at 25 subcarriers, resulting in remarkable 18.102% increase.

The enhancement highlights the efficacy of the MVD-DRL model in optimizing resource management and enhancing IoT connectivity, particularly in disaster-prone and non-terrestrial environments. The consistent outperformance of MVD-DRL over other models at each increment of subcarriers underscores its capability to handle the dynamic demands and optimize network performance effectively.

The results presented in Fig. 4(a) depict the NSC achieved by different DRL models. Starting with GRM, the NSC rises from 3.445 with 5 sensors to 27.334 with 25 sensors, reflecting an enhancement of 12.321%. DSAC shows a similar trend, starting at 4.497 and increasing to 28.386, which is an enhancement of about 531.24%. MADDPG exhibits steady growth as well, with the NSC improving from 5.549 to 29.438, marking an enhancement of 12.58%. MARDPG also performs well, increasing from 6.601 to 30.49, which is an enhancement of 12.916%. The MVD-DRL model, however, shows the significant improvement. It starts with an NSC of 7.653 at 5 sensors and reaches 31.542 at 25 sensors, resulting in remarkable enhancement of 12.191%. The key advantage of MVD-DRL lies in its ability to manage resource allocation more effectively in dynamic, non-terrestrial environments, optimizing IoT connectivity with fewer sensors but achieving higher

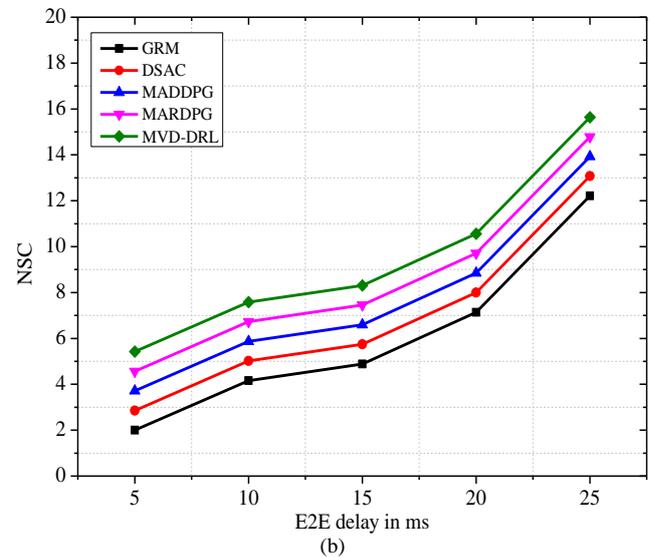
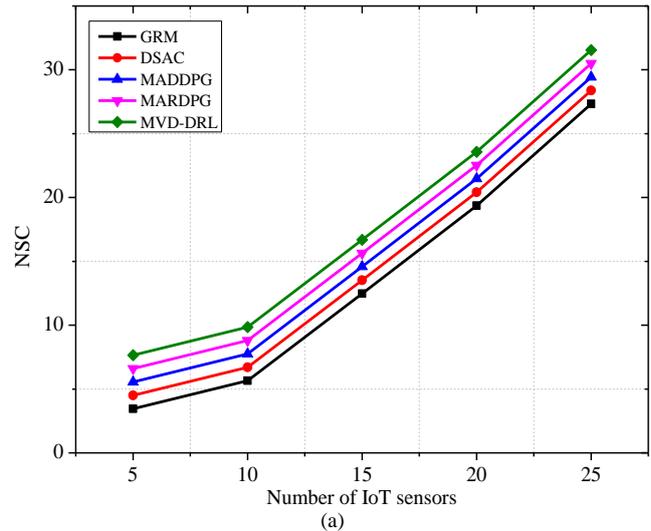


Fig. 4. NSC results comparison with (a) varying amount of IoT sensors; (b) with varying E2E delay in microseconds.

overall performance. This makes MVD-DRL more reliable for real-world applications where efficiency and scalability are critical, especially in scenarios requiring robust performance across various network sizes.

The results in Fig. 4(b) illustrate the NSC across different DRL models. Starting with GRM, the NSC increases from 1.999 to 12.218 as the E2E delay increases from 5 to 25 ms. This represents an impressive enhancement of 11.36%. DSAC shows an improvement from 2.855 to 13.074, marking an enhancement of 37.76%. MADDPG follows a similar trend, with the NSC rising from 3.711 to 13.93, which translates to an increase of 27.25%. MARDPG exhibits an increase as well, going from 4.567 to 14.786, reflecting an enhancement of 22.75%. The MVD-DRL model shows the highest performance increase, starting from an NSC of 5.423 at 5 ms E2E delay and reaching 15.642 at 25 ms E2E delay with the enhancement of 18.38%. While the MVD-DRL might seem lower compared to other models, it is important to note that MVD-

DRL achieves the highest absolute NSC values across all delay conditions, indicating its superior capability in maintaining high connectivity under varying network delays.

4.2 Ablation Study on the Proposed and Existing Approaches

Table 4 presents the results of an ablation study conducted to compare the performance of the proposed and existing approaches with respect to the amount of subcarriers. The study highlights the effectiveness of MVD-DRL in managing the dynamic allocation of subcarriers to optimize IoT connectivity in non-terrestrial networks. MVD-DRL consistently outperforms GRM across varying amounts of subcarriers in terms of NSC results. Starting with 5 subcarriers, MVD-DRL exhibits 13.3% enhancement over GRM. This advantage persists as the amount of subcarriers increases, with MVD-DRL achieves enhancement of 9.6%, 6%, 4% and 3.1% over 10 to 25 subcarriers, respectively. The model's superior scalability and efficiency in resource management enable the performance gains, positioning MVD-DRL as robust solution for optimizing network performance. Starting with 5 subcarriers, GRM exhibits the highest computational complexity at 21.572%, followed by DSAC, MADDPG, and MARDPG with complexities of 17.587%, 13.602%, and 9.617% respectively. In contrast, MVD-DRL shows the least complexity at 5.632%, shows a decrease of 74% compared to GRM. This advantage persists across different amounts of subcarriers, with MVD-DRL consistently demonstrating lower complexities ranging from 65.4% to 74% compared to GRM. Starting with 5 subcarriers, GRM exhibits the highest signal overhead at 18.086%, while MVD-DRL records the least at 9.878%, represent enhancement of 45.4% compared to GRM. As the amount of subcarriers increases, GRM's signal overhead rises, reaches 24.194% at 25 subcarriers. In contrast, MVD-DRL maintains lower signal overheads, ranging from 9.878% to 15.986%, shows efficiency in minimizing overhead across different subcarrier amounts. MVD-DRL model shows significant reduction in signal overhead compared to other models, with decreases ranging from 33.9% to 45.4% compared to GRM. Starting with 5 subcarriers, GRM exhibits the shortest convergence time at 17.891 seconds, followed closely by other models. As the amount of subcarriers increases, convergence times decrease for all models. The convergence times decrease consistently with higher network complexity, indicating improved efficiency in reaching convergence.

Table 5 presents the results of an ablation study conducted to compare the performance of the proposed and existing approaches with respect to the amount of IoT sensor nodes. For NSC comparison across varying amounts of IoT sensor nodes, MVD-DRL consistently outperforms other models. Starting with 50 IoT sensor nodes, MVD-DRL achieves an NSC of 7.777, a 174.3% increase over GRM's NSC of 2.837. This trend continues with 100, 150, 200, and 250 IoT sensor nodes, where MVD-DRL maintains its lead with NSC values of 10.381, 16.110, 23.689,

and 30.003, respectively. Compared to GRM, MVD-DRL shows improvements ranging from 19.8% to 174.3%, shows its consistent superiority in optimizing IoT connectivity across various scenarios. The computational complexity analysis highlights the efficiency of the MVD-DRL model in resource utilization across different amounts of IoT sensor nodes. Starting with 50 nodes, MVD-DRL exhibits the lowest complexity at 5.757%, a 63.7% decrease compared to GRM. This trend continues with 100, 150, 200, and 250 nodes, where MVD-DRL maintains its efficiency with complexities ranging from 6.67% to 8.55%, shows decreases from GRM ranging from 54.2% to 63.7%. DSAC, MADDPG, and MARDPG also shows reductions in complexity compared to GRM, further highlighting the scalability and effectiveness of MVD-DRL in managing computational resources across varying scenarios. The signal overhead analysis shows the efficiency of the MVD-DRL model in minimizing overhead across different amounts of IoT sensor nodes.

With 50 nodes, MVD-DRL shows the lowest overhead at 10.003%, a 39.8% decrease from GRM. This trend continues with 100, 150, 200, and 250 nodes, where MVD-DRL maintains its efficiency with overhead ranging from 11.379% to 16.111%, marking decreases from GRM ranging from 29.1% to 39.8%. DSAC, MADDPG, and MARDPG exhibit reductions in overhead compared to GRM shows the effectiveness of MVD-DRL in managing signal overhead across different scenarios. The convergence time analysis shows the efficiency of the MVD-DRL model in reaching convergence across different amounts of IoT sensor nodes. With 50 nodes, MVD-DRL exhibits the longest convergence time at 18.648 seconds, followed by reductions as the amount of nodes increases, maintaining its position with 250 nodes at 13.649 s. DSAC, MADDPG, MARDPG, and GRM also show reductions in convergence time across scenarios. This comparative trend indicates the robustness of MVD-DRL in handling complex scenarios while maintaining convergence efficiency.

4.3 Fairness Comparison

We compare the fairness of the present DRL models for handling resources in IoT sensor nodes in connected sensor networks. Figure 5(a) shows how MVD-DRL improves E2E delay across different IT sensor node counts. MVD-DRL technique reduces time by 72.8% over the GRM algorithm with 50 nodes. As node counts increase, MVD-DRL outperforms other approaches by reducing latency by 57.5% to 72.8%. This discovery shows that MVD-DRL optimizes E2E delay in many circumstances. Throughput across a range of IoT nodes that collect data shows that MVD-DRL performs better in Fig. 5(b). Beginning with 50 nodes, MVD-DRL achieves throughput of 84.756, marking a 20.6% increase compared to GRM. As the amount of nodes increases, MVD-DRL outperforms other models, with throughput improvements ranging from 20.6% to 27.7% across different node counts. It highlights MVD-DRL's efficiency in enhancing throughput and its robustness across diverse scenarios.

DRL models	Amount of subcarriers									
	5	10	15	20	25	5	10	15	20	25
	NSCs					Computational complexity (%)				
GRM	6.752	9.356	15.085	22.664	28.978	21.572	22.485	23.398	23.926	24.365
DSAC	6.977	9.581	15.310	22.889	29.203	17.587	18.500	19.413	19.941	20.380
MADDPG	7.202	9.806	15.535	23.114	29.428	13.602	14.515	15.428	15.956	16.395
MARDPG	7.427	10.031	15.760	23.339	29.653	9.617	10.530	11.443	11.971	12.410
MVD-DRL	7.652	10.256	15.985	23.564	29.878	5.632	6.545	7.458	7.986	8.425
	Signal overhead (%)					Convergence of DRL model (s)				
GRM	18.086	19.462	21.858	22.466	24.194	17.891	15.913	15.353	13.391	12.892
DSAC	16.034	17.410	19.806	20.414	22.142	18.049	16.071	15.511	13.549	13.050
MADDPG	13.982	15.358	17.754	18.362	20.090	18.207	16.229	15.669	13.707	13.208
MARDPG	11.930	13.306	15.702	16.310	18.038	18.365	16.387	15.827	13.865	13.366
MVD-DRL	9.878	11.254	13.650	14.258	15.986	18.523	16.545	15.985	14.023	13.524

Tab. 4. Ablation study of proposed and existing approaches with respect to amount of subcarriers.

DRL models	Amount of IoT sensor nodes									
	50	100	150	200	250	50	100	150	200	250
	NSCs					Computational complexity (%)				
GRM	2.837	5.441	11.170	18.749	25.063	15.841	16.754	17.667	18.195	18.634
DSAC	4.072	6.676	12.405	19.984	26.298	13.320	14.233	15.146	15.674	16.113
MADDPG	5.307	7.911	13.640	21.219	27.533	10.799	11.712	12.625	13.153	13.592
MARDPG	6.542	9.146	14.875	22.454	28.768	8.278	9.191	10.104	10.632	11.071
MVD-DRL	7.777	10.381	16.110	23.689	30.003	5.757	6.670	7.583	8.111	8.550
	Signal overhead (%)					Convergence of DRL model (s)				
GRM	16.611	17.987	20.383	20.991	22.719	13.312	11.334	10.774	8.812	8.313
DSAC	14.959	16.335	18.731	19.339	21.067	13.896	11.918	11.358	9.396	8.897
MADDPG	13.307	14.683	17.079	17.687	19.415	14.480	12.502	11.942	9.980	9.481
MARDPG	11.655	13.031	15.427	16.035	17.763	15.064	13.086	12.526	10.564	10.065
MVD-DRL	10.003	11.379	13.775	14.383	16.111	18.648	16.670	16.110	14.148	13.649

Tab. 5. Ablation study of the proposed and existing approaches with respect to amount of IoT sensor nodes.

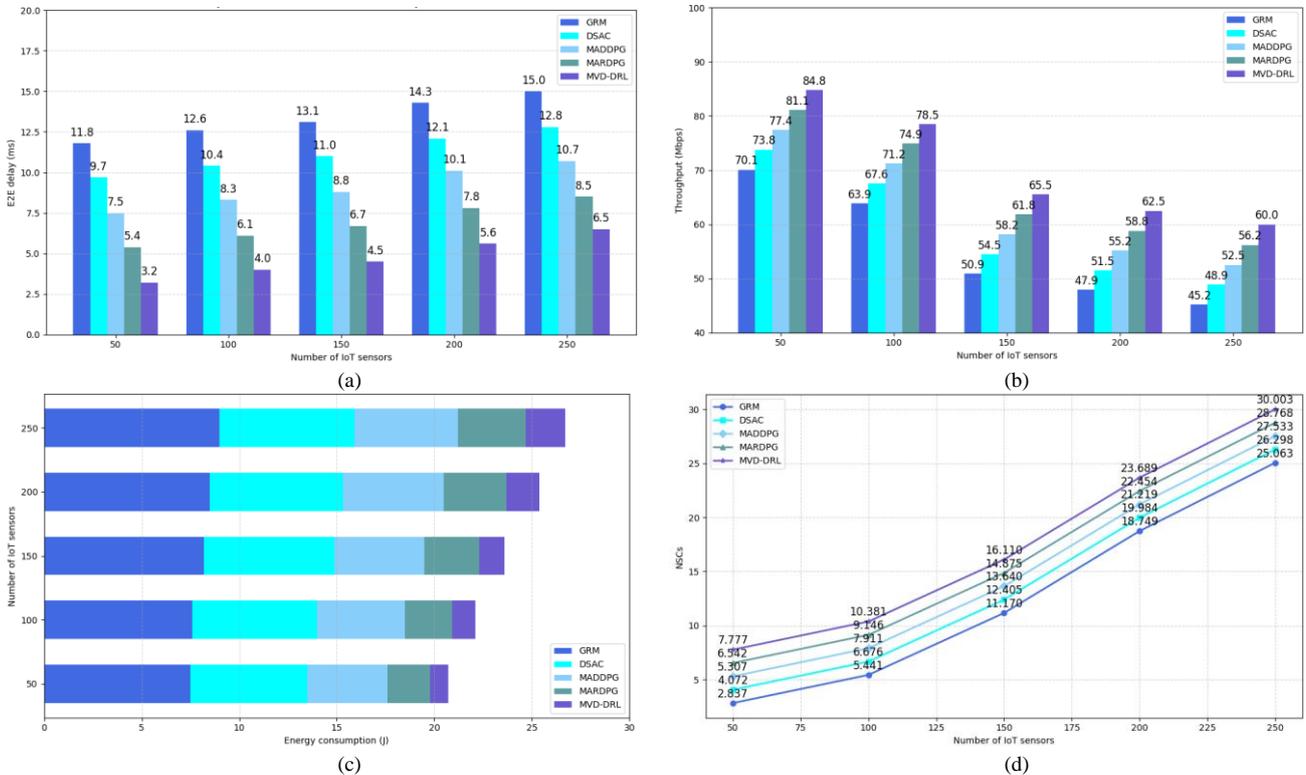


Fig. 5. Measure of (a) E2E delay; (b) throughput; (c) energy consumption, and (d) NSCs with varying amount of IoT sensor nodes.

In Fig. 5(c), the comparison of energy consumption across varying amounts of IoT sensor nodes underscores the efficiency of MVD-DRL. Beginning with 50 nodes, MVD-DRL demonstrates the lowest energy consumption at 0.986, marking a reduction compared to GRM's consumption of 7.594. As the amount of nodes increases, MVD-DRL consistently outperforms other models, with energy consumption reductions ranging from 87% to 75.4% across different node counts. This highlights MVD-DRL's effectiveness in minimizing energy consumption and its robustness in optimizing resource utilization across diverse scenarios.

Figure 5(d) illustrates the NSC comparison across varying IoT sensor node counts. Beginning with 50 nodes, MVD-DRL shows the highest NSC at 7.856, a 218.6% increase over GRM's 2.468. As nodes increase to 250, MVD-DRL maintains its lead with an NSC of 31.457, 20.8% higher than GRM's 26.069. DSAC, MADDPG, and MARDPG exhibit similar trends, with enhancement ranging from 13.5% to 21.0% compared to GRM. Overall, the comparison showcases the performance variations of DRL models across different node counts, with increases ranging from 20.8% to 218.6% compared to GRM.

5. Conclusion

An adaptive resource optimization approach tailored for disaster-resilient IoT connectivity in non-terrestrial environments using deep reinforcement learning (DRL) techniques is presented. By employing the Chaotic Plum Tree (CPT) algorithm for efficient clustering of IoT nodes and optimizing UAV deployment through the non-linear smooth optimization (NLSO) algorithm, the proposed methodology ensures robust and sustainable network connectivity while meeting stringent delay and QoS requirements. Central to our approach is the multi-variable double deep reinforcement learning (MVD-DRL) model, which effectively manages dynamic resource allocation, mitigates congestion, and optimizes transmission power to enhance overall network performance and access success probability. Simulation results demonstrate that the MVD-DRL model achieves significant performance gains compared to existing approaches. Specifically, the end-to-end delay is reduced by 64.486%, 57.66%, 47.586%, and 31.222% compared to GRM, DSAC, MADDPG, and MARDPG models, respectively. The throughput is improved by 20.816%, 15.612%, 10.408%, and 5.204%, while energy consumption is reduced by 80.619%, 75.727%, 67.531%, and 50.978% compared to the same models. Furthermore, the number of successful connections (NSCs) is increased by 28.013%, 21.01%, 14.006%, and 7.003%, respectively. The proposed work presents a promising solution for enhancing disaster-resilient IoT connectivity in non-terrestrial networks, offering substantial improvements in network performance, resource efficiency, and service reliability under challenging environments. Moreover, potential real-world constraints such as signal attenuation due to envi-

ronmental factors, UAV flight endurance, hardware limitations of IoT nodes, interference in dynamic spectrum environments, and mobility-induced latency have also been considered. These practical challenges may affect the overall system performance in real transmission environments. Therefore, as part of future work, we intend to implement the proposed methodology in a real-time testbed using physical UAV and IoT platforms to validate its feasibility, robustness, and adaptability under realistic operating conditions.

References

- [1] CHEN, Q., MENG, W., HAN, S., et al. Spatio-temporal service analysis in multi-layer non-terrestrial networks. *Journal of Communications and Information Networks*, 2024, vol. 9, no. 1, p. 43–55. DOI: 10.23919/jcin.2024.10494943
- [2] SHAMSABADI, A. A., YADAV, A., YANIKOMEROGLU, H. Enhancing next-generation urban connectivity: Is the integrated HAPS-terrestrial network a solution? *IEEE Communications Letters*, 2024, vol. 28, no. 5, p. 1112–1116. DOI: 10.1109/lcomm.2024.3370698
- [3] ZHOU, Y., LEI, L., ZHAO, X., et al. Decomposition and meta-DRL based multi-objective optimization for asynchronous federated learning in 6g-satellite systems. *IEEE Journal on Selected Areas in Communications*, 2024, vol. 42, no. 5, p. 1115–1129. DOI: 10.1109/jsac.2024.3365902
- [4] KHENNOUFA, F., ABDELLATIF, K., KARA, F., et al. Error performance analysis of UAV-mounted RIS for NOMA systems with practical constraints. *IEEE Communications Letters*, 2024, vol. 28, no. 4, p. 887–891. DOI: 10.1109/lcomm.2024.3361378
- [5] KIM, Y. J., HA, S. W., CHO, Y. S. A Doppler-tolerant synchronization method for 5G NR-based non-terrestrial networks. *IEEE Communications Letters*, 2024, vol. 28, no. 3, p. 617–621. DOI: 10.1109/lcomm.2024.3353774
- [6] ABBASI, O., YANIKOMEROGLU, H. UxNB-enabled cell-free massive MIMO with HAPS-assisted sub-THz backhauling. *IEEE Transactions on Vehicular Technology*, 2024, vol. 73, no. 5, p. 6937–6953. DOI: 10.1109/tvt.2023.3347140
- [7] XU, J., FAN, X., JIAN, H., et al. YoloOW: A spatial scale adaptive real-time object detection neural network for open water search and rescue from UAV aerial imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 2024, vol. 62, p. 1–15. DOI: 10.1109/tgrs.2024.3395483
- [8] HORYNA, J., KRÁTKÝ, V., PRITZL, V., et al. Fast swarming of UAVs in GNSS-denied feature-poor environments without explicit communication. *IEEE Robotics & Automation Letters*, 2024, vol. 9, no. 6, p. 5284–5291. DOI: 10.1109/ra.2024.3390596
- [9] SINGLA, A., CALVERAS, A., BETORZ, F., et al. Enhancing satellite non-terrestrial networks through advanced constellation management: optimizing in-orbit resources for NB-IoT. *IEEE Open Journal of the Communications Society*, 2024, vol. 5, p. 2113–2131. DOI: 10.1109/ojcoms.2024.3384265
- [10] AL-HRAISHAWI, H., REHMAN, J. U., RAZAVI, M., et al. Characterizing and utilizing the interplay between quantum technologies and non-terrestrial networks. *IEEE Open Journal of the Communications Society*, 2024, vol. 5, p. 1937–1957. DOI: 10.1109/ojcoms.2024.3380508
- [11] BOQUET, G., MARTINEZ, B., ADELANTADO, F., et al. Low-power satellite access time estimation for internet of things services over nonterrestrial networks. *IEEE Internet of Things*

- Journal*, 2024, vol. 11, no. 2, p. 3206–3216. DOI: 10.1109/jiot.2023.3298017
- [12] CAO, Y., LIEN, S. Y., LIANG, Y. C., et al. Collaborative computing in non-terrestrial networks: A multi-time-scale deep reinforcement learning approach. *IEEE Transactions on Wireless Communications*, 2024, vol. 23, no. 5, p. 4932–4949. DOI: 10.1109/twc.2023.3323554
- [13] GAO, Y., YAN, Z., ZHAO, K., et al. Joint optimization of server and service selection in satellite-terrestrial integrated edge computing networks. *IEEE Transactions on Vehicular Technology*, 2024, vol. 73, no. 2, p. 2740–2754. DOI: 10.1109/tvt.2023.3320187
- [14] GUPTA, A., TRIVEDI, A., PRASAD, B. Multi-UAV and IRS placement for secure data transmission in NOMA-enabled wireless networks. *AEÜ International Journal of Electronics and Communications*, 2024, vol. 178, p. 1–8. DOI: 10.1016/j.aeue.2024.155259
- [15] JUNG, D. H., RYU, J. G., CHOI, J. Satellite clustering for non-terrestrial networks: Orbital configuration-dependent outage analysis. *IEEE Wireless Communications Letters*, 2024, vol. 13, no. 2, p. 550–554. DOI: 10.1109/lwc.2023.3335918
- [16] KIM, E., KIM, J., KIM, J. H., et al. HiMAQ: Hierarchical multi-agent Q-learning-based throughput and fairness improvement for UAV-aided IoT networks. *Journal of Network and Computer Applications*, 2024, vol. 223, p. 1–11. DOI: 10.1016/j.jnca.2023.103813
- [17] KIM, J. B., LEE, I. H., JUNG, H. LEO satellite-aided over-the-air computation for unmanned aerial vehicle swarm sensing. *IEEE Communications Letters*, 2024, vol. 28, no. 1, p. 143–147. DOI: 10.1109/lcomm.2023.3333892
- [18] LIN, K., ULLAH, M. A., ALVES, H., et al. Energy efficiency optimization for subterranean LoRaWAN using a reinforcement learning approach: A direct-to-satellite scenario. *IEEE Wireless Communications Letters*, 2024, vol. 13, no. 2, p. 308–312. DOI: 10.1109/lwc.2023.3327833
- [19] LIU, Y., HUANG, C., CHEN, G., et al. Deep learning empowered trajectory and passive beamforming design in UAV-RIS enabled secure cognitive non-terrestrial networks. *IEEE Wireless Communications Letters*, 2024, vol. 13, no. 1, p. 188–192. DOI: 10.1109/lwc.2023.3325066
- [20] MOSTAFA, A. F., ABDEL-KADER, M., GADALLAH, Y. A UAV-based coverage gap detection and resolution in cellular networks: A machine-learning approach. *Computer Communications*, 2024, vol. 215, p. 41–50. DOI: 10.1016/j.comcom.2023.12.010
- [21] ABOSHOSHA, A., HAGGAG, A., GEORGE, N., et al. IoT-based data-driven predictive maintenance relying on fuzzy system and artificial neural networks. *Scientific Reports*, 2023, vol. 13, p. 1–13. DOI: 10.1038/s41598-023-38887-z
- [22] ALNAKHLI, M. Optimizing spectrum efficiency in 6g multi-UAV networks through source correlation exploitation. *EURASIP Journal on Wireless Communications and Networking*, 2024, no. 6, p. 1–25. DOI: 10.1186/s13638-023-02332-6
- [23] BIRABWA, D. J., RAMOTSOELA, D., VENTURA, N. Multi-agent deep reinforcement learning for user association and resource allocation in integrated terrestrial and non-terrestrial networks. *Computer Networks*, 2023, vol. 231, p. 1–22. DOI: 10.1016/j.comnet.2023.109827
- [24] CHAHED, H., USMAN, M., CHATTERJEE, A., et al. AIDA—A holistic AI-driven networking and processing framework for industrial IoT applications. *Internet of Things*, 2023, vol. 22, p. 1–22. DOI: 10.1016/j.iot.2023.100805
- [25] JEREMIAH, S. R., CAMACHO, D., PARK, J. H. Maximizing throughput in NOMA-enabled industrial IoT network using digital twin and reinforcement learning. *Journal of Advanced Research*, 2024, vol. 66, p. 59–70. DOI: 10.1016/j.jare.2024.04.021
- [26] KUMAR, M., MUKHERJEE, P., VERMA, S., et al. A smart privacy-preserving framework for industrial IoT using hybrid meta-heuristic algorithm. *Scientific Reports*, 2023, vol. 13, p. 1–17. DOI: 10.1038/s41598-023-32098-2
- [27] LI, J., XUE, H., WU, M., et al. Energy efficiency performance in RIS-based integrated satellite–aerial–terrestrial relay networks with deep reinforcement learning. *EURASIP Journal on Advances in Signal Processing*, 2023, no. 121, p. 1–20. DOI: 10.1186/s13634-023-01070-7
- [28] PANDEY, A. K., SAXENA, R., AWASTHI, A., et al. Privacy preserved data sharing using blockchain and support vector machine for industrial IoT applications. *Measurement: Sensors*, 2023, vol. 29, p. 1–7. DOI: 10.1016/j.measen.2023.100891
- [29] PAWASE, C. J., CHANG, K. Demodulation reference signal (DM-RS) based channel estimation for non-terrestrial networks to support high mobility. *ICT Express*, 2024, vol. 10, no. 1, p. 46–52. DOI: 10.1016/j.icte.2023.05.004
- [30] YIN, Y., HUANG, C., XIONG, N. N., et al. Joint dynamic routing and resource allocation in satellite-terrestrial integrated networks. *Computer Networks*, 2023, vol. 231, no. C, p. 1–17. DOI: 10.1016/j.comnet.2023.109823
- [31] DING, F., BAO, C., ZHOU, D., et al. Towards autonomous resource management architecture for 6G satellite-terrestrial integrated networks. *IEEE Network*, 2024, vol. 38, no. 2, p. 113–121. DOI: 10.1109/mnet.2024.3354308
- [32] HE, M., WU, H., ZHOU, C., et al. Resource slicing with cross-cell coordination in satellite-terrestrial integrated networks. *arXiv (Cornell University)*, 2024, p. 1–6. DOI: 10.48550/arxiv.2404.13158
- [33] LI, J., CHAI, R., LIU, C., et al. Energy-aware joint route selection and resource allocation in heterogeneous satellite networks. *IEEE Transactions on Vehicular Technology*, 2024, vol. 73, no. 8, p. 12067–12081. DOI: 10.1109/tvt.2024.3381479
- [34] TONG, M., LI, S., HAN, W., et al. Online learning-based offloading decision and resource allocation in mobile edge computing-enabled satellite-terrestrial networks. *China Communications*, 2024, vol. 21, no. 3, p. 230–246. DOI: 10.23919/jcc.2023.0043.202403
- [35] SHI, J., CHEN, X., ZHANG, Y., et al. Joint optimization of task offloading and resource allocation in satellite-assisted IoT networks. *IEEE Internet of Things Journal*, 2024, vol. 11, no. 21, p. 34337–34348. DOI: 10.1109/jiot.2024.3398055
- [36] SHI, J., YANG, H., CHEN, X., et al. Resource allocation for integrated satellite-terrestrial networks based on RSMA. *IET Communications*, 2025, no. 19, p. 1–12. DOI: 10.1049/cmu2.12745
- [37] SUN, J., WANG, H., SUN, J., et al. An online integrated satellite–terrestrial IoT task offloading and service deployment strategy. *Internet of Things*, 2024, no. 26, p. 1–20. DOI: 10.1016/j.iot.2024.101210
- [38] TANG, J., LI, J., ZHANG, L., et al. Opportunistic content-aware routing in satellite-terrestrial integrated networks. *IEEE Transactions on Mobile Computing*, 2024, vol. 23, no. 11, p. 10460–10474. DOI: 10.1109/tmc.2024.3377729
- [39] LIN, T., LUO, Z. A self-attention based dynamic resource management for satellite-terrestrial networks. *China Communications*, 2024, vol. 21, no. 4, p. 136–150. DOI: 10.23919/jcc.2023-0489.202404
- [40] VAEZI, M., AZARI, A., KHOSRAVIRAD, S. R., et al. Cellular, wide-area, and non-terrestrial IoT: A survey on 5G advances and the road toward 6G. *IEEE Communications Surveys and Tutorials*, 2022, vol. 24, no. 2, p. 1117–1174. DOI: 10.1109/comst.2022.3151028

- [41] MOHAMMADISARAB, A., NOURUZI, A., KHALILI, A., et al. Resilient disaster relief in industrial IoT: UAV trajectory design and resource allocation in 6G non-terrestrial networks. *IEEE Open Journal of the Communications Society*, 2024, vol. 5, p. 1827–1845. DOI: 10.1109/ojcoms.2024.3376335

About the Authors ...

Fathe JERIBI is an Associate Professor at the College of Engineering and Computer Science, Jazan University, Saudi Arabia. He received his Ph.D. in Information Technology from Towson University, Maryland, USA and MS in Computer Science and Information Technology from Sacred Heart University, Connecticut, USA. His areas of interest are machine learning, software engineering, com-

puter networks, SDN, wireless ad hoc networks, IoT, and distributed computing.

R. John MARTIN (corresponding author) a distinguished educator in Computer Science, holds expertise in Computational Intelligence, Machine Vision, Healthcare AI, Sustainable Computing and Signal Processing, spanning a 27-year career in higher education, research, and educational leadership. His Ph.D. in Computer Science from Bharathiar University, India focused on an innovative machine learning model for epileptic seizure detection using EEG signals. Currently at Jazan University, Saudi Arabia, he's known for his pedagogical talent and administrative leadership. His impactful works feature in prestigious journals of IEEE and the ACM. Notably, Martin holds international patents and copyrights, showcasing ground-breaking healthcare AI innovations.