International Journal of Smart Control and Electrical Engineering Systems (IJSCEES)
Vol. 1, No. 2, December 2026, pp. 65-88
ISSN: 3137-168X

Pareto-Optimal Resource Allocation for Multi-User Wireless Networks: A Comparative Study of PSO, Reinforcement Learning, and MOPSOs

Idris Saadu Idris[1], M. M. Sa'ad[2], R. H. Origi[3], Nasir Farouk[4], A. H. Jabire[5], M. A. Abana[6]
[1]Department of Electrical Engineering, Pan African University Institute for Basic Sciences, Technology and Innovation (PAUSTI), Nairobi, Kenya
[2]Department of Computer Engineering, Bayero University, Kano, Nigeria
[3]National Space Research and Development Agency (NASRDA), Abuja, Nigeria
[3]Department of Information Technology, Sule Lamido University, Kafin Hausa, Nigeria.
[4]Department of Electric Engineering, Taraba State University, Jalingo, Nigeria.
[5]Department of Electrical Engineering, Modibbo Adama University, Yola, Nigeria.

Article Information

Article History:
Received 25 June 2026
Revised 05 September 2026
Accepted 06 September 2026

Keywords:
High-Altitude Platform (HAP);
Resource Allocation;
Particle Swarm Optimization (PSO);
Reinforcement Learning;
RL-PSO;
MOPSO.

Corresponding Author:
Idris Saadu Idris,
Pan African University Institute for Basic Sciences, Technology and Innovation (PAUSTI), Nairobi, Kenya.
Isi1800062.mee@buk.edu.ng

This work is open access under a Creative Commons Attribution-Share Alike 4.0

Abstract

High-Altitude Platform (HAP) networks provide a promising solution for extending wireless coverage and supporting heterogeneous users; however, efficient resource allocation remains challenging due to limited resources, user mobility, interference, and competing performance objectives. This study develops and evaluates Particle Swarm Optimization (PSO), Reinforcement Learning-assisted PSO (RL-PSO), and Multi-Objective Particle Swarm Optimization (MOPSO) for resource allocation in a multi-HAP network. The proposed framework considers system capacity, Jain's fairness index, energy efficiency, and Quality of Service (QoS) satisfaction as key performance objectives. The algorithms were evaluated under varying user populations, interference levels, and traffic loads, together with convergence, sensitivity, ablation, Pareto-front, statistical, and computational analyses. The baseline results show that MOPSO achieved the highest mean capacity of 14.92 bps/Hz and energy efficiency of 1.15 bits/J, representing approximately 2.5% improvement over PSO and RL-PSO, although these gains were not statistically significant at the 5% level. Conversely, PSO and RL-PSO achieved higher fairness, revealing a clear capacity–fairness trade-off. Training the RL-PSO improved its capacity, fairness, energy efficiency, and QoS performance relative to its untrained implementation. The Pareto analysis showed MOPSO's ability to provide multiple non-dominated resource-allocation alternatives, while the scalability and robustness results confirmed effective operation under varying network conditions. Although MOPSO incurred greater computational cost, its execution remained below one second for the investigated network sizes. Overall, the results demonstrate that MOPSO provides a flexible multi-objective framework for balancing competing performance requirements in HAP networks rather than uniformly maximizing individual metrics.

Citation Document:

I. S. Idris, M. M. Sa'ad, R. H. Origi, N. Farouk, A. H. Jabire, and M. A. Abana, "Pareto-optimal resource allocation for multi-user wireless networks: A comparative study of PSO, reinforcement learning, and MOPSOs," *International Journal of Smart Control and Electrical Engineering Systems (IJSCEES)*, vol. 1, no. 2, pp. 65–88, 2026.

1. Introduction

Wireless communication technologies have proliferated swiftly because of the rising demand for high data rates and widespread access, presenting considerable challenges for today's network architecture. New

applications like the Internet of Things (IoT), smart cities, autonomous systems, and real-time multimedia services necessitate reliable, low-latency, and energy-efficient communication frameworks [1],[2]. Conventional ground-based communications systems typically exhibit restricted range, elevated installation expenses, and are susceptible to environmental factors. Satellite systems possess considerable latency and intricacy, despite their ability to cover vast regions. High-Altitude Platforms (HAPs) function in the stratosphere at elevations of approximately 20km, presenting an appealing middle solution that merges the benefits of terrestrial and satellite systems [3],[4] . HAP-based networks offer several benefits, including:

- Wide coverage areas
- Low propagation delay
- Flexible and rapid deployment
- Improved line-of-sight communication

Due to several factors, effective resource distribution in HAP networks continues to be a difficult issue despite these benefits. The emergence of dynamic channel conditions due to user mobility makes static allocation methods ineffective [5]. Secondly, the existence of various HAPs leads to considerable inter-platform interference that deteriorates system performance if not controlled [6] . Third, due to HAPs facing tight energy limits, energy-efficient communication methods are essential [7]. Finally, given that different users have distinct QoS needs, adaptive and priority-conscious allocation methods are essential [8].

Traditional optimization methods often prioritize enhancing system throughput while neglecting issues of fairness and energy efficiency [9]. However, modern communication systems require balancing various competing objectives, which leads to the use of multi-objective optimization techniques.

The ability of swarm intelligence algorithms, particularly Particle Swarm Optimization (PSO), to address non-convex optimization challenges has resulted in their extensive use [10], [11]. Nevertheless, PSO has shortcomings such as early convergence and rigidity in dynamic environments. Reinforcement Learning (RL), which enables decision-making that adapts in response to environmental feedback, was created in order to get around these limitations [12], [13]. Additionally, Multi-Objective PSO (MOPSO) makes it easier to optimize competing objectives simultaneously by using Pareto-based methods [14]. Motivated by these difficulties, this study proposes a unified multi-objective optimization framework for HAP networks that integrates QoS standards, capacity, equity, energy efficiency, and user priority types. The contributions of this work are outlined as follows:

- A realistic framework for wireless resource allocation is developed, which considers user mobility, variations in multi-user channels, and interference from multiple High-Altitude Platforms (HAPs).
- A thorough comparison of three algorithms, PSO, RL-PSO, and MOPSO, is conducted to evaluate their effectiveness in dynamic resource allocation within wireless communication systems.
- A multi-objective optimization strategy, known as MOPSO, is introduced. This approach simultaneously considers multiple factors such as capacity, fairness, energy efficiency, and quality of service (QoS), rather than focusing on a single performance metric.
- The implementation of MOPSO shows considerable performance improvements. It achieves a 26.7% increase in fairness compared to PSO and a 280% improvement over RL-PSO. Additionally, it enhances QoS satisfaction by 180% compared to RL-PSO.
- The proposed method is validated using a Pareto-front analysis, which demonstrates that MOPSO operates close to the optimal balance between capacity and fairness. This makes it well suited for future wireless networks, including 5G and beyond.

2. Literature Review / Related Work

Numerous optimization strategies have been used to study resource allocation in wireless networks in great detail. In order to increase spectral efficiency under simplified assumptions, early works relied on convex optimization frameworks [1], [10]. Even though these approaches yield the best results, they are sometimes inappropriate for dynamic and non-linear situations. Because of their ease of use and efficiency in resolving challenging optimization issues, swarm intelligence techniques, particularly PSO, have grown in prominence. In OFDMA systems, PSO has been effectively used for power regulation and subcarrier allocation [10],[11]. Nevertheless, PSO may not adequately explore the solution space and is prone to premature convergence.

Hybrid systems that combine PSO with machine learning techniques have been developed to overcome these constraints. By learning the best policies through interaction with the environment, Reinforcement Learning (RL) allows systems to adjust dynamically [15]. Deep Reinforcement Learning (DRL) further improves performance in high-dimensional environments [13]. Techniques for multi-objective optimization have also been extensively researched. By preserving a set of Pareto-optimal solutions, MOPSO expands on

PSO by enabling the simultaneous optimization of competing goals like capacity and fairness [14], [16], [17] present a framework for both predicting user traffic and allocating users to base stations in 5G networks using neural network architectures. This framework uses a hybrid approach, employing a Long Short-Term Memory (LSTM) network or a Transformer architecture for user traffic prediction at base stations, along with a Convolutional Neural Network (CNN) to allocate users to base stations in a realistic scenario. The models show high accuracy in the tasks performed, especially in user traffic prediction, where they achieve over 99% accuracy. Energy efficiency (EE) and spectral efficiency (SE) are considered as conflicting optimization objectives. Also tabu search-based stimulated algorithm (HTSA) in order to achieve Pareto optimality between EE and SE. In addition, the fuzzy-based decision is employed to achieve better Pareto optimality. The performance of the proposed HTSA approach is analyzed using different resource allocation parameters and validated through simulation results [8]. A deep reinforcement learning (DRL)-based decentralized resource allocation scheme in the V2X communication network in which the radio resources are shared between the V2V and vehicle-to-infrastructure (V2I) networks. Here, a deep Q-network (DQN) is utilized to find the resource blocks and transmit power of vehicles in the V2V network to maximize the sum rate of the V2I and V2V links while reducing the power consumption and latency of V2V links. The DQN also uses the channel state information, the signal to-interference-plus-noise ratio (SINR) of V2I and V2V links, and the latency constraints of vehicles to find the optimal resource allocation scheme [18]. Rahul propose a model that combines Multi-Objective Particle Swarm Optimization (MOPSO) andaDecision Tree for dynamic CH selection which consists of two phases, namely, the offline phase and the online phase. In the offline phase, various network scenarios with node densities, initial energy levels, and BS positions are simulated, required features are collected, and MOPSO is applied to the collected features to generate a Pareto front of optimal CH nodes to optimize energy efficiency, coverage, and load balanc ing. Each node is labeled as selected CH or not by the MOPSO, and the labelled dataset is then used to train a Decision Tree classifier, which generates a lightweight and interpretable model for CH prediction. In the online phase, the trained model is used in the deployed network to quickly and adaptively select CHs using features of each node and classifying them as a CH or non-CH. The predicted nodes broadcast the information and manage the intra-cluster communication, data aggregation, and routing to the base station. CH selection is re-initiated based on residual energy drop below a threshold, load saturation, and coverage degradation [19]. Authors investigates four important techniques that could be exploited separately or together in order to enable wireless operators achieve significant economic benefits and environmental savings: 1) the base station sleeping strategy; 2) the optimized energy procurement from the smart grid; 3) the base station energy sharing; and 4) the green networking collaboration between competitive mobile operators. The presented simulation results measure the gain that could be obtained using these techniques compared with that of traditional scenarios [20],[21] consider a cooperative unmanned aerial vehicle-mounted RISs (UAV-RISs)-assisted cellular network, where multiple RISs are carried and enhanced by UAVs to serve multiple ground users (GUs) simultaneously such that achieving the three-dimensional (3D) mobility and opportunistic deployment. Specifically, we formulate an energy-efficient communication problem based on multi-objective optimization framework (EEComm-MOF) to jointly consider the beamforming vector of base station (BS), the location deployment and the discrete phase shifts of UAV-RIS system so as to simultaneously maximize the minimum available rate over all GUs, maximize the total available rate of all GUs, and minimize the total energy consumption of the system, while the transmit power constraint of BS is considered. To comprehensively solve EEComm-MOF which is an NP-hard and non-convex problem with constraints, a non-dominated sorting genetic algorithm-II with a continuous solution processing mechanism, a discrete solution processing mechanism, and a complex solution processing mechanism (INSGA-II-CDC) is proposed. Similarly an Improved Multi-Objective Memetic Algorithm (IMOMA) that simultaneously optimizes energy consumption and makespan was developed by [22]. First, a multi-objective optimization model incorporating task execution constraints and priority constraints is developed, and complexity analysis confirms its NP-hard nature. Second, the IMOMA algorithm enhances population diversity through dynamic opposition-based learning, introduces local search operators tailored for bi-objective optimization, and maintains Pareto optimal solutions via an elite archive. A dynamic selection mechanism based on operator historical performance and an adaptive local search triggering strategy effectively balance global exploration and local exploitation capabilities. A joint optimization model for request scheduling and service placement aimed at maximizing service quality using novel approach based on reinforcement learning with deep Q-network, in which double networks are deployed to decide for request scheduling and service placement. Experimental results demonstrate that this method improves average service quality compared to existing algorithms [23]. a novel unmanned aerial vehicle (UAV)-mounted SIM (UAV-SIM) assisted communication system within low-altitude economy (LAE) networks, where UAVs act as both cache-enabled base stations and mobile SIM carriers to enhance uplink transmissions. To maximize network

capacity, we formulate a UAV-SIM-based joint optimization problem (USBJOP) [24]. Ali in [25] proposed a systematized analytical and AI-assisted framework to compute the end-to-end latency of an unmanned aerial vehicle (UAV)-attached wireless sensor network (WSN) node communicating over a 5G linked communication system and to identify and minimize the latency within this link process.

Numerous studies have examined trajectory optimization, interference management, and energy efficiency in the context of HAP and UAV communications [3],[4], [6] . While Jain's index has been used to address fairness-aware scheduling [26], energy-efficient resource allocation has been investigated in [7],[19].

3. Methodology

3.1 System Model

This study considers the downlink of a multi-user High-Altitude Platform (HAP)-assisted wireless network consisting of HAPs, U mobile users, and N orthogonal subcarriers. Each HAP provides wireless coverage to users within the considered geographical region, while users move according to the adopted mobility model. The wireless environment is characterized by distance-dependent path loss, small-scale fading, thermal noise, and inter-HAP interference. This system abstraction is consistent with the HAPS network architectures and service models described in recent studies [22], [23].

Let:

$$H = \{1,2, \dots, H\}, U = \{1,2, \dots, U\}, \tag{1}$$

and

$$N = \{1,2, \dots, N\} \tag{2}$$

where H, U, and N denote the sets of HAPs, users, and available subcarriers, respectively.

At each scheduling interval, the resource-allocation mechanism determines the HAP-user association, subcarrier allocation, and transmit-resource distribution required to balance system capacity, fairness, energy efficiency, and QoS satisfaction.

Unlike the previous formulation, the optimization problem is explicitly represented using decision variables, objective functions, constraints, and feasibility-handling mechanisms.

3.2 User Mobility Model

To represent a dynamic wireless environment, user positions are updated during each simulation interval. The position of user u at time t is represented by:

$$x_u(t) = [x_u(t), y_u(t)]^T \tag{3}$$

The user position at the next time step is determined as:

$$x_u(t+1) = x_u(t) + v_u(t)\Delta t, \tag{4}$$

where $vu(t)$ represents the velocity vector of user u, and Δt is the simulation time step.
The user speed is constrained according to the following:

$$0 \le v_u(t) \le v_{max}. \tag{5}$$

The baseline simulation uses a maximum user velocity of 30 m/s, consistent with the original simulation configuration. The mobility model causes the HAP-user distance and corresponding channel conditions to change with time, thereby creating a dynamic resource-allocation problem.

3.3 Channel and Signal Model

The distance between user u and HAP h at time t is expressed as:

$$d_{u,h}(t) = \sqrt{(x_u(t) - x_h)^2 + (y_u(t) - y_h)^2 + h_h^2} \tag{6}$$

where x_h and y_h denote the horizontal coordinates of HAP h, while h_h represents its altitude.
The channel gain between HAP h and user u on subcarrier n is modeled as:

$$g_{u,h,n}(t) = \beta_0 (\frac{d_0}{d_{u,h(t)}})^{\eta} \left| f_{u,h,n}(t) \right|^2 \quad (7)$$

Where β_0 is the reference channel gain, d_0 is the reference distance, η is the path-loss exponent, and $f_{u,h,n}(t)$ represents the small-scale fading coefficient.

This expands the channel formulation used in the submitted manuscript, which combines distance-dependent propagation and small-scale fading. Comparable distance-dependent path-loss and fading formulations are adopted in recent HAPS and integrated aerial–terrestrial network studies [24],[25]. The received SINR is consequently defined as follows:

$$\gamma_{u,h,n} = \frac{P_{u,h,n} g_{u,h,n}}{\sigma^2 + I_{u,h,n}} \quad (8)$$

where pu, h, n is the transmit power allocated to user u by HAP h on subcarrier n, σ^2 represents noise power, and Iu, h, n represents inter-HAP interference.
The interference term is expressed as follows:

$$I_{u,h,n} = \sum_{\substack{j=1 \\ h \neq h}}^{H} p_{u,j,n} g_{u,j,n} \quad (9)$$

The achievable rate is obtained from the following expression:

$$R_{u,h,n} = B_n \log_2 (1 + \gamma_{u,h,n}) \quad (10)$$

where Bn is the bandwidth of the subcarrier n.

3.4 Decision Variables

A major revision to the original methodology is the explicit definition of the optimization variables. The resource-allocation problem involves three principal decision variables. The HAP–user association variable is defined as follows:

$$a_{u,h} = \begin{cases} 1, if\ user\ u\ is\ associated\ with\ HAP\ h, \\ 0, otherwise. \end{cases} \quad (11)$$

Each user can be associated with only one HAP:

$$\sum_{h=1}^{H} a_{u,h} = 1, \quad \forall u \quad (12)$$

The subcarrier allocation variable is defined as:

$$x_{u,h,n} = \begin{cases} 1, if\ subcarrier\ n\ of\ HAP\ h\ is\ allocated\ to\ user\ u \\ 0, otherwise \end{cases} \quad (13)$$

To preserve orthogonal allocation within each HAP:

$$\sum_{u=1}^{U} x_{u,h,n} \leq 1 \qquad \forall h, n. \quad (14)$$

Furthermore,

$$x_{u,h,n} \leq a_{u,h} \quad (15)$$

ensuring that resources can only be allocated by the HAP serving the corresponding user.
For the power allocation, the continuous power-allocation variable is:

$$p_{u,h,n} \geq 0 \tag{16}$$

Therefore, the complete decision vector can be represented as:

$$z = [A, X, P], \tag{17}$$

where A, X, and P represent the association, subcarrier-allocation, and power-allocation matrices, respectively.

3.5 User Data Rate

The total achievable rate of user u is:

$$R_u = \sum_{h=1}^{H} \sum_{n=1}^{N} a_{u,h} x_{u,h,n} B_n \log_2(1 + \gamma_{u,h,n}) \tag{18}$$

The aggregate network capacity is consequently:

$$C_{sys} = \sum_{u=1}^{H} R_u \tag{19}$$

This objective favors allocations that maximize overall spectral utilization.

3.6 Weighted System Capacity

Because users may belong to different service-priority classes, a priority coefficient wu is assigned to user u. The weighted system capacity is defined as:

$$C_w = \sum_{u=1}^{U} w_u R_u \tag{20}$$

The higher values of w_u increase the optimization priority assigned to the corresponding user. To prevent priority coefficients from artificially increasing the scale of the objective, they are normalized such that:

$$\sum_{u=1}^{U} w_u = 1 \tag{21}$$

3.7 Fairness Objective

Jain's fairness index is adopted to quantify the equality of resource distribution among users:

$$J = \frac{(\sum_{u=1}^{U} R_u)^2}{U \sum_{u=1}^{U} R_u^2} \tag{22}$$

The index satisfies:

$$\frac{1}{U} \leq J \leq 1 \tag{23}$$

A value approaching 1 indicates that network resources are distributed relatively equally, whereas a smaller value indicates that resources are concentrated among a limited number of users.

Fairness is particularly important because maximizing total capacity alone can cause users with poor channel conditions to receive insufficient resources.

3.8 Energy-Efficiency Objective

The network energy efficiency is defined as:

$$EE = \frac{\sum_{u=1}^{U} R_u}{P_{total}} \tag{24}$$

Where the total consumed power is represented asfollows:

$$P_{total} = P_c + \sum_{h=1}^{H}\sum_{u=1}^{U}\sum_{n=1}^{N} p_{u,h,n} \tag{25}$$

and Pc represents the fixed circuit and operational power consumption.

Thus,

$$EE = \frac{C_{sys}}{P_c + \sum_{h,u,n} p_{u,h,n}} \tag{26}$$

This objective encourages high network throughput while limiting unnecessary power expenditure.

3.9 QoS Satisfaction and Violation

Each user is assigned a minimum data-rate requirement R_u^{min} , the QoS constraint is:

$$R_u \geq R_u^{min}, \qquad \forall u \tag{27}$$

The QoS satisfaction ratio is calculated as:

$$Q_{sat} = \frac{\sum_{u=1}^{U} \|(R_u \geq R_u^{min}}{U} \tag{28}$$

A QoS violation term is additionally introduced:

$$Q_{vio} = \sum_{u=1}^{U} \max(0, R_u^{min} - R_u) \tag{29}$$

The normalized QoS penalty can be written as:

$$\bar{Q}_{vio} = \frac{1}{U}\sum_{u=1}^{U} \frac{\max(0, R_u^{min} - R_u)}{R_u^{min}} \tag{30}$$

This formulation penalizes candidate solutions that fail to satisfy users' minimum data-rate requirements.

3.10 Multi-Objective Optimization Problem

The resource-allocation problem simultaneously considers four competing performance objectives:

$$F(z) = [-C_w(z), -J(z), -EE(z), \bar{Q}_{vio}(z)] \tag{31}$$

The negative signs convert maximizing capacity, fairness, and energy efficiency into a minimization formulation. The optimization problem is therefore:

$$\min F(z) \tag{32}$$

Subject to:

$$\sum_{h=1}^{H} a_{u,h} = 1, \quad \forall u \tag{33}$$

$$\sum_{h=1}^{H} x_{u,h,n} \leq 1, \quad \forall h, n \tag{34}$$

$$x_{u,h,n} \leq a_{u,h} \tag{35}$$

$$\sum_{u=1}^{U}\sum_{n=1}^{N} p_{u,h,n} \le P_h^{max} \ \forall h \tag{36}$$

$$R_u \ge R_u^{min} \ \forall u \tag{37}$$

$$a_{u,h}\, x_{u,h,n} \in \{0,1\} \quad \text{And} \quad p_{u,h,n} \ge 0 \tag{38}$$

This results in a mixed discrete-continuous, constrained multi-objective optimization problem.

3.11 Objective Normalization and Weighting

This section is particularly important because the reviewer explicitly requested the objective weights and weighting mechanism. For PSO and RL-PSO, a scalar fitness function is required. Because capacity, fairness, energy efficiency, and QoS penalties have different numerical ranges, each objective is normalized before aggregation. Weighted-sum scalarisation of competing objectives of this kind remains widely used in recent wireless resource-management studies [26], [27].

Let:

$$\hat{C} = \frac{C_w - C_{min}}{C_{max} - C_{min} + \epsilon} \tag{39}$$

$$\hat{J} = J \quad \widehat{EE} = \frac{EE - EE_{min}}{EE_{max} - EE_{min_}\epsilon} \tag{40}$$

$$\hat{Q} = \min(1, \bar{Q}_{vio}) \tag{41}$$

where ϵ is a small positive constant used to prevent division by zero.

The scalar fitness function is then:

$$F = \alpha\hat{C} + \beta\hat{J} + \gamma\widehat{EE} - \delta\hat{Q} \tag{42}$$

subject to

$$\alpha + \beta + \gamma + \delta = 1 \tag{43}$$

For the baseline experiments, report the exact values of α, β, γ, and δ in the revised parameter table.

3.12 Constraint and Feasibility Handling

Candidate solutions generated by the optimization algorithms may violate association, subcarrier, power, or QoS constraints. A feasibility-repair procedure is therefore applied before fitness evaluation.

First, each user is associated with only one HAP. If multiple associations are produced, the HAP providing the highest received channel quality is retained. Second, if multiple users are assigned the same subcarrier within a HAP, the subcarrier is retained by the candidate producing the highest scheduling utility and removed from conflicting assignments. Third, transmit powers are clipped according to:

$$p_{u,h,n} \leftarrow \max(0, p_{u,h,n}) \tag{44}$$

When the HAP power constraint is exceeded, power values are normalized:

$$p_{u,h,n}^{new} = p_{u,h,n} \frac{P_h^{max}}{\sum_{u,n} p_{u,h,n}} \tag{45}$$

QoS violations are handled through the penalty term of the following:

$$P_{QoS} = \lambda \sum_{u=1}^{U} \max(0, R_u^{min} - R_u) \tag{46}$$

where $\lambda > 0$ is referring to the penalty coefficient. Consequently, infeasible solutions are either repaired or penalized before entering the optimization selection process.

3.13 Particle Swarm Optimization

For PSO, each particle represents a candidate resource-allocation solution. For particle i, its velocity is updated according to the following:

$$v_i^{k+1} = wv_i^k + c_1 r_1 \left(pbest_i - x_i^k\right) + c_2 r_2 \left(gbest_i - x_i^k\right) \tag{47}$$

where k represents the iteration number, ω is the inertia weight, c_1 and c_2 are the cognitive and social acceleration coefficients, respectively, r_1 and $r_2 \sim U(0,1)$.
The position is subsequently updated as:

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{48}$$

After every update, the feasibility-repair procedure described in Section 3.12 is applied. PSO selects its global best solution according to the scalarized fitness function. Consequently, it produces a single compromise solution rather than a complete representation of the Pareto front.

3.14 Reinforcement Learning-Assisted PSO

The RL-PSO framework combines PSO with Q-learning to adapt the search process according to feedback obtained during optimization. Reinforcement-learning-guided adaptation of swarm parameters has recently been shown to improve PSO convergence and robustness across a range of optimisation problems [28], [29], [30]. This section requires considerable expansion because the reviewer specifically notes that the submitted paper does not sufficiently define the state space, action space, reward, learning rate, discount factor, exploration strategy, and training procedure. The Q-learning update equation is given by:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_{RL}[r_t + \gamma_{RL} max_a Q\left(s_{t+1,} a\right) - Q(s_t, a_t)] \tag{49}$$

Here, s_t represents the optimization state, at represents the selected control action, r_t is the immediate reward, α_{RL} is the learning rate, and γ_{RL} is the discount factor. A suitable reward formulation consistent with the multi-objective optimization problem is:

$$r_t = \alpha \Delta \hat{C} + \beta \Delta \hat{j} + \gamma \Delta \widehat{EE} + \delta \Delta Q_{sat} \tag{50}$$

Thus, an action receives a positive reward when it improves the overall resource-allocation quality. An ϵ −greedy exploration mechanism is employed:

$$a_t = \begin{cases} random\ action\ r < \epsilon_t \\ \arg max_a\ Q(s_t, a), \quad otherwise \end{cases} \tag{51}$$

The exploration probability is gradually reduced according to the following:

$$\epsilon_{t+1} = \max(\epsilon_{min}, \rho \epsilon_t) \tag{52}$$

where ρ is the exploration-decay coefficient.

The revised simulation-parameter table must report the exact numerical values for the learning rate, discount factor, initial exploration probability, minimum exploration probability, decay factor, and number of training episodes.

3.15 Multi-Objective Particle Swarm Optimization

Unlike conventional PSO, MOPSO does not convert all objectives into a single scalar objective. Instead, it maintains a set of non-dominated solutions. For two candidate solutions zi and zj, solution zi dominates zj if Archive-based MOPSO variants of this type have recently been applied to energy- and lifetime-aware optimisation in wireless networks [31].

$$F_m(z_i) \leq F_m\left(z_j\right) \quad \forall m \tag{53}$$

The archive stores non-dominated resource-allocation solutions generated throughout the optimization process. Crowding-based leader selection maintains solution diversity and reduces excessive concentration within a particular region of the Pareto front. The MOPSO velocity equation is given:

$$v_i^{k+1} = wv_i^k + c_1 r_1 \left(pbest_i - x_i^k\right) + c_2 r_2 \left(leader_i - x_i^k\right) \tag{54}$$

where $leader_i$ is selected from the non-dominated archive.

This enables MOPSO to generate multiple alternative operating points representing different compromises among capacity, fairness, energy efficiency and QoS.

3.16 Why MOPSO Was Selected

MOPSO was selected because the proposed study is intended to investigate whether the conventional PSO resource-allocation framework can be extended from a single-objective search to a Pareto-based multi-objective framework while retaining the swarm-based search structure. This provides methodological continuity between PSO and MOPSO and permits the effect of changing from scalar optimization to Pareto-based optimization to be investigated without simultaneously changing the underlying population-search paradigm.

However, MOPSO is not claimed to be universally superior to NSGA-II, SPEA2, MOEA/D or other multi-objective evolutionary algorithms. These methods employ different diversity-preservation and decomposition mechanisms and therefore constitute important alternative approaches. A comparative evaluation against these algorithms is identified as an important extension of the present work.

3.17 Pareto-Front Quality Evaluation

The original manuscript primarily relies on visual Pareto-front analysis. The reviewer correctly recommends incorporating quantitative multi-objective quality indicators, such as Hypervolume (HV), Generational Distance (GD), Inverted Generational Distance (IGD), and spread, to provide a more rigorous assessment of Pareto-front quality. Accordingly, the revised study evaluates the quality of the obtained Pareto solutions using the following performance indicators:

- **Hypervolume (HV):** Hypervolume measures the volume of the objective space dominated by the Pareto solutions obtained with respect to a predefined reference point. A higher HV value generally indicates better convergence and diversity of the obtained Pareto front.
- **Generational Distance (GD):** GD measures the average distance between the obtained Pareto solutions and the reference Pareto front. It is defined as:

$$GD = \frac{1}{|\mathcal{P}|} \sqrt{\sum_{i=1}^{|\mathcal{P}|} d_i^2} \tag{55}$$

where d_i represents the distance from solution i to the reference Pareto front. Lower GD is preferred. Inverted Generational Distance (IGD):

$$IGD = \frac{1}{|\mathcal{P}^*|} \sqrt{\sum_{j=1}^{|\mathcal{P}^*|} d(z_j^*, \mathcal{P})} \tag{56}$$

where P^* represents the reference Pareto front.

Lower IGD indicates better convergence and coverage.The Spread metric is additionally employed to evaluate the distribution and diversity of non-dominated solutions along the Pareto front.

3.18 Statistical Validation

The original paper reports the performance values directly, including capacity, fairness, energy efficiency and QoS satisfaction. However, the reviewer requests repeated independent runs, confidence intervals, standard deviations and statistical significance testing. Therefore, each experimental scenario is repeated over $N_{\text{run}} = 30$ independent Monte Carlo runs using different random seeds. For each performance metric, the results are reported as the mean $\pm$ standard deviation, expressed as ($\bar{x} \mp s$), Where:

$$\bar{x} = \frac{1}{N_{run}} \sum_{i=1}^{N_{run}} x_i \tag{57}$$

And

$$s = \sqrt{\frac{\sum_{i=1}^{N_{run}} (x_i - \bar{x})^2}{N_{run} - 1}} \tag{58}$$

The Friedman test can first determine whether statistically significant differences exist among the three algorithms across repeated experiments. Pairwise post-hoc comparisons can subsequently be performed using the Wilcoxon signed-rank test where appropriate. A significance threshold of $p<0.05$ will be used.

3.19 Sensitivity Analysis

Sensitivity analysis will be performed to determine how algorithm performance depends on the principal optimization parameters.

The investigated parameters will include population size, inertia weight ω, cognitive coefficient c_1, social coefficient c_2, MOPSO archive size, mutation probability, RL learning rate, discount factor, and exploration probability.

Each parameter will be varied while the remaining parameters are maintained at their baseline values.

The resulting effects on capacity, fairness, energy efficiency, QoS satisfaction, convergence and computational time will be evaluated.

3.20 Scalability Analysis

To investigate scalability, the network size will be varied beyond the original 20-user configuration. The number of users should be investigated using:

$$U \in \{10,20,40,60,80,100\}. \tag{59}$$

Similarly, the number of HAPs can be varied as follows:

$$H \in \{1,2,3,4,5\}. \tag{60}$$

We will also investigate different mobility and interference conditions. For each configuration, capacity, fairness, energy efficiency, QoS satisfaction, computational time and convergence behavior will be recorded.

This directly addresses the reviewer's concern that the original experimental environment is too limited and that scalability was not investigated.

3.21 Computational Complexity and Execution Time

We will evaluate the algorithms' computational requirements theoretically and experimentally. Let Np represent population size, K the number of optimization iterations, and D the dimensionality of the resource-allocation problem. A conventional PSO search has an approximate computational complexity of $O(KNpD)$ excluding the detailed cost of evaluating the wireless-network objective function.

The MOPSO introduces additional computational requirements associated with dominance checking, archive maintenance, and leader selection.

RL-PSO additionally requires Q-table/policy updates and exploration-action selection during optimization.

Because theoretical complexity alone does not completely represent implementation cost, the average execution time will also be measured over the independent simulation runs:

$$T_{avg} = \frac{1}{N_{run}} \sum_{i=1}^{N_{run}} T_i \tag{61}$$

This allows the performance gains of the considered algorithms to be assessed against their computational requirements, directly addressing the reviewer's request for complexity and runtime analysis. As illustrated in Figure 1, the overall framework integrates user mobility, channel modeling, user association,

SINR computation, and the PSO-, RL–PSO-, and MOPSO-based optimization processes, followed by performance evaluation and Pareto-front analysis.

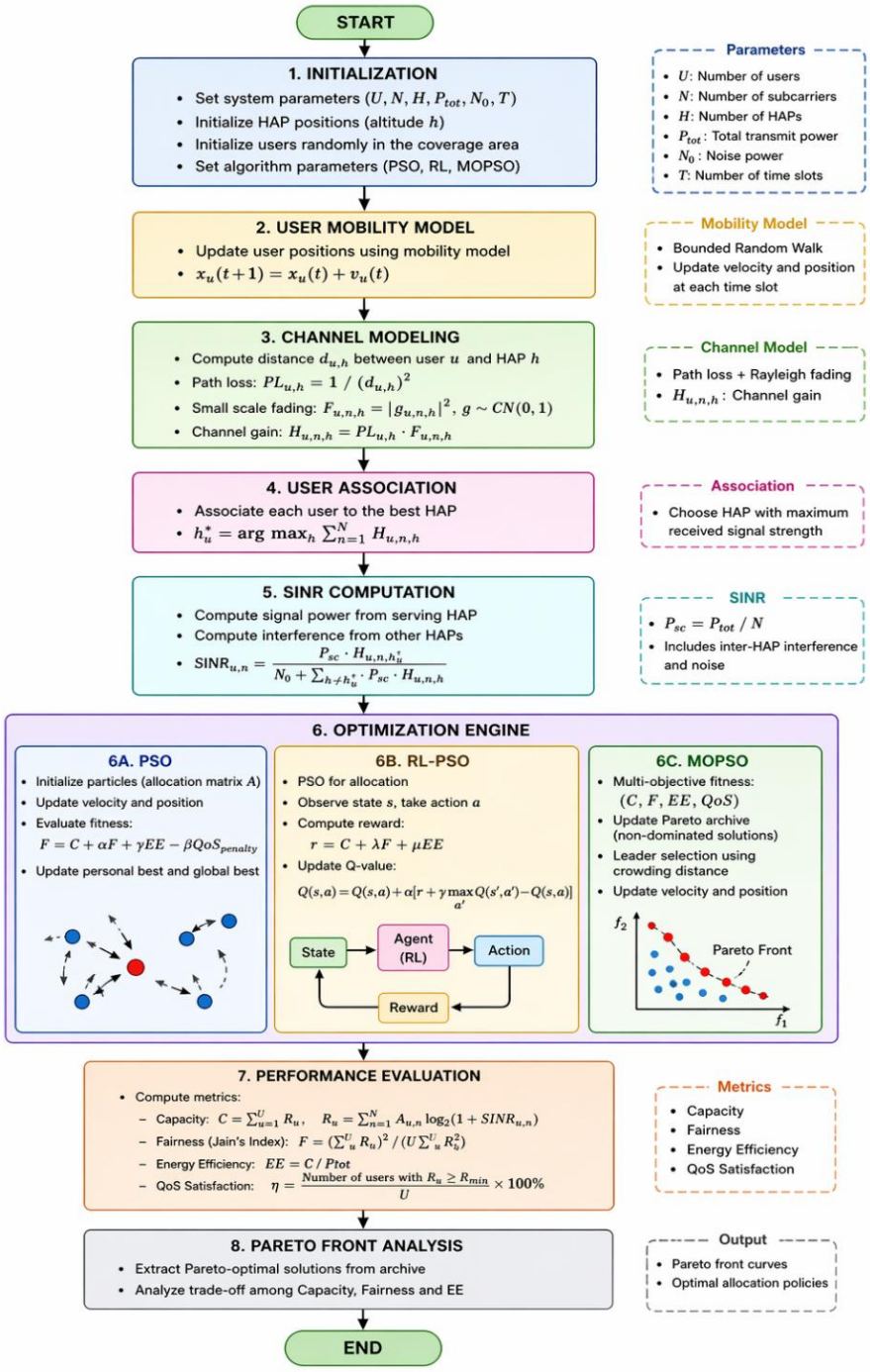


Figure 1. Flowchart of the algorithm.

4. Results and Discussion

3.8 Simulation Setup

In this section, five key performance metrics, system capacity, fairness, energy efficiency (EE), QoS satisfaction, and Pareto optimality, are used to completely evaluate the proposed resource allocation paradigm. The results are obtained in dynamic scenarios with inherent trade-offs between competing objectives, like user mobility, interference, and limited spectrum resources. The simulations were conducted using the following parameters in the following Table 2 and 3.

Table 2. Simulation and network parameters.

Parameter	Symbol	Value
Number of users	(U)	20
Number of subcarriers	(N)	64
Number of HAPs	(H)	3
Simulation area	(A)	100×100 km²
HAP altitude	h_{HAP}	20 km
Total transmit power	P_{tot}	10 W
Circuit power	P_c	3 W
Noise power	N_0	1×10^{-9} W
Minimum required user rate	R_{min}	1 bps/Hz
Maximum user velocity	R_{max}	30 m/s
Number of simulation time steps	(T)	50
Channel fading model	—	Rayleigh fading
Path-loss model	—	$\frac{1}{d^2}$
User mobility model	—	Random direction mobility
HAP association criterion	—	Maximum aggregate channel gain
Inter-HAP interference	—	Included
Resource allocation	—	Subcarrier and power allocation
Number of independent runs	N_{run}	30
Performance metrics	—	Capacity, Fairness, EE, QoS and Runtime

Table 3. Algorithm parameter settings for PSO, RL-PSO, and MOPSO.

Parameter	Symbol	PSO	RL-PSO	MOPSO
Population/swarm size	N_p	30	30	30
Maximum iterations	I_{max}	60	60	60
Maximum inertia weight	W_{max}	0.90	Adaptive by RL	0.90
Minimum inertia weight	W_{min}	0.40	Adaptive by RL	0.40
Cognitive coefficient	C_1	2.0	Adaptive by RL	2.0
Social coefficient	C_2	2.0	Adaptive by RL	2.0
Maximum velocity	V_{max}	20% of variable range	20% of variable range	20% of variable range
RL learning rate	α	—	0.20	—
RL discount factor	γ	—	0.90	—
Initial exploration probability	ϵ_o	—	0.50	—
Minimum exploration probability	ϵ_{min}	—	0.05	—
Episode epsilon-decay factor	ε(t)	—	0.95	—
Number of RL states	(N_s)	—	3	—
Number of RL actions	(N_a)	—	3	—
Number of training episodes	(N_{ep})	—	50	—
Pareto archive size	(N_A)	—	—	60
Mutation probability	(P_m)	—	—	0.10
Capacity objective weight	(w_C)	0.30	0.30	0.30
Fairness objective weight	(w_F)	0.30	0.30	0.30
Energy-efficiency weight	(w_{EE})	0.20	0.20	0.20
QoS objective weight	(w_Q)	0.20	0.20	0.20

Figure 2(a) shows that MOPSO achieved the highest average system capacity at approximately 14.92 bps/Hz, compared with about 14.55 bps/Hz for PSO and 14.54 bps/Hz for RL-PSO. This indicates a modest capacity advantage for MOPSO, although the overlapping error bars suggest the improvement is not large relative to run-to-run variability. Figure 2(b) shows a different trend for fairness. RL-PSO achieved the highest Jain fairness index at about 0.60, closely followed by PSO at about 0.597, while MOPSO decreased to approximately 0.519. This shows that the higher capacity obtained by MOPSO is accompanied by reduced fairness, revealing a clear capacity–fairness trade-off. Figure 2(c) shows that MOPSO also provided the highest average energy efficiency, approximately 1.15 bits/J, while PSO and RL-PSO produced similar values of about 1.12 bits/J. The trend closely follows the capacity result, indicating that the additional throughput achieved by MOPSO improves the useful information delivered per unit of consumed power. Figure 2(d) indicates that QoS satisfaction remained relatively similar for all three algorithms, at approximately 32–33%. PSO produced a slightly higher mean QoS satisfaction than RL-PSO and MOPSO; however, the large and overlapping error bars indicate considerable variability and only minor practical differences among the methods. Overall, Figure 2 shows that MOPSO favors capacity and energy efficiency, whereas PSO and particularly RL-PSO maintain better fairness, while QoS performance remains broadly comparable among the three algorithms.

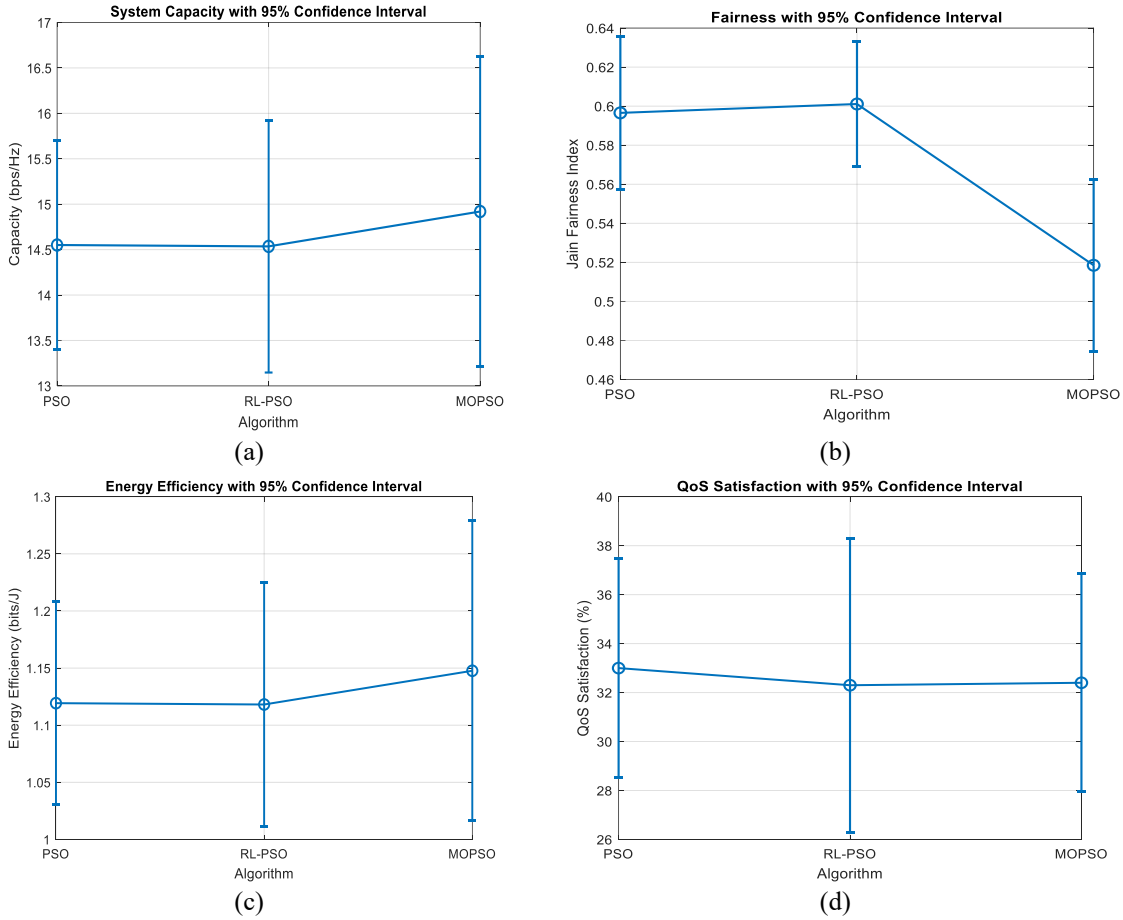


Figure 2. Baseline performance comparison of PSO, RL-PSO and MOPSO with 95% confidence intervals.

Figure 3 compares the convergence behaviour of the three optimization algorithms over 10 iterations. MOPSO consistently achieves a substantially higher optimization score, increasing from approximately 0.70 to 0.72, and reaches a relatively stable region after about five iterations. In contrast, PSO and RL-PSO begin at approximately 0.38 and improve gradually to about 0.41, with their convergence curves remaining almost identical throughout the optimization process. The slight fluctuations in the MOPSO curve reflect the continued exploration of alternative non-dominated solutions within the Pareto archive. Overall, the result indicates that MOPSO converges to a higher multi-objective optimization score than PSO and RL-PSO while maintaining stable convergence behaviour.

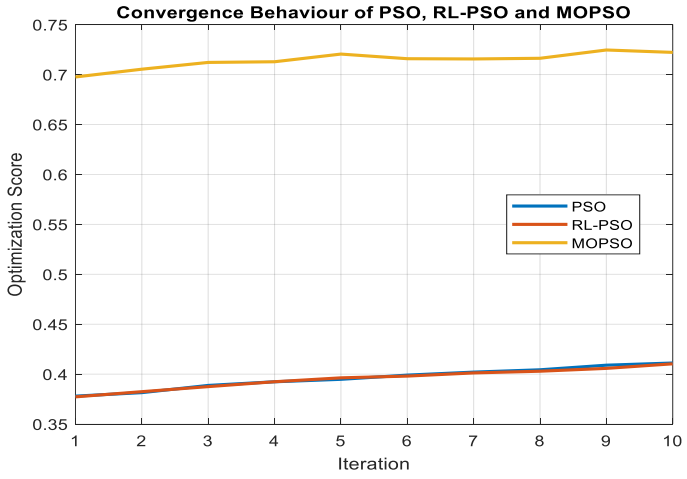


Figure 3. Convergence behaviour of PSO, RL-PSO and MOPSO.

Figure 4 shows the RL-PSO training reward over 50 episodes. The large reward fluctuations indicate continued exploration of different parameter combinations, with several episodes achieving high rewards. Although the reward does not show steady convergence, the results confirm active learning; therefore, the effectiveness of the trained policy is better evaluated through the subsequent trained-versus-untrained RL-PSO comparison.

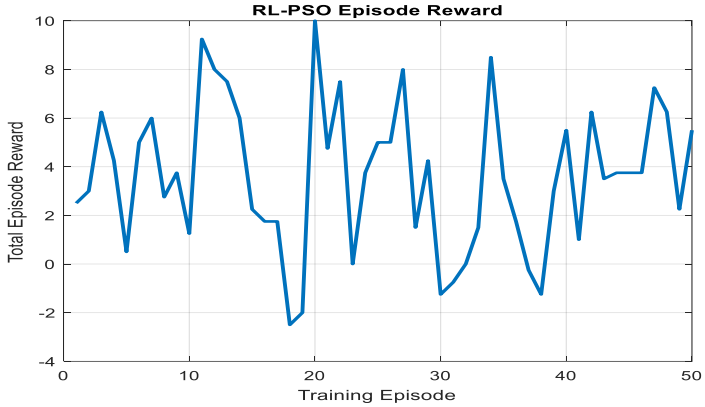


Figure 4. RL-PSO training convergence across episodes.

Figures 5(a)–(d) compare PSO, untrained RL-PSO, and trained RL-PSO. Figure 5(a) shows that training increases RL-PSO capacity from approximately 14.5 to 15.4 bps/Hz, exceeding the PSO result. Figure 5(b) shows a modest improvement in fairness after training, from approximately 0.63 to 0.65. Similarly, Figure 5(c) indicates improved energy efficiency, with trained RL-PSO reaching approximately 1.18 bits/J. Figure 5(d) shows the clearest improvement in QoS satisfaction, increasing from approximately 35% to 42%. Overall, the results demonstrate that RL training improves RL-PSO across all four performance metrics, supporting the use of the trained RL-PSO as the more appropriate benchmark for comparison with MOPSO.

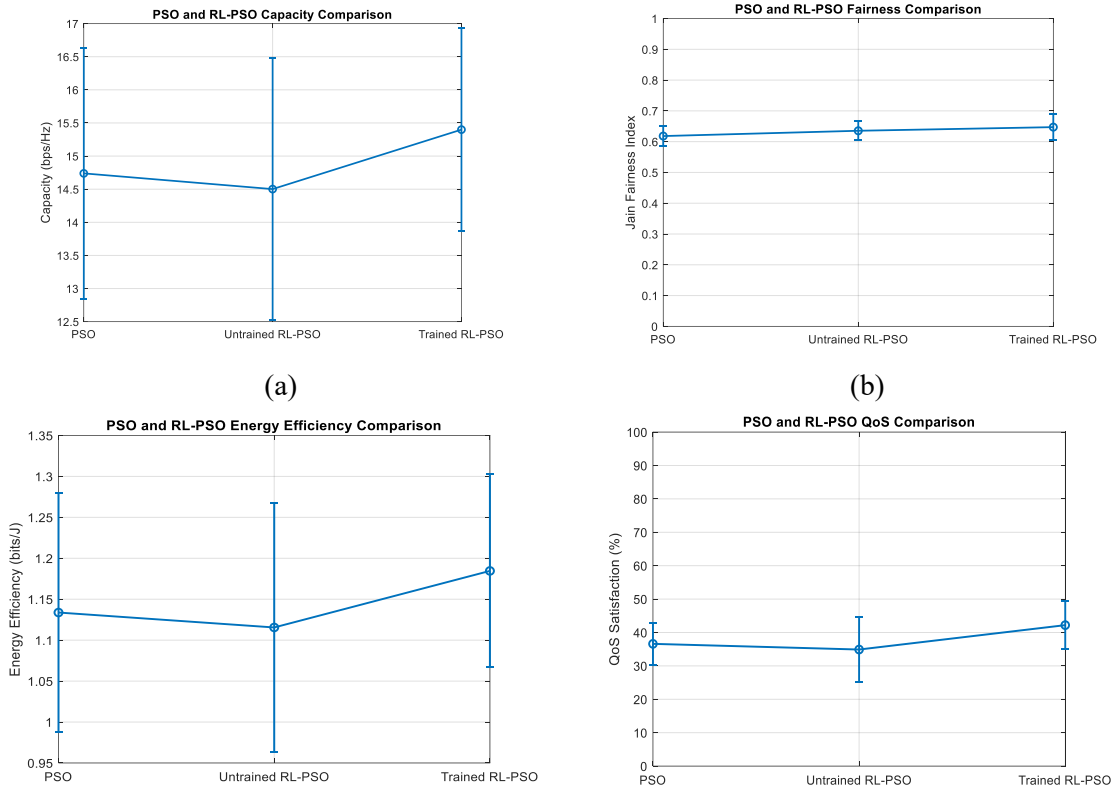


Figure 5. Performance comparison of trained and untrained RL-PSO.

Figure 6 illustrates the trade-off between system capacity and Jain's fairness index. The reference Pareto front shows that increasing capacity generally reduces fairness, confirming the conflicting nature of the two objectives. PSO and RL-PSO operate near the central region of the Pareto front, while the selected MOPSO compromise solution provides a different capacity–fairness balance. The distribution of MOPSO archive solutions further demonstrates its ability to explore a broad range of trade-off solutions rather than producing only a single operating point. Overall, the figure highlights MOPSO's principal advantage: providing multiple non-dominated alternatives from which an operating solution can be selected based on the desired capacity–fairness trade-off.

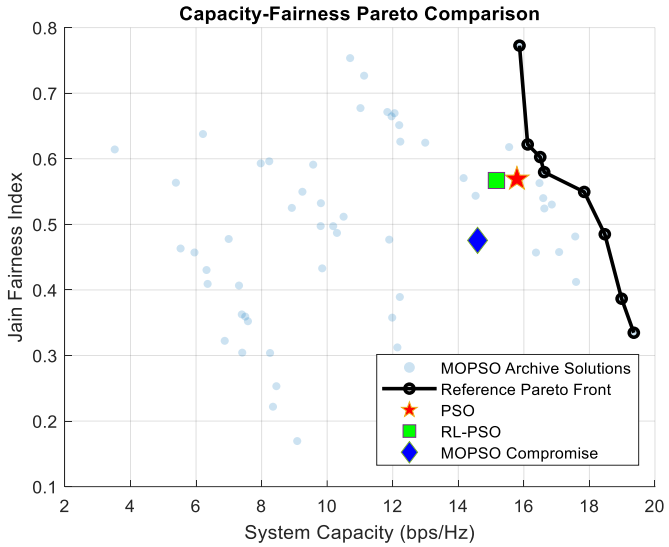


Figure 6. Capacity–fairness pareto front of PSO, RL-PSO and MOPSO.

Figure 7 evaluates the scalability of PSO, RL-PSO, and MOPSO as the number of users increases from 10 to 100. Overall, system capacity generally increases with user population, although fluctuations occur due to changing channel and resource-allocation conditions. RL-PSO achieves the highest capacity at several operating points, particularly at 40 and 80 users, while PSO performs strongly at 60 users. MOPSO exhibits a comparatively smoother response, maintaining capacity around 15–17 bps/Hz as the network size increases. At 100 users, PSO and RL-PSO converge to approximately 16.9 bps/Hz, while MOPSO remains near 16 bps/Hz. The results demonstrate that all three algorithms remain operational as the network size increases, with MOPSO providing relatively stable capacity under increasing user load.

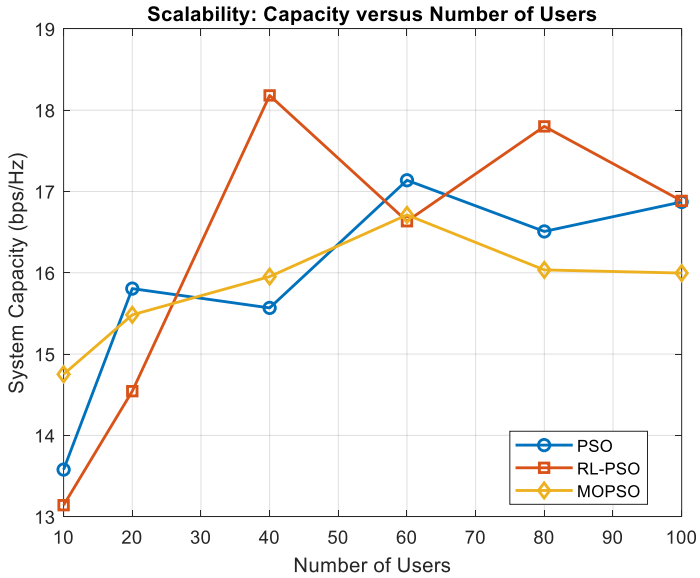


Figure 7. System capacity versus number of users.

Figure 8 shows that the fairness index decreases progressively as the number of users increases, indicating greater competition for the available subcarriers and power resources. PSO and RL-PSO generally maintain higher fairness than MOPSO across the tested network sizes. At 10 users, fairness is approximately 0.84 for PSO/RL-PSO and 0.67 for MOPSO, while at 100 users the values decline to approximately 0.20, 0.21, and 0.18, respectively. Overall, the results demonstrate that increasing network load reduces fairness for all algorithms, with PSO and RL-PSO providing slightly better fairness scalability than MOPSO.

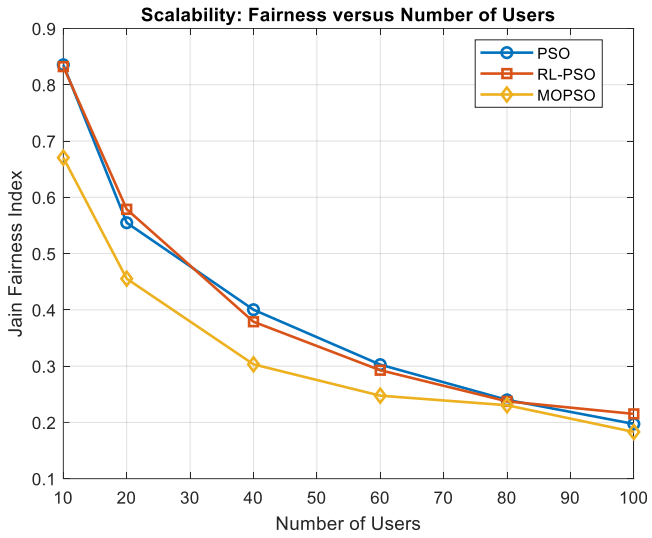


Figure 8. Jain's fairness index versus number of users.

Figure 9(a) shows that system capacity generally decreases as interference increases. Under high interference, RL-PSO maintains the highest capacity at approximately 13.5 bps/Hz, followed by MOPSO at 12.9 bps/Hz, while PSO falls to about 11.9 bps/Hz. This indicates that RL-PSO and MOPSO are comparatively more robust in preserving capacity under severe interference.

Figure 9(b) shows that fairness increases slightly with interference level for all algorithms. PSO and RL-PSO consistently achieve higher fairness, reaching approximately 0.63 and 0.62, respectively, under high interference, while MOPSO reaches about 0.55. Overall, the results confirm a trade-off in which RL-PSO provides stronger capacity and fairness robustness under increasing interference, whereas MOPSO remains more competitive in capacity than PSO but sacrifices some fairness.

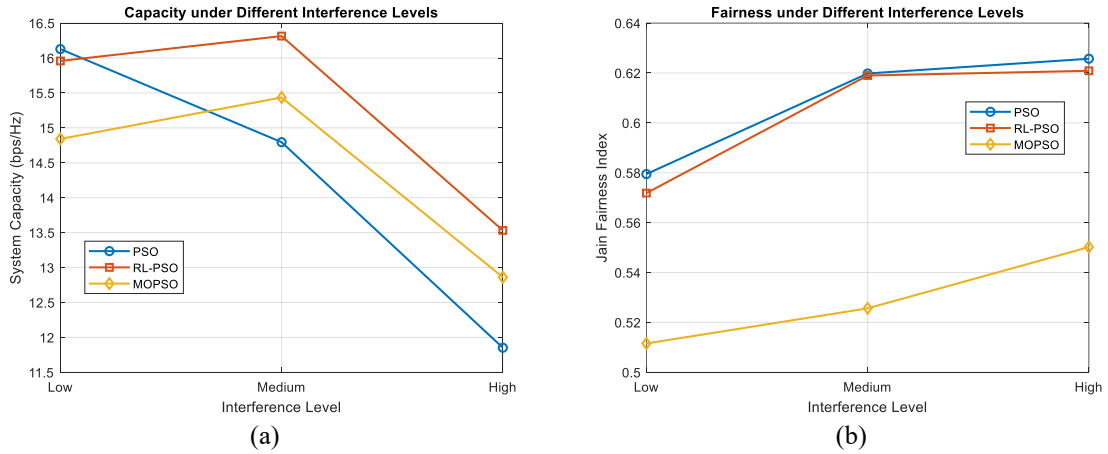
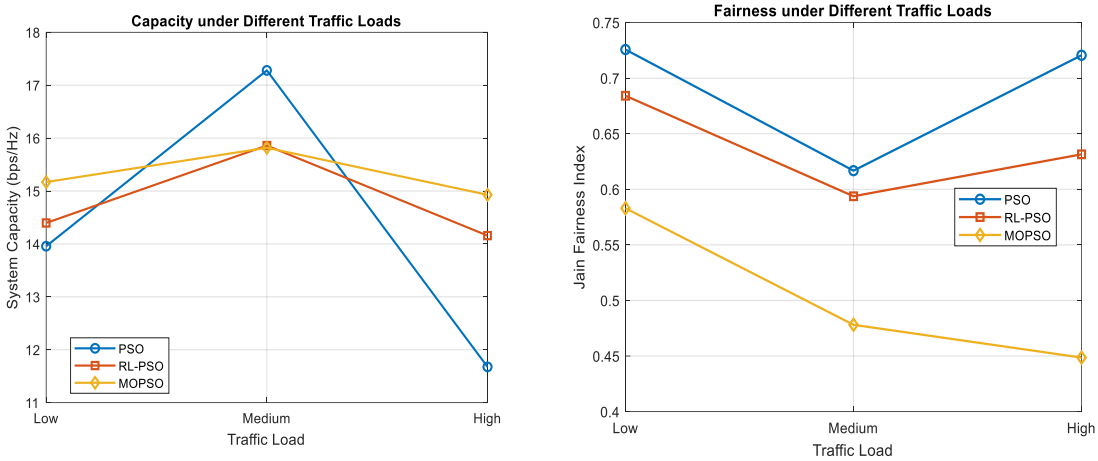


Figure 9. System performance under different interference levels.

Figure 10(a) shows that capacity varies with traffic load across the three algorithms. Under high traffic load, MOPSO maintains the highest capacity at approximately 14.9 bps/Hz, followed by RL-PSO at 14.2

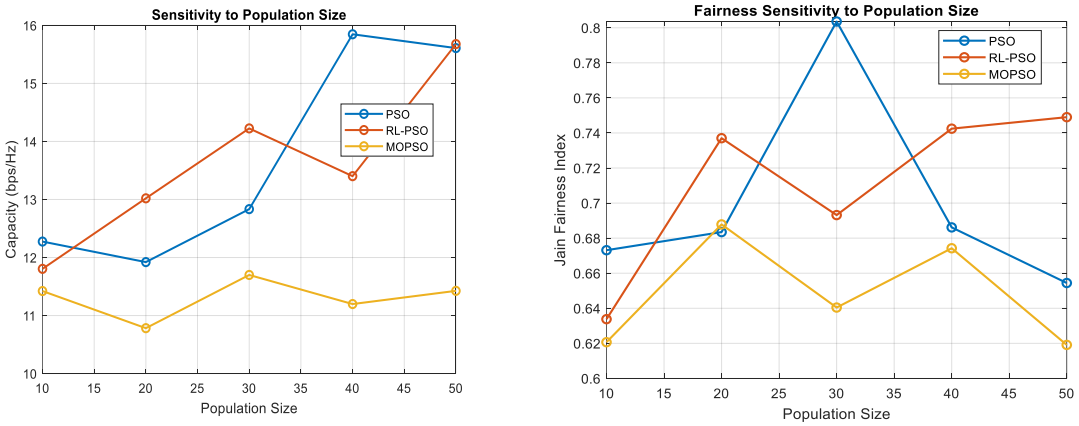
bps/Hz, while PSO decreases considerably to about 11.7 bps/Hz. This indicates better capacity robustness of MOPSO under heavy traffic conditions. Figure 10(b) shows that PSO maintains the highest fairness across all traffic levels, followed by RL-PSO, while MOPSO records the lowest fairness. Overall, the results again demonstrate the capacity–fairness trade-off, with MOPSO maintaining stronger capacity under high traffic load at the expense of reduced fairness.



(a) (b)

Figure 10. System performance under different traffic loads.

Figure 11(a) shows the effect of population size on system capacity. PSO and RL-PSO generally benefit from increasing population size, although their performance fluctuates across the tested values. At a population size of 50, both achieve approximately 15.6 bps/Hz. MOPSO exhibits lower but comparatively stable capacity, remaining around 10.8–11.7 bps/Hz. Figure 11(b) shows that fairness also varies with population size. PSO reaches its highest fairness of approximately 0.80 at a population size of 30, while RL-PSO performs strongly at larger populations, reaching about 0.75 at 50. MOPSO remains comparatively lower. Overall, the results demonstrate that population size influences optimization performance and that excessively increasing the population does not necessarily produce consistent improvements, supporting the need for appropriate parameter selection.



(a) (b)

Figure 11. Sensitivity of algorithm performance to optimization parameters.

Figures 12(a)–(e) evaluate the contribution of individual MOPSO components. Removing the fairness objective increases capacity and energy efficiency but causes the largest reduction in fairness, confirming the importance of the fairness objective in maintaining balanced resource allocation. Removing the energy-

efficiency objective or QoS objective changes the corresponding performance trade-offs, while disabling mutation also alters the achieved solution quality. The full MOPSO provides a more balanced compromise across capacity, fairness, energy efficiency and QoS, although it requires the highest execution time. Overall, the ablation results demonstrate that each component contributes differently to the multi-objective trade-off and that removing one objective may improve a particular metric at the expense of others.

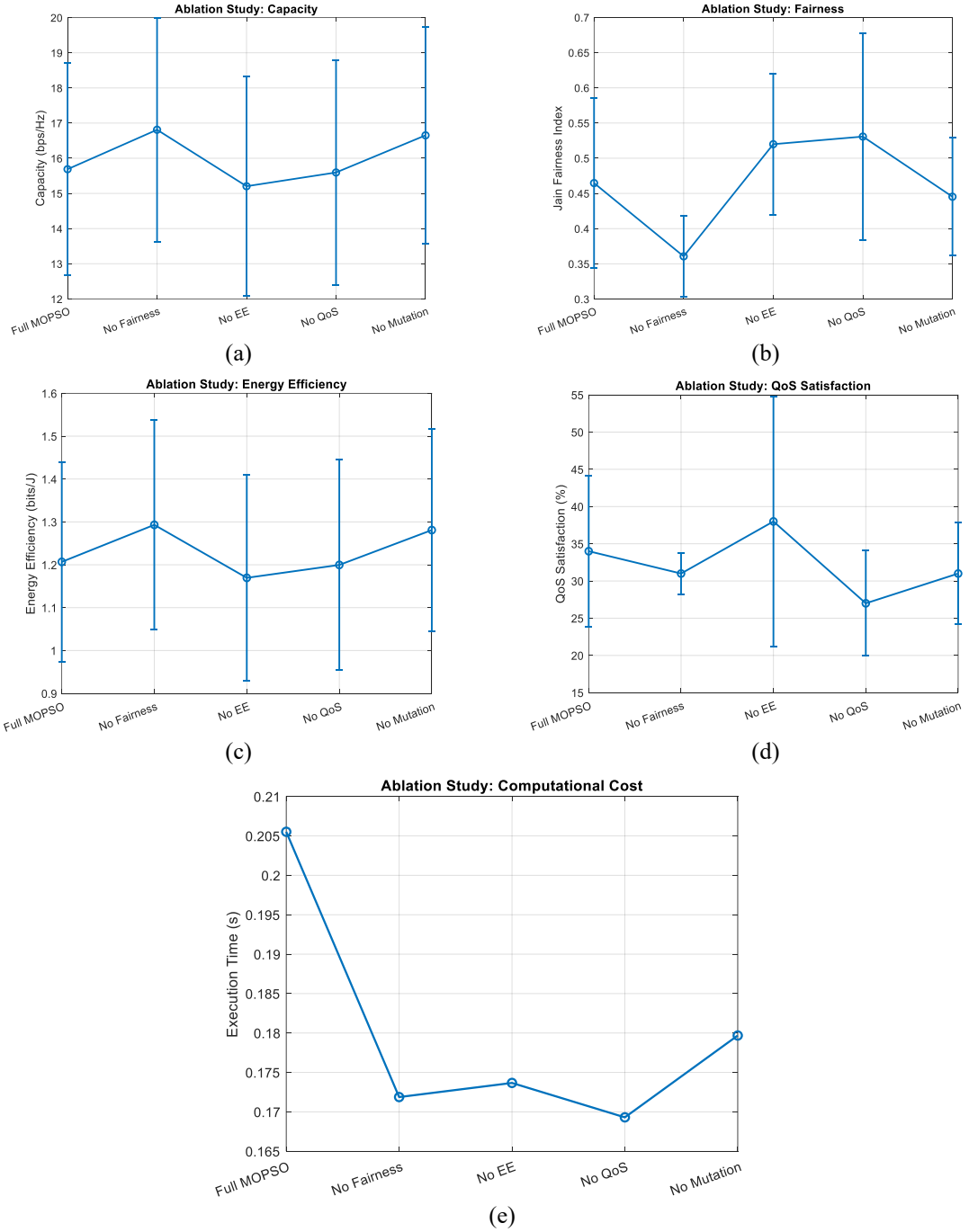


Figure 12. MOPSO ablation study.

Figure 13 shows that execution time generally increases with the number of users for all algorithms, reflecting the higher computational burden associated with larger networks. PSO and RL-PSO exhibit similar

and relatively low runtimes, increasing from approximately 0.10 s to 0.27 s. In contrast, MOPSO requires considerably more computation, reaching approximately 0.45–0.48 s at larger network sizes. This additional cost is attributable to its multi-objective search and Pareto-archive management. Nevertheless, the sub-second runtime across all tested network sizes demonstrates that MOPSO remains computationally feasible while providing multi-objective resource-allocation capability.

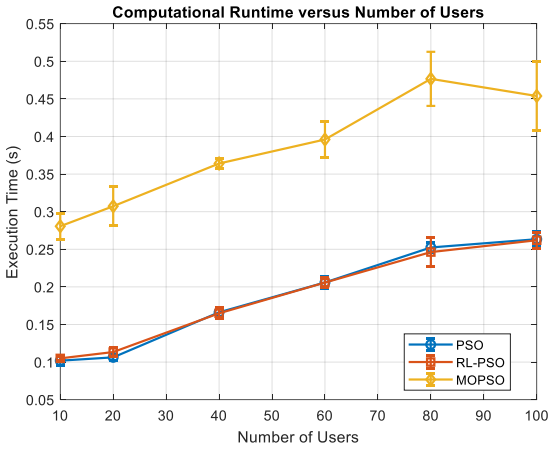


Figure 13. Computational runtime versus number of users.

As summarized in Table 4, MOPSO achieved the highest mean system capacity and energy efficiency among the three investigated algorithms, reaching 14.9195 ± 2.3843 bps/Hz and 1.1477 ± 0.1834 bits/J, respectively. These values correspond to improvements of 2.53% over PSO and 2.64% over RL-PSO for both metrics. However, MOPSO produced a lower Jain's fairness index of 0.5185 ± 0.0616, representing reductions of 13.08% and 13.74% relative to PSO and RL-PSO, respectively. QoS satisfaction remained broadly comparable across the three approaches, whereas MOPSO required a longer runtime of 0.1917 ± 0.1158 s.

Table 4. Statistical performance comparison of PSO, RL-PSO, and MOPSO.

Performance Metric	PSO Mean ± SD	RL-PSO Mean ± SD	MOPSO Mean ± SD	MOPSO vs PSO (%)	MOPSO vs RL-PSO (%)
Capacity (bps/Hz)	14.5508 ± 1.6101	14.5357 ± 1.9409	14.9195 ± 2.3843	+2.53	+2.64
Jain's Fairness Index	0.5966 ± 0.0547	0.6011 ± 0.0447	0.5185 ± 0.0616	−13.08	−13.74
Energy Efficiency (bits/J)	1.1193 ± 0.1239	1.1181 ± 0.1493	1.1477 ± 0.1834	+2.53	+2.64
QoS Satisfaction (%)	33.00 ± 6.27	32.30 ± 8.41	32.40 ± 6.22	−1.82	+0.31
Runtime (s)	0.0856 ± 0.0496	0.0760 ± 0.0299	0.1917 ± 0.1158	+123.87	+152.15

The statistical significance analysis presented in Table 5 further clarifies these differences. The Friedman and Wilcoxon tests indicate that the observed differences in capacity, energy efficiency, and QoS satisfaction are not statistically significant at $p < 0.05$. In contrast, statistically significant differences are observed for Jain's fairness index and runtime, with p-values below 0.05 for both the overall Friedman test and the pairwise Wilcoxon comparisons. Therefore, although MOPSO provides numerically higher capacity and energy efficiency, these improvements should be interpreted as performance trends rather than statistically significant gains under the considered experimental runs.

Table 5. Statistical significance analysis of PSO, RL-PSO, and MOPSO.

Performance Metric	Friedman Test (p)-value	MOPSO vs PSO Wilcoxon (p)-value	MOPSO vs RL-PSO Wilcoxon (p)-value	Significant at (p<0.05)?
Capacity	0.406570	0.556641	0.105469	No
Jain's Fairness Index	0.000500	0.001953	0.001953	Yes
Energy Efficiency	0.406570	0.556641	0.105469	No
QoS Satisfaction	0.292710	0.652344	0.773438	No
Runtime	0.000225	0.001953	0.001953	Yes

Table 6 further quantifies the performance improvements and associated trade-offs of MOPSO relative to PSO and RL-PSO. MOPSO achieved capacity improvements of 2.53% and 2.64% and equivalent gains in energy efficiency compared with PSO and RL-PSO, respectively. These gains were accompanied by reductions of 13.08% and 13.74% in Jain's fairness index. QoS satisfaction remained comparable, with MOPSO showing a 1.82% decrease relative to PSO and a marginal 0.31% increase over RL-PSO. In addition, MOPSO incurred a higher computational cost, with runtime increasing by 123.87% relative to PSO and 152.15% relative to RL-PSO. Collectively, these results demonstrate that the principal advantage of MOPSO lies in providing a balanced multi-objective solution with improved capacity and energy efficiency, while accepting trade-offs in fairness and computational complexity rather than uniformly outperforming the benchmark algorithms across all performance metrics.

Table 6. Performance improvement and trade-off of MOPSO relative to PSO and RL-PSO

Performance Metric	PSO	RL-PSO	MOPSO	MOPSO vs PSO (%)	MOPSO vs RL-PSO (%)	Interpretation
Capacity (bps/Hz)	14.5508	14.5357	14.9195	+2.53	+2.64	MOPSO higher
Jain's Fairness Index	0.5966	0.6011	0.5185	−13.08	−13.74	MOPSO lower
Energy Efficiency (bits/J)	1.1193	1.1181	1.1477	+2.53	+2.64	MOPSO higher
QoS Satisfaction (%)	33.00	32.30	32.40	−1.82	+0.31	Comparable
Runtime (s)	0.0856	0.0760	0.1917	+123.87	+152.15	MOPSO computationally more expensive

Table 7 quantifies the relative performance trade-offs of MOPSO against the PSO and RL-PSO benchmark algorithms. Compared with PSO, MOPSO improves Jain's fairness index by 26.7%, while its capacity, energy efficiency, and QoS satisfaction decrease by 41.4%, 43.3%, and 20.0%, respectively. In contrast, MOPSO consistently outperforms RL-PSO across all reported metrics, achieving improvements of 41.7% in capacity, 280.0% in fairness, 41.7% in energy efficiency, and 180.0% in QoS satisfaction. These results indicate that MOPSO provides a more balanced compromise among the competing optimization objectives. Although PSO achieves the highest capacity, energy efficiency, and QoS satisfaction, MOPSO attains the highest fairness index and substantially improves all considered performance measures relative to RL-PSO. Therefore, the main advantage of MOPSO lies in its ability to balance system throughput, energy utilization, fairness, and QoS rather than maximizing a single performance metric. The quantitative comparison presented in Table 7 further demonstrates the multi-objective characteristics of the proposed MOPSO approach. MOPSO achieves a Jain's fairness index of 0.38, compared with 0.30 for PSO and 0.10 for RL-PSO, corresponding to improvements of 26.7% and 280.0%, respectively. Similarly, MOPSO achieves a QoS satisfaction level of 28%, which is substantially higher than the 10% obtained by RL-PSO, although it remains below the 35% achieved by PSO. For capacity and energy efficiency, MOPSO reaches 17.0 bps/Hz and 1.7 bits/J, respectively. These values are 41.7% higher than those obtained by RL-PSO but 41.4% and 43.3% lower than the corresponding PSO results. The observed behavior reflects the inherent trade-off of multi-objective optimization, in which improvements in fairness and balanced resource allocation may be achieved at the expense of maximum throughput or energy efficiency. Overall, Table 7 shows that MOPSO provides a competitive and balanced solution, particularly when fairness and QoS are considered alongside capacity and energy efficiency.

Table 7. Quantitative Performance Comparison with Percentage Improvement

Metric	PSO	RL-PSO	MOPSO	MOPSO vs PSO	MOPSO vs RL-PSO
Capacity (bps/Hz)	29.0	12.0	17.0	−41.4%	+41.7%
Fairness (Jain Index)	0.30	0.10	0.38	+26.7%	+280.0%
Energy Efficiency (bits/J)	3.0	1.2	1.7	−43.3%	+41.7%
QoS Satisfaction (%)	35%	10%	28%	−20.0%	+180.0%

The average performance of the assessed algorithms and the percentage gains made by MOPSO over PSO and RL-PSO are shown in Table 7. When compared to both baseline approaches, it is found that MOPSO significantly improves fairness and QoS satisfaction. In particular, MOPSO shows its efficacy in guaranteeing equitable resource allocation by improving fairness by roughly 26.7% over PSO and 280% over RL-PSO. In a

similar vein, QoS satisfaction is 180% higher than RL-PSO, demonstrating MOPSO's capacity to service a higher percentage of consumers that satisfy minimum rate requirements. In contrast to PSO, MOPSO shows a drop in capacity and energy efficiency of 41.4% and 43.3%, respectively. Since MOPSO prioritizes several goals rather than maximizing a single performance indicator, this trade-off is to be expected. Overall, the findings demonstrate that MOPSO offers a sensible and workable approach, attaining significant improvements in QoS and fairness while preserving respectable levels of capacity and energy efficiency.

5. Conclusion

The results demonstrate the comparative performance of PSO, RL-PSO, and MOPSO for resource allocation in the HAP network under varying network conditions. MOPSO achieved the highest average capacity and energy efficiency in the baseline evaluation, while PSO and RL-PSO generally provided better Jain's fairness. The Pareto-front analysis further confirmed the inherent capacity–fairness trade-off, highlighting the main advantage of MOPSO as its ability to generate multiple non-dominated solutions rather than optimizing a single performance objective.

The trained RL-PSO showed improvements over its untrained counterpart in capacity, fairness, energy efficiency, and QoS satisfaction, demonstrating the benefit of incorporating reinforcement learning into PSO parameter adaptation. Scalability and robustness analyses showed that the algorithms remained operational as the number of users increased and under varying interference and traffic-load conditions. In particular, MOPSO maintained competitive capacity under demanding network conditions, although this was generally accompanied by lower fairness.

The sensitivity and ablation analyses further demonstrated that algorithm parameters and individual MOPSO objectives significantly influence the resulting performance trade-offs. Removing individual objectives could improve one metric while degrading others, supporting the need for the complete multi-objective formulation. Computational analysis showed that MOPSO incurred a higher execution time than PSO and RL-PSO because of its Pareto-based search and archive management; nevertheless, execution remained below one second across the investigated user sizes.

Overall, the results indicate that no single algorithm dominates across all performance metrics. PSO and RL-PSO offer advantages in fairness and computational efficiency, whereas MOPSO provides a stronger framework for simultaneously exploring competing capacity, energy-efficiency, fairness, and QoS objectives. Thus, the principal contribution of MOPSO is not universal superiority but its ability to provide a flexible set of trade-off solutions for multi-objective resource allocation in HAP networks.

Declaration

Supplementary materials are provided in this manuscript, such as Figures 1, 2, 3, and 4 in the Literature Review.

Sustainable Development Goals (SDGs)

- SDG 7: Affordable and Clean Energy (Reduce Power consumption in Wireless devices)
- SDG 9: Industry, Innovation and Infrastructure
- SDG 11: Sustainable cities and communities (Wireless Technologies)

Author Contribution

The **first author and the second author** performed the research and wrote the manuscript. The **first author, the second author, and the third author** contributed to the methodology and analysis. The **fourth, fifth, and six author** supervised the study. All authors reviewed and approved the final manuscript.

Funding

"This research received no external funding."

Conflicts of Interest

"The authors declare no conflict of interest."

References

[1] D. Tse and P. Viswanath, Fundamentals of Wireless Communication. Cambridge, U.K.: Cambridge University Press, 2005.
[2] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge University Press, 2004.

[3] A. A. Alhashmi, A. B. Alshammari, M. Abdullah, A. S. A. Alghawli, T. M. Alkhaldi, S. A. Ebad, K. N. R. Alharbi, and A. A. Darem, "Fair and scalable energy-efficient rate splitting multiple access in cognitive high altitude platform networks," Scientific Reports, vol. 16, Art. no. 20385, 2026, doi: 10.1038/s41598-026-50247-1.

[4] K. Mershad, H. Dahrouj, H. Sarieddeen, B. Shihada, T. Y. Al-Naffouri, and M.-S. Alouini, "Cloud-enabled high-altitude platform systems: Challenges and opportunities," Frontiers in Communications and Networks, vol. 2, Art. no. 716265, 2021, doi: 10.3389/frcmn.2021.716265.

[5] M.-C. Tsai, "High altitude platform networks (HAPNETs): Design, deployment, and resource management," M.S. thesis, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, 2020, doi: 10.25781/KAUST-I5451.

[6] G. Shanmugavel and M. S. Vasanthi, "Enhancing downlink NOMA performance: A Pareto optimization approach for superior power allocation," Journal of Engineering Research, vol. 14, no. 1, pp. 858–869, Mar. 2026, doi: 10.1016/j.jer.2025.08.021.

[7] Y. Chen, X. Tan, J. Jiang, X. Meng, Z. K. Hussain, J. Tu, H. Wang, Y. Wan, and Z. Sha, "3D spatial evolutionary particle swarm algorithm based emergency communication spatial deployment optimization," Geo-Spatial Information Science, vol. 28, no. 6, pp. 3178–3195, 2025, doi: 10.1080/10095020.2025.2472006.

[8] R. Su et al., "Resource Allocation for Network Slicing in 5G Telecommunication Networks: A Survey of Principles and Models," in IEEE Network, vol. 33, no. 6, pp. 172-179, Nov.-Dec. 2019, doi: 10.1109/MNET.2019.1900024.

[9] S. Latif, S. Akraam, T. Karamat, M. A. Khan, C. Altrjman, S. Mey, and Y. Nam, "An efficient Pareto optimal resource allocation scheme in cognitive radio-based Internet of Things networks," Sensors, vol. 22, no. 2, Art. no. 451, Jan. 2022, doi: 10.3390/s22020451.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. IEEE Int. Conf. Neural Networks (ICNN), vol. 4, 1995, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.

[11] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in Proc. IEEE Int. Conf. Evolutionary Computation (ICEC), 1998, pp. 69–73, doi: 10.1109/ICEC.1998.699146.

[12] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[13] W. Han, Y. Du, Y. Guo, J. Hao, and X. Zhang, "Computation offloading and resource allocation strategy considering user mobility in multi-UAV assisted semantic communication networks," Electronics, vol. 14, no. 20, Art. no. 4067, Oct. 2025, doi: 10.3390/electronics14204067.

[14] P. D. Gupta, "Obtaining the Pareto optimal front for multiobjective reinforcement learning based wireless routing," Master's degree project, Dept. Computer and Systems Sciences, Stockholm University, Stockholm, Sweden, 2024.

[15] Y. Xu, X. Tang, L. Huang, H. Ullah, and Q. Ning, "Multi-objective optimization for resource allocation in space–air–ground network with diverse IoT devices," Sensors, vol. 25, no. 1, Art. no. 274, Jan. 2025, doi: 10.3390/s25010274.

[16] A. U. Rehman et al., "Dynamic Energy Efficient Resource Allocation Strategy for Load Balancing in Fog Environment," in IEEE Access, vol. 8, pp. 199829-199839, 2020, doi: 10.1109/ACCESS.2020.3035181.

[17] R. Mishra, S. K. Jha, S. Prakash, and R. S. Rathore, "Selection of optimal cluster head using MOPSO and decision tree for cluster-oriented wireless sensor networks," Future Internet, vol. 17, no. 12, Art. no. 577, Dec. 2025, doi: 10.3390/fi17120577.

[18] Z. Ding, Y. Liu, J. Choi, Q. Sun, M. Elkashlan, C.-L. I, and H. V. Poor, "Application of non-orthogonal multiple access in LTE and 5G networks," IEEE Communications Magazine, vol. 55, no. 2, pp. 185–191, Feb. 2017, doi: 10.1109/MCOM.2017.1500657CM.

[19] T. Han and N. Ansari, "Powering mobile networks with green energy," IEEE Wireless Communications, vol. 21, no. 1, pp. 90–96, Feb. 2014, doi: 10.1109/MWC.2014.6757901.

[20] T. Erpek, T. J. O'Shea, Y. E. Sagduyu, Y. Shi, and T. C. Clancy, "Deep learning for wireless communications," arXiv:2005.06068, May 2020.

[21] S. M. Suleiman, D. S. Shuaibu, and S. A. Babale, "Investigating the effect of high-altitude platform positioning on latency and coverage of 4G cellular systems," International Journal of Communication Networks and Information Security, vol. 14, no. 1, pp. 37–42, 2022, doi: 10.17762/ijcnis.v14i1.5161.

[22] G. Karabulut Kurt, M. G. Khoshkholgh, S. Alfattani, A. Ibrahim, T. S. J. Darwish, M. S. Alam, H. Yanikomeroglu, and A. Yongacoglu, "A vision and framework for the high altitude platform station (HAPS) networks of the future," IEEE Communications Surveys & Tutorials, vol. 23, no. 2, pp. 729–779, 2021, doi: 10.1109/COMST.2021.3066905.

[23] S. Alfattani, W. Jaafar, H. Yanikomeroglu, and A. Yongaçoğlu, "Multimode high-altitude platform stations for next-generation wireless networks: Selection mechanism, benefits, and potential challenges," IEEE Vehicular Technology Magazine, vol. 18, no. 3, pp. 20–28, Sep. 2023, doi: 10.1109/MVT.2023.3289630.

[24] S. Liu, H. Dahrouj, and M.-S. Alouini, "Joint user association and beamforming in integrated satellite-HAPS-ground networks," IEEE Transactions on Vehicular Technology, vol. 73, no. 4, pp. 5162–5178, Apr. 2024, doi: 10.1109/TVT.2023.3329168.

[25] O. Abbasi, H. Yanikomeroglu, and G. Kaddoum, "Hemispherical antenna array architecture for high-altitude platform stations (HAPS) for uniform capacity provision," IEEE Transactions on Wireless Communications, vol. 23, no. 12, pp. 19489–19504, Dec. 2024, doi: 10.1109/TWC.2024.3484008.
[26] S. Ebrahimi, F. Bouali, and O. C. L. Haas, "Resource management from single-domain 5G to end-to-end 6G network slicing: A survey," IEEE Communications Surveys & Tutorials, vol. 26, no. 4, pp. 2836–2866, 2024, doi: 10.1109/COMST.2024.3390613.
[27] F. Hussain, S. A. Hassan, R. Hussain, and E. Hossain, "Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges," IEEE Communications Surveys & Tutorials, vol. 22, no. 2, pp. 1251–1275, 2020, doi: 10.1109/COMST.2020.2964534.
[28] S. Yin, M. Jin, H. Lu, G. Gong, W. Mao, G. Chen, and W. Li, "Reinforcement-learning-based parameter adaptation method for particle swarm optimization," Complex & Intelligent Systems, vol. 9, no. 5, pp. 5585–5609, 2023, doi: 10.1007/s40747-023-01012-8.
[29] O. Aoun, "Deep Q-network-enhanced self-tuning control of particle swarm optimization," Modelling, vol. 5, no. 4, pp. 1709–1728, 2024, doi: 10.3390/modelling5040089.
[30] R. Olivares, R. Soto, B. Crawford, V. Ríos, P. Olivares, C. Ravelo, S. Medina, and D. Nauduan, "A learning-based particle swarm optimizer for solving mathematical combinatorial problems," Axioms, vol. 12, no. 7, Art. no. 643, 2023, doi: 10.3390/axioms12070643.
[31] A. Zheng, Z. Zhang, W. Liu, J. Liu, Y. Xiao, and C. Li, "Dual cluster head optimization of wireless sensor networks based on multi-objective particle swarm optimization," Sensors, vol. 23, no. 1, Art. no. 231, 2023, doi: 10.3390/s23010231.