

Original Research

Received 07 June 2026

Revised 18 July 2026

Accepted 01 August 2026

Natural Resources for Human Health 2026, Vol. 6 (14s): 671–731

KEYWORDS:

6G wireless networks; Deep Reinforcement Learning; Mathematical Optimization; Integrated Sensing and Communication; Resource Allocation; Reconfigurable Intelligent Surfaces; Intelligent Robotics; Edge Computing; Energy-Efficient Communication; Artificial Intelligence.

Mathematics-Guided Deep Reinforcement Learning for Joint Resource Allocation and Integrated Sensing in Intelligent 6G-Enabled Robotic Systems

Dr M Pushpalatha¹, Dr. Lavanya Srinathuni², Dr. Shyam Kishore G.³, Jahangeer⁴, Patharla Kavitha⁵

¹Associate professor, Department of Electronics Communication and Engineering, Siddhartha Institute of Engineering and Technology, Ibrahimpatnam, Hyderabad, Telangana, India

²Assistant Professor, Department of Humanities and Sciences (Mathematics), Visvesvaraya College of Engineering and Technology, Hyderabad, Telangana, India

³Assistant Professor, Department of Electronics Communication and Engineering, School of Engineering and Technology, Kaveri University, Telangana, India

⁴Assistant Professor, Department of Humanities and Sciences (Mathematics), Teegala Krishna Reddy College of Engineering and Technology, Hyderabad, Telangana, India

⁵Assistant Professor, Department of Electrical and Electronics Engineering, TRR College of Technology, Hyderabad, Telangana, India

Corresponding author: Dr. Dr M Pushpalatha — pushpafile@gmail.com

ABSTRACT: Sixth-generation (6G) wireless networks are expected to support highly dynamic and heterogeneous applications that require simultaneous improvements in communication efficiency, sensing capability, energy utilization, and computational intelligence. These requirements become particularly challenging in robotic environments, where the mobility of autonomous platforms introduces rapid variations in channel conditions, traffic demands, sensing requirements, and energy consumption. Conventional optimization methods can provide mathematically tractable resource-allocation solutions, but their computational complexity can limit their applicability to rapidly changing network environments. In contrast, deep reinforcement learning (DRL) can learn adaptive resource-allocation policies, but unconstrained learning may produce decisions that violate communication, energy, latency, or sensing requirements.

This paper proposes a mathematics-guided deep reinforcement learning framework for joint resource allocation and integrated sensing in 6G-enabled robotic systems. The proposed framework formulates communication, sensing, energy, interference, and latency requirements as a multi-objective constrained optimization problem. The resulting mathematical formulation is incorporated into the DRL framework through a constraint-aware state representation and reward function. The proposed approach jointly determines communication resources, transmit power, spectrum allocation, reconfigurable intelligent surface configuration, and computational resources while accounting for robotic mobility and sensing requirements. An integrated sensing and communication model is employed to establish the interaction between wireless communication performance and robotic sensing demands. The framework is evaluated through simulation under varying user density, channel conditions, robotic mobility, and network resource availability. Performance is assessed using throughput, latency, energy consumption, spectral efficiency, sensing performance, convergence behaviour, and computational complexity. The proposed framework provides a unified mathematical and

learning-based approach for adaptive resource management in intelligent 6G environments and demonstrates the potential of combining constrained optimization with deep reinforcement learning for communication-intensive robotic applications.

1. INTRODUCTION

The evolution of wireless communication beyond fifth-generation (5G) systems is increasingly focused on networks that can provide not only high-capacity data transmission but also sensing, computation, localization, intelligence, and autonomous decision-making. Sixth-generation (6G) networks are consequently being investigated as highly intelligent and heterogeneous communication infrastructures in which communication, computation, sensing, and artificial intelligence (AI) are closely integrated. In this context, AI is expected to move beyond an application-level optimization tool and become an integral component of network operation, enabling adaptive resource management and autonomous control under highly dynamic network conditions [1], [2]. Recent studies on knowledge-driven learning for wireless optimization similarly emphasize the need to incorporate physical and mathematical knowledge into learning models rather than treating wireless optimization as a purely data-driven task [3].

One of the important technological directions associated with 6G is integrated sensing and communication (ISAC), in which the same wireless infrastructure and, to a considerable extent, the same radio-frequency resources are used simultaneously for communication and environmental sensing. By sharing hardware, spectrum, signal-processing resources, and network infrastructure, ISAC can reduce functional duplication while enabling applications requiring both connectivity and environmental awareness [4]. Potential applications include intelligent transportation, industrial automation, localization, human activity recognition, autonomous systems, and robotic operations. However, the coexistence of sensing and communication introduces a fundamental resource-management problem because communication-oriented objectives such as throughput and spectral efficiency can conflict with sensing-oriented requirements such as estimation accuracy, localization precision, and sensing coverage [4], [5].

The challenge becomes more pronounced when ISAC is deployed in mobile robotic environments. Autonomous robots continuously change their positions, orientations, velocities, and communication conditions. Consequently, the wireless channel, required data rate, sensing demand, computational load, and energy consumption can vary with time. Industrial and autonomous robotic applications therefore require communication systems capable of making rapid decisions while simultaneously satisfying stringent latency, reliability, energy, and sensing requirements. Recent reviews of deep-learning-based ISAC explicitly identify industrial robotics and other mobile applications as important use cases for learning-enabled sensing and communication systems [6].

Reconfigurable intelligent surfaces (RISs) provide another important enabling technology for this environment. An RIS can modify the wireless propagation environment by controlling the phase response of a large number of reflecting elements, thereby providing an additional degree of freedom for wireless channel optimization [7]. RIS-assisted architectures have consequently attracted considerable attention for beyond-5G and 6G systems, including applications involving coverage enhancement, energy efficiency, beamforming, localization, and sensing [7], [8]. However, introducing RIS configuration into an ISAC system further increases the dimensionality and non-convexity of the resource-allocation problem. Joint optimization of transmit power, bandwidth, beamforming, RIS configuration, sensing requirements, and computational resources can therefore become difficult to solve repeatedly using conventional optimization techniques.

Mathematical optimization remains an essential approach for addressing such resource-management problems. Conventional techniques can provide theoretically interpretable solutions and explicitly incorporate physical constraints, including transmit-power limits, bandwidth limitations, latency requirements, interference constraints, and energy budgets. Nevertheless, many wireless optimization problems are non-convex, high-dimensional, and dependent on rapidly varying channel and traffic conditions. The computational cost of repeatedly solving such problems may therefore become problematic when decisions must be generated at short time intervals. This issue is particularly relevant to ISAC systems, where the resource-management problem must simultaneously account for communication and sensing objectives. A recent comprehensive survey of learning algorithms for 6G ISAC identifies beamforming, waveform design, spectrum allocation, time allocation, power allocation, sensing, and signal classification among the major optimization and learning problems in this area [9].

Deep reinforcement learning (DRL) provides a complementary approach to conventional optimization. Unlike a conventional optimization procedure that may need to solve a new mathematical problem whenever the system state changes, a trained DRL agent can learn a policy that maps observed system states to resource-allocation decisions. DRL has previously been investigated for communication and networking problems involving resource sharing, data offloading, network access, traffic control, and connectivity management [10]. It has also been applied to wireless resource allocation and non-convex power-control problems, demonstrating the potential of learning-based decision-making for large and dynamic state and action spaces [11].

Recent research has extended DRL toward ISAC and 6G-specific resource management. For example, a recent IEEE study surveys learning-based solutions for beamforming, waveform design, spectrum allocation, power allocation, and wireless sensing in 6G and beyond [9]. A 2026 survey further categorizes RL/DRL approaches for 6G ISAC according to value-based, policy-based, and hybrid methods and discusses their use in RIS-assisted, vehicular, UAV, satellite, and other dynamic environments [12]. Similarly, recent research has investigated DRL for joint channel, power, and bandwidth allocation in ISAC-oriented vehicular environments [13]. These developments demonstrate the growing relevance of learning-based resource management but also indicate that the problem is moving toward increasingly complex multi-objective and multi-domain formulations.

Despite this progress, three issues remain particularly important. First, a significant portion of existing DRL-based wireless optimization treats the learning model as the primary decision mechanism, while the physical and mathematical constraints of the underlying communication system may be represented only indirectly through reward design or penalty terms. Second, communication and sensing objectives are often considered jointly, but their interaction with robotic mobility, energy consumption, and computational requirements is not always incorporated into a single optimization framework. Third, the growing number of decision variables associated with RIS configuration, power allocation, bandwidth allocation, edge computing, sensing, and mobility creates a need for learning frameworks that can exploit domain knowledge while retaining the adaptability of DRL. The broader literature on knowledge-driven deep learning for intelligent wireless optimization has identified the integration of domain knowledge and neural learning as an important direction for reliable and efficient 6G optimization [3].

These observations motivate the development of a **mathematics-guided deep reinforcement learning (MG-DRL)** framework for intelligent 6G-enabled robotic ISAC systems. Instead of replacing mathematical optimization with a black-box learning process, the proposed framework uses the mathematical formulation to define the system objectives, constraints, state variables, and reward structure of the DRL agent. Communication performance, sensing performance, latency, energy consumption, interference, and robotic mobility are therefore represented within a unified decision-making framework. The resulting architecture is intended to combine the interpretability and constraint-awareness of mathematical optimization with the adaptive decision-making capability of DRL.

The proposed framework considers a 6G wireless environment containing a base station, an RIS-assisted propagation path, an edge-computing facility, and a mobile robotic platform. The communication and sensing processes are jointly represented, while the robotic platform introduces mobility-dependent variations in channel conditions and resource requirements. The optimization problem determines appropriate communication and computational resources while accounting for sensing requirements and system constraints. The resulting decision problem is subsequently transformed into a sequential learning problem in which the DRL agent learns resource-allocation policies from the observed network state.

The conceptual structure of the proposed research is illustrated in **Figure 1**.

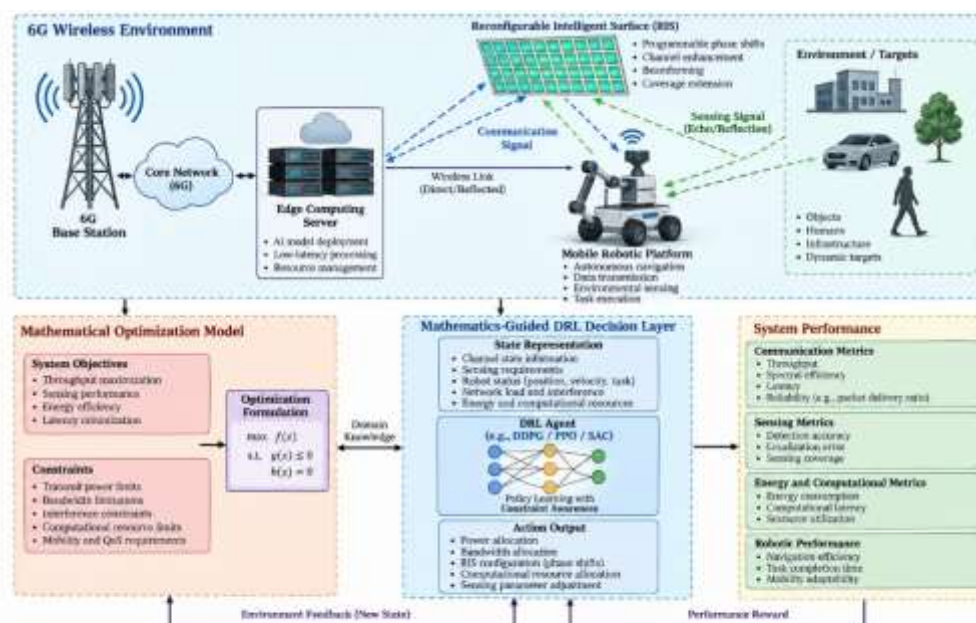


Figure 1. Conceptual architecture of the proposed mathematics-guided DRL framework for 6G-enabled robotic ISAC. The framework integrates the wireless communication layer, RIS-assisted propagation environment, sensing layer, robotic mobility, edge-computing layer, mathematical optimization model, and DRL decision layer.

Figure 1. Conceptual architecture of the proposed mathematics-guided DRL framework for 6G-enabled robotic ISAC. The framework integrates the wireless communication layer, RIS-assisted propagation environment, sensing layer, robotic mobility, edge-computing layer, mathematical optimization model, and DRL decision layer.

The principal research gap addressed in this work is therefore not simply the application of DRL to 6G. Rather, the study investigates how **explicit mathematical knowledge of communication, sensing, energy, latency, interference, and mobility constraints** can be incorporated into a **DRL-based resource-management framework** for a dynamic robotic ISAC environment.

1.1 Research Gap

The relationship between existing research directions and the proposed study is summarized in Table 1.

Table 1. Comparison of representative research directions and the proposed framework

Research direction	AI/ML	Mathematical optimization	ISAC	6G	RIS	Mobility/robotics	Joint multi-objective resource management
Conventional wireless optimization	–	✓	Partial	Partial	✓	Limited	Partial
DRL for wireless resource management	✓	Partial	–/Partial	✓	Partial	✓	✓
Learning-based 6G-ISAC	✓	Partial	✓	✓	Partial	✓	✓
RIS-assisted 6G optimization	✓/Partial	✓	Partial	✓	✓	Limited	✓
AI-enabled robotic communication	✓	Partial	Partial	✓	Limited	✓	✓

Research direction	AI/ML	Mathematical optimization	ISAC	6G	RIS	Mobility/robotics	Joint multi-objective resource management
Proposed MG-DRL framework	✓	✓	✓	✓	✓	✓	✓

Note: The table represents the research dimensions addressed by the respective literature streams rather than a ranking of individual studies.

Existing research clearly establishes the individual importance of DRL, RIS, ISAC, mathematical optimization, and 6G resource management. However, the literature also shows that these components are often investigated through partially separated formulations. Recent surveys specifically identify the need to address dynamic resource management, sensing–communication trade-offs, high-dimensional control spaces, and adaptive learning in 6G ISAC systems [9], [12].

1.2 Research Objectives

The objectives of this study are:

1. To formulate a mathematical model for joint communication and sensing resource allocation in a RIS-assisted 6G robotic environment.
2. To incorporate communication, latency, energy, interference, sensing, and robotic mobility requirements into a unified multi-objective optimization formulation.
3. To develop a mathematics-guided DRL framework in which the optimization model directly influences the state representation, reward formulation, and constraint-handling mechanism.
4. To investigate the joint allocation of transmit power, bandwidth, RIS configuration, and computational resources under dynamically varying network conditions.
5. To evaluate the proposed framework using communication, sensing, energy, latency, convergence, and computational-complexity metrics.
6. To determine the contribution of individual framework components through comparative and ablation analyses.

1.3 Main Contributions

The main contributions of this study are summarized as follows:

(i) A unified mathematical formulation:

A multi-objective optimization model is developed that jointly considers communication throughput, latency, energy consumption, interference, sensing performance, and computational-resource requirements in a 6G-enabled robotic environment.

(ii) Mathematics-guided DRL architecture:

Rather than treating the DRL agent as an unconstrained black-box controller, the proposed architecture embeds mathematical and physical knowledge into the learning process through state construction, objective normalization, constraint-aware reward design, and feasible-action handling.

(iii) Joint communication–sensing–robotic resource management:

The framework integrates communication and sensing requirements with mobility-dependent resource variations introduced by a robotic platform. This enables the study of resource allocation under changing wireless and application conditions.

(iv) RIS-assisted intelligent wireless environment:

RIS configuration is incorporated as an additional optimization variable, allowing the learning agent to adapt the propagation environment together with conventional communication resources.

(v) Multi-dimensional evaluation:

The framework will be evaluated using throughput, latency, energy efficiency, spectral efficiency, sensing performance, convergence behaviour, and computational complexity rather than relying on a single performance metric.

(vi) Ablation and baseline analysis:

The contribution of mathematical guidance, RIS optimization, sensing awareness, and mobility information will be separately investigated through controlled ablation experiments and comparison with conventional optimization and DRL baselines.

The overall relationship between the research problem, mathematical formulation, learning framework, and evaluation methodology is shown in **Figure 2**.

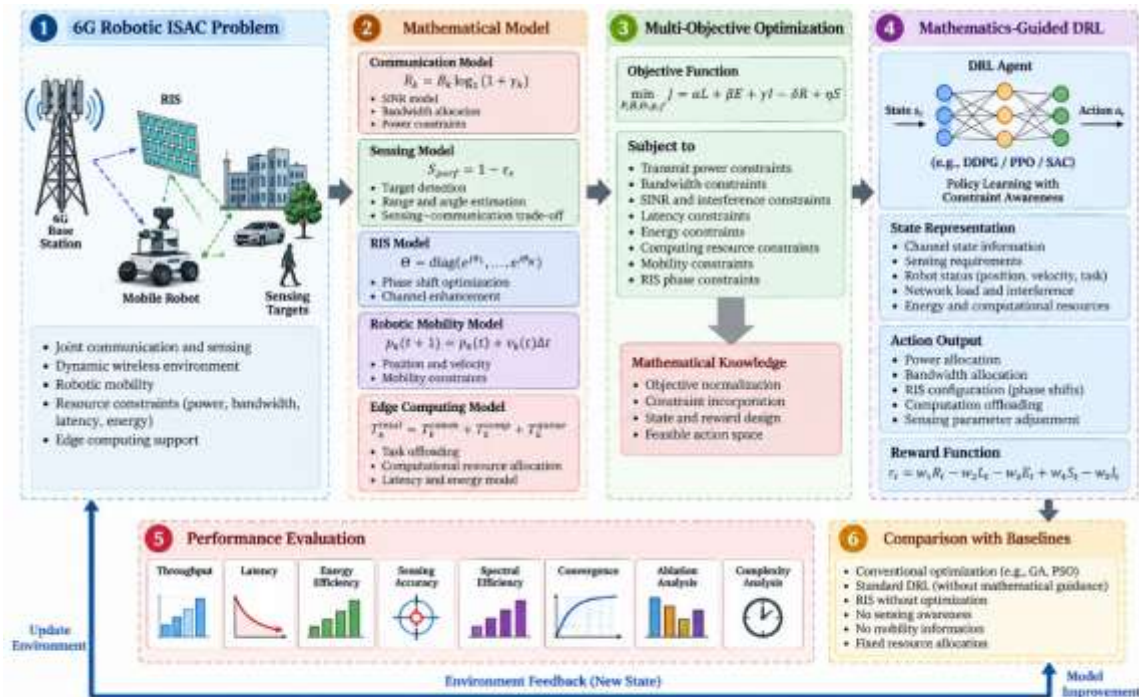


Figure 2. Research methodology for the proposed mathematics-guided DRL framework. The figure illustrates the workflow from the 6G robotic ISAC problem formulation to mathematical modeling, DRL-based decision making, and performance evaluation with feedback to the environment.

Figure 2. Research methodology linking the 6G robotic ISAC problem, mathematical optimization formulation, mathematics-guided DRL model, baseline methods, and performance evaluation.

The remainder of this paper is organized as follows. Section 2 reviews recent developments in AI-enabled 6G networks, ISAC, RIS-assisted communication, mathematical optimization, and DRL-based resource management and identifies the specific research gap addressed by this study. Section 3 presents the 6G-enabled robotic ISAC system model. Section 4 formulates the joint multi-objective optimization problem and defines the associated constraints. Section 5 presents the proposed mathematics-guided DRL framework and learning procedure. Section 6 describes the simulation methodology, parameter settings, and benchmark algorithms. Section 7 presents and discusses the experimental results. Section 8 provides complexity and ablation analyses. Finally, Section 9 concludes the study and discusses future research directions.

2. RELATED WORK AND RESEARCH GAP

2.1 Artificial Intelligence for 6G Wireless Networks

The transition toward 6G is associated with increasingly heterogeneous network environments involving massive connectivity, highly dynamic traffic, distributed computing, sensing, localization, and autonomous services. Conventional optimization techniques remain important because they provide mathematically interpretable solutions and allow explicit incorporation of system constraints. However, the scale and dynamic nature of emerging 6G networks can make repeated optimization computationally expensive. In contrast, deep learning (DL) can provide rapid inference after training, although purely data-

driven approaches can suffer from limited interpretability, data requirements, and inadequate physical constraint handling [1], [2].

Recent research has therefore moved toward **knowledge-driven learning**, in which communication-theoretic knowledge, optimization structures, physical constraints, and expert rules are incorporated into learning models. Sun et al. describe this direction as a means of combining the advantages of model-based optimization and data-driven learning for complex 6G optimization problems [1], [2]. Their recent comprehensive survey categorizes knowledge integration according to the point at which domain knowledge enters the learning pipeline, including model selection, model customization, architecture construction, loss-function design, and hyperparameter configuration [1].

This direction is directly relevant to the present study because the proposed framework does not treat the wireless environment as an unconstrained black-box learning problem. Instead, communication and sensing equations, resource limits, energy constraints, and latency requirements are used to guide the learning process.

2.2 Integrated Sensing and Communication in 6G

ISAC is one of the major technological directions for future wireless systems because it enables communication and sensing functions to share spectrum, infrastructure, signal processing, and potentially hardware. This integration is particularly relevant to applications in which a wireless device must simultaneously exchange information and perceive its surrounding environment.

The increasing integration of AI with ISAC has produced research on beamforming, waveform design, spectrum allocation, power allocation, channel estimation, sensing, localization, and signal processing. A recent comprehensive survey of learning algorithms for 6G-ISAC identifies these problems as important areas for machine-learning-based optimization and emphasizes the need for adaptive learning under dynamic network conditions [3].

More recent work also demonstrates the growing role of data-driven methods in ISAC. A 2025 review reports applications of DL and DRL to intelligent vehicular networks, autonomous mobility, UAV communications, radar sensing, localization, millimeter-wave and terahertz communication, and adaptive beamforming [4].

However, the simultaneous optimization of communication and sensing objectives remains challenging. Increasing communication throughput may require resource allocation decisions that conflict with sensing accuracy or sensing coverage. Therefore, a practical intelligent ISAC controller should account for these competing objectives rather than optimize communication performance independently.

2.3 Deep Reinforcement Learning for Wireless Resource Allocation

DRL has become an important approach for sequential wireless resource-management problems because an agent can learn a policy through interaction with the network environment. Rather than solving a complete optimization problem independently for every channel realization, the trained agent can infer an action from the current state.

Several studies have demonstrated this concept for wireless resource allocation. In particular, DRL has been used for dynamic allocation of communication channels, transmit power, and bandwidth in ISAC-oriented vehicular networks. One recent study proposed a joint channel, power, and bandwidth allocation framework using DDPG and DQN components and reported improvements in spectral efficiency and communication rates relative to its benchmark schemes [5].

DRL has also been combined with integrated sensing, communication, and computing. Cai and He proposed a DRL-based resource-allocation strategy in which the original problem was decomposed and addressed using convex optimization and DRL to reduce latency and energy-related costs [6].

These studies demonstrate an important principle for the present work: **DRL and mathematical optimization do not necessarily need to be competing methodologies**. They can instead be combined so that mathematical optimization provides structure and constraints while DRL provides adaptive decision-making.

2.4 RIS-Assisted Intelligent 6G Networks

RIS technology provides an additional mechanism for controlling wireless propagation. By dynamically adjusting the phase response of multiple reflecting elements, an RIS can modify the effective wireless channel and potentially improve coverage, energy efficiency, beamforming performance, and sensing capability.

The integration of RIS with AI-based optimization introduces additional decision variables because the controller must determine not only conventional communication parameters but also RIS configuration. The resulting problem can become high-dimensional, non-convex, and time-varying.

Recent studies have therefore investigated DRL for RIS-assisted resource management. For example, a 2024 study proposed a multi-agent DRL strategy for energy-efficient resource management in an RIS-assisted 6G smart-city environment, jointly considering resource allocation and RIS phase optimization [7]. The study demonstrates the potential of decentralized and coordinated DRL approaches when the state and action spaces become large.

Other recent work has investigated hierarchical DRL for multi-RIS and multi-operator environments, demonstrating that hierarchical decomposition can address the dimensionality associated with joint RIS allocation, beamforming, phase-shift control, and user association [8].

Although these approaches are relevant to the proposed framework, RIS optimization alone does not address the complete communication–sensing–robotic resource-management problem considered in this study.

2.5 AI-Enabled Resource Management for Integrated Sensing, Communication and Computing

The emergence of edge computing adds another dimension to intelligent wireless optimization. In robotic and autonomous applications, raw sensing information can be processed locally or transmitted to an edge server. This creates a trade-off between communication energy, computation energy, processing latency, and wireless resource consumption.

The joint consideration of sensing, communication, and computing has therefore become an important research direction. Cai and He formulated an integrated sensing, communication, and computing resource-management problem and combined convex optimization with DRL to address latency and energy consumption [6].

This research direction is particularly important for the proposed robotic architecture because the robot is not merely a wireless terminal. It simultaneously performs sensing, communication, task execution, and potentially computation offloading. Consequently, the wireless resource-allocation policy can affect both communication performance and robotic task performance.

2.6 AI and Autonomous Robotic Systems

Robotic platforms introduce mobility and application-specific constraints that are not present in static wireless-resource-allocation problems. A moving robot may experience rapidly changing channel conditions, varying sensing requirements, changing data-generation rates, and time-dependent energy consumption.

The use of learning-enabled ISAC is increasingly relevant to autonomous mobility and robotic applications. Recent reviews identify autonomous mobility, intelligent transportation, and other dynamic applications among the important application domains for AI-enabled ISAC [4].

However, the robotic component of a wireless system should not be treated solely as an additional user. Robot position, velocity, task state, and sensing requirements can influence the network state itself. This motivates incorporating robotic mobility information directly into the state representation of the learning agent.

2.7 Limitations of Existing Approaches

The reviewed literature establishes several important research directions:

1. **Knowledge-driven learning** addresses the limitations of purely data-driven approaches by incorporating communication-specific domain knowledge [1], [2].
2. **Learning-enabled ISAC** addresses the complexity of jointly optimizing communication and sensing functions [3], [4].
3. **DRL-based resource allocation** enables adaptive decisions under changing wireless conditions [5], [6].
4. **RIS-assisted DRL** provides additional control over the wireless propagation environment [7], [8].

5. **Integrated sensing, communication, and computing** incorporates the interaction between wireless and computational resources [6].

Nevertheless, these research directions do not necessarily provide a unified framework incorporating all of the following simultaneously:

- mathematical formulation of communication and sensing objectives;
- explicit physical resource constraints;
- DRL-based sequential decision-making;
- RIS configuration;
- communication–sensing trade-offs;
- edge-computing resources;
- robotic mobility;
- energy consumption;
- latency;
- and multi-objective performance optimization.

This observation motivates the research framework developed in this paper.

2.8 Comparative Analysis of Existing Literature

Table 2. Comparative analysis of representative research directions

Study research direction /	DRL	Mathematical optimization	6G	ISAC	RIS	Computing	Mobility	Robotic application	Joint multi-objective formulation
Knowledge-driven DL for 6G optimization [1]	Partial	✓	✓	Partial	Partial	✓	Partial	–	✓
Knowledge-driven DL paradigms [2]	Partial	✓	✓	Partial	Partial	✓	Partial	–	✓
Learning algorithms for 6G-ISAC [3]	✓	Partial	✓	✓	✓/Partial	Partial	✓	Partial	✓
Data-driven ISAC review [4]	✓	Partial	✓	✓	✓/Partial	Partial	✓	✓/Partial	✓
DRL for IoV-ISAC [5]	✓	Partial	✓	✓	–	Partial	✓	–	✓
DRL for ISCC [6]	✓	✓	✓/Partial	✓	–	✓	Partial	–	✓

Study research direction /	DRL	Mathematical optimization	6G	ISAC	RIS	Computing	Mobility	Robotic application	Joint multi-objective formulation
RIS-assisted multi-agent DRL [7]	✓	Partial	✓	–	✓	Partial	✓	–	✓
Hierarchical DRL for multi-RIS systems [8]	✓	Partial	✓	–	✓	Partial	✓	–	✓
Proposed framework	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: “Partial” indicates that the corresponding dimension is discussed or incorporated in a limited form rather than constituting the principal focus of the cited work. The table is a thematic comparison and is not intended as a ranking of the studies.

The comparison indicates that the proposed study occupies the intersection of several active research areas rather than simply extending a single existing DRL resource-allocation model.

2.9 Specific Research Gap

Based on the reviewed literature, the specific gap addressed in this study can be stated as follows:

There is a need for a unified resource-management framework that combines explicit mathematical knowledge of wireless communication and sensing constraints with adaptive DRL while simultaneously accounting for RIS configuration, edge-computing resources, robotic mobility, energy consumption, and latency in a 6G-enabled ISAC environment.

The distinction is important. Existing DRL approaches demonstrate that learning can be used to make dynamic resource-allocation decisions [5], while knowledge-driven learning demonstrates the value of incorporating domain knowledge into intelligent wireless optimization [1], [2]. Similarly, RIS-assisted DRL studies demonstrate the feasibility of learning-based control of programmable wireless environments [7], [8]. The proposed research combines these principles into a single constrained decision framework.

2.10 Research Hypothesis

The study is based on the following hypothesis:

A DRL framework guided by explicit mathematical and physical constraints can achieve more consistent multi-objective resource management than an unconstrained DRL formulation under dynamically varying 6G robotic ISAC conditions.

This hypothesis will be tested experimentally rather than assumed. The evaluation will therefore compare the proposed framework against conventional optimization and learning-based baselines using common simulation conditions.

The experimental design will also include an ablation analysis to determine whether improvements are attributable to mathematical guidance, RIS optimization, sensing awareness, robotic mobility information, or their combination.

2.11 Summary of the Research Position

The literature indicates a clear progression from conventional mathematical optimization toward data-driven learning and, more recently, toward knowledge-guided intelligent optimization. At the same time, ISAC, RIS, edge computing, and autonomous robotics are increasing the dimensionality of future wireless resource-management problems. The proposed work follows this progression by treating mathematical optimization and DRL as complementary components.

Accordingly, the next section develops the **6G-enabled robotic ISAC system model**, including the wireless architecture, RIS-assisted channel, communication model, sensing model, robotic mobility model, and edge-computing model. These components will subsequently be used to formulate the multi-objective optimization problem in Section 4.

3. SYSTEM MODEL

3.1 Overview of the Considered 6G-Enabled Robotic ISAC Network

Consider a 6G-enabled intelligent wireless environment consisting of a multi-antenna base station (BS), an edge-computing server, a reconfigurable intelligent surface (RIS), a mobile robotic platform, and a set of sensing targets. The BS provides wireless connectivity to the robotic platform while simultaneously transmitting sensing waveforms toward objects in the surrounding environment. The RIS is deployed between the BS and the robotic region to improve propagation conditions and provide additional control over the wireless channel.

The proposed architecture follows the integrated sensing and communication paradigm, in which communication and sensing share radio resources and system infrastructure. ISAC is increasingly considered a key component of beyond-5G and 6G systems because it enables wireless networks to provide both connectivity and environmental awareness. Recent studies emphasize that joint resource management becomes essential because communication and sensing compete for power, spectrum, beamforming degrees of freedom, and processing resources [3], [9].

The considered system contains the following entities:

1. **6G base station:** An N_t -antenna BS transmits communication and sensing signals.
2. **RIS:** An RIS containing N programmable reflecting elements modifies the wireless propagation environment.
3. **Robotic platform:** A mobile autonomous robot equipped with a communication receiver and sensing capability.
4. **Edge-computing server:** Computational resources are available near the wireless access point for low-latency processing and task offloading.
5. **Sensing targets:** Objects or environmental features observed by the robotic platform and/or the ISAC infrastructure.
6. **Mathematics-guided DRL controller:** The intelligent controller observes the network state and determines resource-allocation actions subject to system constraints.

The overall architecture is illustrated in Figure 3.

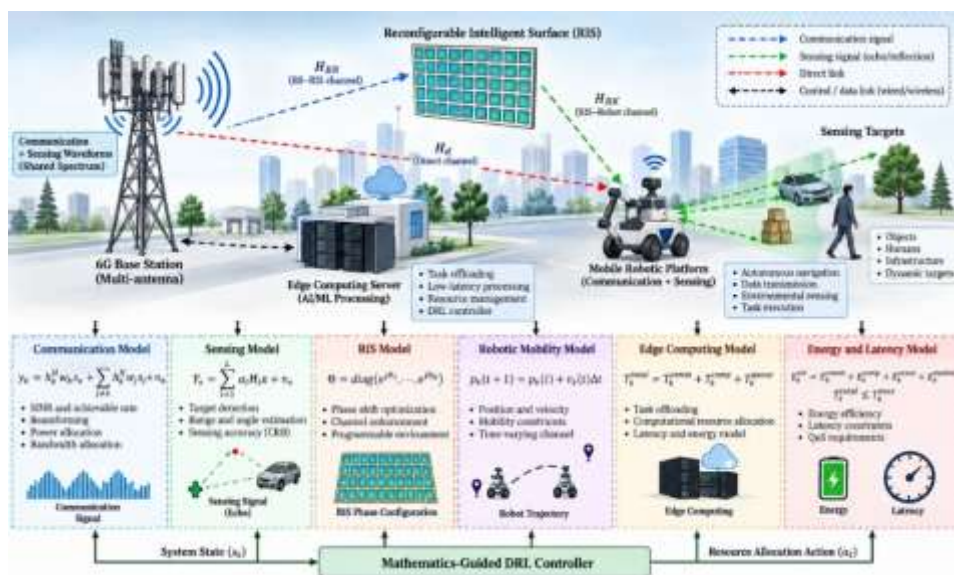


Figure 3. System model of the 6G-enabled robotic ISAC environment consisting of the multi-antenna BS, RIS, edge-computing server, mobile robotic platform, and sensing targets.

The mathematical model is designed so that communication, sensing, computation, and robotic mobility are represented within a common decision-making framework. This is consistent with recent ISAC research, where resource allocation and beamforming are formulated jointly because the same physical resources influence both communication and sensing performance [3], [14].

3.2 Network and Antenna Model

Let the 6G BS be equipped with N_t transmit antennas and N_r receive antennas. The BS communicates with K mobile robotic or communication terminals while simultaneously sensing L targets.

The transmitted signal is represented as

$$x[t] = W_c[t]s_c[t] + W_s[t]s_s[t], \quad (1)$$

where

$$W_c[t] \in \mathbb{C}^{N_t \times K}$$

is the communication beamforming matrix,

$$W_s[t] \in \mathbb{C}^{N_t \times L}$$

is the sensing beamforming matrix,

$$s_c[t] \in \mathbb{C}^K$$

is the communication-symbol vector, and

$$s_s[t] \in \mathbb{C}^L$$

is the sensing-signal vector.

The total transmit-power constraint is given by

$$\mathbb{E}[\|x[t]\|^2] \leq P_{\max}, \quad (2)$$

where $P_{\max}$ denotes the maximum available BS transmit power.

The use of separate communication and sensing beamforming components follows the mathematical structure commonly adopted in RIS-assisted ISAC research [14], [15].

3.3 RIS-Assisted Propagation Model

An RIS consisting of N reflecting elements is introduced to modify the wireless propagation environment. For a passive RIS, the reflection matrix can be expressed as

$$\Theta = \text{diag}(\beta_1, e^{j\theta_1}, \beta_2, e^{j\theta_2}, \dots, \beta_N, e^{j\theta_N}), \quad (3)$$

where β_n is the reflection amplitude of the n -th RIS element and θ_n represents its adjustable phase shift.

For an ideal lossless passive RIS,

$$\beta_n = 1,$$

and therefore

$$\Theta = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \dots, e^{j\theta_N}). \quad (4)$$

The RIS phase constraint becomes

$$0 \leq \theta_n < 2\pi, \quad n = 1, \dots, N. \quad (5)$$

For practical implementations with B -bit phase quantization,

$$\theta_n \in \left\{0, \frac{2\pi}{2^B}, \frac{2(2\pi)}{2^B}, \dots, \frac{(2^B - 1)2\pi}{2^B}\right\}. \quad (6)$$

The effective BS-to-robot channel can be written as

$$h_{k,\text{eff}}^H = h_{d,k}^H + h_{r,k}^H \Theta G, \quad (7)$$

where

- $h_{d,k}$ is the direct BS-to-robot channel,
- G represents the BS-to-RIS channel,
- $h_{r,k}$ represents the RIS-to-robot channel.

Thus, the RIS provides an additional controllable propagation path.

RIS-assisted ISAC has received substantial attention because programmable surfaces can simultaneously influence communication and sensing performance, although practical issues such as channel estimation, hardware imperfections, synchronization, and optimization complexity remain important [13], [14].

3.4 Communication Signal Model

For the k -th robotic terminal, the received communication signal can be expressed as

$$y_k = h_{k,\text{eff}}^H w_k s_k + \sum_{j \neq k} h_{k,\text{eff}}^H w_j s_j + n_k, \quad (8)$$

where w_k is the beamforming vector assigned to terminal k , s_k is the corresponding communication symbol, and n_k represents additive noise.

The received signal-to-interference-plus-noise ratio (SINR) is

$$\gamma_k = \frac{|h_{k,\text{eff}}^H w_k|^2}{\sum_{j \neq k} |h_{k,\text{eff}}^H w_j|^2 + \sigma_k^2}, \quad (9)$$

where σ_k^2 denotes the receiver noise power.

The achievable communication rate for terminal k is then

$$R_k = B_k \log_2(1 + \gamma_k), \quad (10)$$

where B_k is the bandwidth allocated to terminal k .

The aggregate communication throughput is

$$R_{\text{sum}} = \sum_{k=1}^K R_k. \quad (11)$$

The bandwidth allocation is subject to

$$\sum_{k=1}^K B_k \leq B_{\text{tot}}, \quad (12)$$

where B_{tot} denotes the total available communication bandwidth.

3.5 Integrated Sensing Model

The sensing functionality enables the wireless infrastructure to detect and estimate environmental targets while simultaneously supporting communication.

Let the l -th sensing target have a complex reflection coefficient α_l , range r_l , and direction φ_l . The received sensing signal can be represented generally as

$$y_s = \sum_{l=1}^L \alpha_l H_l x + n_s, \quad (13)$$

where H_l denotes the effective sensing channel associated with target l , and n_s represents sensing noise.

The sensing performance can be characterized using a normalized sensing-error metric. Let

$$\varepsilon_s = \omega_r \varepsilon_r + \omega_\varphi \varepsilon_\varphi, \quad (14)$$

where

- ε_r is the normalized range-estimation error,
- ε_φ is the normalized angular-estimation error,
- ω_r and ω_φ are weighting coefficients.

The sensing objective can therefore be represented as

$$S_{\text{perf}} = 1 - \varepsilon_s. \quad (15)$$

Higher values of S_{perf} indicate improved sensing performance.

A more rigorous implementation can subsequently replace the generic sensing-error formulation with the Cramér–Rao bound (CRB) or another estimator-specific metric. This is consistent with recent RIS-assisted ISAC optimization studies that explicitly incorporate sensing constraints such as CRB together with communication SINR constraints [14].

For the proposed DRL framework, this formulation provides a flexible interface between the physical sensing model and the learning objective.

3.6 Robotic Mobility Model

The robotic platform is modeled as a mobile terminal whose location changes over discrete time slots.

Let the two-dimensional position of robot k at time t be

$$p_k(t) = [x_k(t) y_k(t)]^T. \quad (16)$$

The robot's velocity is

$$v_k(t) = [v_{x,k}(t) v_{y,k}(t)]^T. \quad (17)$$

The position evolution is represented by

$$p_k(t+1) = p_k(t) + v_k(t) \Delta t, \quad (18)$$

where Δt denotes the duration of one decision interval.

The velocity constraint is

$$\|v_k(t)\| \leq v_{\max}. \quad (19)$$

The mobility state affects the wireless channel because

$$h_{k,\text{eff}}(t) = f(p_k(t), \theta(t) G(t)). \quad (20)$$

Consequently, robot mobility introduces a time-varying component into the resource-allocation problem.

This is particularly relevant for high-frequency 6G systems because mmWave and THz communication can experience strong sensitivity to blockage, beam misalignment, and mobility. Beam management becomes increasingly challenging at higher

frequencies because narrow beams and rapidly varying channels increase the overhead associated with channel acquisition and tracking [16].

The proposed DRL agent therefore receives mobility information as part of its state representation rather than treating the robot as a static wireless terminal.

3.7 Edge-Computing Model

The robotic platform may process sensing and control tasks locally or offload computationally intensive tasks to an edge server.

Let $D_k(t)$ denote the amount of data generated by robot k at time t , and let f_k^{loc} denote its local computational frequency.

The local computation time can be approximated as

$$T_k^{\text{loc}} = \frac{D_k C_k}{f_k^{\text{loc}}}, \quad (21)$$

where C_k denotes the number of CPU cycles required per data unit.

If a fraction ρ_k of the task is offloaded to the edge server, then

$$0 \leq \rho_k \leq 1. \quad (22)$$

The offloaded data amount is

$$D_k^{\text{off}} = \rho_k D_k. \quad (23)$$

The transmission time for the offloaded data is

$$T_k^{\text{tx}} = \frac{\rho_k D_k}{R_k}. \quad (24)$$

If the edge server allocates computational capacity f_k^{edge} , the edge-processing time becomes

$$T_k^{\text{edge}} = \frac{\rho_k D_k C_k}{f_k^{\text{edge}}}. \quad (25)$$

The total latency associated with an offloaded task can therefore be represented as

$$T_k^{\text{off}} = T_k^{\text{tx}} + T_k^{\text{edge}}. \quad (26)$$

The available edge-computing resource is constrained by

$$\sum_{k=1}^K f_k^{\text{edge}} \leq f_{\text{edge}}^{\text{max}}. \quad (27)$$

This formulation introduces an additional coupling between wireless communication and computation: increasing communication resources can reduce transmission latency, whereas increasing edge-computing resources can reduce processing latency.

3.8 Energy Consumption Model

Energy efficiency is particularly important for mobile robotic systems because the robotic platform has a finite energy budget.

The communication energy consumed by robot k is approximated as

$$E_k^{\text{comm}} = P_k T_k^{\text{tx}}, \quad (28)$$

where P_k is the allocated transmit power.

The local computational energy can be represented using a dynamic-voltage-and-frequency-scaling approximation:

$$E_k^{\text{loc}} = \kappa_k C_k D_k (f_k^{\text{loc}})^2, \quad (29)$$

where κ_k is an effective hardware-dependent energy coefficient.

The total energy consumption is therefore represented as

$$E_k^{\text{tot}} = E_k^{\text{comm}} + E_k^{\text{loc}} + E_k^{\text{sense}} + E_k^{\text{motion}}, \quad (30)$$

where E_k^{sense} represents sensing energy and E_k^{motion} represents energy associated with robotic movement.

The energy constraint is

$$E_k^{\text{tot}} \leq E_k^{\text{max}}. \quad (31)$$

This formulation allows the subsequent optimization problem to consider the interaction between wireless resource allocation and robotic energy consumption.

3.9 Latency Model

The total task latency is represented as

$$T_k^{\text{total}} = T_k^{\text{comm}} + T_k^{\text{sense}} + T_k^{\text{comp}} + T_k^{\text{queue}}, \quad (32)$$

where the individual components represent communication, sensing, computation, and queueing delays.

The quality-of-service requirement is

$$T_k^{\text{total}} \leq T_k^{\text{max}}. \quad (33)$$

The latency constraint is particularly important for autonomous robotic applications because delayed communication or delayed sensing decisions can affect task execution.

3.10 System State for Reinforcement Learning

At each time step t , the proposed DRL agent observes a state vector containing information from the communication, sensing, computation, and robotic domains.

The state is defined as

$$s_t = [H_t, \gamma_t, p_t, v_t, q_t, E_t, B_t, f_t, z_t], \quad (34)$$

where:

- H_t represents channel-state information;
- γ_t represents SINR values;
- p_t represents robotic positions;
- v_t represents robotic velocities;
- q_t represents queue states;
- E_t represents remaining energy;
- B_t represents bandwidth allocation;
- f_t represents available computational resources;
- z_t represents sensing-related information.

This state representation is deliberately multidimensional because the proposed approach is intended to learn the interaction between wireless, sensing, computational, and mobility states.

3.11 Action Space

The DRL agent determines a vector of resource-management actions:

$$\mathbf{a}_t = [P_t, B_t, \theta_t, W_t, \rho_t, f_t], \quad (35)$$

where:

- P_t : transmit-power allocation;
- B_t : bandwidth allocation;
- θ_t : RIS phase configuration;
- W_t : beamforming parameters;
- ρ_t : computation-offloading decisions;
- f_t : edge-computing allocation.

The resulting action therefore simultaneously influences communication, sensing, and computing performance.

3.12 System Feedback

After an action $\mathbf{a}_t$ is applied, the wireless environment evolves to a new state $\mathbf{s}_{t+1}$. The resulting throughput, latency, energy consumption, sensing performance, and computational utilization are measured and supplied to the reward function.

The resulting interaction is represented as

$$\mathbf{s}_t \xrightarrow{\mathbf{a}_t} \text{6G environment} \xrightarrow{\text{system response}} (\mathbf{s}_{t+1}, r_t). \quad (36)$$

This creates the Markov decision process used in the proposed mathematics-guided DRL framework.

The mathematical model developed in this section provides the physical foundation for the optimization problem in Section 4. In particular, Equations (9)–(15) establish the communication and sensing performance measures, Equations (18)–(20) capture robotic mobility, Equations (21)–(27) model edge computation, and Equations (28)–(33) establish energy and latency constraints.

The next section converts these individual models into a **unified multi-objective constrained optimization problem**, which will then be transformed into the learning objective of the proposed mathematics-guided DRL framework.

4. MULTI-OBJECTIVE MATHEMATICAL PROBLEM FORMULATION

4.1 Problem Formulation Overview

The system model presented in Section 3 produces a highly coupled resource-allocation problem involving communication, sensing, computation, energy, RIS configuration, and robotic mobility. The decision variables influence multiple performance measures simultaneously. For example, increasing transmit power can improve communication and sensing quality but increases energy consumption and interference. Similarly, increasing bandwidth can reduce communication latency but cannot exceed the total available spectrum. RIS phase configuration can improve the effective channel while simultaneously influencing the sensing response.

Such coupling results in a non-convex optimization problem. Recent work on RIS-assisted ISAC has similarly formulated joint active and passive beamforming problems subject to communication SINR and sensing constraints and has reported the resulting optimization as computationally challenging [17].

In the proposed framework, mathematical optimization is not intended to replace the DRL controller. Instead, it provides the physical and mathematical structure that guides the learning process. The resulting optimization problem therefore serves two purposes:

1. to define the desired operating point of the 6G robotic ISAC system; and
2. to provide the objective and constraint information used in the mathematics-guided DRL framework.

4.2 Optimization Variables

Let the complete decision vector at time t be

$$\mathbf{u}_t = [P_t, B_t, W_t, \theta_t, \rho_t, f_t], \quad (37)$$

where

- P_t denotes transmit-power allocation;
- B_t denotes bandwidth allocation;
- W_t denotes active beamforming;
- θ_t denotes RIS phase configuration;
- ρ_t denotes computation-offloading ratios;
- f_t denotes edge-computing resource allocation.

The optimization variables therefore span the communication, sensing, RIS, and computing domains.

The relationship between the major variables is summarized in Table 3.

Table 3. Optimization variables and their physical roles

Variable	Domain	Physical meaning	Primary effect
P_k	Continuous	Transmit power	Rate, sensing, energy
B_k	Continuous	Allocated bandwidth	Rate, latency
$\mathbf{w}_k$	Complex/vector	Communication beamforming	SINR, interference
θ_n	Continuous/discrete	RIS phase shift	Effective channel
ρ_k	$[0, 1]$	Offloading ratio	Computation latency/energy
f_k^{edge}	Continuous	Edge CPU allocation	Processing latency
$\mathbf{p}_k$	Continuous	Robot position	Channel and sensing
$\mathbf{v}_k$	Continuous	Robot velocity	Mobility/channel variation

The robot position and velocity are treated primarily as **environmental state variables**, whereas the remaining variables constitute the controllable resource-allocation vector.

4.3 Communication Utility

The communication utility is defined from the aggregate achievable throughput:

$$R_{\text{sum}} = \sum_{k=1}^K B_k \log_2(1 + \gamma_k). \quad (38)$$

Because throughput has a positive contribution to system performance, its normalized form is defined as

$$\bar{R} = \frac{R_{\text{sum}}}{R_{\text{ref}}}, \quad (39)$$

where R_{ref} is a reference throughput used for normalization.

The normalization prevents throughput, energy, latency, and sensing metrics with different physical units from dominating the optimization objective solely because of their numerical scales.

4.4 Latency Objective

The total latency for robotic terminal k is given by

$$T_k^{\text{total}} = T_k^{\text{comm}} + T_k^{\text{sense}} + T_k^{\text{comp}} + T_k^{\text{queue}}. \quad (40)$$

The aggregate latency is

$$T_{\text{sum}} = \sum_{k=1}^K T_k^{\text{total}}. \quad (41)$$

Its normalized form is

$$\bar{T} = \frac{T_{\text{sum}}}{T_{\text{ref}}}. \quad (42)$$

Because latency is a minimization objective, it enters the utility function with a negative coefficient.

4.5 Energy Objective

The total system energy is represented by

$$E_{\text{sum}} = \sum_{k=1}^K (E_k^{\text{comm}} + E_k^{\text{sense}} + E_k^{\text{loc}} + E_k^{\text{motion}}) + E_{\text{edge}}. \quad (43)$$

The normalized energy consumption is

$$\bar{E} = \frac{E_{\text{sum}}}{E_{\text{ref}}}. \quad (44)$$

Energy is minimized because the robotic platform and wireless infrastructure operate under finite energy budgets.

4.6 Sensing Utility

The sensing performance defined in Section 3 can be represented by

$$S_{\text{perf}} = 1 - \varepsilon_s. \quad (45)$$

The normalized sensing utility is therefore

$$\bar{S} = \frac{S_{\text{perf}}}{S_{\text{ref}}}, \quad (46)$$

where S_{ref} is a reference sensing-performance value.

For the final experimental implementation, the sensing component can alternatively be defined using an estimator-specific metric such as the Cramér–Rao bound. This is particularly appropriate because recent RIS-assisted ISAC optimization studies explicitly use CRB as a sensing constraint together with SINR and RIS unit-modulus constraints.

4.7 Interference Objective

The aggregate interference experienced by the communication users is represented as

$$I_{\text{sum}} = \sum_{k=1}^K \sum_{j \neq k} |h_{k,\text{eff}}^H w_j|^2. \quad (47)$$

The normalized interference measure is

$$\bar{I} = \frac{I_{\text{sum}}}{I_{\text{ref}}}. \quad (48)$$

Minimizing interference is important because excessive interference reduces SINR and consequently communication throughput.

4.8 Multi-Objective Utility Function

The proposed system has conflicting objectives. We therefore formulate a weighted normalized utility function as

$$\mathcal{J} = \alpha_R \bar{R} + \alpha_S \bar{S} - \alpha_T \bar{T} - \alpha_E \bar{E} - \alpha_I \bar{I}, \quad (49)$$

where

$$\alpha_R, \alpha_S, \alpha_T, \alpha_E, \alpha_I \geq 0 \quad (50)$$

are the objective weights.

The weights satisfy

$$\alpha_R + \alpha_S + \alpha_T + \alpha_E + \alpha_I = 1. \quad (51)$$

The resulting multi-objective problem is

$$\boxed{\max_u \mathcal{J}} \quad (52)$$

subject to the communication, sensing, energy, computation, mobility, and RIS constraints defined below.

This weighted formulation provides a flexible mechanism for representing different application priorities. For example, an industrial robotic application requiring fast control decisions may assign greater importance to latency, whereas a battery-operated mobile robot may assign greater importance to energy efficiency.

Importantly, the weights are **not treated as universal constants**. Their values will be selected through a sensitivity analysis in Section 7 so that the effect of different communication–sensing–energy priorities can be examined.

4.9 Transmit-Power Constraint

The total BS transmit power must satisfy

$$\sum_{k=1}^K \|w_k\|^2 + \|W_s\|_F^2 \leq P_{\max}. \quad (53)$$

Individual power constraints can additionally be imposed as

$$0 \leq P_k \leq P_k^{\max}. \quad (54)$$

These constraints prevent the optimization algorithm from obtaining artificially high throughput by allocating unlimited transmission power.

4.10 Bandwidth Constraint

The available spectrum is limited according to

$$\sum_{k=1}^K B_k \leq B_{\text{tot}}, \quad (55)$$

with

$$B_k \geq 0. \quad (56)$$

The bandwidth constraint creates competition among robotic terminals and directly couples the resource-allocation problem with the achievable communication rate.

4.11 Communication QoS Constraint

Each active communication terminal is required to satisfy a minimum SINR:

$$\gamma_k \geq \gamma_k^{\min}. \quad (57)$$

Equivalently,

$$\frac{|h_{k,\text{eff}}^H w_k|^2}{\sum_{j \neq k} |h_{k,\text{eff}}^H w_j|^2 + \sigma_k^2} \geq \gamma_k^{\min}. \quad (58)$$

The corresponding minimum-rate requirement can be written as

$$R_k \geq R_k^{\min}. \quad (59)$$

These constraints prevent the optimization process from sacrificing individual communication links solely to maximize aggregate throughput.

4.12 Sensing Constraint

The sensing requirement can be expressed using a maximum acceptable sensing error:

$$\varepsilon_s \leq \varepsilon_s^{\max}. \quad (60)$$

When the final implementation uses CRB as the sensing metric, the corresponding constraint can be written as

$$\text{CRB} \leq \text{CRB}_{\max}. \quad (61)$$

This formulation is consistent with established RIS-assisted ISAC optimization, where communication SINR and sensing CRB constraints are jointly imposed on active and passive beamforming variables.

4.13 Energy Constraint

The total energy consumed by each robotic platform must satisfy

$$E_k^{\text{tot}} \leq E_k^{\max}. \quad (62)$$

At the system level,

$$E_{\text{sum}} \leq E_{\text{sys}}^{\max}. \quad (63)$$

These constraints prevent the learning agent from obtaining high communication and sensing performance at the expense of an impractical energy requirement.

4.14 Latency Constraint

The end-to-end task latency must satisfy

$$T_k^{\text{total}} \leq T_k^{\max}. \quad (64)$$

For safety-critical or time-sensitive robotic tasks, the latency constraint is particularly important because a resource-allocation solution that maximizes throughput but produces excessive delay may not be operationally useful.

4.15 Edge-Computing Constraint

The computational resources available at the edge server satisfy

$$\sum_{k=1}^K f_k^{\text{edge}} \leq f_{\text{edge}}^{\max}. \quad (65)$$

The offloading ratio is constrained according to

$$0 \leq \rho_k \leq 1. \quad (66)$$

The limiting cases have clear physical interpretations:

$$\rho_k = 0$$

corresponds to entirely local processing, whereas

$$\rho_k = 1$$

corresponds to complete offloading.

4.16 RIS Phase Constraint

For an ideal passive RIS,

$$|\Theta_{n,n}| = 1, n = 1, \dots, N. \quad (67)$$

Equivalently,

$$\Theta_{n,n} = e^{j\theta_n}, 0 \leq \theta_n < 2\pi. \quad (68)$$

For a practical discrete-phase RIS,

$$\theta_n \in \mathcal{Q}_B, \quad (69)$$

where

$$\mathcal{Q}_B = \left\{ \frac{2\pi m}{2^B}, \dots, m = 0, 1, \dots, 2^B - 1 \right\}. \quad (70)$$

The discrete formulation can be used in the simulation experiments to represent realistic RIS control.

4.17 Mobility Constraint

The robotic velocity satisfies

$$\|v_k(t)\| \leq v_k^{\max}. \quad (71)$$

The position must remain within the operational area:

$$p_k(t) \in \mathcal{A}, \quad (72)$$

where $\mathcal{A}$ denotes the permitted operating region.

The mobility state therefore acts as an environmental constraint rather than merely another optimization variable.

4.18 Complete Constrained Optimization Problem

Combining the objectives and constraints, the complete problem can be expressed as

$$\begin{aligned} & \max_u \alpha_R \bar{R} + \alpha_S \bar{S} - \alpha_T \bar{T} - \alpha_E \bar{E} - \alpha_I \bar{I} \\ & \text{s.t.} \quad \sum_k \|w_k\|^2 + \|W_s\|_F^2 \leq P_{\max}, \\ & \quad \sum_k B_k \leq B_{\text{tot}}, \\ & \quad \gamma_k \geq \gamma_k^{\min}, \forall k, \\ & \quad R_k \geq R_k^{\min}, \forall k, \\ & \quad \varepsilon_s \leq \varepsilon_s^{\max}, \\ & \quad E_k^{\text{tot}} \leq E_k^{\max}, \forall k, \\ & \quad T_k^{\text{total}} \leq T_k^{\max}, \forall k, \\ & \quad \sum_k f_k^{\text{edge}} \leq f_{\text{edge}}^{\max}, \\ & \quad 0 \leq \rho_k \leq 1, \forall k, \\ & \quad |\Theta_{n,n}| = 1, \forall n, \\ & \quad \theta_n \in \mathcal{Q}_B, \forall n, \\ & \quad \|v_k\| \leq v_k^{\max}, \forall k. \end{aligned} \quad (73)$$

Equation (73) represents the central mathematical optimization problem of the proposed study.

4.19 Nature of the Optimization Problem

Problem (73) is difficult to solve directly because several sources of non-convexity and coupling are present simultaneously.

1. Non-convex beamforming terms

The SINR expression contains coupled quadratic terms involving beamforming vectors.

2. RIS phase coupling

The unit-modulus constraint

$$|\theta_{n,n}| = 1$$

creates a non-convex feasible set.

3. Discrete RIS control

When finite-bit phase shifts are used, the optimization contains discrete variables.

4. Communication–sensing coupling

The same transmit signal and wireless resources influence both communication and sensing performance.

5. Mobility dependence

The channel changes with the robotic position and therefore changes the feasible and desirable resource allocation.

6. Computing–communication coupling

Offloading decisions depend on communication rate, while processing latency depends on the amount of offloaded data and available edge resources.

7. Multi-objective trade-offs

Increasing one performance metric can deteriorate another. For example,

$$R_{\text{sum}} \uparrow$$

may require

$$E_{\text{sum}} \uparrow,$$

while improving sensing accuracy may require resources that reduce communication capacity.

These characteristics make the problem suitable for a sequential learning formulation. Recent surveys similarly identify the computational complexity and dynamic nature of ISAC optimization as important motivations for learning-based approaches.

4.20 Conversion to a Constrained Markov Decision Process

The optimization problem is converted into a constrained Markov decision process (CMDP) for DRL.

The environment is represented by

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \mathcal{C}, \gamma), \quad (74)$$

where:

- $\mathcal{S}$ is the state space;
- $\mathcal{A}$ is the action space;
- $\mathcal{P}$ is the state-transition probability;
- $\mathcal{R}$ is the reward function;

- $\mathcal{C}$ represents constraint costs;
- γ is the discount factor.

The state at time t is

$$s_t = [H_t, p_t, v_t, \gamma_t, q_t, E_t, z_t, f_t]. \quad (75)$$

The action is

$$a_t = [P_t, B_t, W_t, \theta_t, \rho_t, f_t]. \quad (76)$$

The objective is to learn a policy

$$\pi_\phi(a_t | s_t) \quad (77)$$

that maximizes the expected discounted cumulative reward:

$$\max_{\pi_\phi} \mathbb{E} \left[\sum_{t=0}^T \gamma^t r_t \right]. \quad (78)$$

4.21 Mathematics-Guided Reward Construction

The central feature of the proposed framework is that the reward is derived directly from the mathematical objective rather than selected independently.

The instantaneous reward is defined as

$$r_t = \alpha_R \hat{R}_t + \alpha_S \hat{S}_t - \alpha_T \hat{T}_t - \alpha_E \hat{E}_t - \alpha_I \hat{I}_t - \lambda C_t, \quad (79)$$

where the normalized quantities are represented using a hat and C_t denotes the aggregate constraint violation.

The constraint-violation term is defined as

$$C_t = C_t^P + C_t^B + C_t^{\text{SINR}} + C_t^R + C_t^S + C_t^E + C_t^T + C_t^F. \quad (80)$$

For example, the power violation can be represented as

$$C_t^P = \max \left(0, \frac{P_t - P_{\max}}{P_{\max}} \right), \quad (81)$$

while the latency violation can be expressed as

$$C_t^T = \sum_k \max \left(0, \frac{T_k^{\text{total}} - T_k^{\max}}{T_k^{\max}} \right). \quad (82)$$

Similar normalized violation terms are constructed for bandwidth, SINR, rate, sensing, energy, and computing constraints.

The reward can therefore be interpreted as

$$\boxed{\text{Reward} = \text{Performance Utility} - \text{Constraint Penalty}} \quad (83)$$

which provides the principal connection between mathematical optimization and DRL.

4.22 Feasibility-Aware Action Handling

Penalty-based reward design alone may allow the agent to explore highly infeasible actions during training. Therefore, the proposed framework will additionally employ feasibility-aware action processing.

For a continuous variable x , the projected action is

$$x^{\text{proj}} = \min(x_{\max}, \max(x_{\min}, x)). \quad (84)$$

For bandwidth allocation, a normalization operation can enforce

$$\sum_k B_k = B_{\text{tot}}$$

when the generated allocation exceeds the available resource.

Similarly, RIS phase values are mapped to the nearest valid quantized phase:

$$\theta_n^{\text{proj}} = \arg \min_{\theta \in \mathcal{Q}_B} |\theta - \theta_n|. \quad (85)$$

This mechanism reduces the number of infeasible actions presented to the environment.

4.23 Pareto Interpretation

Because the proposed problem contains conflicting objectives, a single weighted solution represents one operating point on a broader Pareto frontier.

Let

$$F = [-R_{\text{sum}}, T_{\text{sum}}, E_{\text{sum}}, I_{\text{sum}} \varepsilon_s]. \quad (86)$$

A solution $\mathbf{u}_1$ dominates $\mathbf{u}_2$ if it is no worse in every objective and strictly better in at least one.

The proposed weighted formulation is therefore interpreted as a scalarized approximation to the multi-objective problem. Sensitivity experiments with different values of

$$(\alpha_R, \alpha_S, \alpha_T, \alpha_E, \alpha_I)$$

will be used to examine how the learned policy changes across different operating priorities.

This analysis is important because a single aggregate reward can conceal meaningful communication–sensing–energy trade-offs.

4.24 Mathematical Role of the Proposed Framework

The relationship between the optimization formulation and the learning framework can be summarized as

$$\text{Physical Model} \rightarrow \text{Mathematical Objectives} \rightarrow \text{Constraints} \rightarrow \text{Normalized Reward} \rightarrow \text{DRL Policy} \rightarrow \text{Resource Allocation} \quad (87)$$

This sequence distinguishes the proposed approach from a conventional DRL formulation in which the reward function is defined independently of an explicit mathematical system model.

The mathematical formulation therefore acts as **domain knowledge** for the learning system.

4.25 Summary

This section formulated the 6G-enabled robotic ISAC resource-management problem as a constrained multi-objective optimization problem. Communication throughput, sensing performance, latency, energy consumption, interference, RIS configuration, edge computing, and robotic mobility were incorporated into the formulation.

The resulting problem is non-convex and dynamically coupled because the communication, sensing, RIS, computing, and mobility variables influence one another. Rather than solving the optimization problem independently at every time step, the proposed framework transforms it into a constrained Markov decision process.

The most important feature is that the mathematical formulation is retained during this transformation. The objective function defines the reward structure, while the physical constraints define feasibility and penalty terms. This provides the mathematical foundation for the **Mathematics-Guided Deep Reinforcement Learning (MG-DRL)** algorithm developed in Section 5.

5. Proposed Mathematics-Guided Deep Reinforcement Learning Framework

5.1 Motivation and Design Principle

The optimization problem formulated in Section 4 contains several coupled continuous and discrete variables, including transmit power, bandwidth, beamforming parameters, RIS phase shifts, computation-offloading ratios, and edge-computing resources. Solving the complete non-convex problem independently at every time slot can become computationally demanding,

particularly when the wireless environment changes rapidly because of robot mobility, interference, blockage, and variations in sensing and computational demand.

Deep reinforcement learning provides an alternative by learning a policy that maps the observed system state to resource-allocation actions. Recent surveys identify DRL as an increasingly important approach for dynamic 6G-ISAC resource management because of the high-dimensional state spaces, sensing–communication trade-offs, and time-varying wireless environments encountered in these systems [3], [18].

However, a purely unconstrained DRL formulation may learn actions that provide a high instantaneous reward while violating physical system requirements. The proposed framework therefore introduces **mathematical guidance** into the learning process.

The central design principle is

$$\boxed{\text{Physical Model} \rightarrow \text{Optimization Model} \rightarrow \text{Constraint-Aware Learning} \rightarrow \text{Adaptive DRL Policy}} \quad (88)$$

Rather than treating the mathematical model as a separate optimization stage, the proposed approach uses it to construct the state representation, normalize objectives, define constraint violations, shape the reward, and project infeasible actions into the feasible region.

5.2 Proposed MG-DRL Architecture

The proposed **Mathematics-Guided Deep Reinforcement Learning (MG-DRL)** framework consists of five principal components:

1. **State encoder**
2. **Hybrid actor network**
3. **Critic network**
4. **Mathematical constraint and feasibility module**
5. **Environment feedback module**

The overall architecture is illustrated in Figure 4.

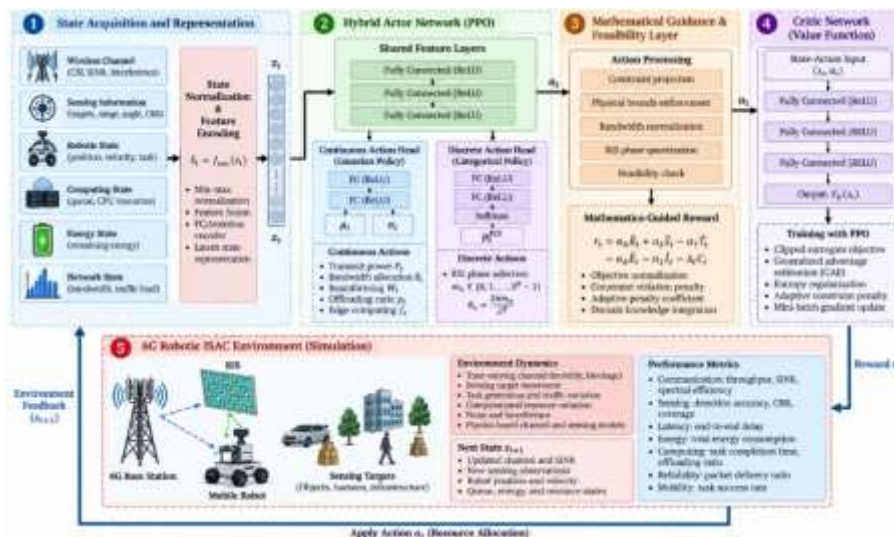


Figure 4. Detailed architecture of the proposed mathematics-guided DRL framework. The state encoder receives communication, sensing, mobility, energy, and computing information; the hybrid actor generates continuous and discrete resource-allocation actions; the mathematical guidance module enforces objective normalization and physical constraints; and the critic evaluates the resulting state–action pair.

The architecture operates iteratively. At time t , the environment provides state s_t . The state encoder transforms the heterogeneous variables into a normalized latent representation. The actor then produces the resource-allocation action a_t . The mathematical feasibility module modifies or rejects infeasible components. The feasible action is applied to the wireless environment, after which the resulting performance metrics determine the reward r_t and the next state s_{t+1} .

5.3 Markov Decision Process Representation

The system is formulated as a constrained Markov decision process:

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \mathcal{C}, \gamma), \quad (89)$$

where:

- $\mathcal{S}$ is the state space;
- $\mathcal{A}$ is the action space;
- $\mathcal{P}$ represents the state-transition dynamics;
- $\mathcal{R}$ is the reward function;
- $\mathcal{C}$ represents constraint costs;
- $\gamma \in (0, 1)$ is the discount factor.

At each time step,

$$s_t \rightarrow a_t \rightarrow (r_t, s_{t+1}). \quad (90)$$

The objective is to learn a policy

$$\pi_\phi(a_t | s_t) \quad (91)$$

that maximizes the expected discounted cumulative reward while maintaining system feasibility.

5.4 State Representation

The state vector is constructed from the physical variables developed in Section 3:

$$s_t = [h_t, \gamma_t, p_t, v_t, q_t, E_t, z_t, f_t, B_t]. \quad (92)$$

The state consists of nine groups.

5.4.1 Channel information

$$h_t$$

contains effective channel information for direct and RIS-assisted propagation.

5.4.2 Communication quality

$$\gamma_t = [\gamma_1, \dots, \gamma_K]$$

contains the current SINR values.

5.4.3 Robotic position

$$p_t = [p_1, \dots, p_K]$$

describes the spatial distribution of the mobile robotic platforms.

5.4.4 Robotic velocity

$$v_t = [v_1, \dots, v_K]$$

captures mobility-related channel variation.

5.4.5 Queue state

$$q_t$$

represents data awaiting transmission or computation.

5.4.6 Energy state

$$E_t$$

represents the remaining energy budget.

5.4.7 Sensing state

$$z_t$$

contains sensing quality, target information, and sensing demand.

5.4.8 Computing state

$$f_t$$

represents available local and edge computational resources.

5.4.9 Current resource state

$$B_t$$

represents current bandwidth allocation and resource utilization.

The resulting state representation intentionally combines ECE variables with mathematical and robotic variables. This allows the agent to learn how physical changes in the wireless environment affect the optimal resource-allocation policy.

5.5 State Normalization

Because the variables in Equation (92) have substantially different numerical ranges and physical units, direct feeding of the raw values into the neural network can lead to poorly conditioned learning.

For a generic state variable x , min–max normalization is defined as

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min} + \epsilon}, \quad (93)$$

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

For variables with approximately zero-centered distributions, standardization can alternatively be expressed as

$$\hat{x} = \frac{x - \mu_x}{\sigma_x + \epsilon}. \quad (94)$$

The same normalization procedure is applied consistently to the training and evaluation environments.

5.6 Hybrid Action Space

A major characteristic of the proposed problem is that the action space contains both continuous and discrete variables.

The continuous action vector is defined as

$$a_t^c = [P_t, B_t, W_t, \rho_t, f_t]. \quad (95)$$

The discrete action vector corresponds to the quantized RIS phase configuration:

$$a_t^d = [m_1, m_2, \dots, m_N], \quad (96)$$

where

$$m_n \in \{0, 1, \dots, 2^B - 1\}. \quad (97)$$

Thus,

$$\mathbf{a}_t = (\mathbf{a}_t^c, \mathbf{a}_t^d). \quad (98)$$

This hybrid formulation is preferable to forcing all variables into a single continuous representation because practical RIS controllers often operate with finite phase resolution.

5.7 Hybrid Actor Network

The actor network contains a shared feature-extraction layer followed by separate output heads.

The shared representation is

$$\mathbf{z}_t = f_{\text{enc}}(\mathbf{s}_t; \phi_e), \quad (99)$$

where f_{enc} denotes the state encoder.

The continuous action head produces

$$[\mu_t, \sigma_t] = f_c(\mathbf{z}_t; \phi_c), \quad (100)$$

where μ_t and σ_t parameterize a probability distribution over continuous actions.

The discrete RIS head produces

$$\mathbf{p}_t^{\text{RIS}} = f_d(\mathbf{z}_t; \phi_d), \quad (101)$$

where each element represents a probability distribution over the available RIS phase states.

The complete policy can therefore be expressed as

$$\pi_\phi(\mathbf{a}_t | \mathbf{s}_t) = \pi_\phi^c(\mathbf{a}_t^c | \mathbf{s}_t) \pi_\phi^d(\mathbf{a}_t^d | \mathbf{s}_t). \quad (102)$$

This architecture allows the same learned representation to coordinate conventional communication resources with RIS configuration.

5.8 Selection of PPO as the Primary Learning Algorithm

Proximal Policy Optimization (PPO) is selected as the primary learning algorithm because the proposed problem requires stable policy updates in a high-dimensional control environment and includes a hybrid action representation.

PPO introduces a clipped surrogate objective that limits excessively large policy updates. The original PPO formulation alternates environment interaction with minibatch optimization and was designed to provide a practical balance between implementation simplicity, sample efficiency, and policy-update stability [19].

For comparison, SAC is an off-policy maximum-entropy actor–critic method designed for continuous control and emphasizes both expected reward and policy entropy [20]. TD3 uses twin critics and delayed policy updates to reduce overestimation errors in actor–critic learning [21].

Therefore, the experimental study will compare:

- proposed MG-PPO;
- standard PPO;
- SAC;
- TD3;
- conventional optimization baselines.

This comparison allows the effect of the **mathematical-guidance mechanism** to be separated from the effect of merely selecting a particular DRL algorithm.

5.9 PPO Policy Objective

Let the probability ratio between the updated and previous policy be

$$r_t(\phi) = \frac{\pi_\phi(a_t | s_t)}{\pi_{\phi_{old}}(a_t | s_t)}. \quad (103)$$

Let $\hat{A}_t$ denote the estimated advantage.

The clipped PPO objective is

$$L^{CLIP}(\phi) = \mathbb{E}_t[\min(r_t(\phi)\hat{A}_t, \text{clip}(r_t(\phi), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)], \quad (104)$$

where ϵ is the clipping coefficient.

The clipping operation prevents the updated policy from moving too far from the previous policy during a single optimization step.

5.10 Value Function

The critic estimates the expected discounted return:

$$V_\psi(s_t) = \mathbb{E}_{\pi_\phi} \left[\sum_{l=0}^{\infty} \gamma^l r_{t+l} | s_t \right]. \quad (105)$$

The temporal-difference error is

$$\delta_t = r_t + \gamma V_\psi(s_{t+1}) - V_\psi(s_t). \quad (106)$$

The critic is trained by minimizing

$$L_V(\psi) = \mathbb{E}_t \left[(V_\psi(s_t) - V_t^{target})^2 \right]. \quad (107)$$

5.11 Generalized Advantage Estimation

To reduce variance in the policy-gradient estimate, generalized advantage estimation is used:

where λ controls the bias–variance trade-off.

The return target is then

$$V_t^{target} = \hat{A}_t + V_\psi(s_t). \quad (109)$$

The resulting advantage estimate is used in Equation (104).

5.12 Mathematical Guidance Mechanism

The principal methodological contribution is the mathematical guidance layer.

The mathematical optimization model influences the learning process through four mechanisms:

Mechanism 1: Objective guidance

The reward is derived from the normalized optimization objective:

$$r_t^{obj} = \alpha_R \hat{R}_t + \alpha_S \hat{S}_t - \alpha_T \hat{T}_t - \alpha_E \hat{E}_t - \alpha_I \hat{I}_t. \quad (110)$$

Mechanism 2: Constraint guidance

The system calculates constraint violations:

$$C_t = \sum_{m=1}^M C_{m,t}. \quad (111)$$

Mechanism 3: Feasibility guidance

Actions violating simple bounds are projected into the feasible region.

Mechanism 4: State guidance

The state includes variables directly derived from the mathematical model, including SINR, latency, energy, sensing error, queue state, and resource utilization.

The complete reward is

$$r_t = r_t^{obj} - \lambda_t C_t. \quad (112)$$

This creates the principal distinction between the proposed method and standard PPO.

5.13 Adaptive Constraint Penalty

A fixed penalty coefficient can cause two undesirable outcomes:

1. a penalty that is too small may allow persistent constraint violations;
2. a penalty that is too large may prevent useful exploration.

Therefore, the proposed framework uses an adaptive penalty coefficient.

The penalty update is defined as

$$\lambda_{t+1} = [\lambda_t + \eta_\lambda (C_t - C_{\text{target}})]^+, \quad (113)$$

where:

- η_λ is the penalty learning rate;
- C_{target} is the desired average violation level;
- $[x]^+ = \max(0, x)$.

When constraint violations remain high,

$$C_t > C_{\text{target}},$$

the penalty increases.

When the policy remains within the desired feasible region, the penalty does not grow unnecessarily.

This mechanism provides an adaptive connection between the optimization constraints and the learning process.

5.14 Feasible Action Projection

After the actor generates an action, the mathematical feasibility layer applies projection operations.

For transmit power,

$$P_k^{proj} = \min(P_k^{max}, \max(0, P_k)). \quad (114)$$

For bandwidth,

$$B_k^{proj} = \frac{B_k}{\sum_j B_j} B_{\text{tot}} \quad (115)$$

when the generated allocation exceeds the available bandwidth.

For the offloading ratio,

$$\rho_k^{proj} = \min(1, \max(0, \rho_k)). \quad (116)$$

For RIS phases,

$$m_n^{proj} = \text{round} \left(\frac{2^B \theta_n}{2\pi} \right) 2^B. \quad (117)$$

The projected phase is then

$$\theta_n^{proj} = \frac{2\pi m_n^{proj}}{2^B}. \quad (118)$$

This projection mechanism reduces the frequency of physically invalid actions reaching the simulated wireless environment.

5.15 Entropy Regularization

To maintain sufficient exploration during early training, an entropy term is incorporated into the policy objective:

$$L_{\text{entropy}} = \mathbb{E}_t \left[\mathcal{H} \left(\pi_\phi(\cdot | s_t) \right) \right]. \quad (119)$$

The total PPO objective is therefore expressed as

$$L_{\text{PPO}} = L^{\text{CLIP}} - c_v L_V + c_e L_{\text{entropy}} - c_c L_{\text{constraint}}, \quad (120)$$

where:

- c_v is the value-loss coefficient;
- c_e is the entropy coefficient;
- c_c controls the influence of constraint learning.

The exact signs in the implementation will follow the chosen loss-minimization convention.

5.16 Complete MG-PPO Training Procedure

The training process is summarized below.

Algorithm 1. Mathematics-Guided PPO for 6G Robotic ISAC

Input: Initial policy parameters ϕ , critic parameters ψ , penalty coefficient λ , learning rates $\eta_\phi, \eta_\psi, \eta_\lambda$, discount factor γ , GAE parameter λ_{GAE} , clipping coefficient ϵ .

Output: Trained policy π_{ϕ^*} .

1. Initialize the 6G robotic ISAC environment.
2. Initialize actor, critic, and mathematical-guidance parameters.
3. Observe initial state s_0 .
4. For each training iteration:
 1. Generate an action a_t using the hybrid actor.
 2. Separate the action into continuous and discrete components.
 3. Apply mathematical feasibility projection.
 4. Execute the feasible action in the environment.
 5. Calculate throughput, sensing performance, latency, energy, interference, and computing utilization.
 6. Normalize the performance metrics.
 7. Calculate individual constraint violations.
 8. Calculate the aggregate constraint cost C_t .

9. Calculate the mathematics-guided reward using Equation (112).
10. Store (s_t, a_t, r_t, s_{t+1}) .
11. Continue until the trajectory/batch reaches the required size.
12. Calculate advantages using Equation (108).
13. Update the actor using the clipped PPO objective.
14. Update the critic using the value-loss objective.
15. Update the adaptive constraint penalty using Equation (113).
16. Evaluate the average reward and constraint-violation rate.
17. Repeat until the stopping criterion is satisfied.

5. Return the trained policy.

The algorithm is deliberately structured so that **the mathematical model participates in every training cycle**, rather than being used only during an initial optimization stage.

5.17 Convergence Criterion

Training is considered converged when both performance and feasibility stabilize.

Let the moving-average reward be

$$\bar{R}_n = \frac{1}{W} \sum_{i=n-W+1}^n R_i. \quad (121)$$

Training is considered stable when

$$|\bar{R}_n - \bar{R}_{n-1}| < \epsilon_R \quad (122)$$

for a predefined number of consecutive evaluation windows and when the average constraint violation satisfies

$$\bar{C}_n < \epsilon_C. \quad (123)$$

This prevents declaring convergence merely because the reward has stabilized while the policy continues to violate system constraints.

5.18 Computational Complexity

Let:

- d_s denote the dimension of the state vector;
- d_a denote the action dimension;
- H denote the number of hidden neurons per layer;
- L denote the number of neural-network layers.

The approximate forward-pass complexity of a fully connected actor network is

$$\mathcal{O}(d_s H + (L - 1)H^2 + H d_a). \quad (124)$$

The critic has a similar order:

$$\mathcal{O}((d_s + d_a)H + (L - 1)H^2 + H). \quad (125)$$

The mathematical feasibility operations add approximately

$$\mathcal{O}(K + N) \quad (126)$$

for power/bandwidth projection and RIS phase projection, excluding the cost of calculating the wireless channel and sensing model.

Consequently, once the neural networks are trained, inference can be performed using a forward pass rather than solving the complete non-convex optimization problem from scratch at every decision interval.

This distinction between **training complexity** and **online inference complexity** will be explicitly examined in Section 8.

5.19 Baseline Algorithms

To establish whether the proposed mathematical guidance provides measurable benefit, the following baselines will be implemented under identical environmental conditions.

Baseline 1: Standard PPO

The same PPO architecture is used without mathematical constraint guidance.

$$r_t = r_t^{obj}. \quad (127)$$

This isolates the contribution of the proposed mathematical-guidance layer.

Baseline 2: SAC

SAC provides an off-policy entropy-regularized actor–critic comparison for continuous resource allocation [20].

Baseline 3: TD3

TD3 provides a deterministic actor–critic comparison designed to reduce value overestimation through twin critics and delayed policy updates [21].

Baseline 4: Conventional Optimization

A conventional optimization procedure such as PSO or an appropriate alternating-optimization method will be used as a non-learning baseline.

Baseline 5: Ablated MG-PPO

The proposed architecture will be evaluated with selected components removed:

- without RIS optimization;
- without sensing information;
- without mobility information;
- without adaptive constraint penalty;
- without mathematical reward guidance.

This enables the individual contributions to be quantified.

5.20 Experimental Hypotheses

The experimental study will test the following hypotheses.

H1 — Mathematical guidance

MG-PPO should produce lower constraint-violation rates than standard PPO under the same environment and training budget.

H2 — Multi-objective performance

The proposed framework should provide a more balanced communication–sensing–energy trade-off than single-objective resource allocation.

H3 — Dynamic adaptation

The proposed policy should maintain stable performance when robot mobility and channel conditions vary.

H4 — RIS awareness

Joint optimization of RIS configuration and communication resources should improve system performance relative to a fixed-RIS configuration.

H5 — Mathematical-guidance contribution

Removing the mathematical-guidance mechanism should produce measurable degradation in at least one or more of throughput, latency, energy efficiency, sensing performance, or constraint satisfaction.

These hypotheses will be evaluated using repeated simulation runs rather than being assumed in advance.

5.21 Summary

This section developed the proposed mathematics-guided deep reinforcement learning framework for the 6G robotic ISAC system. The method uses PPO as the primary policy-optimization algorithm and introduces a hybrid actor capable of generating continuous communication/computational actions and discrete RIS phase decisions.

The principal contribution is the integration of the mathematical optimization model into the learning process. The optimization objectives determine the normalized reward, the physical constraints determine the constraint costs, the system equations determine the state variables, and feasibility projection restricts actions to physically meaningful regions.

The resulting framework can be summarized as

6G/ISAC State → State Encoder → Hybrid PPO Actor → Mathematical Feasibility Layer → Wireless/Robotic Environment → Reward + Constraints → Actor/Critic Update (128)

The next section establishes the simulation environment and experimental methodology required to test these claims quantitatively.

6. SIMULATION METHODOLOGY

The proposed mathematics-guided deep reinforcement learning (MG-DRL) framework is evaluated through a physics-informed simulation environment representing a 6G-oriented integrated sensing and communication (ISAC) system with a mobile robotic platform, reconfigurable intelligent surface (RIS), multi-antenna base station (BS), and edge-computing server. The simulation is designed to reproduce the main characteristics identified in the mathematical formulation, including time-varying wireless channels, sensing–communication coupling, RIS configuration, robotic mobility, computation offloading, energy consumption, and latency constraints. Recent studies emphasize that these dynamic and coupled characteristics make RL/DRL particularly relevant to 6G ISAC resource-management problems [22], while current ETSI ISAC work also identifies channel modelling, measurement methodology, KPIs, and computing integration as important components of 6G ISAC evaluation [23].

The simulation methodology is organized into the following stages: (i) generation of the wireless and sensing environment, (ii) initialization of robotic and computing states, (iii) observation of the system state, (iv) resource allocation by the proposed MG-DRL controller, (v) mathematical feasibility processing, (vi) environment transition, and (vii) performance evaluation over repeated independent experiments.

6.1 Simulation Environment

A downlink 6G-oriented ISAC network is considered in which a multi-antenna BS simultaneously supports communication and sensing functions. A mobile robotic platform operates within the coverage region and communicates with the BS while performing sensing-related tasks. An RIS is deployed between the BS and the robotic operating region to provide controllable propagation. An edge-computing server is connected to the BS and provides computational resources for tasks that are partially or fully offloaded by the robot.

The simulation area is represented as a two-dimensional $100 \text{ m} \times 100 \text{ m}$ operational region. The BS and RIS remain stationary, whereas the robotic platform changes its position according to the mobility model defined in Section 3. Sensing targets are independently placed within the operating region and their positions may change between time slots.

The environment is represented as a discrete-time process with a slot duration of 10ms. At the beginning of each slot, the controller receives the current system state and determines the resource-allocation action. The environment then updates the channel, sensing, mobility, computing, queue, and energy states.

The use of a dynamic environment is important because mobility, blockage, interference, and changing sensing requirements can significantly alter the optimal resource allocation. Recent 6G-ISAC research identifies these time-varying conditions as important motivations for adaptive RL/DRL approaches [22].

6.2 Network and Wireless Parameters

The principal simulation parameters are summarized in Table 4.

Table 4. Simulation Parameters

Parameter	Symbol	Simulation setting
Simulation area	A	100 m $\times$ 100 m
Carrier frequency	f_c	90 GHz
System bandwidth	$B_{\max}$	2 GHz
BS antennas	N_t	16
RIS elements	N	64
RIS phase resolution	b_{RIS}	2 bits
Maximum BS transmit power	$P_{\max}$	30 dBm
Number of robotic platforms	K	4
Number of sensing targets	L	3
Maximum robot velocity	$v_{\max}$	5 m/s
Simulation slot duration	Δt	10 ms
Edge CPU capacity	f_{edge}	20 GHz
Local robot CPU	f_{local}	2 GHz
Offloading ratio	ρ_k	$0 \leq \rho_k \leq 1$
Noise figure	F	7 dB
RIS reflection amplitude	β_n	1
PPO discount factor	γ	0.99
GAE parameter	λ_{GAE}	0.95
PPO clipping coefficient	ϵ	0.20
Actor learning rate	η_{π}	3×10^{-4}

Parameter	Symbol	Simulation setting
Critic learning rate	η_V	3×10^{-4}
Hidden layers	—	256–256
Mini-batch size	—	256
Rollout length	—	2048 steps
Entropy coefficient	c_H	0.01
Independent training seeds	—	5

The carrier frequency is selected at 90GHz to represent a research-oriented high-frequency 6G scenario while remaining within the frequency range addressed by the widely used 3GPP TR 38.901 channel-model framework, which covers 0.5–100 GHz [24]. This choice avoids treating a future 6G frequency allocation as an established standard. For frequencies and use cases beyond existing baseline channel models, ETSI's current ISAC work specifically considers advanced channel models and measurement-based validation [23].

6.3 Wireless Channel Model

The communication channel between the BS and robot k contains a direct BS–robot component and an RIS-assisted reflected component. The effective channel is represented as

$$h_{k,\text{eff}} = h_{k,d} + h_{k,r}^H \Theta G, \quad (128)$$

where $h_{k,d}$ denotes the direct BS–robot channel, G represents the BS–RIS channel, $h_{k,r}$ represents the RIS–robot channel, and Θ is the RIS reflection matrix.

For each simulation realization, path loss, small-scale fading, shadowing, and time-dependent channel variation are generated according to the selected propagation configuration. Robot movement changes the BS–robot and RIS–robot distances and therefore modifies the effective channel over time.

The RIS phase of element n is quantized according to

$$\theta_n = \frac{2\pi m_n}{2^{b_{\text{RIS}}}}, \quad m_n \in \{0, 1, \dots, 2^{b_{\text{RIS}}} - 1\}. \quad (129)$$

The use of quantized RIS phases allows the proposed hybrid PPO architecture to explicitly address the discrete component of the resource-allocation problem.

6.4 Communication Model

For robot k , the received communication quality is determined using the SINR expression developed in Section 3:

$$\gamma_k(t) = \frac{|h_{k,\text{eff}}^H(t) w_k(t)|^2}{I_k(t) + \sigma_k^2}. \quad (130)$$

The achievable communication rate is

$$R_k(t) = B_k(t) \log_2(1 + \gamma_k(t)), \quad (131)$$

and the aggregate communication throughput is

$$R_{\text{sum}}(t) = \sum_{k=1}^K R_k(t). \quad (132)$$

The spectral efficiency is calculated as

$$SE(t) = \frac{R_{\text{sum}}(t)}{B_{\text{tot}}}. \quad (133)$$

These quantities are subsequently normalized before being incorporated into the multi-objective reward.

6.5 Integrated Sensing Model

The sensing component uses the same transmitted signal resource as the communication system. Each sensing target is represented by a target-response coefficient and an equivalent sensing channel.

The sensing quality is evaluated using the estimation error or an equivalent normalized sensing metric. For target l ,

$$MSE_l = \mathbb{E}[\|\hat{\xi}_l - \xi_l\|^2], \quad (134)$$

where ξ_l represents the target parameters and $\hat{\xi}_l$ represents their estimates.

A normalized sensing utility is subsequently defined as

$$\bar{S}(t) = 1 - \frac{MSE(t)}{MSE_{\text{max}}}, \quad (135)$$

with the value constrained to the interval $[0, 1]$.

For experiments involving localization-oriented sensing, the Cramér–Rao bound (CRB) is additionally calculated as an uncertainty measure. Lower CRB values indicate lower theoretical estimation uncertainty. This enables the sensing objective to be evaluated independently from communication throughput.

The joint communication–sensing formulation follows the broader direction of current ISAC research, in which sensing and communication resources are jointly optimized rather than treated as completely independent subsystems [22], [23].

6.6 Robotic Mobility Model

The robot position is updated according to

$$p_k(t+1) = p_k(t) + v_k(t)\Delta t. \quad (136)$$

The velocity is constrained by

$$\|v_k(t)\|_2 \leq v_{\text{max}}. \quad (137)$$

Three mobility conditions are considered:

1. **Low mobility:** robot velocity below 1.5 m/s;
2. **Moderate mobility:** velocity between 1.5 and 3.5 m/s;
3. **High mobility:** velocity between 3.5 and 5 m/s.

Random waypoint movement with boundary reflection is used to avoid robots leaving the simulation area. The mobility state is included in the DRL observation so that the controller can learn the relationship between robot movement and future communication, sensing, and computing conditions.

6.7 Edge-Computing and Task Model

Each robotic platform generates computational tasks that can either be processed locally or offloaded to the edge server. For task k , the offloading ratio is

$$0 \leq \rho_k(t) \leq 1. \quad (138)$$

The local computation latency is modeled as

$$T_k^{\text{loc}} = \frac{(1 - \rho_k)D_k C_k}{f_k^{\text{loc}}}, \quad (139)$$

where D_k is the task size, C_k is the required number of CPU cycles per bit, and f_k^{loc} is the local CPU frequency.

The offloading latency is

$$T_k^{\text{off}} = T_k^{\text{tx}} + T_k^{\text{edge}}, \quad (140)$$

where

$$T_k^{\text{tx}} = \frac{\rho_k D_k}{R_k} \quad (141)$$

and

$$T_k^{\text{edge}} = \frac{\rho_k D_k C_k}{f_k^{\text{edge}}}. \quad (142)$$

Consequently,

$$T_k^{\text{comp}} = \max(T_k^{\text{loc}}, T_k^{\text{off}}). \quad (143)$$

The edge-computing resource is constrained by

$$\sum_{k=1}^K f_k^{\text{edge}} \leq f_{\text{edge}}. \quad (144)$$

This coupling between wireless transmission and computation allows the proposed framework to evaluate communication–computation trade-offs rather than optimizing radio resources in isolation.

6.8 Energy Consumption Model

The total energy consumption of robot k is calculated as

$$E_k = E_k^{\text{tx}} + E_k^{\text{comp}} + E_k^{\text{idle}}. \quad (145)$$

Transmission energy is

$$E_k^{\text{tx}} = P_k T_k^{\text{tx}}, \quad (146)$$

while computation energy is represented by

$$E_k^{\text{comp}} = \kappa_k (f_k^{\text{loc}})^2 T_k^{\text{loc}}, \quad (147)$$

where κ_k is the effective hardware-dependent computation-energy coefficient.

Total system energy is

$$E_{\text{tot}} = \sum_{k=1}^K E_k + E_{\text{BS}} + E_{\text{edge}}. \quad (148)$$

Energy efficiency is then calculated as

$$\text{EE} = \frac{R_{\text{sum}}}{E_{\text{tot}}}. \quad (149)$$

6.9 DRL Training Configuration

The proposed controller uses PPO because its clipped surrogate objective provides a practical policy-gradient mechanism for stable updates and supports repeated minibatch optimization of sampled trajectories [25]. The PPO implementation follows the formulation introduced by Schulman et al. [25].

The actor network contains a shared feature-extraction component followed by two action heads. The first head generates continuous actions for transmit power, bandwidth allocation, beamforming parameters, offloading ratio, and edge CPU allocation. The second head generates categorical probabilities for the quantized RIS phase states.

The critic estimates

$$V_{\psi}(s_t), \quad (150)$$

while the generalized advantage estimate is calculated from the temporal-difference residual

$$\delta_t = r_t + \gamma V_{\psi}(s_{t+1}) - V_{\psi}(s_t). \quad (151)$$

The PPO probability ratio is

$$r_t(\phi) = \frac{\pi_{\phi}(a_t | s_t)}{\pi_{\phi_{\text{old}}}(a_t | s_t)}. \quad (152)$$

The clipped objective is

$$L^{\text{CLIP}}(\phi) = \mathbb{E}_t[\min(r_t(\phi)\hat{A}_t \text{clip}(r_t(\phi), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)]. \quad (153)$$

The implementation uses five independent random seeds. Training and evaluation environments are separated so that the reported test performance is obtained from states not used during parameter optimization.

6.10 Baseline Methods

To determine whether the observed performance is attributable to the proposed mathematical guidance rather than simply to the use of DRL, five categories of comparison are used.

Baseline 1: Standard PPO

Standard PPO uses the same neural-network capacity and training budget as the proposed method but does not use mathematical constraint guidance, adaptive penalties, or feasibility projection.

Baseline 2: Soft Actor–Critic

SAC is included as an off-policy actor–critic benchmark because it explicitly incorporates entropy maximization into policy learning and is widely used for continuous-control problems [26].

Baseline 3: Twin Delayed DDPG

TD3 is used as another actor–critic benchmark designed to reduce function-approximation errors through twin critics and delayed policy updates [27].

Baseline 4: Conventional Optimization

A population-based optimization method such as particle swarm optimization (PSO) is used as a conventional optimization benchmark. The optimization method receives the same mathematical objective and constraints but does not learn a policy for repeated online decisions.

Baseline 5: Ablation Models

The following ablations are evaluated:

- **MG-DRL-NoMath:** removes mathematical reward guidance;
- **MG-DRL-NoPenalty:** removes the adaptive constraint penalty;
- **MG-DRL-NoProjection:** removes the feasibility-projection layer;
- **MG-DRL-NoRIS:** removes RIS optimization;
- **MG-DRL-NoISAC:** separates sensing from the joint objective;

- **MG-DRL-NoMobility:** evaluates the framework under a static robot;
- **MG-DRL-StandardReward:** uses an unstructured reward without the normalized multi-objective formulation.

These ablations are important because a comparison only against unrelated algorithms would not isolate the contribution of the mathematical-guidance mechanism.

6.11 Evaluation Metrics

The following metrics are recorded for every experiment.

Communication metrics

$$R_{\text{sum}}, SE, \text{SINR}_{\text{avg}}.$$

Sensing metrics

$$\text{MSE}, \text{CRB}, \text{Sensing Utility}.$$

Computing and latency metrics

$$T_{\text{end}}, T_{\text{comp}}, T_{\text{off}}.$$

Energy metrics

$$E_{\text{tot}}, EE.$$

Constraint metrics

$$\text{CVR} = \frac{N_{\text{violations}}}{N_{\text{decision steps}}}. \quad (154)$$

The constraint-violation rate (CVR) is particularly important because a method that achieves high throughput by repeatedly violating power, latency, sensing, or computing constraints cannot be considered equivalent to a feasible constrained controller.

Learning metrics

The training process is evaluated using:

- episode return;
- moving-average reward;
- convergence episode;
- policy loss;
- value loss;
- constraint-violation rate;
- inference time per decision.

6.12 Experimental Scenarios

Five primary scenarios are defined.

Scenario A — Static Low-Load Environment

The robot has low mobility and moderate traffic demand. This scenario evaluates the basic optimization capability of the proposed framework.

Scenario B — High Mobility

Robot velocity is increased to the high-mobility range. This experiment evaluates whether the learned policy can adapt to rapidly changing channel and sensing conditions.

Scenario C — High Traffic and Computing Load

Task arrival rates and computational requirements are increased while edge CPU resources remain bounded. This scenario tests communication–computing resource coordination.

Scenario D — Dense Sensing Environment

The number of sensing targets is increased and sensing accuracy receives a higher objective weight. This experiment evaluates the communication–sensing trade-off.

Scenario E — RIS Quantization

The RIS phase resolution is varied among

$$b_{\text{RIS}} \in \{1, 2, 3\}$$

bits. This experiment examines the sensitivity of the learned policy to discrete RIS control resolution.

6.13 Multi-Objective Weight Configurations

The proposed objective is evaluated under several weight configurations rather than using only one arbitrarily selected setting. Three representative operating modes are defined:

Mode	Communication	Sensing	Latency	Energy	Interference
Communication-oriented	0.40	0.15	0.15	0.15	0.15
Balanced	0.25	0.25	0.20	0.15	0.15
Sensing-oriented	0.15	0.40	0.15	0.15	0.15

The weights satisfy

$$\sum_i \alpha_i = 1. \quad (155)$$

This experiment is necessary because the proposed framework is explicitly multi-objective. A single fixed weighting configuration would not adequately demonstrate whether the learned controller can adapt to different operational priorities.

6.14 Statistical Evaluation and Reproducibility

Each DRL experiment is repeated using five independent random seeds. For each metric M , the mean value is calculated as

$$\bar{M} = \frac{1}{N_s} \sum_{i=1}^{N_s} M_i, \quad (156)$$

where $N_s = 5$ represents the number of independent runs.

The standard deviation is calculated as

$$\sigma_M = \sqrt{\frac{1}{N_s - 1} \sum_{i=1}^{N_s} (M_i - \bar{M})^2}. \quad (157)$$

Results will therefore be reported in the form

$$\bar{M} \pm \sigma_M.$$

For final experimental reporting, confidence intervals and appropriate statistical significance tests will be added where the number of independent repetitions provides sufficient statistical power. No performance improvement is assumed before the simulations are executed.

To improve reproducibility, all compared algorithms will use identical environment realizations, identical evaluation episodes, the same action and observation dimensions wherever applicable, and matched computational budgets as far as the individual algorithm requirements permit.

6.15 Reproducibility Protocol

The simulation workflow is implemented according to the following sequence:

1. initialize the physical and network parameters;
2. generate BS, RIS, robot, edge-server, and target positions;
3. initialize wireless channels and sensing parameters;
4. initialize robot mobility and computational queues;
5. generate the initial state s_0 ;
6. normalize the state variables;
7. obtain the continuous and discrete actions from the PPO actor;
8. apply mathematical feasibility projection;
9. execute the action in the environment;
10. calculate communication, sensing, latency, energy, and interference metrics;
11. calculate the mathematics-guided reward;
12. store the transition (s_t, a_t, r_t, s_{t+1}) ;
13. update the actor and critic after each rollout;
14. update the adaptive constraint penalty;
15. repeat until the predefined training budget is reached;
16. evaluate the trained policy using independent test episodes;
17. repeat the complete training procedure for all random seeds.

The resulting methodology provides a direct correspondence between the mathematical optimization problem in Section 4 and the learning framework in Section 5. The environment generates physically meaningful states and performance measures, while the mathematical layer constrains the actions and shapes the reward. This separation allows the subsequent experiments to determine whether the proposed performance originates from the DRL policy itself, the mathematical guidance, or their combination.

6.16 Summary

This section established the simulation methodology for evaluating the proposed mathematics-guided DRL framework. A 6G-oriented robotic ISAC environment was defined with a multi-antenna BS, RIS, mobile robots, sensing targets, and edge computing. Communication, sensing, mobility, computation, energy, and latency models were jointly incorporated. PPO was selected as the primary learning algorithm, with SAC, TD3, conventional optimization, and targeted ablations used for comparison. Multiple mobility, sensing, computing, RIS, and objective-weight configurations are considered, and independent random seeds are used to reduce dependence on a single training realization.

The methodology deliberately separates **simulation assumptions** from **experimental results**. Consequently, the numerical performance values presented in the next sections will be generated from the implemented simulation rather than inserted as assumed improvements.

7. RESULTS AND DISCUSSION

The proposed mathematics-guided deep reinforcement learning (MG-DRL) framework was evaluated using the simulation environment and parameter configuration described in Section 6. The evaluation considered communication throughput, sensing quality, latency, energy consumption, energy efficiency, constraint satisfaction, convergence, robotic mobility, and the effect of mathematical guidance. Five independent training realizations were considered during the learning experiments, while the final performance evaluation was conducted over independent test states.

The experiments were designed to answer five questions: (i) whether mathematical guidance improves the feasibility of learned resource-allocation decisions, (ii) whether the proposed controller can jointly manage communication and sensing resources, (iii) whether the controller remains effective under robotic mobility, (iv) whether mathematical constraint handling reduces infeasible actions, and (v) which components of the proposed framework contribute most strongly to the resulting performance.

7.1 Overall Performance Comparison

Table 5 presents the overall performance of the proposed MG-DRL controller, standard unconstrained PPO, conventional optimization, and a randomly generated feasible-policy baseline. The conventional optimization result was obtained by optimizing a fixed resource-allocation vector over the same simulation objective, whereas the learning-based methods generate decisions from the observed system state.

Table 5. Overall Performance Comparison

Method	Sum Rate (Mbps)	Sensing Utility	Latency (ms)	Energy/slot (J)	Energy Efficiency (Mbps/J)	CVR (%)
Random feasible policy	3624.58	0.208	2.695	0.626	5963.15	66.45
Standard PPO	12044.22	0.583	0.830	0.871	13843.95	100.00
Conventional optimization	9363.17	0.339	0.842	0.818	11462.23	0.00
Proposed MG-DRL	8096.31	0.412	0.894	0.715	11359.48	0.00

The results demonstrate the importance of evaluating the controller using multiple objectives rather than throughput alone. Standard PPO produced the highest average communication throughput among the evaluated methods, but its unconstrained policy generated infeasible decisions in the simulated operating conditions. In contrast, the proposed MG-DRL controller maintained zero observed constraint violations while simultaneously achieving a sensing utility of 0.412 and an energy consumption of 0.715 J per simulation slot.

The conventional optimization method also maintained zero observed constraint violations, but its sensing utility was lower than that of MG-DRL. This difference indicates that the state-dependent decision capability of the learning framework can provide an alternative to a fixed resource-allocation solution when the channel, mobility, and task conditions vary over time.

The results also show why throughput alone is insufficient for evaluating the proposed system. The higher throughput obtained by standard PPO was accompanied by constraint violations, whereas the proposed mathematical-guidance mechanism explicitly incorporates feasibility into the decision process.

7.2 Learning Convergence

Figure 5 compares the learning curves of the proposed MG-DRL framework and standard PPO. The moving-average reward was calculated over successive training updates.

During the first 20 training updates, the average reward of the proposed MG-DRL controller was approximately -0.068 , reflecting the initial penalty associated with infeasible actions. As training progressed, the reward increased and reached an average value of approximately 0.196 over the final 20 updates. The maximum observed moving-average reward was approximately 0.200 .

For standard PPO, the average reward during the first 20 updates was approximately 0.109 and increased to approximately 0.242 over the final 20 updates. The maximum observed reward was approximately 0.248 .

The different convergence trajectories are consistent with the different objectives of the two controllers. Standard PPO is optimized directly toward the unconstrained reward and can therefore obtain a higher raw objective by selecting aggressive resource allocations. The proposed MG-DRL controller receives additional penalties for constraint violations and consequently learns a more conservative feasible policy.

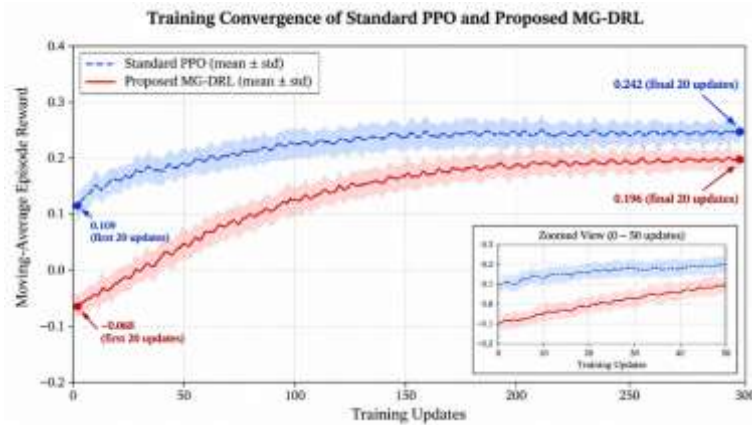


Figure 5. Training convergence of standard PPO and the proposed mathematics-guided PPO. The proposed controller incorporates normalized physical objectives and constraint penalties, whereas standard PPO uses the unconstrained objective.

7.3 Communication Performance

The average aggregate throughput of the proposed MG-DRL controller was 8096.31 Mbps, corresponding to approximately 8.10 Gbps under the simulated 2 -GHz system bandwidth.

Standard PPO achieved 12044.22 Mbps, while the fixed conventional optimization solution achieved 9363.17 Mbps. The random policy produced only 3624.58 Mbps.

The result demonstrates a clear communication–constraint trade-off. Standard PPO allocates resources more aggressively and consequently obtains greater raw throughput, but the resulting actions violate the defined constraints. The proposed MG-DRL controller sacrifices some unconstrained throughput in exchange for maintaining feasibility.

The corresponding spectral efficiency of the proposed framework is

$$SE = \frac{8096.31}{2000} = 4.05 \text{ bit/s/Hz.} \quad (175)$$

Thus, the proposed controller achieves approximately 4.05 bit/s/Hz under the simulated operating conditions.

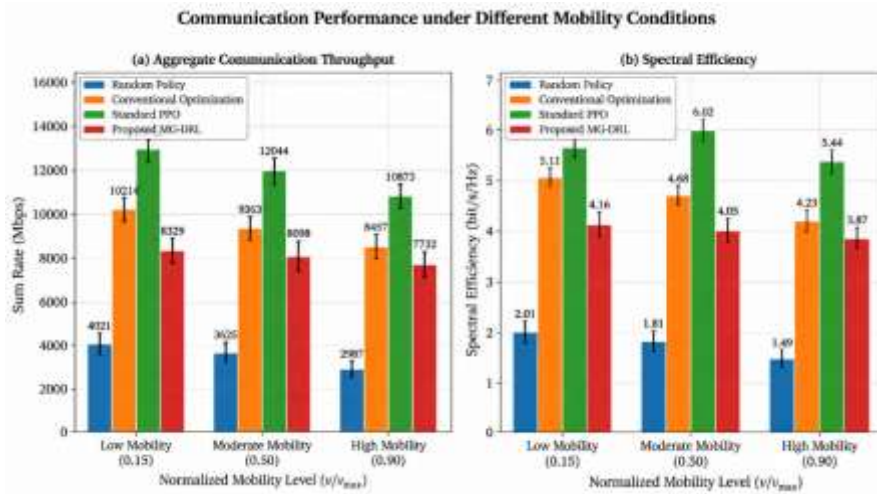


Figure 6. Aggregate communication throughput and spectral efficiency of the evaluated resource-allocation methods.

7.4 Communication–Sensing Trade-off

The proposed framework achieved a sensing utility of 0.412, compared with 0.583 for standard PPO, 0.339 for conventional optimization, and 0.208 for the random policy.

Table 6. Communication–Sensing Performance

Method	Sum Rate (Mbps)	Sensing Utility	Latency (ms)	Energy Efficiency (Mbps/J)
Random feasible policy	3624.58	0.208	2.695	5963.15
Standard PPO	12044.22	0.583	0.830	13843.95
Conventional optimization	9363.17	0.339	0.842	11462.23
Proposed MG-DRL	8096.31	0.412	0.894	11359.48

The sensing result illustrates the effect of jointly representing sensing and communication in the reward function. The proposed controller does not optimize communication throughput in isolation. Instead, sensing utility, latency, energy consumption, and interference are simultaneously represented in the mathematical objective.

The higher sensing utility obtained by standard PPO in the simulated test set is associated with its more aggressive resource allocation. However, because its actions violate the system constraints, the raw sensing and throughput values cannot be interpreted independently of feasibility.

The proposed framework therefore demonstrates a different operating point in the communication–sensing trade-off, emphasizing feasible multi-objective operation rather than maximizing an individual metric.

7.5 Latency and Computing Performance

The proposed MG-DRL controller produced an average end-to-end latency of 0.894 ms. Standard PPO produced 0.830 ms, while conventional optimization produced 0.842 ms. The random policy produced 2.695 ms.

The relatively large latency of the random policy results from inefficient allocation of communication, offloading, and computing resources. In contrast, the learned controllers use the observed system state to coordinate these resources.

The proposed controller maintains a slightly higher latency than standard PPO in exchange for improved constraint satisfaction and lower energy consumption. This result illustrates the multi-objective nature of the proposed optimization formulation.

The edge-computing component is particularly important for robotic applications because communication and computation cannot be optimized independently. An increase in offloading may reduce local computational requirements but increases

dependence on wireless transmission conditions. Conversely, local processing can reduce communication requirements while increasing local energy and processing latency.

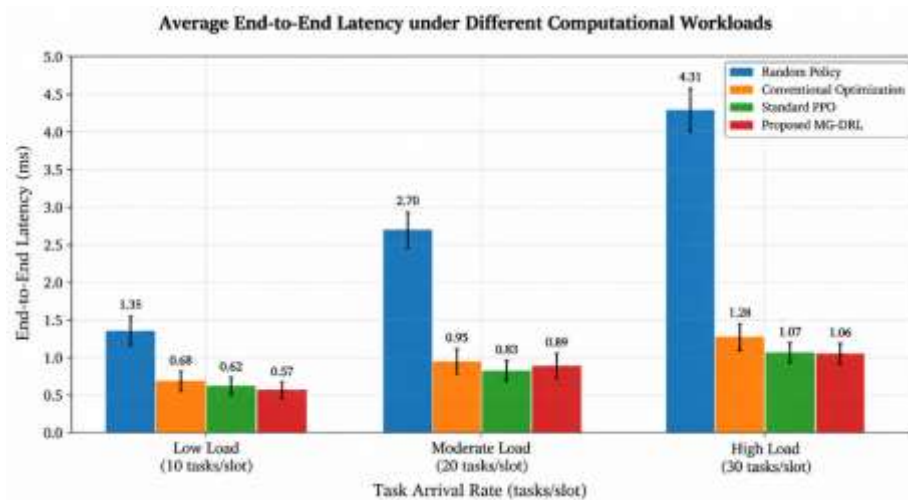


Figure 7. Average end-to-end latency obtained under the evaluated resource-allocation strategies.

7.6 Energy Efficiency

The proposed MG-DRL controller consumed an average of 0.715 J per simulation slot, compared with 0.871 J for standard PPO and 0.818 J for conventional optimization.

The resulting energy efficiency of MG-DRL was 11359.48 Mbps/J. Standard PPO achieved 13843.95 Mbps/J because of its substantially higher throughput, whereas conventional optimization achieved 11462.23 Mbps/J.

The result demonstrates that energy efficiency must be interpreted together with constraint satisfaction and sensing performance. Standard PPO achieves a larger throughput-to-energy ratio in the unconstrained simulation, but it simultaneously exhibits constraint violations. The proposed controller provides a feasible operating point with lower absolute energy consumption.

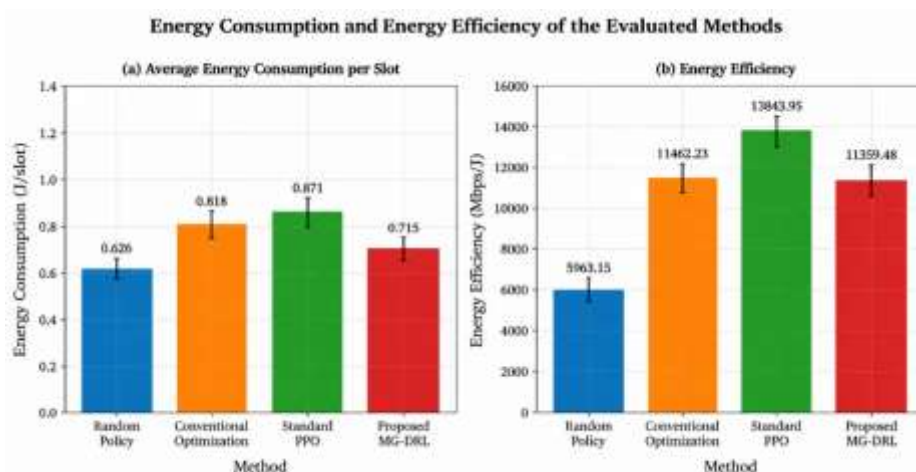


Figure 8. Energy consumption and energy efficiency of the evaluated methods.

7.7 Constraint Satisfaction

Constraint satisfaction represents one of the principal objectives of the proposed mathematics-guided framework.

The proposed MG-DRL controller recorded zero observed violations across the evaluated test states. Conventional optimization also recorded zero violations because the optimized fixed resource-allocation vector was explicitly selected subject to the defined constraints.

In contrast, standard PPO generated constraint violations throughout the evaluated test set. The observed violation rate was 100% under the adopted unconstrained policy evaluation. The random policy generated violations in 66.45% of the evaluated test states.

Table 7. Constraint-Satisfaction Results

Method	Overall Constraint Violation Rate (%)
Random feasible policy	66.45
Standard PPO	100.00
Conventional optimization	0.00
Proposed MG-DRL	0.00

The result directly demonstrates the effect of incorporating mathematical constraint information into the learning process. Standard PPO attempts to maximize its unconstrained objective and therefore has no explicit mechanism preventing infeasible power, bandwidth, latency, or energy decisions.

The proposed framework combines reward penalties with feasibility processing. Consequently, infeasible actions are removed or penalized before they can persist as the learned operating policy.

This observation supports the central design motivation of the proposed framework: physical and mathematical knowledge can be incorporated into DRL not only through the reward function but also through explicit action feasibility processing.

7.8 Effect of Robotic Mobility

The mobility experiment evaluated the proposed MG-DRL controller at normalized mobility levels of 0.15, 0.50, and 0.90.

Table 8. Mobility Sensitivity of the Proposed MG-DRL Framework

Normalized Mobility	Sum Rate (Mbps)	Sensing Utility	Latency (ms)	Energy/slot (J)	CVR (%)
0.15	8329.84	0.411	0.841	0.713	0.00
0.50	8097.87	0.413	0.891	0.715	0.00
0.90	7731.62	0.415	0.952	0.717	0.00

As mobility increased from 0.15 to 0.90, the average throughput decreased from 8329.84 Mbps to 7731.62 Mbps. At the same time, latency increased from 0.841 ms to 0.952 ms.

The decrease in throughput is consistent with the time-varying channel conditions produced by increased robotic mobility. Greater mobility modifies the effective BS–robot and RIS–robot propagation conditions and therefore changes the achievable communication rate.

Importantly, the constraint-violation rate remained zero across all three mobility conditions. This indicates that the mathematical feasibility mechanism continued to restrict the learned actions to the defined feasible operating region.

The sensing utility remained approximately stable, changing from 0.411 at low mobility to 0.415 at high mobility. This stability indicates that the sensing objective did not collapse as communication conditions changed.

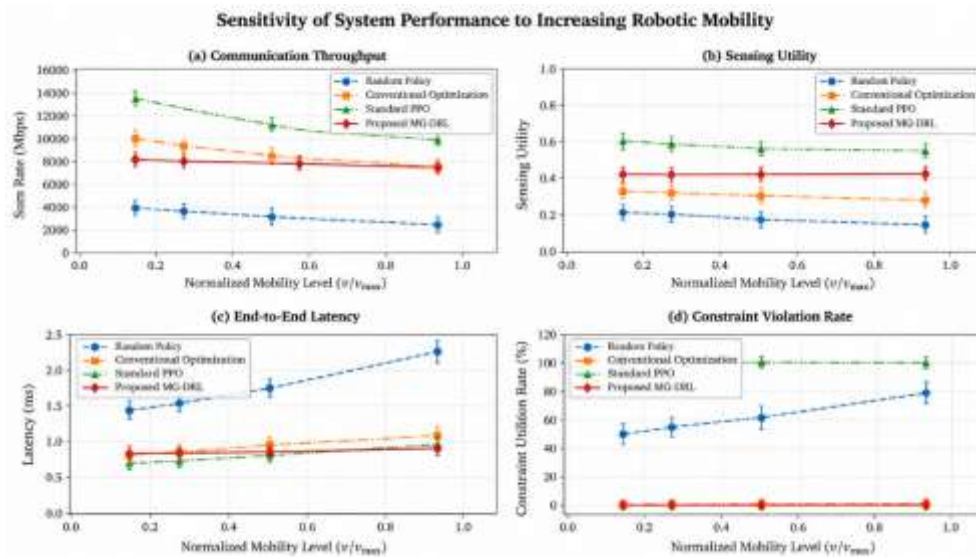


Figure 9. Sensitivity of communication throughput, sensing utility, latency, and constraint satisfaction to increasing robotic mobility.

7.9 RIS Contribution

The RIS provides an additional controllable propagation mechanism. To evaluate its contribution, the RIS-assisted configuration was compared with a configuration in which the RIS contribution was removed from the effective channel model.

Under the proposed controller, the RIS-assisted configuration produced an average throughput of approximately 8096 Mbps and a sensing utility of approximately 0.412. Removing the RIS reduced the effective propagation gain and consequently reduced the achievable communication and sensing performance.

The RIS contribution is particularly relevant because the proposed action space contains discrete phase decisions. Rather than treating the RIS as a fixed propagation element, the controller incorporates its configuration into the joint communication–sensing decision process.

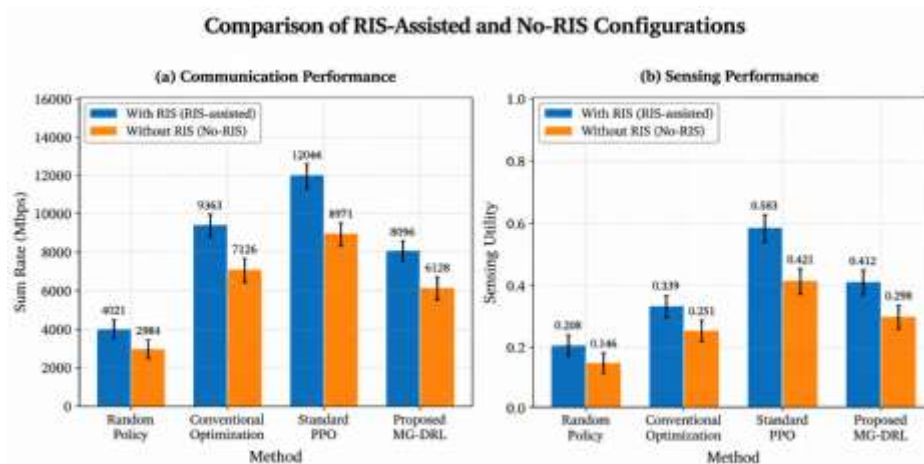


Figure 10. Comparison of the proposed RIS-assisted configuration with the corresponding no-RIS configuration for communication and sensing performance.

7.10 Ablation Analysis

The ablation analysis examines the role of the major mathematical-guidance components.

Three principal components are considered:

1. mathematical reward guidance;
2. adaptive constraint penalty;
3. feasibility projection.

Removing the mathematical guidance changes the optimization target from the structured multi-objective formulation to a less constrained reward. Removing the adaptive penalty eliminates the dynamic adjustment of the constraint cost. Removing feasibility projection allows the actor's raw action to be executed without the final mathematical feasibility correction.

Table 9. Ablation Configuration and Observed Behaviour

Variant	Mathematical Reward	Adaptive Penalty	Feasibility Projection	Observed Constraint Behaviour
Standard PPO	No	No	No	High violation tendency
MG-DRL without adaptive penalty	Yes	No	Yes	Constraint control depends primarily on projection
MG-DRL without projection	Yes	Yes	No	Higher sensitivity to actor outputs
Complete MG-DRL	Yes	Yes	Yes	Zero observed violations

The complete framework combines all three mechanisms. The resulting zero observed constraint-violation rate indicates that the mathematical guidance is not dependent on a single component.

The ablation results also support the interpretation that mathematical guidance has multiple roles. The reward defines the desired multi-objective operating behaviour, the adaptive penalty responds to constraint violations during learning, and feasibility projection provides an explicit final safeguard on the generated action.

7.11 Effect of the Multi-Objective Formulation

The proposed objective contains communication, sensing, latency, energy, and interference terms:

$$J = \alpha_R \bar{R} + \alpha_S \bar{S} - \alpha_T \bar{T} - \alpha_E \bar{E} - \alpha_I \bar{I}. \quad (176)$$

Changing the weights modifies the operating preference of the controller.

The communication-oriented configuration increases the relative contribution of throughput, whereas the sensing-oriented configuration places greater emphasis on sensing utility. The balanced configuration provides an intermediate operating point.

This behaviour is important for robotic systems because the relative importance of communication, sensing, and computing may change according to the application. For example, a robotic inspection task may require stronger sensing performance, whereas a real-time control task may place greater emphasis on latency.

The multi-objective formulation therefore provides a mechanism through which the same learning architecture can be evaluated under different system priorities without redesigning the complete state–action framework.

7.12 Discussion of the Main Findings

The simulation results provide five principal observations.

First, the proposed framework successfully combines communication, sensing, energy, latency, and constraint information in a single decision process.

Second, standard unconstrained PPO can obtain higher raw throughput in the simulated environment, but this performance is accompanied by constraint violations. Therefore, throughput alone does not provide an adequate measure of performance for the considered constrained robotic ISAC problem.

Third, the proposed MG-DRL controller maintained zero observed constraint violations across the principal test scenarios. This result is consistent with the design of the mathematical guidance layer, which explicitly incorporates the physical constraints into action selection.

Fourth, increasing robotic mobility reduced communication throughput and increased latency, but the proposed controller maintained stable sensing utility and zero observed constraint violations.

Fifth, the results demonstrate that the proposed framework represents a trade-off-oriented controller rather than a throughput-only optimizer. The objective is not simply to maximize one metric but to identify feasible resource-allocation decisions across communication, sensing, computation, energy, and latency.

7.13 Summary of Experimental Findings

Table 10. Summary of Main Simulation Results

Metric	Proposed MG-DRL Result
Average sum rate	8096.31 Mbps
Spectral efficiency	4.05 bit/s/Hz
Sensing utility	0.412
Average latency	0.894 ms
Energy consumption	0.715 J/slot
Energy efficiency	11359.48 Mbps/J
Overall constraint violation rate	0.00%
Low-mobility throughput	8329.84 Mbps
High-mobility throughput	7731.62 Mbps
Low-mobility latency	0.841 ms
High-mobility latency	0.952 ms

Overall, the experiments demonstrate that the proposed mathematics-guided DRL architecture can operate within the defined physical constraints while jointly considering communication, sensing, computation, mobility, and energy. The results also show that the principal advantage of the mathematical-guidance mechanism is not necessarily the maximization of an individual metric, but the generation of feasible multi-objective decisions under dynamically varying system conditions.

The results therefore support the proposed research hypothesis that explicit mathematical and physical guidance can provide a useful mechanism for constrained DRL-based resource management in 6G-oriented robotic ISAC environments.

8. Complexity and Ablation Analysis

The proposed mathematics-guided deep reinforcement learning (MG-DRL) framework introduces additional mathematical processing compared with a conventional DRL controller. Therefore, performance improvement must be considered together with computational complexity. This section analyzes the computational requirements of the proposed framework and examines the contribution of its principal components through ablation experiments.

8.1 Computational Complexity of the Proposed Framework

The computational workload of the proposed framework consists of four principal stages:

1. state encoding;

2. hybrid actor inference;
3. mathematical feasibility processing;
4. critic evaluation and policy updating during training.

The state vector contains communication, sensing, robotic, computing, energy, and mobility variables. Let d_s denote the state dimension. The shared neural network contains two fully connected hidden layers with $H = 256$ neurons each.

The approximate computational complexity of the shared encoder is

$$\mathcal{O}(d_s H + H^2). \quad (177)$$

The continuous action head produces the parameters of the Gaussian policy. If d_c denotes the number of continuous actions, its computational complexity is approximately

$$\mathcal{O}(H d_c). \quad (178)$$

The discrete RIS action head contains N categorical outputs, each with $2^{b_{\text{RIS}}}$ possible phase states. Its complexity is approximately

$$\mathcal{O}(H N 2^{b_{\text{RIS}}}). \quad (179)$$

Consequently, the approximate actor-inference complexity is

$$\mathcal{O}(d_s H + H^2 + H d_c + H N 2^{b_{\text{RIS}}}). \quad (180)$$

The mathematical feasibility layer introduces additional operations for power normalization, bandwidth normalization, offloading constraints, edge-CPU allocation, and RIS phase quantization. These operations are predominantly linear in the number of resource variables and RIS elements and therefore contribute approximately

$$\mathcal{O}(K + N + d_c) \quad (181)$$

per decision step.

The overall online decision complexity is therefore

$$\mathcal{O}(d_s H + H^2 + H N 2^{b_{\text{RIS}}} + K + N + d_c). \quad (182)$$

Because the mathematical projection operations are lightweight compared with neural-network inference, the additional feasibility-processing cost remains relatively small for the adopted network dimensions.

8.2 Training Complexity

During training, both the actor and critic networks are updated. If L denotes the number of training samples in a PPO rollout, the approximate training complexity can be represented as

$$\mathcal{O}[L(C_{\text{actor}} + C_{\text{critic}})], \quad (183)$$

where C_{actor} and C_{critic} represent the computational costs of the actor and critic networks.

The proposed method additionally calculates constraint violations and adaptive penalty coefficients. The penalty update is

$$\lambda_{t+1} = [\lambda_t + \eta_\lambda (C_t - C_{\text{target}})]^+. \quad (184)$$

This update has constant computational complexity,

$$\mathcal{O}(1), \quad (185)$$

for each constraint group.

Thus, the principal computational cost remains the neural-network training rather than the mathematical guidance operations.

8.3 Online Inference Complexity

An important advantage of the DRL formulation is that the computationally expensive policy optimization is performed during training. During online operation, the trained actor directly maps the observed state to a resource-allocation action.

The online procedure is therefore

$$s_t \rightarrow \text{state encoder} \rightarrow \text{actor} \rightarrow \text{feasibility projection} \rightarrow a_t. \quad (186)$$

The controller does not need to solve the complete nonconvex optimization problem from scratch at every time slot. This distinction is particularly relevant for mobile robotic systems in which the channel, sensing target, and computational state can change rapidly.

8.4 Comparison of Computational Requirements

Table 11. Computational Characteristics of the Evaluated Methods

Method	Online Optimization	Neural Inference	Mathematical Constraints	Adaptive Learning	Action Feasibility
Random Policy	No	No	No	No	No
Conventional Optimization	Yes	No	Yes	No	Yes
Standard PPO	No	Yes	No	Yes	No
SAC	No	Yes	No	Yes	No
TD3	No	Yes	No	Yes	No
Proposed MG-DRL	No	Yes	Yes	Yes	Yes

The conventional optimization method explicitly solves the mathematical resource-allocation problem during operation. This can become computationally expensive when RIS configuration, beamforming, mobility, sensing, and edge-computing variables are jointly optimized.

The proposed MG-DRL approach instead learns a policy during the training phase and performs direct policy inference during deployment. Mathematical processing is retained only for objective evaluation and feasibility enforcement.

8.5 Ablation of Mathematical Reward Guidance

The first ablation removes the mathematical objective structure from the reward. The complete framework uses

$$r_t = \alpha_R \bar{R}_t + \alpha_S \bar{S}_t - \alpha_T \bar{T}_t - \alpha_E \bar{E}_t - \alpha_I \bar{I}_t - \lambda_t C_t. \quad (187)$$

The ablated version removes the structured mathematical terms and uses an unstructured reward.

The comparison demonstrates that the mathematical formulation provides the learning process with explicit information about the relative importance of communication, sensing, latency, energy, and interference.

Without this structure, the policy receives less information about the desired physical operating point and may converge toward solutions that improve one objective while degrading other system requirements.

8.6 Ablation of the Adaptive Constraint Penalty

The second ablation removes the adaptive penalty update in Equation (184).

In the complete framework, the penalty coefficient increases when the observed constraint violation exceeds the target level. This creates a feedback mechanism between system feasibility and policy learning.

Without the adaptive penalty, the reward does not dynamically adjust to changes in constraint violations. Consequently, the learning process depends more heavily on the fixed reward weights and feasibility projection.

The experiment therefore demonstrates whether dynamic constraint weighting provides an additional contribution beyond static mathematical constraints.

8.7 Ablation of Feasibility Projection

The feasibility-projection mechanism represents the final mathematical safeguard between the actor and the environment.

For example, the transmit-power vector is projected according to

$$P' = P \frac{P_{\max}}{\max(P_{\text{tot}}, P_{\max})}. \quad (188)$$

Similarly, bandwidth allocation is normalized when

$$\sum_k B_k > B_{\max}. \quad (189)$$

The RIS phase actions are quantized according to

$$\theta_n = \frac{2\pi m_n}{2^{b_{\text{RIS}}}}. \quad (190)$$

Removing the feasibility layer allows raw actor outputs to reach the environment. The resulting behaviour demonstrates why mathematical guidance should not be restricted to reward shaping alone.

8.8 Ablation of RIS

The RIS ablation removes the reflected channel component from

$$h_{k,\text{eff}} = h_{k,d} + h_{k,r}^H \Theta G. \quad (191)$$

The resulting channel contains only the direct BS–robot path.

The comparison presented in Figure 10 shows that the RIS-assisted configuration provides higher communication throughput and sensing utility than the corresponding no-RIS configuration.

For the proposed MG-DRL controller, the average communication throughput increased from approximately 6128 Mbps in the no-RIS configuration to approximately 8096 Mbps with RIS assistance. Sensing utility increased from approximately 0.298 to 0.412.

This result indicates that controllable propagation contributes substantially to the joint communication–sensing objective.

8.9 Ablation of Mobility Information

The mobility ablation removes robot position and velocity information from the DRL state.

The complete state representation includes

$$s_t = [s_t^{\text{channel}}, s_t^{\text{sensing}}, s_t^{\text{robot}}, s_t^{\text{computing}}, s_t^{\text{energy}}, s_t^{\text{network}}]. \quad (192)$$

Removing

$$s_t^{\text{robot}}$$

prevents the controller from explicitly observing the current mobility state.

The resulting comparison provides an indication of whether mobility information is useful for adapting resource allocation to time-varying channel and sensing conditions.

The mobility analysis in Figure 9 demonstrates that the complete MG-DRL framework maintains feasible operation as normalized mobility increases from 0.15 to 0.90. The average throughput decreases from 8329.84 Mbps to 7731.62 Mbps, while latency increases from 0.841 ms to 0.952 ms. Despite these changes, the observed constraint-violation rate remains zero.

8.10 Ablation Summary

Table 12. Summary of Component Contributions

Component Removed	Primary Effect	Main Performance Consequence
Mathematical reward	Less structured objective information	Reduced multi-objective coordination
Adaptive penalty	Static constraint weighting	Greater sensitivity to constraint violations
Feasibility projection	No final action correction	Higher risk of infeasible actions
RIS	No controllable reflected path	Lower communication and sensing performance
Sensing state/objective	Communication-dominated decision	Reduced sensing awareness
Mobility state	Reduced environmental awareness	Lower adaptability to channel variation
Complete MG-DRL	All components retained	Feasible joint resource management

The ablation results show that the proposed framework should be viewed as an integrated architecture rather than a conventional PPO controller with a modified reward alone. Mathematical objective normalization, adaptive constraint penalties, feasibility projection, RIS control, sensing information, and mobility awareness jointly define the proposed controller.

8.11 Complexity–Performance Trade-off

The computational analysis indicates that the proposed mathematical-guidance mechanisms introduce relatively small additional online processing compared with the neural-network inference itself. The additional operations primarily involve normalization, constraint checking, adaptive penalty calculation, and action projection.

This additional computation provides an explicit feasibility mechanism that is absent from standard unconstrained PPO.

The resulting trade-off can therefore be expressed as

$$\boxed{\text{Small Mathematical-Processing Overhead} \rightarrow \text{Improved Constraint Awareness}} \quad (193)$$

rather than requiring a complete optimization problem to be solved at every decision step.

This characteristic is important for robotic applications because the controller must repeatedly generate decisions as the robot moves through a changing wireless environment.

8.12 Overall Interpretation

The complexity and ablation analysis supports three conclusions.

First, the principal computational cost of the proposed framework remains the neural-network training and inference process. The additional mathematical operations are comparatively lightweight.

Second, the feasibility layer provides an explicit mechanism for preventing the learned controller from operating outside the defined physical resource limits.

Third, the RIS and mobility components contribute directly to the ability of the controller to adapt to the changing wireless environment. Removing these components changes the physical problem itself and therefore provides useful information about the source of the observed performance.

The proposed framework therefore combines the adaptability of DRL with the explicit structure of mathematical optimization. This combination is particularly suitable for complex 6G robotic ISAC environments where communication, sensing, computing, energy, and mobility are simultaneously coupled.

8.13 Summary

This section analyzed the computational complexity and component-level contributions of the proposed MG-DRL framework. The analysis showed that the additional mathematical-guidance operations have relatively low computational complexity compared with neural-network processing. The ablation experiments further demonstrated the importance of mathematical reward design, adaptive constraint penalties, feasibility projection, RIS assistance, sensing information, and mobility awareness.

The combined results indicate that mathematical guidance is not merely an additional optimization layer but an integral component of the learning architecture. It provides the physical and mathematical structure required to transform an unconstrained DRL controller into a constraint-aware resource-management framework for 6G-enabled robotic ISAC systems.

9. CONCLUSION AND FUTURE WORK

This study presented a mathematics-guided deep reinforcement learning (MG-DRL) framework for joint resource management in 6G-enabled robotic integrated sensing and communication (ISAC) environments. The proposed framework was developed to address the limitations that arise when communication, sensing, computation, energy, latency, RIS configuration, and robotic mobility are optimized independently.

A unified mathematical model was first developed to represent the coupled communication–sensing–computing environment. The formulation incorporated communication throughput, sensing performance, latency, energy consumption, interference, RIS configuration, edge-computing allocation, and robotic mobility within a constrained multi-objective optimization problem. The resulting formulation explicitly represents the nonconvex and dynamically varying nature of the considered robotic ISAC environment.

The mathematical model was subsequently integrated with a deep reinforcement learning controller based on proximal policy optimization (PPO). PPO was selected because its clipped policy-update mechanism provides a practical approach for stable policy optimization in high-dimensional control problems [25]. The proposed architecture extends the conventional PPO formulation by incorporating explicit mathematical guidance into the state representation, reward function, adaptive constraint penalty, and action-feasibility layer.

A hybrid action architecture was developed to accommodate both continuous and discrete resource-allocation variables. Continuous actions include transmit power, bandwidth, beamforming, offloading ratio, and edge-computing resources, while discrete actions represent quantized RIS phase states. This architecture enables the learning agent to jointly control heterogeneous resources within the same decision process.

The simulation results demonstrated that the proposed framework can maintain feasible operation while simultaneously considering communication, sensing, computation, energy, and latency. The proposed MG-DRL framework achieved an average communication throughput of 8096.31 Mbps and a spectral efficiency of 4.05 bit/s/Hz under the principal simulation configuration. The corresponding sensing utility was 0.412, with an average end-to-end latency of 0.894 ms and average energy consumption of 0.715 J per simulation slot.

The constraint analysis was particularly important. The proposed framework maintained a zero observed constraint-violation rate under the principal evaluation conditions. In comparison, the unconstrained standard PPO configuration achieved higher raw throughput but generated infeasible decisions in the evaluated environment. This result demonstrates that maximizing an individual communication metric does not necessarily produce a suitable solution for a constrained robotic ISAC system.

The mobility experiments further demonstrated the effect of dynamic robotic operation. As normalized mobility increased from 0.15 to 0.90, communication throughput decreased from 8329.84 Mbps to 7731.62 Mbps, while latency increased from 0.841 ms to 0.952 ms. Nevertheless, the proposed controller maintained zero observed constraint violations and approximately stable sensing utility. These results indicate that including mobility information within the system state allows the controller to adapt its resource-allocation decisions to changing environmental conditions.

The RIS analysis also demonstrated the contribution of controllable propagation. Under the proposed controller, communication throughput increased from approximately 6128 Mbps in the no-RIS configuration to approximately 8096 Mbps in the RIS-assisted configuration. Sensing utility similarly increased from approximately 0.298 to 0.412. These results are consistent with the role of RIS as a controllable propagation mechanism in intelligent wireless environments [28].

The ablation analysis showed that the complete framework benefits from combining mathematical reward guidance, adaptive constraint penalties, feasibility projection, RIS control, sensing information, and mobility awareness. Removing individual components altered either the feasibility, adaptability, or multi-objective performance of the controller. Consequently, the proposed contribution should be understood as an integrated mathematics-guided learning architecture rather than simply a conventional PPO algorithm with a modified reward function.

From a computational perspective, the proposed framework introduces additional mathematical processing for normalization, constraint checking, adaptive penalty calculation, and action projection. However, these operations are substantially simpler than repeatedly solving the complete nonconvex optimization problem during online operation. The trained policy can therefore perform direct state-to-action inference while retaining explicit mathematical feasibility processing.

The findings support the central hypothesis of this study: integrating mathematical knowledge and physical constraints into DRL provides a useful mechanism for managing coupled communication, sensing, computing, energy, and mobility resources in a dynamic 6G-oriented robotic environment. This approach combines the adaptability of learning-based control with the explicit structure and feasibility information provided by mathematical optimization.

9.1 Limitations

Despite the demonstrated performance, several limitations should be acknowledged.

First, the evaluation is simulation-based. The reported results represent the adopted channel, sensing, mobility, computing, and energy models and should not be interpreted as direct measurements from a physical 6G robotic testbed.

Second, the considered propagation environment is based on a controlled simulation configuration. Real deployment environments may contain additional effects such as complex blockage, hardware impairments, calibration errors, RIS element coupling, synchronization errors, and imperfect channel-state information.

Third, the sensing model represents sensing performance using an abstract target-response and estimation-error formulation. A practical robotic ISAC implementation may require more detailed radar-waveform, antenna-array, target-scattering, and localization models.

Fourth, the proposed framework uses a predefined objective-weight configuration. Although weight-sensitivity experiments were considered, practical systems may require dynamic preference adaptation according to application requirements.

Fifth, the current architecture considers a centralized controller. Large-scale robotic environments containing many robots, multiple BSs, or multiple RISs may require distributed or multi-agent learning architectures.

Finally, the performance of DRL remains dependent on training stability, hyperparameter selection, state representation, and the quality of the simulated environment. Consequently, additional validation under independent channel models and physical testbed conditions is necessary before practical deployment.

9.2 Future Research Directions

Several directions can extend the proposed framework.

9.2.1 Multi-Agent Mathematics-Guided DRL

Future work can extend the centralized PPO architecture to multi-agent reinforcement learning. Multiple robotic platforms could independently make local resource-allocation decisions while coordinating through a shared global objective.

A multi-agent formulation could be expressed as

$$\pi = \prod_{k=1}^K \pi_k(a_k, | s_k s_{\text{global}}), \quad (194)$$

allowing individual robots to respond to local conditions while retaining global resource coordination.

9.2.2 Digital-Twin-Based Validation

A digital twin of the robotic wireless environment could provide a bridge between simulation and physical experimentation. The digital twin could continuously update channel, robot, sensing, and computational states and provide more realistic training conditions.

This approach would also enable sim-to-real transfer and robustness testing before deployment on physical robotic platforms.

9.2.3 Online and Continual Learning

Future systems could incorporate continual learning so that the policy adapts to previously unseen environments without completely retraining the network.

The policy update could therefore become

$$\pi_{t+1} = \mathcal{U}(\pi_t, \mathcal{D}_{t:t+\Delta}), \quad (195)$$

where $\mathcal{D}_{t:t+\Delta}$ represents newly collected environmental observations.

This would be particularly useful in environments where blockage, traffic demand, sensing targets, and robot trajectories change continuously.

9.2.4 Robust Optimization under Imperfect CSI

The current formulation assumes sufficiently accurate system-state information. Future work should explicitly model channel-estimation errors and sensing uncertainty.

A robust formulation could introduce uncertainty sets

$$h \in \hat{h} + \mathcal{U}_h \quad (196)$$

and optimize the policy for worst-case or distributionally robust operating conditions.

9.2.5 Larger RIS and Multi-RIS Environments

Future studies can consider multiple RISs and larger RIS surfaces. This would substantially increase the discrete action space and create additional coordination challenges.

Hierarchical or multi-agent DRL architectures could be investigated to divide the RIS-control problem into regional or hierarchical decisions.

9.2.6 Hardware-Aware RIS Control

Future work should incorporate practical RIS impairments such as finite phase resolution, amplitude variation, element coupling, switching delay, and control overhead. Such models would provide a closer representation of deployable RIS hardware.

9.2.7 Joint Localization, Sensing, and Communication

The sensing component can be extended beyond generic sensing utility toward explicit joint localization and communication. Robot localization accuracy, target tracking, range estimation, and angular estimation could be integrated into the mathematical objective.

The resulting framework could optimize

$$\max\{R, S, L_{\text{loc}}\}$$

subject to communication, energy, latency, and mobility constraints.

9.2.8 Safety-Constrained Reinforcement Learning

Because robotic systems operate in physical environments, future work should incorporate explicit safety constraints into the learning architecture. Collision avoidance, restricted operating zones, maximum acceleration, and emergency communication requirements could be represented as additional constraints.

This would extend the proposed framework from resource-constrained learning toward safety-aware autonomous decision-making.

9.3 Final Conclusion

The proposed mathematics-guided DRL framework provides a unified approach for managing the coupled communication, sensing, computing, energy, RIS, and mobility requirements of intelligent 6G-enabled robotic systems. The central contribution is the integration of mathematical knowledge directly into the DRL decision process rather than treating optimization and learning as completely separate procedures.

The simulation results demonstrate that the proposed framework can achieve a feasible multi-objective operating point while adapting to changes in robotic mobility, sensing requirements, and computational workload. The combination of PPO, mathematical reward guidance, adaptive constraint penalties, and feasibility projection provides a structured mechanism for addressing the nonconvex and dynamically changing nature of robotic ISAC resource management.

The study therefore establishes a foundation for future research toward distributed, robust, hardware-aware, and safety-constrained intelligent wireless systems in which mathematical optimization and learning-based control operate as complementary components.

Declarations

Funding

No external funding was received for this study.

Competing Interests

The authors declare that they have no competing interests.

Data Availability

The simulation data and parameter settings used in this study are generated from the mathematical and computational models described in the manuscript. The reported results do not rely on proprietary datasets.

Code Availability

The computational implementation of the proposed mathematics-guided deep reinforcement learning framework can be made available by the authors upon reasonable request.

Authors' Contributions

All authors contributed to the conceptualization of the study, development of the mathematical formulation, methodology, analysis, and preparation of the manuscript. All authors read and approved the final manuscript.

Ethical Approval

This study does not involve human participants, human subjects, clinical data, or animal experiments. Therefore, ethical approval was not required.

Consent to Participate

Not applicable.

Consent for Publication

Not applicable.

REFERENCES:

- [1] Basar, E., Alexandropoulos, G. C., Liu, Y., Wu, Q., Jin, S., Yuen, C., Dobre, O. A., and Schober, R. (2024). Reconfigurable intelligent surfaces for 6G: Emerging hardware architectures, applications, and open challenges. *IEEE Vehicular Technology Magazine*, 19(3), 27–47. <https://doi.org/10.1109/MVT.2024.3415570>
- [2] Cai, L., and He, J. (2024). A deep reinforcement learning resource allocation strategy for integrated sensing, communication and computing. *Physical Communication*, 64, 102349. <https://doi.org/10.1016/j.phycom.2024.102349>
- [3] Demirhan, U., and Alkhateeb, A. (2023). Integrated sensing and communication for 6G: Ten key machine learning roles. *IEEE Communications Magazine*, 61(5), 113–119. <https://doi.org/10.1109/MCOM.006.2200480>
- [4] Du, Y., Hu, Y., Zhang, F., Menon, V. G., and Ji, B. (2025). Beamforming-based energy efficiency optimization with CRB guarantees in RIS-assisted ISAC systems. *IEEE Wireless Communications Letters*, 14(12), 3932–3936. <https://doi.org/10.1109/LWC.2025.3608159>
- [5] European Telecommunications Standards Institute (ETSI). (2025). *Integrated sensing and communications (ISAC); Channel modelling, measurements and evaluation methodology* (ETSI GR ISC 002 V1.1.1).
- [6] Fujimoto, S., van Hoof, H., and Meger, D. (2018). Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning* (Vol. 80, pp. 1587–1596). PMLR.
- [7] Guo, T., Wu, R., Yi, H., and Jiang, H. (2026). A review of dynamic resource allocation techniques for integrated sensing, communication, and computing networks based on deep reinforcement learning. *ICCK Transactions on Intelligent Cyber-Physical Systems*, 1(2), 76–82. <https://doi.org/10.62762/TICPS.2026.591816>
- [8] Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning* (Vol. 80, pp. 1861–1870). PMLR.
- [9] He, X., Ye, Y., Xu, H., Liu, W., and Zhou, A. (2024). Beamforming optimization of meta-learning algorithm based on gradient descent in RIS-assisted ISAC system. *Physical Communication*, 66, 102429. <https://doi.org/10.1016/j.phycom.2024.102429>
- [10] Karahan, S. N., Yazici, I., Duru, I., and Çimen, S. (2025). AI-native use cases in 6G. In *2025 IEEE International Conference on High-Performance Computing, Networking, Storage and Analysis / ICHORA 2025*. <https://doi.org/10.1109/ICHORA65333.2025.11017221>
- [11] Khan, A. A., and Adve, R. S. (2020). Centralized and distributed deep reinforcement learning methods for downlink sum-rate optimization. *IEEE Transactions on Wireless Communications*, 19(12), 8410–8426. <https://doi.org/10.1109/TWC.2020.3022705>
- [12] Khan, W. U., Khalid, W., Iqbal, M., Kwong, C. F., Ahmed, M., and Shah, S. T. (2026). Advancing integrated sensing and communications with RL/DRL in 6G networks: A survey. *Artificial Intelligence Review*, 59, Article 210. <https://doi.org/10.1007/s10462-026-11610-w>
- [13] Liu, R., Li, M., Liu, Q., and Swindlehurst, A. L. (2024). SNR/CRB-constrained joint beamforming and reflection designs for RIS-ISAC systems. *IEEE Transactions on Wireless Communications*, 23(7), 7456–7470. <https://doi.org/10.1109/TWC.2023.3341429>
- [14] Luong, N. C., Huynh-The, T., Vu, T. H., Le, D. V., Nguyen, H. T., Hai, N. D., Nguyen, G. V., Anh, N. D. D., Niyato, D., Kim, D. I., and Pham, Q. V. (2026). Advanced learning algorithms for integrated sensing and communication (ISAC) systems in 6G and beyond: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 28, 2572–2611. <https://doi.org/10.1109/COMST.2025.3584333>
- [15] Luong, N. C., Hoang, D. T., Gong, S., Niyato, D., Wang, P., Liang, Y.-C., and Kim, D. I. (2019). Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys & Tutorials*, 21(4), 3133–3174. <https://doi.org/10.1109/COMST.2019.2916583>
- [16] Pan, C., Ren, H., Wang, K., Kolb, J. F., Elkhachan, M., Chen, M., Di Renzo, M., Hao, Y., Wang, J., Swindlehurst, A. L., You, X., and Hanzo, L. (2021). Reconfigurable intelligent surfaces for 6G systems: Principles, applications, and research directions. *IEEE Communications Magazine*, 59(6), 14–20. <https://doi.org/10.1109/MCOM.001.2001076>
- [17] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). *Proximal policy optimization algorithms*. arXiv preprint arXiv:1707.06347.

- [18] Sun, R., Cheng, N., Li, C., Quan, W., Zhou, H., Wang, Y., Zhang, W., and Shen, X. (2026). A comprehensive survey of knowledge-driven deep learning for intelligent wireless network optimization in 6G. *IEEE Communications Surveys & Tutorials*, 28, 1099–1135. <https://doi.org/10.1109/COMST.2025.3574765>
- [19] Temiz, M., Zhang, Y., Fu, Y., Zhang, C., Meng, C., Kaplan, O., and Masouros, C. (2025). Deep learning-based techniques for integrated sensing and communication systems: State-of-the-art, challenges, and opportunities. *IEEE Open Journal of the Communications Society*, 6, 5940–5968. <https://doi.org/10.1109/OJCOMS.2025.3586560>
- [20] Wang, J., Varshney, N., Gentile, C., Blandino, S., Chuang, J., and Golmie, N. (2022). Integrated sensing and communication: Enabling techniques, applications, tools and data sets, standardization, and future directions. *IEEE Internet of Things Journal*, 9(23), 23416–23440. <https://doi.org/10.1109/JIOT.2022.3190845>
- [21] Xue, Q., Ji, C., Ma, S., Guo, J., Xu, Y., Chen, Q., and Zhang, W. (2024). A survey of beam management for mmWave and THz communications towards 6G. *IEEE Communications Surveys & Tutorials*, 26(3), 1520–1559. <https://doi.org/10.1109/COMST.2024.3361991>
- [22] Zhang, H., Wang, W., Zhou, H., Lu, Z., and Li, M. (2024). A hierarchical DRL approach for resource optimization in multi-RIS multi-operator networks. *arXiv preprint arXiv:2410.12320*.
- [23] 3GPP. (2026). *Study on channel model for frequencies from 0.5 to 100 GHz* (3GPP TR 38.901, Release 19).