ENGINEERING Information Technology & Electronic Engineering
https://www.academax.com/EITEE; https://www.engineering.org.cn
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=11344065
E-mail: eng.itee@zju.edu.cn

Engineering Journals
Available on IEEE Xplore®

Research Article

https://doi.org/10.1631/ENG.ITEE.2026.0047

An adaptive meta-reinforcement learning scheme for maritime dynamic spectrum access and service scheduling

Zhongyang MAO[1,2,3], Jiahuan GENG[1,2,3]✉, Faping LU[1,2,3], Wenbiao TIAN[1,2,3], Jiafang KANG[1,2,3], Yaozong PAN[1,2,3]

[1]*Naval Aviation University, Yantai 264001, China*

[2]*National Key Laboratory of Millimeter-Wave and Terahertz Remote Sensing, Yantai 264001, China*

[3]*Shandong Key Laboratory of Sea and Air Information Perception and Processing Technology, Yantai 264001, China*

Abstract: The rapid proliferation of marine economic activities has led to explosive growth in maritime communication demands. Consequently, the scarcity of spectrum resources, the highly dynamic spectrum environment, and diverse service requirements have become increasingly critical issues. While conventional deep reinforcement learning (DRL) methods perform well for specific training scenarios, they fail to generalize effectively to unknown environments. To address this challenge, this paper investigates a maritime dynamic spectrum access and service scheduling scheme based on adaptive meta-reinforcement learning. First, we construct a channel occupancy model that encompasses both Markov frequency-hopping mode and spread-spectrum frequency-hopping mode. By incorporating a dual-queue mechanism for urgent and normal data packets, we formulate the multi-agent cooperative decision-making problem as a decentralized partially observable Markov decision process (POMDP). Second, to overcome the limited generalization capability of traditional DRL algorithms, we propose a spectrum access and service scheduling algorithm based on a meta-learning paradigm (MetaSASS), which facilitates rapid policy transfer through meta-training across a distribution of tasks. Furthermore, to resolve the adaptation efficiency bottleneck caused by fixed inner-loop learning rates, we design an adaptive meta-reinforcement learning SASS algorithm (AMRLSASS). This algorithm employs a task encoding module to dynamically adjust learning rates, thereby balancing convergence speeds across tasks of varying complexities. Simulation results demonstrate that the proposed AMRLSASS algorithm outperforms baseline algorithms across a series of metrics within unknown environments and effectively validate the superiority of the proposed method in complex maritime environments. The algorithm proposed in this paper provides an effective solution for the rapid adaptation of intelligent spectrum access and service scheduling to unknown scenarios in maritime wireless communication systems.

Key words: Dynamic spectrum access; Adaptive meta-reinforcement learning; Service scheduling; Maritime communication

1 Introduction

Driven by the increasing frequency of maritime economic activities and the rapid evolution of marine wireless technologies, maritime communication traffic has surged exponentially, placing unprecedented pressure on limited spectrum resources (Yang et al., 2022; Nomikos et al., 2023). However, due to the pronounced "tidal effect" of user traffic, exclusive licensing of certain frequency bands, and physical limitations in signal coverage, significant "spectrum holes" remain underutilized within licensed spectra. In critical scenarios such as maritime emergency command and search-and-rescue operations, communication demands often arise abruptly and urgently. Existing systems largely rely on idle channels in unlicensed or shared frequency bands, resulting in highly uncertain and dynamic spectrum access.

For instance, the very high frequency (VHF) maritime mobile band (156–174 MHz), allocated by the International Telecommunication Union (ITU) for maritime communications under the Radio Regulations (ITU, 2024), is essential for distress, safety, and operational coordination. Within this band, certain channels are designated for high-priority safety and emergency services, while other portions of the spectrum may be underutilized or designated for non-safety applications. The ITU has recognized the need for more efficient spectrum utilization and has issued recommendations such as ITU-R M.1842-1-2009 (ITU, 2009), which specifies the characteristics of VHF radio systems for data exchange in maritime mobile service channels, and ITU-R M.1084-5-2012 (ITU, 2012), which outlines interim

✉ Jiahuan GENG, echo_HAU@163.com
Zhongyang MAO, https://orcid.org/0000-0001-6279-1627
Jiahuan GENG, https://orcid.org/0009-0004-9031-0690
Faping LU, https://orcid.org/0000-0002-8172-0311
Wenbiao TIAN, https://orcid.org/0000-0002-3558-7767
Jiafang KANG, https://orcid.org/0000-0003-4177-5622
Yaozong PAN, https://orcid.org/0000-0002-4442-6332

CLC number: TN929.5

Received: Feb. 13, 2026; Revision accepted: June 4, 2026;
Crosschecked: June 12, 2026; Published online: Aug. 4, 2026

solutions for improving spectrum efficiency in this band. This evolving regulatory landscape—moving from static frequency assignments toward more flexible, dynamic sharing paradigms—creates a pressing need for intelligent spectrum management solutions.

Beyond the challenge of spectrum scarcity, the traffic generated in these maritime scenarios is highly heterogeneous. Communication traffic in rescue scenes exhibits notable heterogeneity: high-priority emergency command and rescue signaling impose stringent requirements on transmission reliability and low latency, whereas routine data transmission and media services prioritize high throughput (Saggese et al., 2022; Lyu et al., 2024). Therefore, in complex maritime environments characterized by limited spectrum resources, time-varying channel conditions, and diverse service types, intelligently sensing and accessing spectrum holes while providing differentiated quality-of-service guarantees for varying-priority services has become a pressing issue for improving both spectrum efficiency and communication reliability.

Dynamic spectrum sharing technology, which allows secondary users to opportunistically access unused "spectrum holes" of licensed users, is key to enhancing spectrum utilization. Traditional access strategies based on fixed rules or statistical regularities struggle to cope with such complex and dynamic environments, often leading to low spectrum resource utilization and difficulty in ensuring communication quality, especially for high-dimensional nonconvex problems that are hard to solve directly (So and Soya, 2023; Khisa et al., 2025). In recent years, intelligent learning approaches such as deep reinforcement learning (DRL) have provided new perspectives by optimizing decisions through autonomous interaction with the environment. Moreover, DRL has been shown to outperform traditional static methods in addressing resource allocation challenges in heterogeneous networks, significantly improving spectral efficiency and interference suppression (Upadhyay et al., 2025). Centralized intelligent spectrum coordination algorithms have also been proposed to enable efficient multiplexing of shared spectrum resources among multiple secondary users (SUs) (Atimati et al., 2023, 2025). Despite these advances, most existing models assume the availability of complete or ideal channel state information (Albinsaid et al., 2022; Yuan et al., 2024), which is difficult to obtain with high accuracy and low latency in practical maritime communication environments, posing serious challenges for real-world deployment. Consequently, some researchers have modeled the dynamic spectrum sharing problem as a decentralized partially observable Markov decision process (POMDP) (Sheng TQ et al., 2022; Zhang SG et al., 2023) and employed improved DRL methods to solve it.

To address cooperation and competition among multiple users in wireless communication networks, multi-agent DRL has become a focus of research. Several studies have concentrated on joint optimization of spectrum sensing and access, aiming to reduce inter-user interference through multi-agent collaboration (Zhang YL et al., 2023; Li XH et al., 2024). Researchers have leveraged various algorithms, including deep Q-networks (DQNs) (Ke et al., 2025), multi-agent deep deterministic policy gradient (MADDPG) (Li YZ et al., 2020; Zhang X et al., 2023), proximal policy optimization (PPO) (Wang et al., 2026), twin delayed deep deterministic policy gradient (TD3) (Dong et al., 2022), long short-term memory (LSTM) (Feng et al., 2023; Li YH et al., 2023), and federated learning (Yang et al., 2023), to address multi-agent dynamic spectrum access, channel collisions, and improvements in system throughput and spectrum efficiency. Although these methods have advanced multi-user cooperation, most still assume relatively regular or quasi-static channel dynamics, an idealized assumption far removed from the highly dynamic maritime environment. Traditional DRL models are typically trained for specific tasks and are prone to overfitting; when faced with changes in environmental statistics (e.g., channel distribution and traffic load), their performance can degrade sharply or even lead to failure, necessitating extensive retraining for new tasks—a process incompatible with the stringent reliability and real-time demands of maritime emergency communication.

To mitigate the weak generalization and slow adaptation of DRL strategies in new environments, transfer learning and meta-learning techniques have attracted growing attention. As an example of transfer learning, Sheng HM et al. (2024) proposed a transfer-DQN scheme that reuses knowledge via a weighted experience replay mechanism, partially alleviating performance fluctuations across environment switches; however, its generalization remains limited to tasks that are similar across different environments.

In contrast, meta-reinforcement learning adopts a "learning to learn" paradigm, aiming to enable models to rapidly adapt to new tasks with only a few samples by leveraging prior task experience, thus offering a promising approach for cross-task adaptation. For instance, Lu and Gursoy (2021) employed a model-agnostic meta-learning (MAML) framework to achieve fast adaptation to tasks with varying channel transition probabilities. Similar advantages have been observed in vehicular networks (Huang et al., 2022; Kai et al., 2025) and edge computing scenarios (Niu et al., 2023; Jia et al., 2024). Overall, meta-learning has proven effective in enhancing policy generalization across diverse tasks and environments.

Extending the above meta-reinforcement learning (Meta-RL) approaches to heterogeneous networks and diverse services, several recent studies have designed general access protocols based on meta-learning and mixture-of-experts architectures, achieving fast convergence and near-centralized performance in unseen scenarios (Liu et al., 2023, 2024, 2026). Despite these promising developments, classic meta-learning methods often employ a fixed learning rate during multi-task training. This uniform treatment fails to account for varying task difficulties and cannot dynamically respond to changes in environment and task requirements. Although optimizing the meta-optimizer learning rate via cosine annealing has been shown to improve generalization (Antoniou et al., 2019), this scheduling strategy causes the learning rate to monotonically decrease during training, thereby failing to adapt to actual task dynamics.

In summary, traditional optimization methods and classical reinforcement learning approaches exhibit limitations—such as overidealized model assumptions, insufficient generalization capability, and heavy reliance on prior information—when applied to highly dynamic maritime environments. While meta-learning techniques show promise in rapid adaptation, they lack a tailored mechanism for task-specific adjustment in the complex scenario of maritime multi-agent cooperative spectrum access with differentiated service guarantees. To address these challenges, this paper focuses on dynamic and complex maritime wireless communication scenarios and investigates an adaptive Meta-RL scheme for spectrum access and service scheduling. The aim is to develop an intelligent scheduling framework capable of fast adaptation to environmental changes, supporting multi-user cooperation,

and achieving efficient cross-scene generalization. The main contributions and innovations are summarized as follows:

1. A joint optimization model for heterogeneous dynamic spectrum access and multi-priority service scheduling is constructed, tailored to the requirements of maritime wireless communications. Incorporating realistic maritime spectrum characteristics, a channel occupancy model covering both Markov frequency-hopping pattern and frequency-hopping spread-spectrum (FHSS) pattern is established. A dual-queue buffer mechanism for urgent and normal data packets is designed to accommodate diverse service demands. The joint decision-making problem is formulated as a decentralized POMDP, with a multi-objective reward function that balances transmission success rate, channel conflict suppression, and queue congestion penalty.

2. A meta-reinforcement learning-based architecture for dynamic spectrum access and service scheduling is proposed, effectively overcoming the poor generalization of traditional DRL algorithms. To mitigate their reliance on specific environmental statistics and limited adaptability to unknown scenarios, a meta-learning paradigm is introduced, leading to a spectrum access and service scheduling algorithm based on a meta-learning paradigm (MetaSASS). By meta-training over a wide distribution of heterogeneous tasks, the agent learns general spectrum variation patterns, enabling fast policy fine-tuning with only a few interaction samples when confronted with new channel states and traffic loads. This significantly enhances the algorithm adaptability and reliability in non-stationary maritime spectrum environments.

3. An adaptive meta-learning rate control mechanism based on task-characteristic awareness is designed to further improve learning efficiency and generalization across multiple tasks. To address the slow convergence on simple tasks caused by fixed inner-loop learning rates in MetaSASS, an adaptive Meta-RL SASS algorithm, AMRLSASS, is proposed. The algorithm explicitly perceives the complexity and dynamic nature of the current task through a task-encoding estimation module and dynamically generates the inner-loop learning rate. This mechanism aligns policy update steps with task characteristics, accelerates the convergence of the meta-learning process, and achieves efficient adaptation to complex and varying communication environments.

2 Problem formulation

2.1 Network model

This paper studies the dynamic spectrum access and service scheduling problem in maritime emergency rescue communication scenarios. As illustrated in Fig. 1, the system consists of primary users (PUs) dynamically occupying a subset of K orthogonal channels, alongside M SUs serving as intelligent agents representing different maritime communication nodes with time-varying communication demands. At each decision time slot, each SU can continuously sense L channels ($L<K$). The action of the agent is a discrete value indicating the starting index of the channel selected for aggregation,

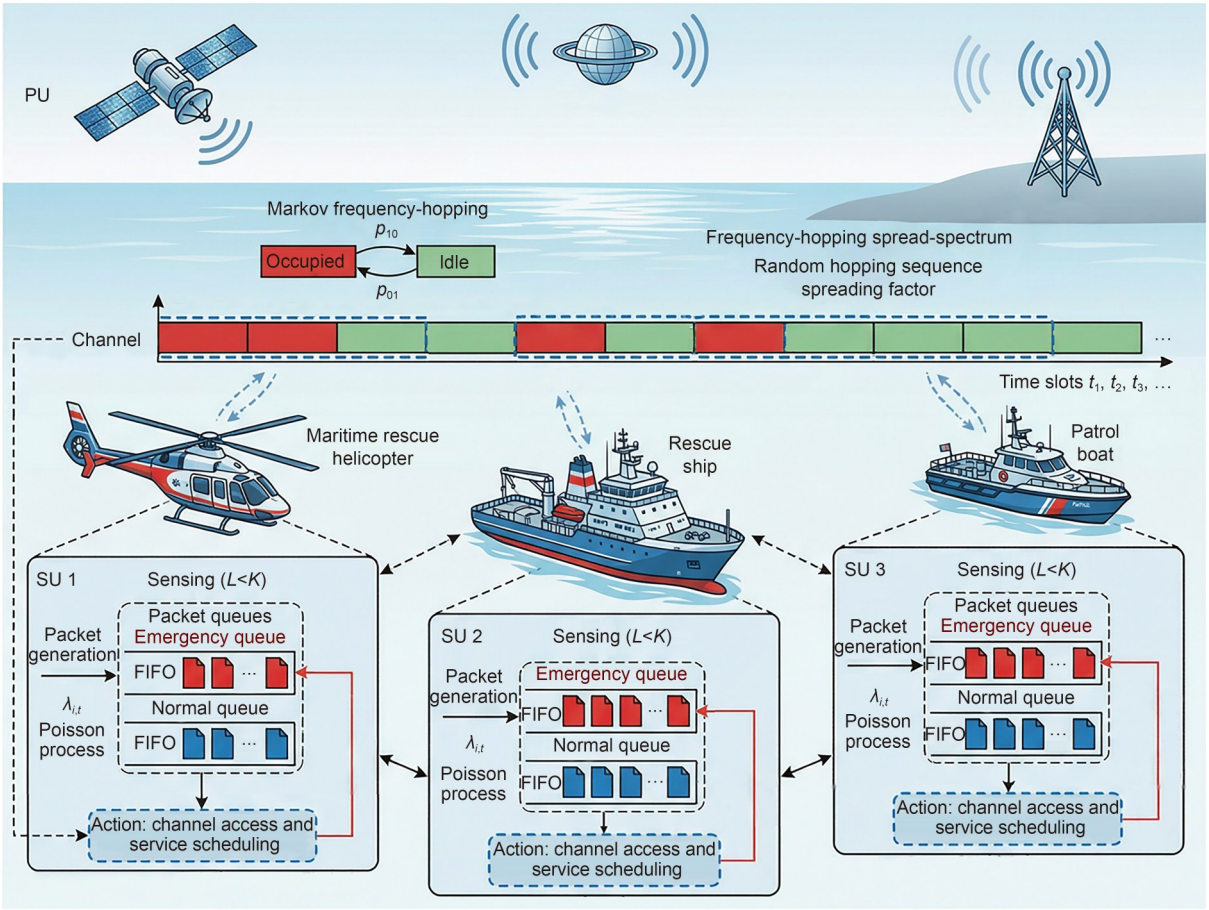

Fig. 1 Architecture of dynamic spectrum access and service scheduling for maritime emergency rescue

thereby determining the channel state available for transmission in the current slot. Due to the lack of global channel information, each agent must make decisions based solely on local observations—including the states of the L sensed channels and its own communication requirements—making this a typical partially observable decision-making process.

Multiple agents operate in an unknown and time-varying channel environment, performing autonomous spectrum access decisions based on their respective communication needs. This constitutes a decentralized dynamic spectrum sharing and multi-objective optimization problem. Since multiple agents compete for limited spectral resources, conflicts arise if they simultaneously select the same channel, leading to degraded communication performance. Therefore, agents must engage in cooperative sensing and distributed decision-making to complete data transmission tasks over channels not occupied by PU, thereby achieving efficient utilization of spectrum resources and reliable communication services.

2.2 Channel model

Accurate modeling of the spectrum occupancy pattern is fundamental to dynamic spectrum access and service scheduling decisions, as it directly impacts the learning efficiency and environmental adaptability of intelligent agents. Existing studies commonly employ stochastic process models to describe the evolution of channel states, with the Markov chain (Li YZ et al., 2020; Sheng TQ et al., 2022) and frequency-hopping sequence model (Rao et al., 2024, 2025) being two typical representatives, corresponding to probabilistic occupancy and sequential hopping spectrum dynamics, respectively. Integrating existing modeling approaches from the literature, this paper abstracts the channel occupancy patterns into the following two categories:

1. Markov frequency-hopping pattern. Under this pattern, the channel state evolution follows a two-state discrete-time Markov chain, where PU occupies channels with certain probabilities. The state of channel c at time slot t is denoted by $\chi_c^t \in \{0, 1\}$, where 0 indicates idle and 1 indicates occupied. The state transition probability matrix $\boldsymbol{P}$ is defined as

$$\boldsymbol{P} = \begin{bmatrix} p_{00} & 1-p_{00} \\ p_{10} & 1-p_{10} \end{bmatrix}, \tag{1}$$

where p_{00} represents the probability that a channel remains idle, and p_{10} denotes the probability that a channel transitions from occupied to idle. The state transition probabilities determine the channel's dynamic characteristics. Different probabilities reflect different channel variation patterns and directly affect communication quality. For instance, in highly dynamic environments, a lower p_{00} and a higher p_{10} lead to frequent channel state changes, necessitating more frequent spectrum access switching by agents to improve spectrum utilization efficiency.

2. FHSS pattern. In this pattern, PUs occupy channels according to a predefined frequency-hopping sequence. The channel state randomly follows a spectrum occupancy pattern defined by a hopping sequence υ and a spreading factor ς, commonly used in wireless sensor networks for anti-jamming transmission. The occupancy state of channel c_υ^t at time slot t is given by

$$\chi_c^t = \begin{cases} 1, & \text{if } c_\upsilon^t \in \left\{ \upsilon^t - \frac{\varsigma}{2}, ..., \upsilon^t + \frac{\varsigma}{2} \right\} \cap \{0, K-1\}, \\ 0, & \text{else.} \end{cases} \tag{2}$$

2.3 Communication traffic model

Maritime emergency rescue services exhibit significant priority differentiation. Each SU generates data packets with varying communication requirements at different times. The generated packets differ in terms of inter-arrival time, size, and timeout period. SUs must dynamically schedule different types of data packets for transmission based on spectrum access decisions, subject to limited spectrum resources and time constraints.

1. Packet generation. The packet generation process simulates the burstiness and priority characteristics of traffic in different scenarios. Packets are categorized into urgent packets (e.g., distress signals and critical commands) and normal packets (e.g., routine communications and Internet services). Following Saggese et al. (2021), the number of packets $N_{i,\text{type}}^t$ generated by SU i at time slot t follows a Poisson process with parameter $\lambda_{i,\text{type}}^t$, denoted as $N_{i,\text{type}}^t \sim \text{Possion}(\lambda_{i,\text{type}}^t)$. The arrival rates for urgent and normal packets are denoted as λ_e and λ_n, respectively.

2. Packet queue and priority. Each SU maintains two independent first-in first-out (FIFO) queues to buffer different types of packets. The queue state $\boldsymbol{q}^t = (l_\text{e}^t, l_\text{n}^t, \tau_\text{e}^t, \tau_\text{n}^t)$ is part of the agent's observation, where l_e^t and l_n^t represent the lengths of the urgent and normal packet queues respectively, while τ_e^t and τ_n^t denote the remaining time-to-live for the two packet types.

3. Packet transmission and discard. Based on its action a_i, an SU selects a starting channel for access and obtains a transmission opportunity from its set of available idle channels. A packet is forcibly discarded if it fails to be transmitted successfully within its time-to-live, incurring a corresponding penalty in the reward function. This mechanism ensures low-latency requirements for critical information.

2.4 System model

This paper formulates the problem of multi-agent dynamic spectrum access and service scheduling optimization as a decentralized POMDP. Each agent aims to learn an optimal policy that maximizes its long-term cumulative reward through continuous interaction with the environment, without access to global state information—such as the true status of all channels or other agents' queue states. The proposed reinforcement learning framework is designed as follows:

1. State space. Each agent i has a local and partially observable observation space $\boldsymbol{o}_i^t$, which includes an encoding of the environmental features sensed at the current time slot h_i^t, the agent's actual action taken in the previous slot a_i^{t-1}, the queue lengths and timeout levels for both urgent and normal data packets q_i^t, and the proportion of urgent packets in the queue u_i^t. The global state $\boldsymbol{s}^t$ encompasses the state information of all agents.

$$\boldsymbol{o}_i^t = [\, h_i^t, a_i^{t-1}, q_i^t, u_i^t \,], \tag{3}$$

$$\boldsymbol{s}^t = [\, \boldsymbol{o}_1^t, \boldsymbol{o}_2^t, ..., \boldsymbol{o}_M^t \,]. \tag{4}$$

2. Action space. The joint action is denoted as $\boldsymbol{a}^t = [\, a_1^t, a_2^t, ..., a_M^t \,]$, where the action a_i^t of agent i at time slot t represents the index of the starting channel selected for sensing.

3. Reward function. The reward function is designed to address multiple optimization objectives, including (1) maximizing the successful transmission rate, thereby prioritizing high-priority urgent packets while considering normal packet transmission, (2) minimizing channel conflicts by coordinating channel selection among agents, thereby improving spectrum utilization efficiency, and (3) minimizing queue congestion and timeout, thereby controlling queue lengths to reduce packet loss due to expiration. To achieve this multi-objective optimization, the reward r_i^t obtained by agent i at time slot t is formulated as

$$\begin{aligned} r_i^t = {} & \omega_{\text{urgent\_succ}} N_{i,\text{urgent\_succ}}^t + \omega_{\text{normal\_succ}} N_{i,\text{normal\_succ}}^t \\ & - \omega_{\text{coll}} N_{i,\text{coll}}^t - \omega_{\text{urgent\_timeout}} N_{i,\text{urgent\_timeout}}^t \\ & - \omega_{\text{normal\_timeout}} N_{i,\text{normal\_timeout}}^t . \end{aligned} \tag{5}$$

Herein $\omega_{\text{urgent\_succ}}$ and $\omega_{\text{normal\_succ}}$ are reward weights for successfully transmitted urgent and normal packets, respectively; ω_{coll} is the penalty weight for channel conflicts; $\omega_{\text{urgent\_timeout}}$ and $\omega_{\text{normal\_timeout}}$ are penalty weights for the timeout of urgent and normal packets, respectively; $N_{i,\text{urgent\_succ}}^t$ and $N_{i,\text{normal\_succ}}^t$ are the numbers of urgent and normal packets successfully transmitted, respectively; $N_{i,\text{coll}}^t$ is the number of channel conflicts; $N_{i,\text{urgent\_timeout}}^t$ and $N_{i,\text{normal\_timeout}}^t$ are the numbers of urgent and normal packets dropped due to timeout for agent i at time slot t, respectively.

The immediate reward r_i^t reflects a comprehensive trade-off among multiple objectives, guiding each agent to optimize system performance. It incorporates rewards for successful transmissions of different packet types, penalties for multi-access conflicts on the same idle channel, and penalties for packet drops due to timeout. The weighting coefficients ω_* are carefully tuned to balance these objectives, adhering to the emergency communication principle of "prioritizing emergency services while maintaining task completion rate and resource utilization efficiency."

Based on the above formulation, in a time-varying and uncertain channel environment, multi-agent distributed cooperative decision-making is employed to maximize the long-term cumulative system utility. Each agent's objective is to maximize its expected discounted cumulative reward:

$$R_i = \max \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_i^t\right], \tag{6}$$

where $\gamma \in [0, 1)$ is the discount factor, T is the total number of time slots, and $\mathbb{E}[\cdot]$ denotes the expectation over the environment's stochastic transition dynamics and the agent's policy.

3 Dynamic spectrum access and service scheduling scheme

3.1 Meta-reinforcement learning-based algorithm: MetaSASS

The proposed MetaSASS algorithm builds upon the MADDPG framework. MADDPG is an actor–critic method that adopts a centralized training with decentralized execution (CTDE) paradigm: during training, a centralized critic leverages global information to learn coordinated policies, whereas during execution, each agent relies only on local observations to make independent decisions. This CTDE architecture is particularly well-suited to the dynamic spectrum access and service scheduling problem with the POMDP formulation in Section 2, where agents must cooperate under partial observability to avoid collisions and manage shared spectrum resources.

In the MADDPG framework, each agent i maintains an actor network π_{θ_i} and a critic network Q_{ϕ_i}, with target networks θ_i' and ϕ_i' updated via soft updates with coefficient τ. The network is trained using a learning rate α_{critic}. Transitions are stored in a shared replay buffer $\mathcal{D}$. With these notations, the baseline MADDPG procedure is provided in Algorithm S1 in the supplementary materials.

However, while MADDPG provides a robust foundation for multi-agent coordination, it suffers from a critical limitation in the context of maritime communications: its policy adaptation is inherently slow. When confronted with new channel dynamics or varying traffic conditions, standard MADDPG requires extensive online interaction to adjust its policy parameters. This lack of cross-task generalization motivates the integration of meta-learning techniques.

More broadly, optimal policies derived from conventional multi-agent reinforcement learning are typically tailored to specific training environments. Consequently, these policies can guarantee only optimal decisions in scenarios that closely match the training distribution. Due to the inherent partial observability of the real-world settings, dynamic inter-agent interactions, and the time-varying nature of environmental parameters, a model trained on a specific scenario often suffers from performance degradation when deployed in an unseen environment. In such cases, the existing policy becomes ineffective, requiring the model to relearn from a substantial number of new interaction samples, highlighting a fundamental lack of generalization capability across diverse tasks.

To address this critical limitation, MetaSASS introduces a meta-learning paradigm that equips the agents with the capability for rapid policy adaptation. The core idea of meta-learning is to train an agent across a diverse set of tasks that share an underlying distribution of spectrum dynamics and scheduling requirements but exhibit distinct characteristics. Through this process, the agent learns not only how to make intelligent decisions within a single specific task but also, more importantly, acquires the ability to rapidly adapt its policy based on limited interaction experiences from a new environment. This equips the agent's policy network with the crucial capacity for fast adaptation to the features of previously unknown environments.

In the meta-learning framework, the computation of per-step rewards follows the same principle as in traditional MADDPG, where the environment calculates the immediate reward based on the current state and action. The key distinction lies in the collection of experience across multiple tasks, each characterized by its own set of environmental parameters (e.g., channel dynamics, packet arrival rates, and timeout thresholds). To enhance the agents' adaptability in highly dynamic and variable maritime environments, the objective for each agent is to maximize the task-adapted cumulative reward across this distribution of tasks. This is formalized as

$$R_i' = \max_{\theta} \mathbb{E}_{\mathcal{T}\sim p(\mathcal{T})}\left[\mathbb{E}_{\mathcal{T}_j\sim p(\mathcal{T}_j|\theta_i',\mathcal{T})}\sum_{t=1}^{T}\gamma^t r_{i,\mathcal{T}_j}^t\right], \tag{7}$$

where $p(\mathcal{T})$ represents the task distribution, with each task $\mathcal{T}_j$ corresponding to a specific configuration of environmental parameters. $r^t_{i,\mathcal{T}_j}$ denotes the immediate reward for agent i at time slot t in task $\mathcal{T}_j$. θ represents the meta-policy network parameters of the agent, while θ'_i refers to the task-specific parameters obtained after a rapid adaptation process using a small amount of experience from task τ. The core optimization objective is to learn optimal initial parameters θ^* that facilitate efficient adaptation. The algorithm operates through a bilevel optimization process consisting of inner and outer loops, as outlined in Algorithm S2 in the supplementary materials.

1. Inner loop. The inner loop characterizes the rapid adaptation process of agents when confronted with a new task. A task $\mathcal{T}_j$ is sampled from the distribution $p(\mathcal{T})$. Starting from the shared meta-initial parameters θ_i, the agent performs E steps of gradient descent to adapt to this specific task:

$$\theta'_i = \theta_i - \alpha \nabla_{\theta_i} \mathcal{L}_{\mathcal{T}_j}(f_{\theta_i}), \tag{8}$$

where α is the inner-loop learning rate, and $\nabla_{\theta_i} \mathcal{L}_{\mathcal{T}_j}(f_{\theta_i})$ is the loss computed on task τ.

2. Outer loop. The objective of the outer loop is to update the meta-parameters θ_i, so that the agent maximizes aggregate performance over all sampled tasks after inner-loop adaptation. Specifically, the gradients computed for each task during the inner loop are aggregated, and this aggregated gradient is used to update the meta-parameters:

$$\theta_i = \theta_i - \beta \nabla_{\theta_i} \sum\nolimits_{\mathcal{T}_j \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_j}(f_{\theta'_i}), \tag{9}$$

where β is the outer-loop learning rate.

Through iterative application of this bilevel optimization process, the meta-policy network parameters θ_i converge to a general-purpose optimal initialization point θ^*_i. This enables any agent to rapidly and effectively adapt to a new task sampled from the distribution $p(\mathcal{T})$, which is crucial for robust operation in the non-stationary and heterogeneous conditions typical of maritime communication scenarios.

3.2 Adaptive meta-reinforcement learning algorithm: AMRLSASS

Conventional meta-learning algorithms typically employ a fixed inner-loop learning rate, which is often insufficient for capturing variability across different tasks. In the practical context of maritime dynamic spectrum access and service scheduling, such variability becomes particularly pronounced. On one hand, Markov frequency-hopping and FHSS patterns exhibit distinctly different channel dynamics. On the other hand, key parameters—such as packet arrival rates and timeout constraints—also vary dynamically across tasks. Specifically, for simple tasks, a fixed step size can slow convergence, failing to fully exploit meta-learning's potential for rapid adaptation. Complex or highly dynamic tasks may cause oscillations during optimization, thereby reducing the overall adaptation efficiency.

To address these issues, this paper proposes an adaptive Meta-RL algorithm. Built upon the MAML framework (Finn et al., 2017), the algorithm introduces a task difficulty estimation mechanism. Through a difficulty estimation module, the inner-loop learning rate is dynamically adjusted, enabling the agent to adopt differentiated adaptation strategies according to the complexity of each task. This approach ensures fast adaptation while improving decision-making efficiency. The pseudo-code of the algorithm is provided in Algorithm S3 in the supplementary materials, and the overall framework of the AMRLSASS algorithm is illustrated in Fig. 2.

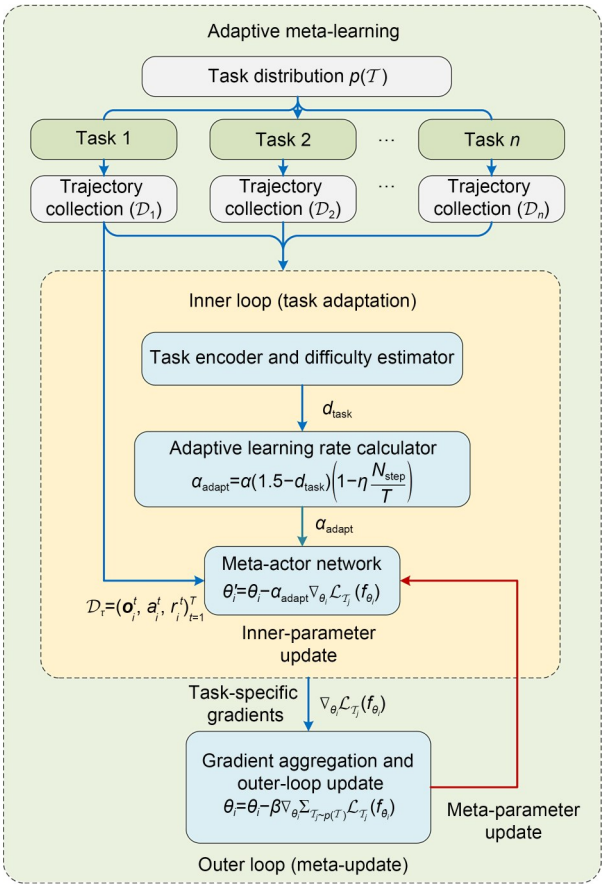

Fig. 2 Framework of the adaptive meta-learning algorithm model

3.2.1 Task-aware and difficulty estimation module

The task-aware and difficulty estimation module extracts task features from trajectory data via self-supervised learning and quantifies task difficulty. It consists of a task encoder (parameterized by φ) and a difficulty prediction head (parameterized by ψ). The architecture of this module is shown in Fig. 3.

1. Trajectory feature extraction

For the trajectory data batch $\mathcal{D}_\tau=(\boldsymbol{o}_i^t, a_i^t, r_i^t)_{t=1}^T$ collected by agent i in task $\mathcal{T}$, a comprehensive feature vector $\boldsymbol{f}_\tau \in \mathbb{R}^{5+L}$ is constructed to characterize the environmental properties of the task. Based on the trajectory data, this feature vector is specifically designed as follows:

(1) Reward statistics. Statistical measures are computed over all reward sequences within the batch, including the mean μ_r, variance σ_r^2, minimum $r_{\min}$, and maximum $r_{\max}$ values. These constitute the first four dimensions of the feature vector:

$$\boldsymbol{f}_\tau[0{:}4] = \left[\mu_r, \sigma_r^2, r_{\min}, r_{\max}\right]. \tag{10}$$

(2) State features. The agent's encoded environmental observation $\boldsymbol{h}_i^t$ at each time step is considered. Its mean distribution across

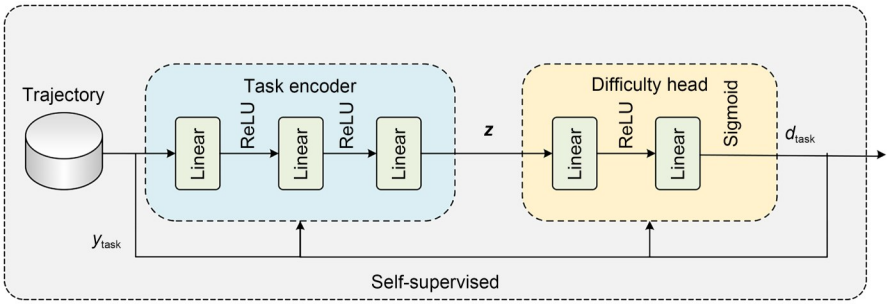

Fig. 3 Framework of the task perception and difficulty estimation module

the entire trajectory batch is computed. This means effectively representing the long-term channel occupancy patterns of the environmental state:

$$\boldsymbol{f}_\tau[4{:}4+L] = \mathbb{E}\left[\boldsymbol{h}_i^t\right]. \tag{11}$$

(3) Trajectory performance. The proportion of samples with positive rewards within the trajectory is calculated. This metric offers an intuitive reflection of the agent's immediate performance in the current task:

$$\boldsymbol{f}_\tau[4+L] = \frac{1}{|\mathcal{D}_\tau| L}\sum_{j=1}^{|\mathcal{D}_\tau|}\sum_{t=1}^{L}\mathbb{I}\left(r_t \geq 0\right), \tag{12}$$

where $\mathbb{I}(\cdot)$ is an indicator function.

2. Task difficulty estimation

After obtaining the feature vector, a task encoder is employed to map it into a latent vector that represents task complexity. The task encoder $\boldsymbol{\varepsilon}_\varphi(\tau)$ maps the feature vector $\boldsymbol{f}_\tau$ to a task representation $\boldsymbol{z}$, with the following neural network structure:

$$\boldsymbol{\varepsilon}_\varphi(\tau) = \boldsymbol{W}_2\partial_1\left(\boldsymbol{W}_1\partial_0\left(\boldsymbol{W}_0\left(\boldsymbol{f}(\tau)\right)+\boldsymbol{b}_0\right)+\boldsymbol{b}_1\right)+\boldsymbol{b}_2, \tag{13}$$

where $\boldsymbol{f}(\tau)$ is the feature vector extracted from trajectory τ, ∂_0 and ∂_1 denote the activation functions, and $\{\boldsymbol{W}_k, \boldsymbol{b}_k\}_{k=0}^{2}$ are the neural network parameters.

To quantify task difficulty, the task representation $\boldsymbol{z}$ is further processed by a difficulty prediction head $\xi_\psi(\boldsymbol{z})$ to produce an estimated difficulty value $d_{\text{task}} \in [0, 1]$:

$$\xi_\psi(\boldsymbol{z}) = \text{sigmoid}\left(\boldsymbol{W}_{\text{d}}\left(\boldsymbol{W}_{\text{z}}\boldsymbol{z}+\boldsymbol{b}_{\text{z}}\right)+\boldsymbol{b}_{\text{d}}\right), \tag{14}$$

where $\boldsymbol{W}_{\text{z}}$ and $\boldsymbol{W}_{\text{d}}$ are weight matrices and $\boldsymbol{b}_{\text{z}}$ and $\boldsymbol{b}_{\text{d}}$ are bias vectors.

To train the aforementioned networks, a self-supervised learning strategy is adopted. This strategy does not rely on external manual labels but automatically generates difficulty labels y_{task} based on the intrinsic statistical properties of the trajectory data. The label generation synthetically considers three dimensions: reward level, reward volatility, and environmental dynamics. It is defined as

$$y_{\text{task}} = w_1\left(1-\frac{\mu_r}{3}\right)+w_2\tanh\left(\sigma_r^2\right)+w_3\delta_{\text{s}}, \tag{15}$$

where the weighting coefficients w_1–w_3 satisfy $\sum_{j=1}^{3} w_j = 1$. In this work, we set $w_1 = 0.3$, $w_2 = 0.3$, and $w_3 = 0.4$. This assignment prioritizes the environmental dynamics factor, reflecting the fact that rapid channel variations pose the primary challenge for fast spectrum access adaptation. The reward-based factors serve as complementary indicators and are assigned equal, moderate weights. δ_{s} represents the state change rate, calculated as the mean absolute difference of the channel state portion in the observation vectors between consecutive timesteps. This metric reflects the speed of channel variation:

$$\delta_{\text{s}} = \frac{1}{|\mathcal{D}_\tau|(L-1)}\sum_{j=1}^{|\mathcal{D}_\tau|}\sum_{t=2}^{L}\left\| \boldsymbol{h}_i^t - \boldsymbol{h}_i^{t-1} \right\|_1, \tag{16}$$

where $\|\cdot\|_{\text{l}}$ denotes the l_1 norm.

To maintain numerical stability and prevent gradient explosion, the computed y_{task} is clipped:

$$y_{\text{task}} \leftarrow \text{clip}\left(y_{\text{task}}, 0.1, 0.9\right). \tag{17}$$

3. Encoder update

The generated difficulty label y_{task} comprehensively accounts for the reward level (where lower values indicate greater difficulty), reward volatility (where higher variance indicates greater difficulty), and environmental dynamics (where faster state variations indicate greater difficulty). By minimizing discrepancy between the predicted difficulty $\hat{d}$ and the self-supervised label y_{task}, the encoder is trained to accurately capture the task difficulty. The training objective employs a mean squared error (MSE) loss function:

$$L_{\text{encoder}} = \frac{1}{|\mathcal{D}_\tau|}\sum_{j=1}^{|\mathcal{D}_\tau|}\left(d_{\text{task},j} - y_{\text{task},j}\right)^2. \tag{18}$$

During training, the parameters of the task encoder φ and the difficulty prediction head ψ are updated via backpropagation. After sufficient training, the model can accurately output a difficulty estimate for a task based on a limited number of interaction trajectories. This capability provides a reliable basis for subsequently and adaptively adjusting the inner-loop learning rate in the Meta-RL framework, enhancing its adaptability to new maritime communication scenarios.

3.2.2 Adaptive learning mechanism

Given the significant variability in task difficulty, a dynamic adjustment of the inner-loop learning rate is essential. The adaptive learning rate is computed adaptively based on the estimated task difficulty d_{task} and the current inner-loop step index N_{step} (out of the total step number T), as follows:

$$\alpha_{\text{adapt}} = \alpha(1.5 - d_{\text{task}})\left(1 - \eta\frac{N_{\text{step}}}{T}\right), \tag{19}$$

where α_{adapt} denotes the adaptive inner-loop learning rate. The term $1.5 - d_{\text{task}}$ is a difficulty modulation factor. $d_{\text{task}} \to 1$ indicates a high-difficulty task, thereby reducing the learning rate to half of its base value. Conversely, $d_{\text{task}} \to 0$ indicates a low-difficulty task. The component $1 - \eta\frac{N_{\text{step}}}{T}$ is a progressive decay factor, with η=0.5, which ensures parameter stability in the later stages of the inner loop. It decays linearly from 1.0 to 0.5 as the step progresses from 0 to T.

Accordingly, the inner-loop parameter update for the adaptive meta-learning process, following Eq. (8), is given by

$$\theta_i' = \theta_i - \alpha_{\text{adapt}}\nabla_{\theta_i}\mathcal{L}_{\mathcal{T}_j}(f_{\theta_i}). \tag{20}$$

3.2.3 Test phase

The test phase is designed to evaluate the fast adaptation capability and generalization performance of the trained meta-model on the entirely novel task. The specific procedure is detailed as follows:

1. Test task sampling. A batch of test tasks $\{\mathcal{T}_j\}$ is sampled from a task pool that shares the same underlying distribution as the training tasks but remains disjointed from the training set. The feature parameter configurations (e.g., spectrum patterns, traffic loads, and timeout constraints) of test tasks are not encountered during the meta-training phase.

2. Fast adaptation. For each sampled test task, first load parameters and perform initialization, then interact with the environment and collect data $\mathcal{D}_\tau$, next calculate task difficulty and adaptive learning rate, and finally perform gradient update to obtain optimized parameters.

3. Performance evaluation. The adapted policy parameter $\pi_{\theta'}$ is deployed to execute multiple evaluation episodes on task τ_{test}. Key performance metrics—including packet success rate, channel utilization, and channel conflict rate—are recorded and analyzed. This comprehensive evaluation verifies the effectiveness and generalizability of the proposed AMRLSASS algorithm.

4 Simulation and performance analysis

4.1 Simulation setup

To comprehensively evaluate the effectiveness of the proposed meta-learning algorithm and its enhanced version, a series of simulations is designed, and the proposed algorithm is compared against several baseline algorithms. All simulations are conducted on the same hardware platform (Intel® Core™ i9-14900HX CPU, NVIDIA RTX 4060 GPU, and 24 GB RAM) and software environment (Python 3.9 and PyTorch 2.7.1). We construct a dynamic spectrum-sharing simulation environment involving one PU and three SUs, simulating the multi-agent collaborative process of spectrum sensing and data transmission in maritime emergency communication scenarios. The core simulation parameters are listed in Table S1 in the supplementary materials. The environment employs two typical spectrum channel occupancy patterns: Markov and FHSS. During training, each algorithm runs 300 episodes, with each episode containing 50 time steps. The compared algorithms include:

1. Random policy. This non-learning algorithm selects a random starting channel for sensing in each time slot, which serves as a performance lower bound.

2. Statistical priority (SP). This algorithm predicts future collision risks based on historical collision frequency. Through extensive interaction with the environment to learn channel dynamic patterns, it maintains channel statistics for each agent, such as idle probability, collision counts, and success counts, calculates the average priority for each channel, and stochastically selects the highest-scoring action in each time slot, thereby asymptotically converging to the optimal policy in steady-state environments.

3. MADDPG. It is a classic multi-agent deep deterministic policy gradient (DDPG) algorithm. During training, the critic network has access to all agents' observations and actions to learn a coordinated policy. During execution, each agent makes decisions based solely on its own local observations.

4. MetaSASS. It is a Meta-RL algorithm and undergoes meta-training on a task distribution encompassing various spectrum patterns to learn a well-initialized meta-policy. When facing a new task, it requires only a few gradient update steps to adapt rapidly.

5. AMRLSASS. Building upon MetaSASS, it introduces a task-aware adaptive learning rate mechanism. This algorithm not only inherits the fast adaptation advantage of meta-learning but also significantly enhances the efficiency and stability of the adaptation process via the adaptive mechanism, making it suitable for application scenarios with significant task heterogeneity and dynamic complexity, such as maritime spectrum access.

Performance is evaluated using eight key metrics: average reward, urgent packet success rate, normal packet success rate, urgent packet timeout rate, normal packet timeout rate, channel utilization rate, average queue length, and channel collision count. These metrics collectively reflect the algorithm's performance in terms of spectrum efficiency, quality of service, system stability, and multi-agent coordination capability.

4.2 Result analysis

4.2.1 Training performance analysis

1. Training task configuration

To comprehensively evaluate the generalization capability and adaptability of the proposed algorithm, the meta-learning algorithm was trained on a set of 16 training tasks encompassing two distinct channel dynamics models: Markov and FHSS (eight tasks per model). This design ensures sufficient diversity within a reasonable difficulty range, providing the meta-learner with rich experiential data to acquire a generalizable policy foundation adaptable to varying spectrum dynamics and traffic loads. In contrast, baseline comparison algorithms were trained solely on the eight tasks from the FHSS pattern. These tasks exhibit systematic variations in key communication parameters, including channel state transition probabilities, service arrival rates, packet timeout limits, hopping sequences, and spreading factors. The sampling ranges for the core parameters defining the task distribution $p(\mathcal{T})$ are summarized in Table S2 in the supplementary materials, aiming to cover a wide spectrum of scenarios potentially encountered in maritime urgent communication environments. Each task type is defined by a set of key environmental parameters: for the Markov pattern, these are the channel state transition

probabilities (p_{00} and p_{10}); for the FHSS pattern, these include the hopping sequence υ and spreading factor ς. The parameters shared by both patterns are the arrival rates for urgent and normal packets (λ_e and λ_n) and their corresponding transmission timeout limits (τ_e and τ_n). These parameters are randomly sampled from their defined ranges for each training task to ensure heterogeneity.

The channel occupancy patterns and the queue variations of the two spectrum models used for training are illustrated in Fig. S1 in the supplementary materials. Fig. S1a depicts the Markov pattern, where channel states exhibit stochastic switching characteristics. Fig. S1b shows the FHSS pattern, where channel occupancy follows a pseudorandom pattern based on a hopping sequence, applicable to anti-jamming and multi-user access scenarios. Fig. S1c presents queue lengths under a primarily urgent packet load; despite the higher arrival rate of urgent packets, the normal packet queue is slightly longer due to its more lenient timeout limit. Fig. S1d displays queue lengths under a primarily normal packet load, where the queue for normal packets grows persistently due to the arrival rate difference, while the urgent packet queue increases slowly.

2. Training performance analysis

To comprehensively evaluate the performance of different algorithms in dynamic maritime emergency communication spectrum access and service scheduling tasks, this paper conducted a comparative analysis involving five algorithms. Table 1 presents a quantitative comparison of key performance metrics obtained over an identical number of training episodes, including the average reward, standard deviation, final reward, and the deviation of the rewards over the last 50 episodes. Furthermore, the training curves illustrating the convergence behavior are depicted in Fig. 4. Collectively, these results demonstrate significant disparities in the performance of different algorithms.

As shown in Table 1, the proposed AMRLSASS algorithm achieves a slightly higher average reward than MetaSASS (37.08 vs. 34.94) with a smaller standard deviation (16.53 vs. 18.58). This indicates that the adaptive learning rate mechanism contributes to further performance optimization and enhanced training stability. From the training curves, the reward for AMRLSASS stabilizes approximately 45 after approximately 100 episodes, with the minimal fluctuation (0.17). While MetaSASS also shows stable convergence, its final reward exhibits slightly higher variance (0.24). The comparison from a zoomed-in view of the curves (where the reward variation across 10 consecutive episodes is less than 1) indicates that

Table 1 Performance comparison of different algorithms

Algorithm	All training episodes		Final training episode	Last 50 training episodes	
	Average reward	Std. reward	Final reward	Average reward	Std. reward
Random	−8.62	25.99	−61	−5.63	24.93
SP	23.24	79.73	−129	9.69	81.46
MADDPG	88.21	40.31	99	113.69	21.88
MetaSASS	34.94	18.58	45.68	45.75	0.24
AMRLSASS	37.08	16.53	45.83	45.74	0.17

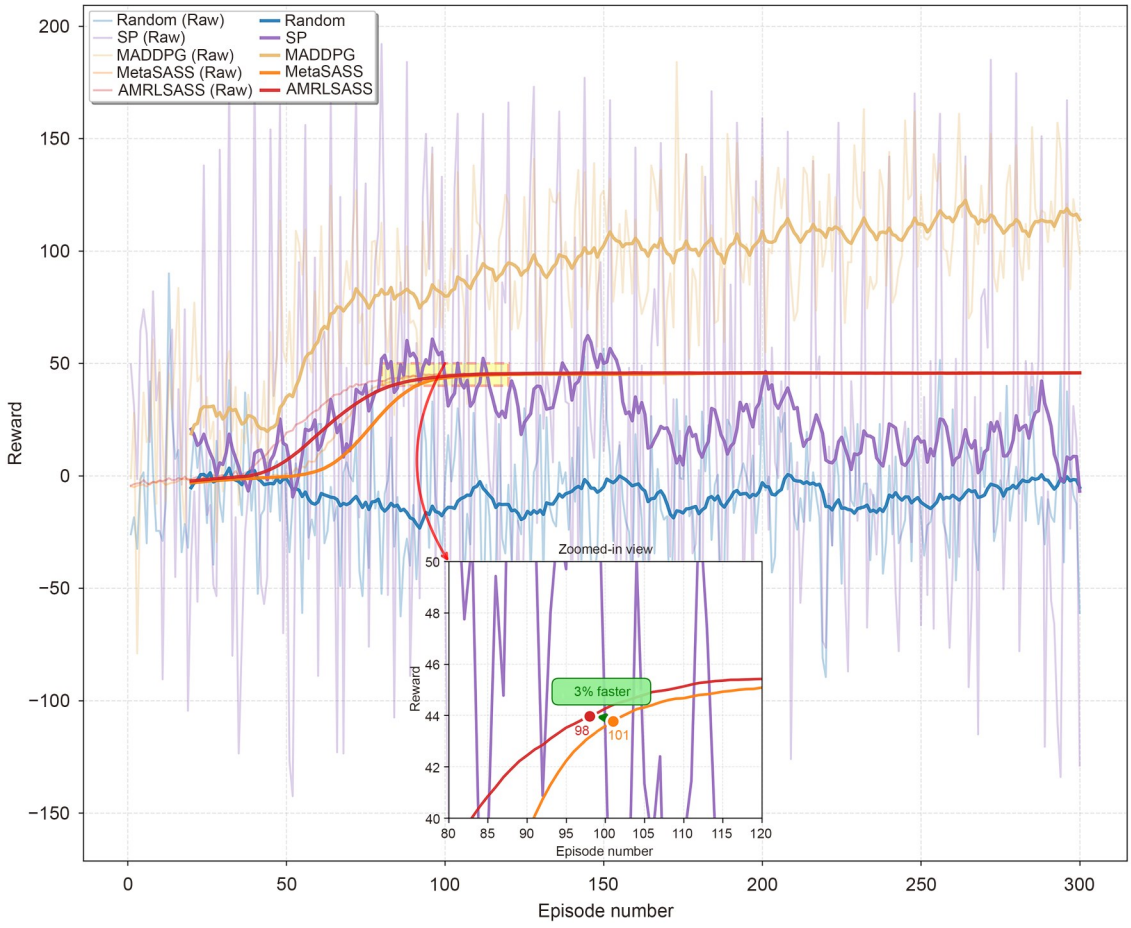

Fig. 4 Training curves of the five algorithms

AMRLSASS achieves a 3% faster convergence speed compared to MetaSASS, validating its effectiveness in reducing training time.

MADDPG, as a multi-agent DRL algorithm, attains a significantly higher average reward than the others, demonstrating its ability to optimize performance through coordinated multi-agent policies on a specific spectrum model. However, its large standard deviation (40.31) indicates intense reward fluctuations during training, indicating instability and challenges in handling highly dynamic environments.

The performances of the SP and Random algorithms are notably inferior. Although SP's average reward (23.24±79.73) is higher than that of Random, it remains substantially lower than those of the meta-learning algorithms. Relying solely on local observations and independent historical statistics without inter-agent coordination, SP frequently suffers from channel conflicts and severe queue backlog, resulting in low and highly variable rewards. The Random algorithm yields the lowest average reward (−8.62±25.99). Lacking any learning process and depending entirely on random channel selection, it cannot leverage environmental feedback to optimize decisions, thus demonstrating the worst performance.

4.2.2 Test performance analysis

1. Test task configuration

To evaluate the generalization performance of different algorithms for dynamic spectrum access and service scheduling in maritime emergency communications, a set of six distinct test tasks, covering two primary spectrum occupancy patterns, was established. The core parameter configurations for these test tasks are detailed in Table S3 in the supplementary materials. These tasks strictly adhere to the principle of out-of-distribution generalization, with their key parameters intentionally set outside the ranges encountered during training. This design effectively assesses the algorithms' adaptability to unforeseen environmental changes. The dynamic variations in spectrum occupancy patterns and agent queue lengths within the test tasks are visualized in Fig. S2 in the supplementary materials.

2. Test performance analysis

The comparative performance of the five algorithms across eight key metrics, under both Markov and FHSS patterns, is illustrated in Figs. S3 and S4 in the supplementary materials, with detailed results provided in Tables 2 and 3, respectively. In both dynamic spectrum environments, the proposed AMRLSASS algorithm demonstrates superior performance across all core metrics. It achieves the highest average reward, urgent packet success rate, and channel utilization. This significant advantage is primarily attributed to its integrated adaptive learning rate module, which dynamically adjusts the inner-loop learning steps based on real-time task difficulty, thereby maintaining high channel utilization while completely avoiding collisions.

The MetaSASS algorithm ranks second, validating the effectiveness of the meta-learning framework. By leveraging a "meta-training and rapid adaptation" approach, MetaSASS learns a generalized policy across a distribution of tasks, enabling it to adapt to new patterns with only a few gradient steps.

In contrast, MADDPG, SP, and Random show substantially poorer overall performance. MADDPG develops a policy specialized for the fixed patterns of its training environment; it lacks the mechanism for rapid policy adjustment when faced with unknown spectrum dynamics, resulting in notably lower rewards and success rates. Non-learning methods (SP and Random) are unable to perform continuous optimization based on environmental feedback. SP relies solely on static historical statistics, while Random selects actions arbitrarily. Consequently, they fail to coordinate resources and avoid collisions effectively, leading to frequent conflicts and queue congestion, indicating potential instability and resulting in negative average rewards and high urgent packet timeout rates. Overall, the results demonstrate that intelligent agents equipped with meta-learning and adaptive mechanisms exhibit a clear generalization advantage in dynamic, multi-task environments.

3. Generalizability analysis

To further validate the algorithms' generalization capability in completely unknown environments, a spectrum pattern switching test scenario was designed. The test consists of 100 steps: the first 50 steps operate under the Markov pattern, and the remaining 50 steps operate under the FHSS pattern. The test parameters are entirely distinct from those used in training. As shown in Fig. 5 and Fig. S5 in the supplementary materials, the generalization performance is evaluated using both episode-level and step-level rewards. The detailed generalization test results for the average reward are presented in Table 4.

Table 2 Performance comparison on the Markov pattern

Algorithm	Average reward	Urgent packet success rate (%)	Normal packet success rate (%)	Urgent packet timeout rate (%)	Normal packet timeout rate (%)	Channel utilization rate (%)	Average queue length	Channel conflict count
AMRLSASS	3.25	61.73	2.99	11.10	0	100	28.29	0
MetaSASS	3.04	60.51	3.66	12.77	0	100	29.21	0
MADDPG	0.29	47.93	2.40	21.20	20.88	80.19	30.62	20.88
SP	−2.08	39.31	0.92	30.42	27.00	70.72	28.54	27.00
Random	−2.85	35.38	1.69	32.11	33.62	61.54	30.00	33.62

Table 3 Performance comparison on the FHSS pattern

Algorithm	Average reward	Urgent packet success rate (%)	Normal packet success rate (%)	Urgent packet timeout rate (%)	Normal packet timeout rate (%)	Channel utilization rate (%)	Average queue length	Channel conflict count
AMRLSASS	4.55	58.89	1.41	5.55	0	100	33.83	0
MetaSASS	4.30	61.00	2.89	7.63	1.88	99.22	32.17	1.88
MADDPG	0.74	47.27	1.27	14.22	33.75	83.37	34.25	33.75
SP	−2.59	35.72	0.46	24.81	53.88	65.78	33.83	53.88
Random	−1.92	40.07	0.79	18.47	65.25	61.55	33.42	65.25

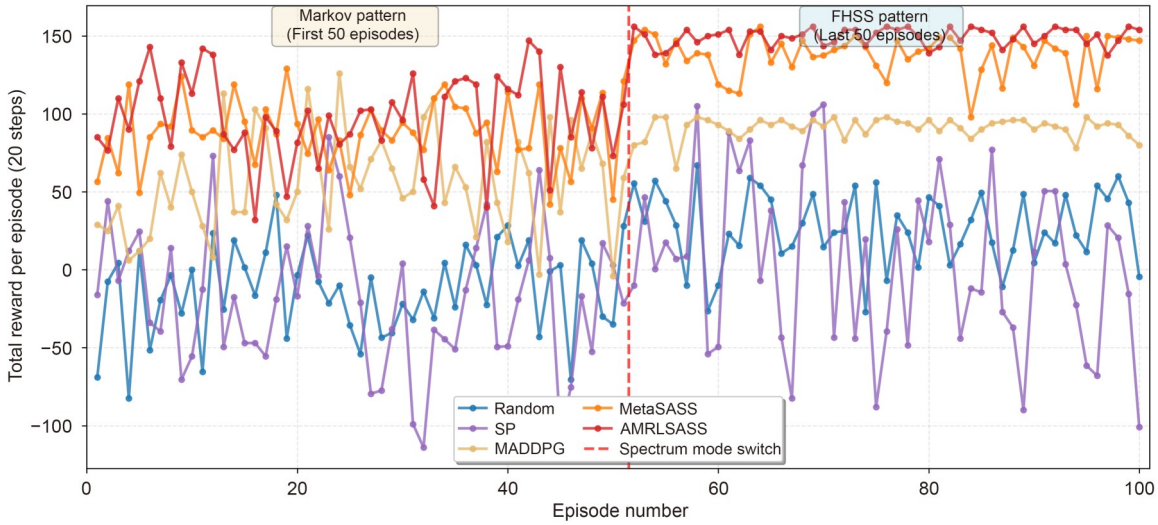

Fig. 5 Comparison of different algorithms in terms of generalization performance (episode-level)

Table 4 Average reward comparison of different algorithms under the generalization test

Algorithm	Markov pattern		FHSS pattern	
	Episode reward	Step reward	Episode reward	Step reward
Random	−13.12	−0.66	26.90	1.34
SP	−18.09	−0.90	4.51	0.23
MADDPG	56.00	2.80	91.24	4.56
MetaSASS	88.67	4.43	137.99	6.90
AMRLSASS	97.86	4.89	149.50	7.48

The results indicate that the AMRLSASS algorithm maintains the best performance under both patterns (Markov: episode reward=97.86, step reward=4.89; FHSS: episode reward=149.50, step reward=7.48). Its adaptive learning rate mechanism effectively enhances its cross-environment generalization performance. The MetaSASS algorithm also demonstrates excellent cross-pattern generalization ability. Notably, the MADDPG algorithm shows a significant performance improvement after switching to the FHSS pattern (episode reward increases from 56.00 to 91.24) and exhibits greater stability. This is because its training process partially encompassed similar dynamic features, granting it a foundational level of environmental adaptability, although its performance remains substantially lower than those of the meta-learning approaches. The conventional methods (SP and Random), due to their lack of learning and adaptive mechanisms, perform poorly under both patterns. This further validates the necessity of intelligent learning algorithms for dynamic spectrum access. A detailed complexity and scalability analysis, including parameter counts and discussion on scaling to larger networks, is provided in Section S4 in the supplementary materials.

5 Conclusions

This study addresses the key challenges in maritime wireless communications, including limited spectrum resources and heterogeneous service demands. We present an in-depth investigation into adaptive Meta-RL for spectrum access and service scheduling. A joint optimization model for heterogeneous dynamic spectrum access and multi-priority service scheduling is developed, integrating Markov and FHSS patterns. Building upon this model, we first propose the MetaSASS algorithm—a Meta-RL-based scheme for dynamic spectrum access and service scheduling. This algorithm effectively resolves the poor generalization of conventional methods in unknown environments. Furthermore, an enhanced algorithm, AMRLSASS, is designed by incorporating an adaptive meta-learning mechanism. By using a task encoder to perceive task complexity in real time and dynamically adjust the inner-loop learning rate, the agent can execute differentiated adaptation strategies tailored to varying task difficulties. This mechanism not only ensures rapid adaptation but also improves decision-making efficiency. Simulation results validate the superior performance of the proposed methods in out-of-distribution test and spectrum pattern switching scenarios. Future work will explore several directions, including the incorporation of imperfect spectrum sensing and node mobility to enhance practical applicability, as well as scaling the proposed method to larger networks.

Supplementary information

The online version contains supplementary materials available at https://doi.org/10.1631/ENG.ITEE.2026.0047.

Acknowledgments

This work was supported by the Shandong Provincial Natural Science Foundation (No. ZR202204300003).

Author contributions

Zhongyang MAO supervised the project and reviewed the paper. Jiahuan GENG conceived and designed the research, processed the data, and drafted, revised, and finalized the paper. Faping LU assisted with the data processing and validation. Wenbiao TIAN contributed to the methodology and investigation. Jiafang KANG and Yaozong PAN helped organize the paper and provided critical feedback.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declaration on the use of generative AI tools

During the preparation of this work, the authors used DeepSeek to improve language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

Albinsaid H, Singh K, Biswas S, et al., 2022. Multi-agent reinforcement learning-based distributed dynamic spectrum access. *IEEE Trans Cogn Commun Netw*, 8(2):1174-1185. https://doi.org/10.1109/TCCN.2021.3120996

Antoniou A, Edwards H, Storkey A, 2019. How to train your MAML. https://arxiv.org/abs/1810.09502

Atimati E, Crawford D, Stewart R, 2023. Intelligent shared spectrum coordination in heterogeneous networks. IEEE Virtual Conf on Communications, p.252-257. https://doi.org/10.1109/VCC60689.2023.10474686

Atimati E, Nyasulu T, Crawford D, et al., 2025. Resource management in dynamic shared spectrum networks. IEEE Int Symp on Dynamic Spectrum Access Networks, p.13-19. https://doi.org/10.1109/DySPAN64764.2025.11115965

Dong L, Qian Y, Xing Y, 2022. Dynamic spectrum access and sharing through actor-critic deep reinforcement learning. *EURASIP J Wirel Commun Netw*, 2022:48. https://doi.org/10.1186/s13638-022-02124-4

Feng MJ, Zhang WH, Krunz M, 2023. Dynamic spectrum access in non-stationary environments: a DRL-LSTM integrated approach. Int Conf on Computing, Networking and Communications, p.159-164. https://doi.org/10.1109/ICNC57223.2023.10074454

Finn C, Abbeel P, Levine S, 2017. Model-agnostic meta-learning for fast adaptation of deep networks. Proc 34th Int Conf on Machine Learning, p.1126-1135.

Huang K, Luo ZZ, Liang L, et al., 2022. Fast spectrum sharing in vehicular networks: a meta reinforcement learning approach. IEEE 96th Vehicular Technology Conf, p.1-5. https://doi.org/10.1109/VTC2022-Fall57202.2022.10012705

ITU, 2009. Characteristics of VHF Radio Systems and Equipment for the Exchange of Data and Electronic Mail in the Maritime Mobile Service RR Appendix 18 Channels. ITU-R M.1842-1-2009.

ITU, 2012. Interim Solutions for Improved Efficiency in the Use of the Band 156–174 MHz by Stations in the Maritime Mobile Service. ITU-R M.1084-5-2012.

ITU, 2024. Table of Transmitting Frequencies in the VHF Maritime Mobile Band. ITU Radio Regulations, Appendix 18.

Jia XY, Wang T, Du X, 2024. Federated multi-objective meta-reinforcement learning for adaptive edge task offloading. IEEE Int Conf on High Performance Computing and Communications, p.482-489. https://doi.org/10.1109/HPCC64274.2024.00071

Kai H, Le L, Shi J, et al., 2025. Meta reinforcement learning for fast spectrum sharing in vehicular networks. *China Commun*, 22(9):320-332. https://doi.org/10.23919/JCC.ja.2023-0192

Ke ZY, Wang XM, Du ZY, et al., 2025. Intelligent frequency reuse for dynamic spectrum anti-jamming: a hybrid-reward-based multi-agent deep reinforcement learning approach. *IEEE Wirel Commun Lett*, 14(3):771-775. https://doi.org/10.1109/LWC.2024.3523221

Khisa S, Elhattab M, Assi C, et al., 2025. Optimizing multi-user uplink cooperative rate-splitting multiple access: efficient user pairing and resource allocation with gradient-based meta learning. *IEEE Trans Commun*, 73(9):7366-7380. https://doi.org/10.1109/TCOMM.2025.3550371

Li XH, Zhang YL, Ding HC, et al., 2024. Intelligent spectrum sensing and access with partial observation based on hierarchical multi-agent deep reinforcement learning. *IEEE Trans Wirel Commun*, 23(4):3131-3145. https://doi.org/10.1109/TWC.2023.3305567

Li YH, Wang Y, Li Y, et al., 2023. Multi-user dynamic spectrum access based on LRQ deep reinforcement learning network. 25th Int Conf on Advanced Communication Technology, p.79-84. https://doi.org/10.23919/ICACT56868.2023.10079537

Li YZ, Zhang WS, Wang CX, et al., 2020. Deep reinforcement learning for dynamic spectrum sensing and aggregation in multi-channel wireless networks. *IEEE Trans Cogn Commun Netw*, 6(2):464-475. https://doi.org/10.1109/TCCN.2020.2982895

Liu ZY, Wang XJ, Zhang Y, et al., 2023. Meta reinforcement learning for generalized multiple access in heterogeneous wireless networks. 21st Int Symp on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, p.570-577. https://doi.org/10.23919/WiOpt58741.2023.10349896

Liu ZY, Wang XJ, Guo K, et al., 2024. Federated meta-RL based multiple access protocol for diverse heterogeneous wireless networks. Int Conf on Future Communications and Networks, p.1-6. https://doi.org/10.1109/FCN64323.2024.10985134

Liu ZY, Wang XJ, Feng CY, et al., 2026. Meta-reinforcement learning with mixture of experts for generalizable multi access in heterogeneous wireless networks. *IEEE Trans Commun*, 74:870-885. https://doi.org/10.1109/TCOMM.2025.3631558

Lu ZY, Gursoy MC, 2021. Dynamic channel access via meta-reinforcement learning. IEEE Global Communications Conf, p.1-6. https://doi.org/10.1109/GLOBECOM46510.2021.9685347

Lyu T, Xu HT, Liu FF, et al., 2024. Computing offloading and resource allocation of NOMA-based UAV emergency communication in marine Internet of Things. *IEEE Int Things J*, 11(9):15571-15586. https://doi.org/10.1109/JIOT.2023.3348164

Niu LW, Chen XF, Zhang N, et al., 2023. Multiagent meta-reinforcement learning for optimized task scheduling in heterogeneous edge computing systems. *IEEE Int Things J*, 10(12):10519-10531. https://doi.org/10.1109/JIOT.2023.3241222

Nomikos N, Gkonis PK, Bithas PS, et al., 2023. A survey on UAV-aided maritime communications: deployment considerations, applications, and future challenges. *IEEE Open J Commun Soc*, 4:56-78. https://doi.org/10.1109/OJCOMS.2022.3225590

Rao N, Xu H, Qi ZS, et al., 2024. Fast adaptive jamming resource allocation against frequency-hopping spread spectrum in wireless sensor networks via meta-deep-reinforcement-learning. *IEEE Trans Aerosp Electron Syst*, 60(6):7676-7693. https://doi.org/10.1109/TAES.2024.3418944

Rao N, Xu H, Qi ZS, et al., 2025. Adaptive jamming decision-making against FHSS communications via inexpert demonstrations assisted meta reinforcement learning. *IEEE Commun Lett*, 29(1):105-109. https://doi.org/10.1109/LCOMM.2024.3502423

Saggese F, Pasqualini L, Moretti M, et al., 2021. Deep reinforcement learning for URLLC data management on top of scheduled eMBB traffic. IEEE Global Communications Conf, p.1-6. https://doi.org/10.1109/GLOBECOM46510.2021.9685777

Saggese F, Moretti M, Popovski P, 2022. NOMA power minimization of downlink spectrum slicing for eMBB and URLLC users. IEEE Wireless Communications and Networking Conf, p.1725-1730. https://doi.org/10.1109/WCNC51071.2022.9771653

Sheng HM, Zhou WJ, Zheng JJ, et al., 2024. Transfer reinforcement learning for dynamic spectrum environment. *IEEE Trans Wirel Commun*, 23(2):1447-1458. https://doi.org/10.1109/TWC.2023.3289502

Sheng TQ, Zhang WS, Ding WJ, et al., 2022. Dynamic spectrum sharing and aggregation scheme based on deep reinforcement learning. Int Wireless Communications and Mobile Computing, p.290-294. https://doi.org/10.1109/IWCMC55113.2022.9825001

So H, Soya H, 2023. Excluded channel selection scheme for dynamic spectrum sharing in various interference environments. *IEEE Access*, 11:69798-69806. https://doi.org/10.1109/ACCESS.2023.3292837

Upadhyay D, Upadhyay A, Venu N, et al., 2025. Deep learning-based spectrum sharing for dynamic resource allocation in 6G cognitive radio networks. 4th Int Conf on Power, Control and Computing Technologies, p.1-6. https://doi.org/10.1109/ICPC2T63847.2025.10958615

Wang QF, Xu WQ, Chen HH, 2026. A heterogeneous-agent deep reinforcement learning approach for dynamic spectrum access in cognitive wireless networks. *IEEE Trans Cogn Commun Netw*, 12:2221-2235. https://doi.org/10.1109/TCCN.2025.3594035

Yang TT, Gao S, Li JB, et al., 2022. Multi-armed bandits learning for task offloading in maritime edge intelligence networks. *IEEE Trans Veh Technol*, 71(4):4212-4224. https://doi.org/10.1109/TVT.2022.3141740

Yang TT, Zhang WS, Bo YL, et al., 2023. Dynamic spectrum sharing based on federated learning and multi-agent actor-critic reinforcement learning. Int Wireless Communications and Mobile Computing, p.947-952. https://doi.org/10.1109/IWCMC58020.2023.10182572

Yuan L, Zhou FH, Wu QH, et al., 2024. Channel prediction-enhanced intelligent resource allocation for dynamic spectrum-sharing networks. IEEE Int Conf on Communications, p.2767-2772. https://doi.org/10.1109/ICC51166.2024.10623083

Zhang SG, Wang Z, Gao GY, et al., 2023. Deep reinforcement learning for UAV-assisted spectrum sharing under partial observability. IEEE 98th Vehicular Technology Conf, p.1-6. https://doi.org/10.1109/VTC2023-Fall60731.2023.10333853

Zhang X, Bhuyan A, Kasera SK, et al., 2023. Distributed power allocation for 6-GHz unlicensed spectrum sharing via multi-agent deep reinforcement learning. IEEE Int Conf on Industrial Technology, p.1-6. https://doi.org/10.1109/ICIT58465.2023.10143125

Zhang YL, Li XH, Ding HC, et al., 2023. A joint scheme on spectrum sensing and access with partial observation: a multi-agent deep reinforcement learning approach. IEEE/CIC Int Conf on Communications in China, p.1-6. https://doi.org/10.1109/ICCC57788.2023.10233366