Layered Trust-Aware Routing with Latency-Security Tradeoff Optimization in Maritime Networks

*Bo Lin, Lianyou Lai**

School of Ocean Information Engineering, Jimei University, China
brseve@jmu.edu.cn, kaikaixinxinlly@jmu.edu.cn

Abstract

Maritime emergency networks require routing policies that provide timely and secure packet delivery under sparse deployment, mobility-induced topology variation, heterogeneous node ownership, and potential adversarial interference. Existing approaches usually optimize communication efficiency or trust evaluation separately, offering limited support for mission-priority traffic under constrained link resources. This article proposes a layered trust-aware routing framework that couples task-hierarchical service control with security-aware deep reinforcement learning. The method first applies global pre-screening to remove infeasible or low-value relay candidates according to policy constraints, resource quotas, link load, energy, mobility, and geometric progress. It then uses a dynamic weighted fusion DQN to select the next hop from the screened candidates based on behavioral reputation, jurisdictional/policy attributes, geo-situational awareness, and Network-Operational Safety. Priority-aware scheduling, retransmission control, and task-aware reward shaping are further embedded into the learning loop. In the mobility-controlled performance evaluation at a node speed of 20 m/s, the proposed method improves throughput by 18.3% and reduces end-to-end delay by 22.7% compared with classical routing baselines. In the separate security-oriented evaluation, the method maintains a packet delivery ratio above 92.6% under a representative adversarial setting with 15% malicious nodes.

Keywords: Maritime emergency communications, DQN, Task-hierarchical scheduling, Secure multi-hop routing

1 Introduction

Maritime communication networks have become a fundamental infrastructure for the safe and efficient operation of maritime transportation, offshore resource development, and fishery activities. In maritime traffic scenarios, vessels rely on communication systems for real-time navigation-state exchange, collision-avoidance warning, and coordinated route scheduling. In offshore engineering, stable communication links are required for remote control of deep-sea equipment, data backhaul from oil and gas platforms, and emergency-response coordination. In fishery operations, maritime networks support fleet communication, sharing of market and environmental information, and long-distance supply coordination. As maritime activities continue to expand in scale, intensity, and autonomy, the reliability and security of maritime communications have become increasingly critical.

However, maritime communication systems differ fundamentally from their terrestrial counterparts due to several intrinsic characteristics of the ocean environment. First, network topology exhibits strong temporal variability driven by vessel mobility, buoy drift, and environmental perturbations, which frequently disrupt links and trigger recurrent route reconfiguration [1]. Second, communication resources are unevenly distributed over both space and time as a consequence of meteorological dynamics, sea-state fluctuations, and geographic constraints. This imbalance is particularly pronounced in open-sea regions, where sparse infrastructure leads to intermittent coverage and limited spectrum availability. Third, maritime network entities are typically operated by multiple stakeholders, including different countries, commercial operators, and regulatory authorities. Such heterogeneity not only increases the complexity of coordination and interoperability, but also raises critical concerns regarding security, policy compliance, and data credibility, especially in emergency scenarios [2]. Consequently, routing in maritime emergency networks cannot be adequately modeled as a pure shortest-path problem; instead, it requires a joint consideration of communication efficiency, trust assurance, and operational constraints.

To cope with these challenges, prior research has explored several complementary directions, which can be broadly categorized into routing design, trust modeling, and intelligent optimization. In terms of routing design, substantial efforts have been dedicated to maintaining reliable connectivity and enabling rapid path recovery under highly dynamic topologies. A prominent approach involves evolving traditional reactive frameworks into multi-metric, context-aware protocols; for instance, the TOPSIS-Aided Ad-hoc On-Demand Distance Vector (TAAODV) protocol optimizes standard path discovery by evaluating neighborhood link stability to suppress route request flooding and dynamically selecting paths based on multi-parameter criteria [3]. In parallel, trust-oriented approaches aim to enhance routing security by evaluating node reliability based on historical interactions or peer

*Corresponding Author: Lianyou Lai; Email: kaikaixinxinlly@jmu.edu.cn
DOI: https://doi.org/10.70003/160792642026072704010

recommendations. Along this line, blockchain-assisted trust frameworks integrate multi-dimensional assessment, probabilistic verification, dynamic state updating, and consensus mechanisms to mitigate routing threats such as malicious packet dropping and false reporting [4]. Furthermore, intelligent optimization techniques formulate routing as a multi-objective decision process, leveraging heuristic or learning-based algorithms to balance latency, energy consumption, and security. As an example, an enhanced ant colony optimization scheme incorporates residual node energy and transmission deviation into its heuristic function, thereby simultaneously improving energy efficiency and routing stability under resource constraints [5].

Although these approaches have achieved promising results in general maritime networking scenarios, they still face notable limitations in maritime emergency communications. Conventional routing protocols remain vulnerable to substantial route-reconstruction delay under rapidly changing topology, which may lead to the loss of critical communication opportunities during rescue or hazard-response operations. Trust-evaluation models are often strongly dependent on historical interaction records, making them less effective for newly deployed rescue devices or temporarily joined nodes that lack sufficient prior trust information. Meanwhile, many intelligent optimization approaches still abstract maritime nodes in an overly simplified manner, without adequately modeling mobility uncertainty, jurisdictional heterogeneity, and location-sensitive security risks. This weak representation may reduce routing robustness and even introduce privacy leakage or compliance violations when communication decisions involve cross-border or policy-constrained maritime regions. In addition, most existing routing schemes remain predominantly path-centric: they optimize which path should be selected, but pay far less attention to which mission should be served first, how much transmission opportunity should be reserved for urgent traffic, and how failures of high-priority traffic should be recovered. Such limitations are particularly restrictive in maritime emergency networks, where critical rescue messages, important coordination traffic, and normal background traffic coexist and compete for scarce link resources.

Motivated by these observations, this paper proposes a task-hierarchical multidimensional security-aware routing framework for maritime emergency communications. The proposed method does not treat the traffic-priority mechanism as an independent optimizer outside the routing framework; instead, it embeds a multi-level task system into the main loop of trust-aware routing. Concretely, each packet first undergoes task-class generation, priority queue scheduling, and link-resource quota allocation; the selected packet is then forwarded by a security-aware Deep Q-Network (DQN) according to a multidimensional Comprehensive Trust Score (CTS); finally, priority-aware error recovery and task-aware reward shaping feed back to routing learning and reputation evolution. In this way, the proposed framework preserves a clear division of functionality: CTS determines which relay is safer, whereas the task hierarchy determines which packet should be served first, how much transmission resource it may consume, how retransmission is handled after failure, and how strongly the learner is encouraged to protect urgent traffic. To improve scalability in large maritime networks, a global-local-global optimization architecture is adopted. A lightweight heuristic first performs coarse-grained candidate screening over the entire network, reducing the original routing decision space to a manageable set of candidate relays. Then, within each screened subspace, the DQN agent learns routing policies from multidimensional state representations under a reward that jointly reflects end-to-end delay, congestion risk, task priority, and compliance cost. Finally, the locally optimized subpaths are projected back to the global layer, where consistency verification and compliance auditing eliminate infeasible or policy-violating relays before feasible subpaths are merged into an end-to-end route. Compared with conventional shortest-path routing, the proposed framework is designed to provide a better balance among efficiency, robustness, service differentiation, security, and policy conformance in highly dynamic maritime emergency networks.

To clearly highlight the novelty of the proposed framework, the main contributions of this paper are summarized as follows:

- We propose a task-hierarchical multidimensional security-aware routing framework for maritime emergency communications, in which packet-service control is incorporated into the main loop of trust-aware routing rather than being treated as an external priority mechanism. Under this design, task-class generation, queue scheduling, quota allocation, routing decision, and error recovery are jointly coordinated within a unified decision framework.
- We develop a multidimensional trust-fusion mechanism based on the CTS, which integrates behavioral reputation, jurisdictional/policy attributes, geo-situational awareness, and network utilization safety into a structured relay-evaluation metric. This design enables the routing process to account not only for communication efficiency, but also for security risk, policy feasibility, and operational trustworthiness in heterogeneous maritime environments.
- We design a global-local-global optimization architecture to improve routing scalability in large maritime networks. A lightweight heuristic is first used to perform coarse-grained candidate screening at the global level, after which a DQN agent learns next-hop selection policies within the reduced candidate subspace. The locally optimized subpaths are then projected back to the global layer for consistency verification and compliance auditing, thereby improving both tractability and policy conformance.
- We further incorporate task-aware service-control mechanisms into the routing-learning loop, including priority queue scheduling, link-resource quota allocation, priority-aware error recovery,

and task-aware reward shaping. As a result, the proposed framework is able to better protect urgent traffic while jointly balancing delay, congestion risk, robustness, and security requirements in highly dynamic maritime emergency scenarios.

Figure 1 provides an intuitive example of the routing behavior considered in this study: the security-aware policy avoids malicious relays even when a shorter path exists.

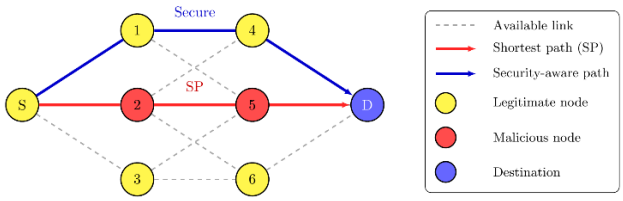


Figure 1. Comparison of security-aware and shortest-path routing

The remainder of this paper is organized as follows. Section 2 reviews related work on hierarchical routing, distributed routing, and learning-based routing. Section 3 presents the system model and problem formulation. Section 4 introduces the proposed task-hierarchical security-aware DQN routing algorithm. Section 5 reports the simulation setup and experimental results. Section 6 discusses the relationship between the proposed method and representative routing, trust, and learning-based algorithms, followed by limitations and future work. Section 7 concludes the paper.

2 Related Work

Hierarchical and cluster-based routing remains an important family of energy-efficient routing mechanisms for resource-constrained wireless networks. Instead of relying on a flat forwarding structure, these methods organize nodes into clusters or tree-like hierarchies so that local members forward traffic to selected aggregation or relay nodes before data are delivered toward the sink. Surveys of LEACH-derived clustering protocols show that such designs can reduce redundant transmissions and prolong network lifetime, but they also expose inherent limitations: cluster-head selection and load distribution strongly affect stability, and overloaded or failed cluster heads may become communication bottlenecks or single points of failure [6]. More recent adaptive hierarchical designs further confirm that hierarchical organization is useful for large-scale sensing networks, but their performance still depends heavily on topology structure, relay-role assignment, and energy-aware maintenance rules [7]. These observations motivate routing mechanisms that retain the scalability benefits of hierarchical organization while avoiding excessive dependence on fixed relay roles.

To alleviate the limitations of centralized or cluster-head-dominated routing, distributed intelligent routing has been increasingly studied in decentralized and autonomous communication networks. A comprehensive survey on deep reinforcement learning (DRL) for communications and networking points out that modern IoT and UAV networks are becoming more decentralized and autonomous, where network entities often need to make local decisions under uncertain network states; it also identifies traffic routing, resource sharing, security, and connectivity preservation as important DRL application areas [8]. From the multi-agent perspective, routing can be formulated as a cooperative sequential decision problem in which each node, relay, or link controller acts as an autonomous agent with partial observations. Recent MARL surveys further emphasize that multi-agent systems must deal with non-stationarity, partial observability, coordination, and scalable training, which are highly consistent with the characteristics of dynamic maritime emergency networks [9-10].

Learning-based routing has therefore become a major research direction for highly dynamic wireless networks. Deep Q-Networks (DQN) demonstrate how value-based reinforcement learning can scale beyond tabular Q-learning by using deep neural networks to approximate action-value functions in high-dimensional decision spaces [11]. In communication networks, DRL has been widely investigated for adaptive network access, wireless caching, data offloading, network security, traffic routing, resource sharing, and data collection [8]. For routing-specific scenarios, Wang et al. developed a multi-agent actor–critic routing method for UAV swarm networks, where shared information is used to support collaborative routing decisions and reduce communication overhead [12]. Kaviani et al. proposed DeepCQ+, which integrates multi-agent deep reinforcement learning into highly dynamic MANET routing and improves robustness across network sizes, mobility conditions, and traffic patterns [13]. Yamin and Permuter modeled wireless routing in integrated access and backhaul networks as a multi-agent partially observable Markov decision process, enabling distributed base-station routing decisions under limited access to real-time network conditions [14].

Despite these advances, the optimization objectives of most existing studies remain predominantly performance-oriented, such as minimizing end-to-end delay or maximizing throughput and packet delivery ratio. In general, they lack an intrinsic multidimensional security-aware design. For example, many existing algorithms implicitly assume that all participating nodes behave benignly, and therefore cannot effectively identify or mitigate selfish nodes, which refuse to forward packets in order to conserve their own resources, or malicious nodes, which may launch blackhole attacks, greyhole attacks, or data-tampering attacks. Even when security is considered, it is often represented by a single metric, such as a simple reputation score derived from historical forwarding behavior. Such a one-dimensional criterion is insufficient for addressing complex and multi-faceted threats. A node may exhibit a high forwarding ratio while still posing substantial risks, such as information leakage or covert manipulation. Moreover, prior approaches rarely incorporate static trust-related information that can be obtained in advance, such as jurisdictional/policy attributes, alliance relationships, or other

geopolitically relevant factors, although such information is highly important in maritime emergency and military communication scenarios.

This limitation opens up new opportunities for routing design. The hierarchical optimization routing strategy proposed in this paper introduces a node trust-evaluation mechanism into the hierarchical routing framework. In this way, the proposed model not only preserves the routing-performance advantages of existing multi-agent approaches, but also enhances communication security and reliability by resisting malicious behavior through trust-aware decision-making. As a result, it can effectively compensate for the security deficiencies of existing schemes. The detailed model design is presented in the following sections.

3 System Model and Problem Formulation

3.1 System Model

The main symbols used in the proposed framework are summarized in Table 1.

Table 1. Main notation used in the proposed framework

Symbol	Description
$\mathcal{G} = (V, E)$	Directed maritime communication graph
$\mathcal{N}_t(i)$, $\mathcal{A}_t(i)$	Neighbor set and screened candidate-relay set of node i
Θ_c	Service profile of traffic class c
$\mathrm{CTS}_{ij}(t)$	Comprehensive trust score from node i to candidate relay j
N_j, $G_{ij}(t)$, $\mathcal{U}_j(t)$, $B_{ij}(t)$	Jurisdictional-policy prior, geo-situational score, network-operational safety, and behavioral reputation
$q_j(t)$, $\xi_j(t)$	Queue occupancy and normalized residual energy of node j
$Q_\theta(s_t, a_t)$	DQN action-value function
$P_{\mathrm{sec}}(j,t)$	CTS-aware security penalty for candidate relay j

To systematically couple physical communication dynamics, multidimensional trust assessment, and routing intelligence, we organize the proposed framework into a three-layer architecture consisting of a Communication Layer, a Fusion Layer, and a Control Layer. In addition, a task-hierarchy mechanism is embedded into the packet-service procedure as a supervisory orchestration logic, rather than being treated as an independent optimization layer. Under this design, the three layers are responsible for environment observation and execution, security-semantic abstraction, and task-aware routing control, respectively.

At each decision epoch, the framework operates in a closed loop that begins with state acquisition and trust fusion. Based on the resulting system representation, packet scheduling and candidate screening are subsequently performed, followed by next-hop selection, packet forwarding with error handling, and the joint update of reward and reputation information. This formulation emphasizes that routing is not treated as an isolated shortest-path selection problem. Rather, relay selection is embedded into a broader service-control loop in which packet priority, transmission opportunity, retry policy, and trust evolution are coordinated in an integrated manner.

The Communication Layer provides the operational substrate of the framework. It consists of heterogeneous maritime entities, including ships, UAVs, buoys, and shore-based stations, and is responsible for local observation, packet transmission, link maintenance, and execution of forwarding actions. This layer therefore serves as the interface between the routing system and the physical maritime environment.

The Fusion Layer acts as an intermediate semantic layer between raw observations and routing control. It converts heterogeneous information collected from the Communication Layer into a structured trust descriptor by jointly evaluating neighboring relays from the perspectives of Behavioral Reputation, Static Jurisdictional/Policy Attribute, Geo-situational Awareness, and Network-Operational Safety. The output of this layer is a CTS, which provides security-oriented semantics for subsequent routing decisions.

The Control Layer is the policy-generation and coordination center of the framework. It receives communication-state features from the Communication Layer and trust descriptors from the Fusion Layer, and then performs task-aware service orchestration, feasible-action screening, and DQN-based next-hop selection. The embedded task-hierarchy logic regulates which packet should be served, how much transmission opportunity it may occupy, how retransmission is handled after failure, and how the reward signal is shaped during learning. The selected action is finally delivered back to the Communication Layer for execution, thereby closing the loop from observation to abstraction, decision, and feedback adaptation.

3.2 Communication Layer

The Communication Layer is defined as the observation-and-actuation layer of the proposed framework. It aggregates heterogeneous maritime entities, including vessels, UAVs, USVs, buoys, satellite terminals, and shore-based stations, and provides the physical interface for sensing network conditions and executing routing commands. Rather than serving as a passive transmission substrate, this layer continuously reports local link quality, queue occupancy, mobility state, residual energy, and forwarding behavior to the upper layers, while applying the next-hop decisions generated by the Control Layer.

The maritime network is represented by a directed graph $\mathcal{G} = (V, E)$, where V denotes the set of participating nodes and E denotes the feasible wireless links. For each node, the local observation is organized into network-state, physical-state, and interaction-behavior features. These features are not expanded into separate equations in the main text; instead, they are used as compact inputs for trust evaluation, candidate screening, and DQN-based routing. The detailed observation vector and auxiliary physical models are provided in Appendix H.

For link modeling, we adopt a two-ray large-scale propagation approximation with log-normal shadowing, followed by SNR and packet-error estimation. This modeling choice is suitable for low-height maritime links under line-of-sight or near-line-of-sight propagation and provides the physical basis for communication-feasibility filtering. In the routing framework, the resulting quantities are used only as link-quality descriptors rather than as independent research contributions.

Delay and energy are handled in the same compact manner. End-to-end delay is decomposed into propagation, transmission, queueing, and processing components, while energy consumption is evaluated through transmission, reception, and local processing costs. Since maritime emergency traffic is bursty and affected by retransmission, priority scheduling, and time-varying link conditions, the queueing component is approximated by a general G/G/1 formulation rather than a restrictive M/M/1 assumption. The full mathematical expressions are summarized in Appendix H to keep the main model focused on the routing architecture.

3.3 Traffic-Priority Task System

The proposed task-priority mechanism is embedded into the trust-aware routing loop rather than introduced as an independent optimization layer. It regulates which packet should be served first, how transmission opportunities are distributed among traffic classes, how retransmission is handled after failure, and how service urgency is reflected in reinforcement learning.

The traffic classes considered in this work are critical, important, and normal. For each class c, the system defines a service profile $\Theta_c = (\pi_c, \rho_c, R_c, b_c, \phi_c)$ according to the urgency and reliability requirements of that class. Here, π_c, ρ_c, and R_c denote the scheduling priority, reserved transmission share, and retry budget, respectively, while b_c and ϕ_c regulate reward bias and queue reinsertion behavior. These parameters are configured in advance by a service-policy library and are not learned online by the DQN agent. More urgent traffic is therefore granted earlier access, stronger transmission support, and more robust retransmission protection. Based on these predefined class profiles, the initial traffic mix is modeled by a probability vector d over the three classes, and each newly generated packet inherits the service parameters of its sampled class.

During packet scheduling, the service score of a packet combines class priority, waiting time, and retry history:

$$S(p) = \pi_p + \alpha_{age}\tau_p + \beta_{retry}n_p. \tag{1}$$

where π_p is inherited from the packet class, τ_p denotes the elapsed in-network time, and n_p is the accumulated retry or congestion count. The local sending queue is sorted in descending order of $S(p)$.

For node u with slot capacity $C_u(t)$, the reserved transmission quota of class c is

$$\begin{aligned} q_{(u,c)}(t) &= \lfloor C_u(t)\rho_c \rfloor, \\ \Delta_u(t) &= C_u(t) - \sum\nolimits_{c\in\mathcal{C}} q_{u,c}(t). \end{aligned} \tag{2}$$

where $\Delta_u(t)$ denotes the residual capacity after class-level reservation. The dispatch set is obtained by selecting high-scored packets within each class quota and then filling the residual capacity according to the global score order:

$$\begin{aligned} D_u(t) &= \bigcup\nolimits_{c\in\mathcal{C}} Top_{q_{u,c}(t)}\left\{ p \in Q_u^{\downarrow}(t) \mid c_p = c \right\}, \\ &\sum\nolimits_{p\in\mathcal{D}u(t)} r_p \le C_u(t). \end{aligned} \tag{3}$$

This design protects high-priority packets while preventing long-waiting lower-priority packets from being permanently starved.

As illustrated in Figure 2, the task hierarchy is positioned outside the CTS definition but inside the end-to-end forwarding loop. The task side determines service eligibility and transmission opportunity, whereas the CTS-driven routing layer evaluates the security suitability of candidate relays. Their integration forms a unified task-aware and security-aware forwarding mechanism.

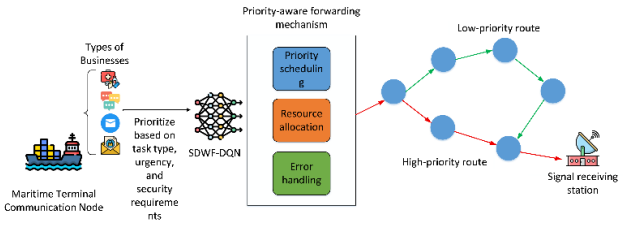


Figure 2. Task-hierarchical service mechanism

3.4 Fusion Layer

The Fusion Layer converts heterogeneous security-related evidence into a compact trust descriptor that can be directly consumed by the routing agent. Instead of relying on a single trust indicator, the proposed framework evaluates each candidate relay from four complementary perspectives: Behavioral Reputation, Static Jurisdictional/ Policy Attribute, Geo-situational Awareness, and Network-Operational Safety. These dimensions jointly describe whether a relay is suitable in terms of observed forwarding behavior, verified authorization prior, geographical consistency, and instantaneous service risk.

For an evaluating node i and a candidate relay j, the security feature vector is denoted as $x_{ij}^{sec}(t) = [N_j, G_{ij}(t), U_j(t), B_{ij}(t)]^{\mathsf{T}}$. The Comprehensive Trust Score (CTS) is defined as

$$\begin{aligned} CTS_{ij}(t) &= \sum\nolimits_{m\in M} \omega_m(t) X_{ij,m}(t), \\ M &= \{jur, geo, net, rep\}. \end{aligned} \tag{4}$$

Here, $X_{ij,m}(t)$ denotes the corresponding component of $x_{ij}^{sec}(t)$, and $\omega_m(t)$ is an adaptive fusion weight. The weights are adjusted according to the instantaneous risk profile of the candidate set, so that dimensions with lower current trust receive greater attention. For dimension m, the candidate-set risk indicator and the corresponding softmax weight are defined as

$$r_m(t) = 1 - \frac{1}{|A_t(i)|} \sum_{j \backslash in A_t(i)} X_{ij,m}(t), \quad \textbf{(5)}$$

The weight normalization satisfies $\omega_m(t) \geq 0$ and $\sum_{m \backslash in M} \omega_m(t) = 1$.

The Behavioral Reputation component captures packet-forwarding reliability from historical interactions and is updated with temporal forgetting. The Static Jurisdictional/Policy Attribute is treated as a bounded policy-compliance prior derived from verified authorization information; it is a heuristic initialization factor rather than a hard security judgment. Geo-situational Awareness evaluates whether the relay location and movement pattern are consistent with the mission context, while Network-Operational Safety describes whether the relay is currently overloaded or likely to introduce delay and packet loss. The component-level realization can be adjusted to different maritime deployments, whereas the CTS aggregation rule in Eqs. (4)–(5) is the core trust-fusion interface used by the routing agent.

3.5 Control Layer

The Control Layer is defined as the decision and coordination layer of the proposed architecture. Its role is to convert communication observations and trust semantics into executable routing actions under task-level service constraints. In the proposed framework, this layer is centered on a DQN-based routing agent, but its functionality is broader than conventional next-hop selection. In addition to routing inference, it also incorporates packet-service orchestration through the embedded task-hierarchy mechanism.

At each decision epoch, the Control Layer receives two categories of inputs. The first category consists of communication-state features delivered by the Communication Layer, such as link condition, queue state, mobility-related variables, and energy indicators. The second category is the trust descriptor generated by the Fusion Layer, namely $CTS_{ij}(t)$ for each candidate relay. By combining these inputs, the Control Layer constructs an augmented routing state with both performance-oriented and security-oriented semantics, upon which the routing policy is learned and executed.

Functionally, the Control Layer completes four tightly coupled tasks. First, it determines packet service eligibility and service order according to the task profile associated with each packet, thereby ensuring that urgent traffic receives differentiated treatment. Second, it applies resource-control logic, including per-class transmission opportunity regulation and candidate-action screening, so that infeasible, noncompliant, or low-value relays can be removed before policy inference. Third, it performs CTS-aware next-hop selection through the DQN agent. Fourth, after each transmission attempt, it updates retry status, reward signals, and control-related statistics according to the forwarding outcome, which in turn influences subsequent routing behavior.

It is important to note that the task-hierarchy mechanism is embedded into the Control Layer as supervisory logic rather than introduced as a separate architectural layer. Accordingly, the Control Layer does not replace trust evaluation performed in the Fusion Layer, nor does it directly execute packet forwarding in the Communication Layer. Instead, it acts as the coordination center that binds service priority, feasible action space, security assessment, routing inference, and post-transmission adaptation into a unified control loop.

With this design, the inter-layer closed loop can be summarized as follows. The Communication Layer senses the environment and reports raw observations; the Fusion Layer transforms these observations into structured trust semantics; the Control Layer performs task-aware routing decisions based on both communication and trust states; and the resulting actions are then executed by the Communication Layer. The transmission outcomes, queue evolution, and forwarding behavior generated during execution are further fed back to update both trust evaluation and control learning. This feedback path enables the proposed framework to adapt jointly to topology variation, traffic urgency, and security risk over time.

3.6 Problem Formulation

Based on the communication, delay, resource, trust, and task-priority models above, the objective of this study is to learn a distributed routing policy that balances reliability, security, and delay under task-dependent service constraints. In the proposed framework, packet service selection precedes next-hop selection; therefore, the routing agent acts only on packets admitted by the local scheduler and evaluates relays from the screened candidate set.

For a path P, we define the Quality of Routing (QoR) as a weighted utility that increases with reliability and security but decreases with delay:

$$U^{QoR}(P) = \omega_S S(P) + \omega_T T(P) - \omega_D D(P) . \quad \textbf{(6)}$$

The long-term goal is to maximize the time-averaged QoR of all active flows by learning a feasible policy π^* over task-admitted packets and screened relay candidates:

$$\pi^\star = \arg\max_{\pi} E_\pi \left[\sum_t \sum_{p \backslash in D_u(t)} U^{QoR}\left(P_{p,t}\right) \right]. \quad \textbf{(7)}$$

This optimization is subject to neighbor feasibility, task admission, queue-capacity, residual-energy, and link-quality constraints:

$$
\begin{aligned}
&a_i(t) \in N_i(t), \quad p \in D_u(t), \\
&0 \le q_i(t) \le Q_{\max,i}, \quad 0 \le \xi_i(t) \le 1, \\
&\gamma_{ij}(t) \ge \gamma_{min}.
\end{aligned} \tag{8}
$$

These constraints are enforced by the candidate pre-screening mask, the task-aware scheduler, and the CTS-aware DQN control rule.

4 Algorithm Design

This section presents the proposed task-hierarchical multidimensional security-aware DQN framework. The complete forwarding cycle consists of task-oriented service orchestration, global candidate pre-screening, CTS-based security fusion, security-aware next-hop selection, priority-aware error control, and task-aware reward update. The overall procedure is summarized in Algorithm 1.

Algorithm 1: Task-Hierarchical Security-Aware Routing (THS–DQN)

Input: Neighbor set $\mathcal{N}_t(i)$, destination coordinates $\mathbf{x}_{\mathrm{dst}}$, policy library and sensitive maritime region $\mathcal{R}_{\mathrm{sens}}$, task profile set $\{\Theta_c\}_{c\in\mathcal{C}}$, local sending queue $Q_u(t)$
Output: Next-hop decision $a_t^\star$
Initialize Q-network parameters θ, replay buffer $\mathcal{D}$, initial penalty coefficient λ_0, and traffic distribution $\mathbf{d}$;
for $t = 1, 2, \ldots$ **do**
  Task orchestration: sort $Q_u(t)$ according to the task-aware service score, allocate per-class quotas, and construct the dispatch set;
  foreach *dispatched packet* p **do**
    Pre-screening: obtain $\mathcal{A}_t(i)$ and the action mask according to policy, physical reachability, load/energy state, mobility, and geometric progress;
    if $\mathcal{A}_t(i) = \varnothing$ **then**
      apply priority-aware requeueing to p and continue;
    end
    CTS fusion: compute the four trust components for each $j \in \mathcal{A}_t(i)$ and obtain $\mathrm{CTS}_{ij}(t)$;
    Action selection: form the state representation and select $a_t^\star$ according to (13);
    Execute transmission and observe the outcome;
    Priority-aware recovery: update the retry state and requeue, drop, or regenerate p according to its service profile;
    Learning: compute the task-aware reward, store the transition, update the Q-network, and refresh the risk coefficient;
  end
end

4.1 Task-Oriented Service Orchestration

For a packet waiting at node u, the task-hierarchy mechanism first determines whether the packet is eligible for service in the current slot. The scheduler uses the score in Eq. (1), allocates class quotas through Eq. (2), and forms the dispatch set according to Eq. (3). This procedure ensures that urgent packets receive preferential service while lower-priority packets can still be promoted through aging and retry accumulation. Thus, the service-ordering mechanism is explicitly coupled to the subsequent routing decision rather than being left as an external priority queue.

4.2 Global Pre-screening

Let node i be the current forwarding node and let $\mathrm{N}_t(i)$ be its neighbor set. Directly applying the DQN policy to all neighbors would enlarge the action space and expose the learner to infeasible relays. Therefore, a global candidate pre-screening mechanism is executed before policy inference. It sequentially removes candidates that violate policy-compliance requirements, fail the communication-feasibility test, suffer from excessive queue or low residual energy, are predicted to lose connectivity shortly, or provide little destination-oriented progress.

The output of this stage is a reduced candidate set $\mathcal{A}_t(i)$ and a corresponding action mask. The screening process can be written compactly as a sequence of filtering operations:

$$
\begin{aligned}
&\mathcal{N}_t^{(1)}(i) = \left\{ j \in \mathcal{N}_t(i) \middle| \ \chi_{\mathrm{pol}}(j) = 1 \right\}, \\
&\mathcal{N}_t^{(2)}(i) = \left\{ j \in \mathcal{N}_{t^{(1)}}(i) \middle| \ \gamma_{ij}(t) \ge \gamma_{\min} \right\}.
\end{aligned} \tag{9}
$$

$$
\begin{aligned}
\mathcal{N}_t^{(3)}(i) = \{ j \in \mathcal{N}_t^{(2)}(i) \mid \ & q_j(t) \le q_{max}, \\
& \xi_j(t) \ge \xi_{min} \}, \\
\mathcal{N}_t^{(4)}(i) = \left\{ j \in \mathcal{N}_t^{(3)}(i) \middle| \widehat{T}_{ij}(t) \ge T_{min} \right\}&.
\end{aligned} \tag{10}
$$

After policy, link, resource, and mobility filtering, destination-oriented progress is measured by

$$
\Delta d_{ij}(t) = \left| x_i(t) - x_{\mathrm{dst}} \right| - \left| x_j(t) - x_{\mathrm{dst}} \right|. \tag{11}
$$

The final candidate set is obtained by Top-K reduction:

$$
\mathcal{A}_t(i) = \text{Top-}K_{\,j \in \mathcal{N}_t^{(4)}(i)} \Psi\left(\Delta d_{ij}(t), \widehat{\gamma}_{ij}(t), \tau_{ij}(t) \right). \tag{12}
$$

Only the retained relays are passed to the Fusion Layer and the DQN controller. In this way, the framework reduces inference cost, improves training stability, and preserves relays that are simultaneously feasible, policy-admissible, and useful for forwarding.

4.3 Multidimensional Security-Aware Fusion and DQN Decision

For each screened relay $j \in \mathcal{A}_t(i)$, the routing agent

obtains the CTS from the Fusion Layer and imposes a security penalty on low-trust candidates. The final action is selected by a CTS-aware Q-value comparison:

$$a_t^{\star} = \arg\max_{j \in \mathcal{A}_t(i)} \left[Q_\theta(s_t, j) - \lambda_t \left(1 - CTS_{ij}(t)\right)^{\alpha} \right]. \quad \textbf{(13)}$$

where λ_t is a risk-dependent penalty coefficient and α controls the sensitivity to trust degradation. Equivalently, the CTS-aware security penalty is

$$P_{\text{sec}}(j,t) = \lambda_t \left(1 - CTS_{ij}(t)\right)^{\alpha}. \quad \textbf{(14)}$$

The risk coefficient is updated according to the observed environmental risk level $\mathcal{R}_t$:

$$\lambda_{t+1} = \text{clip}((1 - \delta)\,\lambda_t + \delta \mathcal{R}_t, \lambda_{\min}, 1). \quad \textbf{(15)}$$

The state representation combines packet-level task attributes with candidate-relay descriptors such as link quality, link persistence, queue state, geometric progress, and CTS.

4.4 Priority-Aware Error Control and Reward Shaping

After each transmission attempt, the controller updates the packet retry counter and applies class-dependent recovery. Packets that still have retry budget may be reinserted into the queue, whereas packets that exceed the budget are dropped or regenerated according to the emergency-service requirement. Front reinsertion is enabled only for classes whose service profile explicitly activates this behavior.

The reinforcement-learning reward is shaped by three outcome types: successful delivery, successful intermediate forwarding, and failure. Delivery receives a positive reward with task bonus and security penalty; intermediate forwarding considers link delay, queue balance, task bonus, and security penalty; failure receives a class-aware negative reward:

$$r_t = \begin{cases} R_{dst} + b_p - P_{\text{sec}}(j,t), & \textit{delivery}, \\ -w_{ij}(t) - \zeta_q (q_j(t) - q_{eq}) & \\ +\zeta_g g_j(t) + \zeta_b b_p - P_{\text{sec}}(j,t), & \textit{forwarding}, \\ -(F + b_p), & \textit{failure}. \end{cases} \quad \textbf{(16)}$$

This reward design keeps task urgency inside the learning loop without replacing the CTS as the security descriptor.

4.5 Routing Decision and Policy Generation

During online inference, each forwarding opportunity is handled through a coupled procedure that includes task scheduling, quota allocation, candidate pre-screening, CTS fusion, DQN-based routing decision, and priority-aware error control. For each node u at time slot t, the scheduler first forms the dispatch set from the local queue. For each dispatched packet, the routing agent identifies feasible relays, computes their CTS values, and chooses the next hop according to (13). Once transmission is completed, the queue state, retry counter, reward, and reputation statistics are jointly updated.

In high-risk scenarios, the proposed strategy increases the influence of the CTS-based penalty and shifts routing decisions toward higher-trust relays. In low-risk and well-connected scenarios, the policy relies more heavily on performance terms such as delay and queue balance. Meanwhile, task-critical packets benefit from earlier service opportunities, stronger retry protection, and larger reward bonuses. The resulting policy therefore adapts to communication performance, service differentiation, and security assurance at the same time.

4.6 Complexity and Implementation Considerations

The computational complexity of each decision cycle mainly consists of candidate pre-screening, Top-K ranking, CTS computation, and DQN forward inference. If $d(i)$ is the degree of node i and K is the retained candidate size, screening scales linearly with $d(i)$, Top-K ranking scales with $d(i)\log K$, and inference scales with K times the neural-network cost. Since $K \ll d(i)$ in dense neighborhoods, the pre-screening stage bounds the action space and reduces unnecessary Q-value evaluations. The memory overhead comes from the replay buffer, candidate embeddings, and network parameters. In implementation, fixed-length padding and action masks are recommended for candidate-link embeddings to facilitate batch training.

5 Simulation

5.1 Simulation Setup

To more clearly demonstrate the effectiveness of the proposed multidimensional security-aware dynamic weighted fusion framework, we conduct simulation experiments on a grid-based maritime topology with 49 nodes constructed using NetworkX. The simulation platform is deployed on a Windows 11 x64 workstation equipped with an AMD Ryzen 7 7840H processor and 32.00 GB of memory.

Unless otherwise specified, all compared methods are evaluated under the same dynamic maritime topology, traffic-generation setting, and security environment. The reported metrics include packet delivery ratio, average end-to-end delay, hop count, throughput, retransmission ratio, and the proposed CTS. For the task-hierarchy mechanism, the default traffic mix follows the configured distribution over critical, important, and normal traffic. The class-dependent service profiles $\Theta_c = (\pi_c, \rho_c, R_c, b_c, \phi_c)$ are preset according to the same service-policy library for all compared methods whenever task-aware scheduling is enabled, so that the performance differences are attributable to the routing strategy rather than inconsistent task-control parameters. The baselines include conventional Shortest Path (SP), a vanilla DQN-based routing method, and several collaborative retraining variants under dynamic-

topology conditions.

To obtain reliable performance curves under varying traffic loads, the baseline experiment uses six generated-packet loads, i.e., 250, 500, 750, 1000, 1250, and 1500 packets. Each policy is trained once at the maximum load and then tested across the complete load sweep, so that the curves reflect generalization under increasing traffic pressure rather than repeated load-specific retraining. The default DQN and training settings are summarized in Table 2.

Table 2. Default DQN and training hyperparameters

Item	Default setting
Network architecture	MLP with two hidden layers
Hidden units	128 units per layer
Activation function	ReLU
Optimizer	Adam
Learning rate	1×10^{-3}
Discount factor	0.95
Exploration strategy	Decaying ε-greedy
Replay buffer size	1×10^{5} transitions
Mini-batch size	64
Target-network update interval	200 training steps

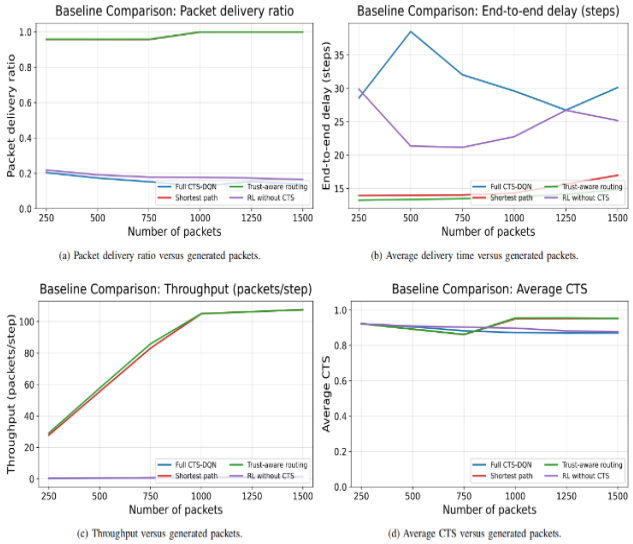


Figure 3. Baseline comparison under traffic load

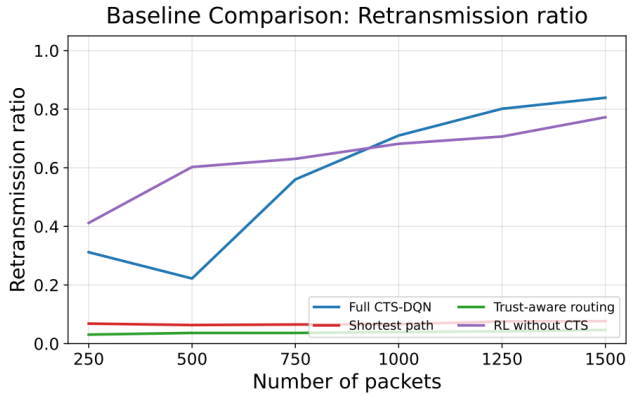


Figure 4. Retransmission ratio under traffic load

5.2 Performance Under Dynamic Topology and Increasing Load

Under slight topology dynamics, four methods are compared: SP, Non-Retrained Collaborative, Global Retrained Collaborative, and Local Retrained Collaborative. Figure 3 reports packet delivery ratio, average delivery time, throughput, and average CTS under the six-load sweep, while Figure 4 reports retransmission behavior.

When the network load is relatively low, topology perturbation has only a limited influence on overall routing performance. However, once the traffic load exceeds 1000 packets, both the SP method and the Non-Retrained Collaborative method exhibit a noticeably higher packet-loss rate. In contrast, the Collaborative methods with either global retraining or local retraining remain comparatively stable after topology perturbation. Specifically, their average delivery ratio stays around 90%, the average delivery delay remains nearly unchanged, and the number of congestion events is reduced by more than 700 compared with the other two methods.

Moreover, although local retraining achieves performance comparable to that of global retraining, it requires substantially less training time. This result indicates that local retraining provides a more favorable trade-off between adaptability and computational overhead in large-scale dynamic maritime networks.

5.3 CTS-Enhanced Security Mechanism

To evaluate the proposed security-aware routing mechanism, we introduce the CTS as an explicit indicator for penalizing insecure relays while preserving routing performance. During simulation, the environment outputs per-hop CTS values and aggregates time-series statistics for visualization and comparative analysis under identical topology and load settings.

Metrics and setup:

In addition to packet delivery ratio, delay, and hop count, we further report:

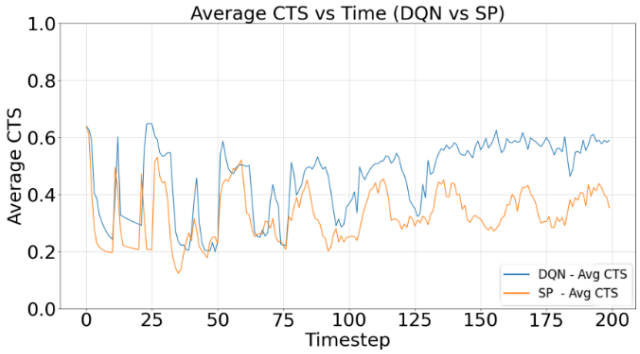
(i) the average CTS over time; and

(ii) the average CTS under different levels of network load.

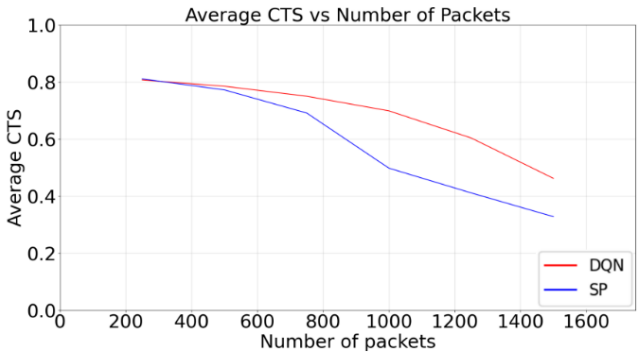
The compared baselines include SP and vanilla DQN, and all strategies operate over the same dynamic maritime graph with identical traffic and security conditions.

As shown in Figure 5(a), the security-aware DQN consistently maintains a higher CTS than SP under the same security environment, indicating that it is more effective at avoiding low-credibility relays. In Figure 5(b), the average CTS decreases with increasing network load for all methods, but the proposed approach exhibits a noticeably smaller decline. This behavior suggests that, even under congestion, the proposed routing policy still tends to favor high-trust paths whenever feasible.

Overall, the CTS trends are consistent with the observed improvements in packet delivery and delay performance, confirming that the proposed fusion-aware routing policy enhances routing security without sacrificing communication efficiency.



(a) Average CTS versus time (DQN vs. SP).



(b) Average CTS versus network load.

Figure 5. CTS comparison across time and network load

6 Discussion

6.1 Comparison with Representative Routing and Learning Algorithms

The simulation results should be interpreted together with the algorithmic differences between the proposed THS-DQN framework and representative algorithms in the literature. Table 3 summarizes the comparison from the perspectives of routing objective, trust modeling, service differentiation, and adaptability to maritime emergency networks. Conventional routing protocols such as AODV [3] provide reactive route discovery and local repair, but they mainly optimize connectivity and path availability. Cluster-based hierarchical routing and LEACH-derived designs reduce local communication overhead through aggregation, but cluster-head selection and relay-role concentration may still create bottlenecks or single points of failure [6-7]. Heuristic and swarm-inspired routing methods, including recent energy-aware maritime or underwater routing schemes [5], are effective at incorporating handcrafted routing metrics, but their decision rules are usually fixed once designed and cannot fully adapt to dynamic trust variation, task urgency, and policy constraints.

Trust and reputation systems provide another line of related work. Recent IoT-, WSN-, and social-IoT-oriented trust-management studies have shifted the focus from early online-service reputation scoring to context-aware trust computation, recommendation credibility, attack-resilient reputation aggregation, and AI-assisted trust modeling [15-19]. Blockchain-based credibility evaluation in the Internet of Ships further improves the integrity and traceability of navigational-event dissemination [4]. These studies are highly relevant to the security motivation of this paper because they show how behavioral evidence, contextual relationships, and tamper-resistant records can be used to assess the credibility of network entities. However, most of them still treat trust assessment as a separate evaluation or dissemination process rather than directly embedding multidimensional trust semantics into packet-level next-hop selection, task scheduling, and reward shaping. Learning-based routing methods, including DQN-based communication-network optimization [8-9], UAV-swarm MARL routing [12], and multi-agent routing in highly dynamic MANET or integrated access and backhaul networks [13-14], improve adaptivity under dynamic environments. However, most of them remain primarily performance-oriented and do not explicitly integrate mission-priority scheduling, jurisdictional/policy feasibility, and multidimensional trust fusion into a unified routing loop.

Table 3. Qualitative comparison with representative algorithms in the literature

Method family	Representative studies	Main capability	Difference from the proposed THS-DQN
Reactive ad hoc routing	AODV [3]	Discovers routes on demand and supports local repair under topology changes.	No trust-aware priority control
Hierarchical clustering	LEACH-derived clustering [6-7]	Reduces communication overhead through cluster-head or tree-based aggregation.	Relay-role bottle-neck dependence
Heuristic and swarm routing	Energy-aware heuristic routing [5]	Optimizes route quality using hand-crafted heuristics such as distance, energy, and transmission deviation.	Handcrafted security-insensitive heuristics
Trust/reputation models	IoT/ WSN/ SIoT trust management [15-19]; blockchain credibility [4]	Evaluates node, object, or event credibility from direct interactions, recommendations, contextual evidence, AI-assisted aggregation, or tamper-resistant records.	Trust assessment separated from routing learning
RL/MARL optimization	DQN/DRL surveys [8-9]; MARL routing [12-14]	Learns adaptive policies in dynamic networks and multi-agent environments.	Security- and mission-agnostic learning

6.2 Limitations and Future Work

Although the proposed framework improves the latency-security tradeoff under the evaluated scenarios, several limitations remain. First, the current validation is still simulation-based. The 49-node grid topology and the current load-sweep experiments provide useful controlled evidence, but they cannot fully represent real maritime networks with irregular vessel trajectories, intermittent satellite or shore links, heterogeneous protocol stacks, and rapidly changing sea-state conditions. Future work will therefore extend the evaluation to larger and more irregular topologies, real AIS-assisted mobility traces, and field-test or emulation platforms.

Second, the physical-layer assumptions are intentionally simplified. The Two-Ray model, AWGN noise, and BPSK modulation are adopted to isolate routing-layer effects and make different routing policies comparable under controlled conditions. Real maritime communication may involve sea-state-dependent fading, atmospheric ducting, co-channel interference, antenna-height variation, and adaptive modulation and coding. Incorporating these channel effects into the simulator is an important next step for validating the robustness of the proposed method.

Third, the current policy library and hyperparameters are manually configured. Although this design makes the task-hierarchy mechanism transparent and controllable, the optimal priority coefficients, resource quotas, retry budgets, CTS weights, and reward coefficients may vary across different rescue missions and network densities. Future work will investigate automatic policy tuning, meta-learning, or constrained reinforcement learning to adapt these parameters while preserving safety and service-priority guarantees.

Fourth, the Static Jurisdictional/Policy Attribute is used only as a bounded and auditable policy-compliance prior. It should not be interpreted as an intrinsic judgment about nationality, ethnicity, or identity. Since inappropriate configuration may introduce unfair routing decisions, future deployments should combine this factor with verifiable authorization records, privacy-preserving trust management, and periodic auditing. Finally, the current baseline set can be further expanded. More recent trust-aware secure routing algorithms, graph-neural-network-based routing, and multi-agent reinforcement-learning methods should be included in future comparisons to evaluate the generality of the proposed THS-DQN framework.

7 Conclusion

This paper addressed maritime emergency routing under dynamic topologies, heterogeneous node attributes, scarce transmission opportunities, and stringent security constraints by coupling multidimensional trust evaluation with task-hierarchical deep reinforcement learning. The proposed framework integrates: (i) a Communication Layer that provides refined delay and energy-related state features; (ii) a Fusion Layer that combines Behavioral Reputation, Static Jurisdictional/Policy Attribute, Geo-situational Awareness, and Network-Operational Safety into the CTS; and (iii) a Control Layer in which task-class generation, priority scheduling, quota allocation, CTS-aware next-hop selection, and priority-aware error recovery are jointly coordinated in a DQN-based loop. Comparative simulations show that the proposed scheme improves throughput by 18.3% and reduces end-to-end delay by 22.7% at a mobility speed of 20 m/s compared with classical baselines. Under adversarial conditions, malicious nodes are modeled as delay-amplifying selective blackhole relays: once selected as the next-hop relay, they double the forwarding delay and drop packets with a fixed probability of 0.10. By dynamically varying the malicious-node ratio in the security experiments, we further evaluate the robustness of the proposed routing policy under different attack intensities. In the representative scenario with 15% malicious nodes, the packet delivery ratio remains above 92.6%, demonstrating the effectiveness of CTS-guided relay selection and priority-aware error recovery against selective dropping and delay manipulation. The discussion further shows that the main advantage of the proposed method lies in the integration of mission-aware service control, multidimensional trust fusion, and learning-based routing. Future work will focus on larger and more realistic maritime scenarios, richer channel and mobility models, automatic parameter tuning, and broader comparisons with trust-aware secure routing and MARL-based approaches.

Appendix H Auxiliary Mathematical Details

This appendix collects auxiliary formulas that support the system model and algorithm implementation. Moving these derivations out of Sections 3 and 4 keeps the main text focused on the proposed architecture and routing logic.

Appendix H.1 Communication, Delay, and Energy Models

For a link from node i to node j, the asymptotic two-ray received-power model beyond the cross-over distance is written as

$$P_{r,ij}(t) = P_{t,i} G_t G_r \frac{h_t^2 h_r^2}{d_{ij}^4(t)}. \tag{17}$$

With log-normal shadowing, the corresponding large-scale path loss is

$$PL_{ij,t}(dB) = 10\log_{10}\left(\frac{d_{ij}^4(t)}{G_t G_r h_t^2 h_r^2}\right) + X_{\sigma_s}. \tag{18}$$

The channel gain, SNR, bit-error probability, and packet-reception probability are computed as

$$g_{ij,t} = 10^{-PL_{ij,t}/10}, \quad \Gamma_{ij,t} = \frac{P_{t,i} g_{ij,t}}{\sigma^2}, \\ P_b = Q\left(\sqrt{2\Gamma_{ij,t}}\right), \quad P_s = \left(1 - P_b\right)^L. \tag{19}$$

The one-hop delay is decomposed into propagation, transmission, queueing, and processing terms. The queueing term follows a G/G/1 approximation:

$$D_{queue,i,t} \approx \frac{\rho_{i,t}}{1-\rho_{i,t}} \cdot \frac{c_{a,i,t}^2 + c_{s,i,t}^2}{2} \cdot \frac{1}{\mu_{i,t}}, \\ \rho_{i,t} = \frac{\lambda_{i,t}}{\mu_{i,t}} < 1. \tag{20}$$

The energy model includes transmission, reception, and local processing costs and is used to construct residual-energy features for candidate screening.

Appendix H.2 Relation to Main Algorithm

The task-scheduling, CTS-weighting, candidate-screening, security-penalty, reward-shaping, and compact optimization formulas are kept in Sections 3 and 4 because they define the core algorithmic logic of the proposed THS-DQN framework. This appendix therefore only retains auxiliary physical-layer, delay, and energy expressions used to construct the state features and feasibility indicators.

References

[1] N. Wakabayashi, I. Jurdana, Maritime Communications and Remote Voyage Monitoring, *2020 International Conference on Broadband Communications for Next Generation Networks and Multimedia Applications (CoBCom)*, Graz, Austria, 2020, pp. 116-123. https://doi.org/10.1109/CoBCom49975.2020.9174182

[2] F. S. Alqurashi, A. Trichili, N. Saeed, B. S. Ooi, M.-S. Alouini, Maritime Communications: A Survey on Enabling Technologies, Opportunities, and Challenges, *IEEE Internet of Things Journal*, Vol. 10, No. 4, pp. 3525-3547, February, 2023. https://doi.org/10.1109/JIOT.2022.3219674

[3] Y. Li, Q. Li, H. Yu, L. Li, TAAODV: TOPSIS-Aided AODV Routing Protocol to Diffuse Mass Information in the Unstable and High-Density UAV Ad Hoc Networks, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 26, No. 10, pp. 17255-17268, October, 2025. https://doi.org/10.1109/TITS.2024.3468007

[4] Y. Wang, T. Ye, E. Zio, T. Wang, B. Wu, A Blockchain-Based Credibility Evaluation Scheme for Navigational Event Dissemination in the Internet of Ships, *Reliability Engineering and System Safety*, Vol. 248, Article No. 110149, pp. 1-15, August, 2024. https://doi.org/10.1016/j.ress.2024.110149

[5] W. Zhu, X. Yang, T. Wu, Y. Qiu, A Routing Algorithm for Underwater Acoustic–Optical Hybrid Wireless Sensor Networks Based on Intelligent Ant Colony Optimization and Energy-Flexible Global Optimal Path Selection, *IEEE Sensors Journal*, Vol. 24, No. 10, pp. 17116-17126, May, 2024. https://doi.org/10.1109/JSEN.2024.3386892

[6] M. Aslam, N. Javaid, A. Rahim, U. Nazir, A. Bibi, Z. A. Khan, Survey of Extended LEACH-Based Clustering Routing Protocols for Wireless Sensor Networks, *Proceedings of the 2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems (HPCC-ICESS)*, Liverpool, UK, 2012, pp. 1232-1238. https://doi.org/10.1109/HPCC.2012.181

[7] A. Farzaneh, M.-A. Badiu, J. P. Coon, LEAST: A Low-Energy Adaptive Scalable Tree-Based Routing Protocol for Wireless Sensor Networks, *arXiv preprint*, arXiv:2211.09443, November, 2022. https://arxiv.org/abs/2211.09443

[8] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, D. I. Kim, Applications of Deep Reinforcement Learning in Communications and Networking: A Survey, *IEEE Communications Surveys & Tutorials*, Vol. 21, No. 4, pp. 3133-3174, Fourthquarter, 2019. https://doi.org/10.1109/COMST.2019.2916583

[9] T. T. Nguyen, N. D. Nguyen, S. Nahavandi, Deep Reinforcement Learning for Multiagent Systems: A Review of Challenges, Solutions, and Applications, *IEEE Transactions on Cybernetics*, Vol. 50, No. 9, pp. 3826-3839, September, 2020. https://doi.org/10.1109/TCYB.2020.2977374

[10] K. Zhang, Z. Yang, T. Başar, Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms, *Handbook of Reinforcement Learning and Control*, Springer, 2021, pp. 321-384. https://doi.org/10.1007/978-3-030-60990-0_12

[11] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, D. Hassabis, Human-Level Control through Deep Reinforcement Learning, *Nature*, Vol. 518, No. 7540, pp. 529-533, February, 2015. https://doi.org/10.1038/nature14236

[12] Z. Wang, H. Yao, T. Mai, Z. Xiong, X. Wu, D. Wu, S. Guo, Learning to Routing in UAV Swarm Network: A Multi-Agent Reinforcement Learning Approach, *IEEE Transactions on Vehicular Technology*, Vol. 72, No. 5, pp. 6611-6624, May, 2023. https://doi.org/10.1109/TVT.2022.3232815

[13] S. Kaviani, B. Ryu, E. Ahmed, K. A. Larson, A. Le, A. Yahja, J. H. Kim, DeepCQ+: Robust and Scalable Routing with Multi-Agent Deep Reinforcement Learning for Highly Dynamic Networks, *Proceedings of the 2021 IEEE Military Communications Conference (MILCOM)*, San Diego, CA, USA, 2021, pp. 31-36. https://doi.org/10.1109/MILCOM52596.2021.9652948

[14] S. Yamin, H. H. Permuter, Multi-Agent Reinforcement Learning for Network Routing in Integrated Access Backhaul Networks, *Ad Hoc Networks*, Vol. 153, Article No. 103347, February, 2024. https://doi.org/10.1016/j.adhoc.2023.103347

[15] Z. Yan, P. Zhang, A. V. Vasilakos, A Survey on Trust Management for Internet of Things, *Journal of Network and Computer Applications*, Vol. 42, pp. 120-134, June, 2014. https://doi.org/10.1016/j.jnca.2014.01.014

[16] J. Guo, I.-R. Chen, J. J. P. Tsai, A Survey of Trust Computation Models for Service Management in Internet of Things Systems, *Computer Communications*, Vol. 97, pp. 1-14, January, 2017. https://doi.org/10.1016/j.comcom.2016.10.012

[17] M. Aaqib, A. Ali, L. Chen, O. Nibouche, IoT Trust and Reputation: A Survey and Taxonomy, *Journal of Cloud Computing*, Vol. 12, No. 1, Article No. 42, March, 2023. https://doi.org/10.1186/s13677-023-00416-8

[18] M. Segata, P. Casari, M. Lestas, A. Papadopoulos, D. Tyrovolas, T. Saeed, G. K. Karagiannidis, C. Liaskos, CoopeRIS: A Framework for the Simulation of Reconfigurable Intelligent Surfaces in Cooperative Driving Environments, *Computer Networks*, Vol. 248, Article No. 110443, June, 2024. https://doi.org/10.1016/j.comnet.2024.110443

[19] W. Fang, C. Zhang, Z. Shi, Q. Zhao, L. Shan, BTRES: Beta-Based Trust and Reputation Evaluation System for Wireless Sensor Networks, *Journal of Network and Computer Applications*, Vol. 59, pp. 88-94, January, 2016. https://doi.org/10.1016/j.jnca.2015.06.013

Biographies

Bo Lin is currently a student with the Department of Artificial Intelligence, School of Information Engineering, Jimei University. His research interests include deep learning, multi-agent reinforcement learning, and their applications in intelligent networking, modeling, decision-making, and optimization.

Lianyou Lai received the M.S. degree in power electronics and power transmission from Jiangsu University, China, in 2004. He is currently an Associate Professor and Head of the Department of Electronic Information Engineering, Jimei University. His research interests include maritime communication networks, secure routing, and intelligent wireless optimization.