Computer Networks 289 (2026) 112648

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Enhancing routing in dynamic and dense networks through UWB swarm ranging for embodied intelligence

Fangyu Zhou, Yiqian Wang, Feng Shan *, Wenjia Wu

School of Computer Science and Engineering, Southeast University, Nanjing 211189, Jiangsu, PR China

ARTICLE INFO

Keywords:
Ultra-wideband
Embodied intelligence
Swarm ranging
Ad hoc network
Multi-point relay selection

ABSTRACT

Embodied intelligence has emerged as a key direction in Artificial Intelligence and is playing an increasingly significant role across various domains. Small embodied agents/robots, such as Unmanned Aerial Vehicles (UAVs), operate in swarm mode based on embodied intelligence and provide greater sensing and processing capabilities compared to individual agents/robots. However, they also introduce new challenges: dynamic and dense networks formed by resource-constrained embodied intelligent agents demand real-time, low forwarding overhead networking protocol. We leverage our previously proposed Ultra-Wideband (UWB) Swarm Ranging Protocol, and (1) propose a UWB-based link quality estimation method that leverages real-time inter-agent ranging to enable faster and more accurate estimation of link states, thereby mitigating the impact of network dynamics. We then (2) formulate a novel Multi-Point Relays (MPR) selection problem that aims to effectively reduce forwarding overhead while ensuring the Quality of Service (QoS) provided by selected relay nodes. An approximation algorithm is proposed, and its approximation ratio is analyzed. We also (3) design an integrated message format that combines ranging and routing functionalities, which is implemented on resource-constrained agents with a 168 MHz MCU and 192 KB memory. Lastly, both simulations and real-world experiments show that, our protocol shortens the round-trip time by approximately 54.7% compared to classic AODV protocol, while improving the packet delivery rate by approximately 17.3% over OLSR protocol. We also open-source our implementation on GitHub, providing a reproducible platform for further research and development: https://github.com/SEU-NetSI/crazyflie-firmware.git.

1. Introduction

In recent years, embodied intelligence has emerged as a key paradigm in artificial intelligence and robotics, emphasizing the tight integration of perception, control, and decision-making through continuous interaction with the physical environment. Unlike purely computational models, embodied systems operate in the real world via closed-loop sensorimotor feedback [1–4].

These systems have demonstrated significant potential in a variety of applications, including autonomous vehicles navigating complex traffic [5,6], rescue agents/robots operating in disaster zones [7,8], unmanned aerial vehicle (UAV) swarms for environmental monitoring [9–11], and deep-sea agents/robots performing mapping and recognition in extreme underwater environments [12,13]. Collectively, these applications underscore the growing importance of embodied intelligence across land, air, and marine domains.

Small embodied agents/robots, such as UAVs [14–16], wheeled platforms [17], and quadrupeds [18], have gained increasingly attention for their low cost, mobility, and scalability. When deployed in swarms, they can surpass monolithic systems by enabling distributed sensing, enhancing fault tolerance, and facilitating parallel task execution. Fig. 1 shows an application scenario in automated logistics. UAVs transport parcels between shelves and trucks, wheeled agents/robots handle ground delivery tasks, and robotic arms on conveyor belts manage sorting and packaging. Inspide of these, Quadruped agents/robots may assist in navigating complex terrains or supporting inspection tasks within warehouses. Together, these heterogeneous agents/robots from a collaborative swarm that improves efficiency and flexibility in modern supply chain operations, while simultaneously posing stringent requirements for reliable inter-agent[1] communication and coordination.

However, this paradigm shift also presents significant challenges, particularly in the form of dynamic and dense communication networks, where frequent, low-latency data exchange is critical for effective coordination. Although Wi-Fi and 5G provide high-density connectivity in everyday scenarios, they are often unsuitable for mobile robotic systems [19–21]. Limited Wi-Fi coverage, particularly in remote

* Corresponding author.
E-mail addresses: zhoufangyu@seu.edu.cn (F. Zhou), yiqianwang@seu.edu.cn (Y. Wang), shanfeng@seu.edu.cn (F. Shan), wjwu@seu.edu.cn (W. Wu).

[1] We use *robots* and *agents* interexchangably in this paper.

https://doi.org/10.1016/j.comnet.2026.112648
Received 14 October 2025; Received in revised form 2 June 2026; Accepted 2 August 2026
Available online 11 August 2026

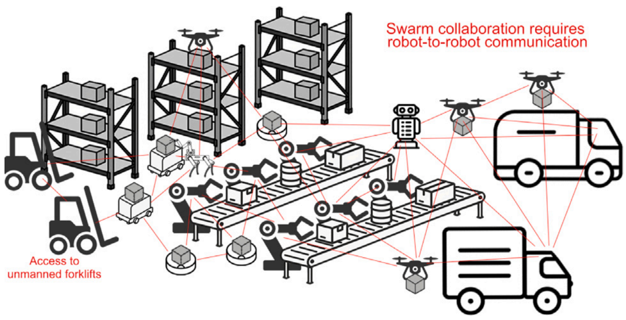


Fig. 1. Illustration of an automated logistics scenario where heterogeneous embodied intelligent agents/robots collaborate as a swarm: UAVs transport parcels between shelves and trucks, wheeled agents/robots carry out ground delivery, robotic arms on conveyor belts perform sorting and packaging, and quadruped agents/robots assist with navigation or inspection. These heterogeneous swarm collaborative operations require reliable inter-agents communication and coordination.

or infrastructure-poor areas, poses a challenge as it cannot accommodate the mobility of agents/robots beyond the range of fixed routers. Despite 5G's broad coverage, its latency and high deployment cost hinder its application in agent/robot swarm systems with high real-time requirements. These constraints underscore the need for lightweight and infrastructure-free self-organizing networks.

On the other hand, high mobility leads to rapid topology changes, while dense deployments result in severe channel contention. These characteristics form what we term ***dynamic and dense networks***, which impose stringent demands on the robustness, responsiveness, and scalability of self-organizing network protocols. Traditional ad hoc protocols, originally designed for static or sparsely populated networks, perform poorly under such conditions. Heavy forwarding overhead and unreliable link estimation severely degrade communication efficiency and task performance.

Conventional multi-hop ad hoc routing protocols, such as Ad-hoc On-Demand Distance Vector (AODV) [22] and Optimized Link State Routing (OLSR) [23], exhibit notable limitations in such environments. AODV's on-demand route discovery reduces maintenance overhead but introduces latency and is susceptible to frequent link breakages. OLSR improves adaptability through periodic broadcasts and proactive routing; however, the selection of Multi-Point Relays (MPRs) depends on statistical measurements of packet loss, which introduce considerable latency and limit real-time responsiveness. Inappropriately selected MPRs often cause the network to fail to meet Quality of Service (QoS) requirements, leading to substantial packet loss and network-wide retransmissions [24]. Prior studies have proposed improvements to MPR selection by considering factors such as a relay node's remaining transmission time within the source's communication range or its residual energy [25,26].

As a conclusion, previous research on conventional multi-hop ad hoc routing protocols does not apply directly to *dynamic and dense networks*. We identify the following key challenges in the context of embodied intelligent agents networking:

- In dynamic networks, traditional link-quality estimation methods introduce latency, delaying routing decisions and undermining real-time responsiveness.
- In dense networks, conventional topology discovery mechanisms incur excessive forwarding overhead, thereby degrading network efficiency.
- Existing routing protocols impose heavy resource demands, making them unsuitable for resource-constrained embodied intelligent agents.

1.1. Motivation and contributions

The motivation for this work stems directly from our previously proposed Ultra-Wideband (UWB) Swarm Ranging Protocols [27,28]. In this protocol, through periodic broadcasting, each agent can simultaneously calculate its distance to neighboring agents.

The core idea of this paper is to incorporate these real-time distance measurements into (1) link-quality estimation, which reduces routing decision delays and achieves real-time responsiveness, and (2) MPR selection, which reduces forwarding overhead and increases network efficiency.

Another motivation comes from the fact that the Swarm Ranging Protocol relies on periodic ranging messages, while ad hoc routing protocols also employ periodic broadcast mechanisms. Thus, **a fundamental idea** is to embed routing control information into ranging messages to enable protocol implementation on resource-constrained agents.

The main contributions of this paper are summarized as follows.

- We propose a UWB-based, distance-aware link quality estimation method that leverages real-time inter-agent ranging to enable faster and more accurate estimation of link states, thereby mitigating the impact of network dynamics.
- We formulate a novel MPR selection problem that aims to effectively reduce forwarding overhead while ensuring the QoS provided by selected relay nodes. An approximation algorithm is proposed, and its approximation ratio is analyzed.
- We design an integrated message format that combines ranging and routing functionalities. Together with a well-structured software framework, a prototype system is implemented on resource-constrained agents with a 168 MHz MCU and 192 KB memory.
- Both simulations and real-world experiments show that, our protocol shortens the round-trip time by about 54.7% compared to classic AODV protocol, while improving the Packet Delivery Rate (PDR) by about 17.3% over OLSR protocol. We also open-source our implementation on GitHub, providing a reproducible platform for further research and development: https://github.com/SEU-NetSI/crazyflie-firmware.git.

The remainder of this paper is organized as follows. Section 2 provides a review of related work. In Section 3, we present the system model and formally define the problem. Our methodology for data link quality estimation is introduced in Section 4. The optimization strategy for MPR selection is discussed in Section 5. Section 6 presents our prototype, including its message format and hardware/software implementation. Experimental evaluation is detailed in Section 7. Section 8 concludes the paper.

2. Related work

In dynamic and dense networks, the highly variable topology and dense node distribution pose significant challenges for efficient routing. Current studies on routing protocols largely focus on optimizing and improving existing traditional protocols. Among them, the mainstream approach is to enhance the OLSR protocol, with improvements generally categorized into two directions: enhancing the link quality estimation mechanism and optimizing the routing mechanism.

2.1. Link quality estimation

The original OLSR protocol proposed by the IETF in 2003 (RFC3626) did not consider link quality. Instead, path cost was solely measured by hop count, where all directly connected nodes were considered equivalent. However, in practice, due to the complexity of wireless channels, links exhibit significant differences in terms of packet loss rate, delay, and bandwidth. Consequently, evaluating routes based solely on hop count often leads to suboptimal decisions. To address this limitation, the open-source implementation olsrd introduced link quality evaluation in 2004 [29]. By measuring the reception probability of HELLO messages periodically exchanged between neighbors, the protocol estimated both link quality and neighbor link quality. The combined metric was further transformed into the Expected Transmission Count (ETX), which better reflects the actual cost of a reliable transmission. Later, in 2014, IETF formally introduced the concept of link metrics in OLSRv2 (RFC7181) [30], validating the effectiveness of ETX-based link quality estimation.

Building upon this foundation, various enhancements have been proposed. For example, Rosati et al. incorporated positional information and relative mobility into ETX computation, leading to the Predictive OLSR (P-OLSR) protocol [31]. By anticipating variations in link quality based on relative velocity, P-OLSR allows proactive adaptation of routing decisions, making it more suitable for dynamic scenarios. However, these approaches mainly rely on mobility trends or historical transmission statistics and are less sensitive to instantaneous variations. In contrast, jointly exploiting instantaneous distance and historical data opens up new possibilities for link-quality estimation in highly dynamic scenarios.

2.2. Routing mechanism

Parallel to improvements in link quality estimation, one major line of research has focused on refining the MPR selection mechanism. In the original OLSR design, the selection strategy relied solely on two-hop neighborhood coverage, without considering dynamic factors such as link stability, node mobility, or resource constraints. To address these limitations, researchers have introduced additional metrics to guide MPR selection while still operating within the two-hop neighborhood framework. For instance, Gangopadhyay et al. [32] incorporated residual energy and energy efficiency into the selection process, thereby improving routing performance in dynamic scenarios. Huang et al. [33] introduced relative position information obtained through GPS and optimized path selection using link duration, which enhanced packet delivery performance. In another approach, Zhang et al. [34] proposed a heuristic algorithm to reduce the number of MPR nodes in the network, aiming to lower redundancy in data dissemination. Despite these improvements, the majority of MPR-related studies continue to rely on two-hop neighbor information, reflecting the inherent design constraints of the mechanism. Following this line, our study also restricts MPR optimization to the two-hop neighborhood.

Another line of research has focused on routing maintenance optimization. Zhou et al. [35] proposed an adaptive algorithm that dynamically adjusts the flooding frequency of topology control messages, while leveraging multiple antennas to sense signal quality for timely routing table updates. Building on the concept of link duration, Hong et al. [36] further introduced a dynamic adjustment of link state maintenance frequency, which helped reduce routing delay and improve the efficiency of channel resource allocation.

2.3. UWB communication

UWB offers a nanosecond-scale temporal resolution and very large occupied bandwidth, enabling centimeter-level time-of-flight ranging and robust operation in rich multi path environments [37]. These properties, together with low transmit power and short pulses, yield accurate, low-latency distance estimates [38] that are comparatively resilient to interference [37]. Standards such as IEEE 802.15.4a [39] and its enhancement 802.15.4z [40,41] explicitly support two-way and secure ranging while allowing data communication in the same physical layer, i.e., ranging and communication can proceed concurrently [42]. Collectively, these capabilities suggest a new design avenue for dynamic, dense multi-agent/robot networks: incorporate real-time UWB ranging into control messaging to reduce link-estimation latency and improve end-to-end reliability under rapid topology changes.

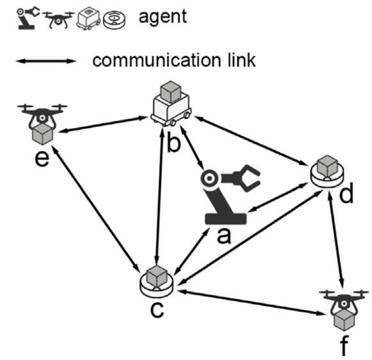


Fig. 2. Redundant transmissions caused by flooding in a simple network topology. When node a broadcasts a packet, its neighbors b, c, and d may all rebroadcast it, leading nodes e and f to receive duplicate copies.

3. System model and problem formulation

3.1. Background

Any ad hoc networking protocol must address the following fundamental question: *how should a packet be forwarded to reach a given destination?* The common approach is to construct a routing table, which specifies the next hop for each possible destination. Such a routing table is typically derived from global topological information, usually computed using a shortest-path routing algorithm.

This leads to a further question: *how can a node obtain global topological information?* A straightforward method is for each node to flood its local neighborhood information throughout the network. However, it often results in excessive redundant transmissions. For example, in Fig. 2, when agent a broadcasts a packet, its neighbors b, c, and d may all rebroadcast the same packet. As a result, nodes e and f receive duplicate copies. While such redundancy can enhance the QoS at the receivers, it also significantly increases overall network forwarding overhead. We therefore aim to address the following key research questions:

- **Multi-point relay selection**. The objective is to select an appropriate subset of one-hop neighbors as forwarding nodes to minimize data packet redundancy while ensuring the guaranteed QoS.
- **Link quality estimation**. To achieve reliable relay selection, as well as build real-time routing table, it is essential to estimate link quality in real time, particularly in terms of packet loss rate.
- **Implementation for resource-constrained agents**. To reduce resource consumption, we investigate the design of an integrated message format that simultaneously supports both swarm ranging and ad hoc routing.

These key research questions fall within the scope of the link layer and the network layer, as illustrated in Fig. 3.

3.2. System model

To reduce transmission redundancy, OLSR introduces the MPR mechanism, which selects a subset of nodes to forward packets. Only the selected one-hop neighbors participate in packet forwarding. However, the classic OLSR [23] does not consider link quality, and therefore cannot guarantee satisfactory Quality of Service (QoS)—in particular, the PDR performance. Although the enhanced OLSRv2 [43] introduces link-quality-related metrics, it incurs a significant delay in obtaining them. Consequently, our work focuses on the broadcast problem of balancing redundancy and PDR in real time.

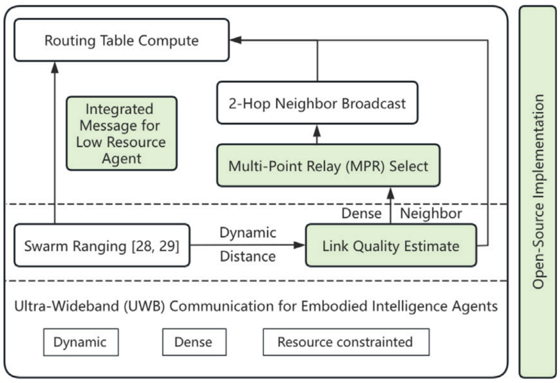


Fig. 3. Base on physical-layer UWB technology and link-layer swarm ranging, our work optimizes link quality estimation and MPR selection, implements and validates the proposed protocol on resource-constrained hardware, and releases the implementation as open source.

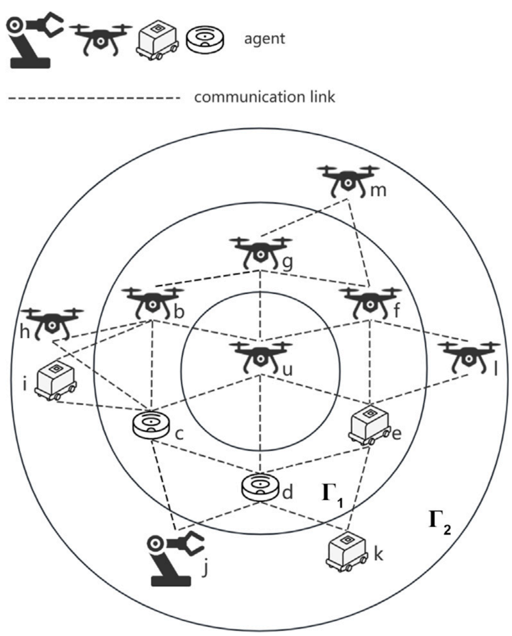


Fig. 4. Illustration of two-hop neighbor PLR calculation in a simple swarm topology. Dashed lines indicate the wireless links, meaning that the connected nodes are a pair of one-hop neighbors.

In this paper, we assume a swarm of embodied agents form a communication network, which can be represented by an undirected graph $G(V,E)$, where V denotes the set of agents and E denotes the set of communication links. Two agents can communicate if and only if their distance is less than R_c, *i.e.,* edge $(u,v)\in E$ if and only if $d_{u,v} < R_c$. Since each agent executes the broadcast process independently, without loss of generality, we assume that the current source agent is u. Let $\Gamma_1 = \{w|(u,w)\in E\}$ denote the set of one-hop neighbors of u, and $\Gamma_2 = \{v|(v,w)\in E, w\in \Gamma_1, v\notin \Gamma_1, v\neq u\}$ denote the set of two-hop neighbors. For instance, Fig. 4 illustrates a simple topology in which the communication links between nodes are represented by double-headed arrows. For node u, $\Gamma_1=\{b,c,d,e,f,g\}$ and $\Gamma_2=\{h,i,j,k,l,m\}$. Inspired by the MPR mechanism in OLSR, node u selects a subset of its one-hop neighbors, denoted as S, to forward the broadcast message. We aim to reduce the transmission redundancy through reduce the number of selected forwarders, *i.e.,* $|S|$.

Although reducing the number of forwarders helps decrease redundancy, it often also leads to reduced QoS, *e.g.,* PDR, from u to its two-hop neighbors. Therefore, we first evaluate the probability that a two-hop neighbor $v\in\Gamma_2$ fails to receive a packet from u, under the given set of forwarders $S\subseteq\Gamma_1$.

Let $P_{u,v}\in[0,1]$ denote the Packet Loss Rate (PLR) on the direct link from node u to node v, and let $Q_{u,v}\in[0,1]$ be the end-to-end PLR that node v fails to receive a packet from node u over two or more hops, $u\neq v$. And the PLR satisfies $PLR+PDR=1$. *A separate section is presented to estimate link quality $P_{u,v}$ in Section 4.*

Assume that failures of different links are independent, which is generally considered acceptable in network models [44–46]. The loss on a single path occurs if any link along it fails. Thus, the PLR on a single two-hop path $u\to w\to v$ is given by $P_{u,w}+(1-P_{u,w})P_{w,v}$, where $w\in\Gamma_1, v\in\Gamma_2$. For a pair of nodes u and $v\in\Gamma_2$, there may exist multiple two-hop paths from u to v. And the packet loss occurs only if all these paths fail. Take Fig. 4 as an example. A packet sent from node u can reach node h via two paths: $u\to b\to h$ and $u\to c\to h$. Since these two paths share no common links, the PLRs along them are independent. Therefore, the overall PLR $Q_{u,h}$ is the product of the PLRs along each path,

$$Q_{u,h} = (P_{u,b}+(1-P_{u,b})P_{b,h})\cdot(P_{u,c}+(1-P_{u,c})P_{c,h}). \tag{1}$$

More generally, given the sender u and receiver $v\in\Gamma_2$, we consider any two paths $u\to w\to v$ and $u\to q\to v$ such that $w,q\in\Gamma_1$ and $w\neq q$. These two paths share no common links because of $(u,w)\neq(u,q)$ and $(w,v)\neq(q,v)$, and thus the PLRs along them are independent. Therefore, given the non-empty set of selected one-hop neighbors $S\subseteq\Gamma_1$, the overall PLR $Q_{u,v}$ can be computed as follows:

$$Q_{u,v} = \prod_{w\in S}\left(P_{u,w}+(1-P_{u,w})P_{w,v}\right), \quad \forall v\in\Gamma_2. \tag{2}$$

Since $Q_{u,v}\in[0,1]$, it follows that each product term in Eq. (2), *i.e.,* $(P_{u,w}+(1-P_{u,w})P_{w,v})$, also lies within $[0,1]$. Moreover, because of $S\subseteq\Gamma_1$, $Q^{min}_{u,v}$ serves as a lower bound of $Q_{u,v}$ when S equals to Γ_1, which can be computed as follows:

$$Q^{min}_{u,v} = \prod_{w\in\Gamma_1}\left(P_{u,w}+(1-P_{u,w})P_{w,v}\right), \quad \forall v\in\Gamma_2. \tag{3}$$

In addition, to ensure the desired QoS, we impose a lower bound on the PDR, which is equivalently expressed as an upper bound on $Q_{u,v}$, given by

$$Q_{u,v}\le\gamma Q^{min}_{u,v}, \quad \forall v\in\Gamma_2, \tag{4}$$

where $\gamma>1$ is a gain factor that controls the acceptable PLR degradation.

The main notations used throughout this paper are summarized in Table 1.

3.3. Problem formulation

Given a node u, with its one-hop neighbor set Γ_1 and two-hop neighbor set Γ_2, we aim to minimize the number of forwarding nodes while ensuring that the PLR for all two-hop neighbors remains below a specified threshold. The broadcast problem can be formulated as follows:

$$\mathcal{P}:\ \min_S |S|, \tag{5}$$

$$\text{s.t.}\ \emptyset\neq S\subseteq\Gamma_1, \tag{5a}$$

$$\prod_{w\in S}\Big(P_{u,w}+(1-P_{u,w})P_{w,v}\Big) \le \min(\gamma Q^{\min}_{u,v},1), \quad \forall v\in\Gamma_2, \tag{5b}$$

where the Constraint (5a) specifies that the selected forwarding nodes must be one-hop neighbors of u, and the Constraint (5b) ensures that the selected S achieves a two-hop PLR no greater than $\min(\gamma Q^{min}_{u,v},1)$.

To simplify the multiplicative Constraint (5b), a logarithmic transformation is applied to convert it into an additive form:

$$\sum_{w\in S}\log_\rho\left(P_{u,w}+(1-P_{u,w})P_{w,v}\right) \ge \log_\rho\min(\gamma Q^{min}_{u,v},1), \quad \forall v\in\Gamma_2, \tag{6}$$

Table 1
Main notations and definitions.

Notation	Definition
G	Undirected graph, formed by agent nodes and communication links between them
V	Set of agent nodes
E	Set of communication links
Γ_1, Γ_2	Set of one-hop and two-hop neighbors of a node, respectively
n	$= \|\Gamma_1\|$, number of one-hop neighbors
k	$= \|\Gamma_2\|$, number of two-hop neighbors
$Q_{u,v}$	End-to-end PLR from node u to node v
$P_{u,v}$	PLR on the direct link from node u to node v
γ	Gain factor for PLR upper bound
$\mathbf{t}$	$= [t_1, t_2, \ldots, t_k]^T$, target vector
$\mathbf{C}$	Contribution matrix
$\mathbf{c}_j$	jth column vector of $\mathbf{C}$
τ	Time slot index
$P_{u,v}^{\tau}$	Observed PLR on the direct data link from node u to node v at time τ
$\hat{P}_{u,v}^{\tau}$	Estimated PLR on direct the data link from node u to node v at time τ
$\overline{P}_{u,v}(d)$	Predicted PLR on the direct data link from node u to node v at distance d
$\widetilde{P}_{u,v}^{\tau}(d)$	Corrected predicted PLR on the direct data link from node u to node v at time τ
$d_{u,v}$	Distance between node u and node v
α	Weighting coefficient for current observation
β	Correction coefficient

where $\rho \in (0,1)$ is a predefined base. This transformation maps the multiplicative terms into the additive logarithmic domain, so that $\log_\rho \min(\gamma Q_{u,v}^{min}, 1)$ serves as the log-scale lower bound for the target two-hop PLR. Let $k = |\Gamma_2|$ be the number of two-hop neighbors of u, and $\Gamma_2(i)$ be the ith two-hop neighbor in Γ_2. Similarly, let $n = |\Gamma_1|$, and $\Gamma_1(j)$ be the jth one-hop neighbor in Γ_1. The target vector $\mathbf{t} = [t_1, t_2, \ldots, t_k]^T$ is constructed to represent the constraint in Eq. (6), where $t_i = \log_\rho \min(\gamma Q_{u,\Gamma_2(i)}^{max}, 1), 1 \le i \le k$. Moreover, in Eq. (6), the left-hand side of the inequality satisfies the lower bound constraint incrementally.

To this end, we define the matrix $\mathbf{C} \in \mathbb{R}^{n \times k}$ to represent the incremental contribution of each one-hop neighbor, and define $\mathbf{c}_j$ as the jth column vector of $\mathbf{C}$. Specifically, $C_{i,j} = \log_\rho(P_{u,\Gamma_1(j)} + (1 - P_{u,\Gamma_1(j)}) P_{w,\Gamma_1(i)})$ represents the incremental value provided by one-hop neighbor $\Gamma_1(j)$ to two-hop neighbor $\Gamma_2(i)$. Accordingly, the Eq. (6) can be rewritten as:

$$\sum_{\Gamma_1(j)\in S} C_{i,j} \ge t_i, \quad \forall i \in [1,k]. \tag{7}$$

Here, we can reformulate the original problem $\mathcal{P}$ as $\mathcal{P}'$.

$$\mathcal{P}' : \min_S |S|, \tag{8}$$

$$\text{s.t.} \quad \emptyset \ne S \subseteq \Gamma_1, \tag{8a}$$

$$\sum_{\Gamma_1(j)\in S} \mathbf{c}_j \ge \mathbf{t}. \tag{8b}$$

By definition, column $\mathbf{c}_j$ represents the contributions of $\Gamma_1(j)$ to all two-hop neighbors, while the ith row of $\mathbf{C}$ specifies the contributions of all one-hop neighbors to $\Gamma_2(i)$. Under this transformation, the original problem is equivalent to selecting a minimum number of columns of $\mathbf{C}$ such that their element-wise sum meets or exceeds the given target vector $\mathbf{t}$.

The key techniques proposed for enhancing routing in dynamic and dense networks through UWB swarm ranging for embodied intelligence are presented in Fig. 5.

4. Data link quality estimation

In embodied intelligence agent networking, the swarm of agents is typically dynamic and dense, leading to highly time-varying wireless communication links between agents. Under such conditions, accurately and promptly estimating intra-swarm link quality is crucial. It enables the routing protocol to maintain an up-to-date network view, select multi-point relays and stable routes, reduce packet loss and latency, and ultimately improve the efficiency of collaborative swarm task execution.

Conventional methods mainly rely on large volumes of probe packets to statistically infer reception success rates. Although effective in static settings, these methods exhibit inherent limitations in highly dynamic swarm environments. The repeated transmission of probes not only introduces latency that hinders the capture of rapid link variations, but also increases communication overhead, which further reduces estimation accuracy and compromises routing reliability when link quality is already poor.

To overcome these limitations, we propose a real-time link quality estimation method that leverages relative distance information obtained during the swarm ranging process, jointly considering the temporal trends in distance variation and historical communication performance. This design enables timely and adaptive estimation without additional probing traffic, offering superior responsiveness and adaptability in resource-constrained and fast-changing swarm scenarios.

To improve the real-time performance of link quality estimation, practical implementations often employ an exponentially weighted moving average (EWMA) that combines historical packet loss data with the most recent observations to estimate the PLR of the data link, as expressed in the following equation,

$$\hat{P}_{u,v}^{\tau} = \alpha P_{u,v}^{\tau} + (1-\alpha)\hat{P}_{u,v}^{\tau-1}, 0 \le \alpha \le 1, \tag{9}$$

where $P_{u,v}^{\tau}$ and $\hat{P}_{u,v}^{\tau}$ denote the observed and estimated PLR on the data link from node u to neighboring node v at time τ, respectively, and α is a weighting coefficient that determines the relative importance assigned to the current observation.

Although EWMA can partially enhance the timeliness of link quality estimation, the observation of PLR still depends on multiple transmissions and receptions of probe packets. When the link quality degrades, resulting in intermittent or continuous loss of probe packets, the local estimation of link quality becomes inaccurate due to the lack of timely updates. To address this issue, our work incorporates relative distance information between nodes into the link quality estimation process. By leveraging the real-time and high-precision distance measurements obtained from swarm ranging, the accuracy and responsiveness of intra-swarm link quality estimation can be significantly improved.

Specifically, under line-of-sight (LoS) conditions with reliable communication quality, we empirically measure the PLRs between nodes at varying distances using UWB communication. Fig. 6 shows the relationship between communication distance and PLR, where the data points represent the average PLR sampled at different communication distances in real-world measurements. When the distance is less than 4 m, the observed PLR values remain close to 0 and nearly constant. Beyond 5 m, however, the PLR fluctuates and increases exponentially, eventually approaching one at approximately 8 m. The relationship between communication distance and PLR is modeled by Eq. (10), enabling real-time prediction of the current link PLR based on relative distance information:

$$\overline{P}_{u,v}(d) = \frac{1}{e^{ad+b} + c}, \tag{10}$$

where $\overline{P}_{u,v}(d)$ denotes the predicted PLR between any pair of one-hop neighbor nodes u and v at their relative distance d. The model is fitted from the aforementioned measurements, with $a = -1.61$, $b = 9.97$, and $c = 1.01$ as fitting parameters.

Considering that swarm networks are often deployed in diverse environments, and that complex scenarios such as urban areas may involve non-line-of-sight (Non-LoS), multipath, and other interference conditions, a purely distance-based model is insufficient to accurately capture instantaneous link quality. Therefore, we introduce a correction mechanism that leverages real-time packet loss observations to improve the accuracy of the predicted PLR by compensating for unmodeled

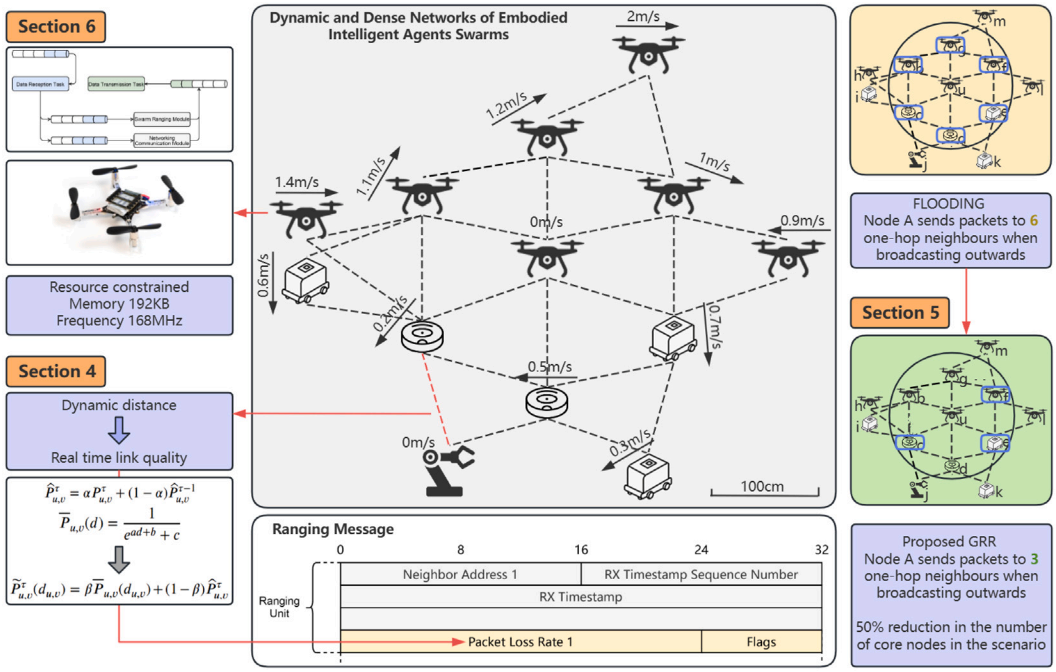


Fig. 5. In dynamic and dense networks (e.g., UAV swarms) where agent positions change rapidly, we estimate link quality based on inter-agent distance and embeds it into ranging messages, from our previously proposed Swarm Ranging Protocol [27,28], thereby reducing overall packet overhead. To reduce network forwarding overhead in dense agent communication, our proposed GRR algorithm (Section 5) minimizes the number of forward nodes while maintaining QoS. Furthermore, the system is implemented on resource-constrained hardware operating at 168 MHz with 192 KB memory, demonstrating its lightweight design.

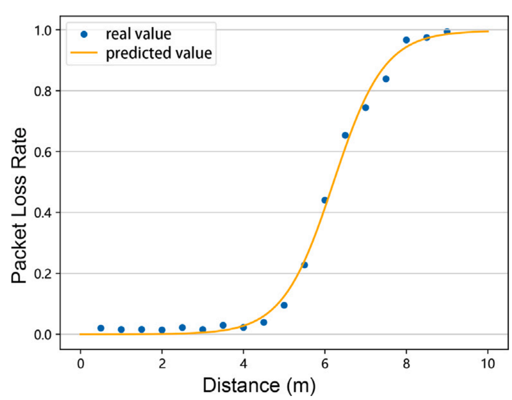


Fig. 6. PLR prediction model based on communication distance. The data points represent measured average PLRs at different distances under LoS conditions using UWB communication. The curve shows the fitted model, where PLR remains near zero below 4 m, increases sharply beyond 5 m, and approaches 1 around 8 m.

impairments. Specifically, the distance-based PLR prediction in Eq. (10) serves as a lightweight trend estimator, while the final PLR used by the protocol is obtained by fusing this prediction with the current packet loss estimation, as formulated in Eq. (11),

$$\widetilde{P}^{\tau}_{u,v}(d_{u,v}) = \beta\overline{P}_{u,v}(d_{u,v}) + (1-\beta)\widehat{P}^{\tau}_{u,v}, \quad 0 \le \beta \le 1, \tag{11}$$

where β is a correction coefficient that quantifies the relative confidence in the predictive model compared to the EWMA-based estimation. This hybrid design enables the prediction model to retain low computational complexity while maintaining robustness and accuracy in dynamic and dense environments.

By combining our proposed prediction model, *i.e.,* Eq. (11), with high-precision relative distance measurements obtained through swarm ranging, the PLR can be predicted in real time. In deployment environments with severe Non-LoS propagation, interference, or multipath effects, the LoS-fitted distance model in Eq. (10) can be recalibrated or replaced by an environment-specific PLR estimator. This substitution does not affect the later use of PLR, which only requires an estimated value for each link.

5. Greedy round-robin selection algorithm and approximation ratio analysis

In this section, we design a greedy round-robin (GRR) selection algorithm to address the broadcast problem $\mathcal{P}'$, and theoretically derive its approximation ratio. In addition, we employ an optimal selection algorithm based on depth-first search to obtain the optimal solution to $\mathcal{P}'$.

5.1. Greedy round-robin selection algorithm

First, we consider a simple case where the target vector is one-dimensional, *i.e.,* there is only one two-hop neighbor, $k = |\Gamma_2| = 1$. Then the solution can be simply obtained by selecting the columns of $\mathbf{C}$ in descending order and accumulating their values until the sum reaches or exceeds the target value t_1. However, in practice, there are often multiple two-hop neighbors. The contribution of a single one-hop neighbor can vary significantly across different two-hop neighbors, making it challenging to establish a unified criterion for ranking all one-hop neighbors. To address this challenge, we propose a GRR selection algorithm that aims to minimize the number of selected one-hop neighbors while ensuring that the target vector is satisfied. The core idea of that iteratively selects the most contributive one-hop neighbor for each two-hop neighbor in a round-robin manner until the target vector is satisfied.

The pseudocode of the GRR selection algorithm is shown in Algorithm 1. S is initially empty. To facilitate the selection of the most

Algorithm 1: *Greedy Round-Robin Selection Algorithm*

Input : contribution matrix $\mathbf{C}$, target vector $\mathbf{t}$
Output: forwarding node set S

```
S = ∅;
for i = 1, 2, …, k do
    Sort the i-th row of rank matrix R such that
      C_{i,R_{i,1}} ≥ C_{i,R_{i,2}} ≥ ⋯ ≥ C_{i,R_{i,n}};
end
for j = 1, 2, …, n do
    for i = 1, 2, …, k do
        if Γ_1(R_{i,j}) ∉ S then S = S ∪ {Γ_1(R_{i,j})} ;
        if Constraint (8b) is satisfied then return S ;
    end
end
return S;
```

contributive one-hop neighbor in each iteration, we introduce an additional rank matrix $\mathbf{R}$ to represent the relative contribution order of each row in $\mathbf{C}$. Specifically, each row satisfies

$$C_{i,R_{i,1}} \geq C_{i,R_{i,2}} \geq \cdots \geq C_{i,R_{i,n}}, \quad \forall i \in [1,k], \tag{12}$$

where the indices in $\mathbf{R}$ indicate the columns of $\mathbf{C}$ sorted in descending order. In each iteration, for each row of $\mathbf{C}$, the node with the largest value $C_{i,R_{i,j}}$ is selected in turn if it has not been selected before; otherwise, it is skipped. The algorithm terminates when the Constraint (8b) is satisfied and then returns S.

5.2. Approximation ratio analysis

Let S be the solution obtained by GRR and S^* be the optimal solution. Theorem 1 provides a problem-specific worst-case approximation bound for GRR, which depends on the number of two-hop neighbors k.

Theorem 1. *For problem $\mathcal{P}'$, let S be the solution obtained by the GRR selection algorithm and S^* be the optimal solution. The GRR selection algorithm achieves an approximation ratio of k, i.e., $\frac{|S|}{|S^*|} \leq k$.*

Proof. When each contribution $C_{i,j} \in \{0,1\}$ and $t_i = 1$, $\mathcal{P}'$ reduces to the classical set cover problem. Here, $C_{i,j}$ is instead a continuous link-quality-dependent contribution, so the classical set-cover proof based on binary coverage cannot be directly applied. Let S_i denote the set of forwarding nodes selected by Algorithm 1 on the ith row of $\mathbf{C}$. It is clear that $\cup_{i=1}^{k} S_i = S$ and $S_i \cap S_{i'} = \emptyset, i \neq i'$. Define l_i as the smallest integer satisfying that $\sum_{j=1}^{l_i} C_{i,R_{i,j}} \geq t_i, \forall i \in [1,k]$. Let l^{max} be the maximum among all l_i, *i.e.*, $l^{max} = \max(l_1, l_2, \ldots, l_k)$.

We prove that l^{max} is a lower bound of $|S^*|$ in the following. Since S^* is a feasible solution, we have $\sum_{\Gamma_1(R_{i,j}) \in S^*} C_{i,R_{i,j}} \geq t_i, \forall i \in [1,k]$. Now, we consider the relaxed problem only for a single row i, making it simpler.

$$\mathcal{P}'_i: \min_{J} |J|, \tag{13}$$

$$\text{s.t.} \ \emptyset \neq J \subseteq \{1,2,\ldots,n\}, \tag{13a}$$

$$\sum_{j \in J} C_{i,R_{i,j}} \geq t_i. \tag{13b}$$

We claim that the optimal solution value is l_i, and $J = \{1,2,\ldots,l_i\}$ is an optimal solution. This claim is proved by contradiction. Suppose on the contrary, that $J = \{1,2,\ldots,l_i\}$ is not optimal. Then, there must exist at least one element $x \in \{l_i+1, l_i+2, \ldots, n\}$ such that each x can replace one or more elements in $\{1,2,\ldots,l_i\}$. Let $Y \subseteq \{1,2,\ldots,l_i\}$ be the set of replaced elements and X be the set of replacements. We have $Y \cap X = \emptyset$ because of $Y \subseteq \{1,2,\ldots,l_i\}$ and $X \subseteq \{l_i+1, l_i+2, \ldots, n\}$. Then the size of new set $J - Y + X$ would be smaller, *i.e.*, $|J - Y + X| < l_i$, thereby $|Y| > |X|$. According to Eq. (12), we have $C_{i,R_{i,x}} \leq C_{i,R_{i,y}}, \forall x \in X, \forall y \in Y$. Select a subset $Z \subseteq Y$ such that $|Z| = |X|$, and thus we have

$$\sum_{j \in J-Y+X} C_{i,R_{i,j}} \leq \sum_{j \in J-Y+Z} C_{i,R_{i,j}}. \tag{14}$$

Because l_i is defined as the smallest integer such that $\sum_{j=1}^{l_i} C_{i,R_{i,j}} \geq t_i$, it follows that $\sum_{j=1}^{l_i-1} C_{i,R_{i,j}} < t_i$. Then we have,

$$\sum_{j \in J-Y+Z} C_{i,R_{i,j}} < \sum_{j=1}^{l_i-1} C_{i,R_{i,j}} < t_i. \tag{15}$$

Hence, according to Eqs. (14) and (15), it can be concluded that $\sum_{j \in J-Y+X} C_{i,R_{i,j}} < t_i$. Therefore, the new set $J - Y + X$ does not satisfy the Constraint (13b), which contradicts the assumption. Thus, $J = \{1,2,\ldots,l_i\}$ is optimal.

Since problem $\mathcal{P}'_i$ only considers one dimension of problem $\mathcal{P}'$, we have

$$|S^*| \geq |\{1,2,\ldots,l_i\}| = l_i, \quad \forall i \in [1,k]. \tag{16}$$

Eq. (16) holds for all i, so, we have $|S^*| \geq l^{max}$. According to the definition of S_i, we have $|S_i| \leq l_i, \forall i \in [1,k]$. Therefore, we have

$$|S| = \sum_{i=1}^{k} |S_i| \leq \sum_{i=1}^{k} l_i \leq k l^{max} \leq k|S^*|, \tag{17}$$

which is equivalent to $\frac{|S|}{|S^*|} \leq k$. □

The time complexity of the greedy algorithm can be analyzed as follows. First, each row of $\mathbf{R}$ must be sorted, which requires $O(kn \log n)$ time, where n and k are the numbers of one-hop and two-hop neighbors, respectively. Subsequently, the algorithm verifies whether each selected node is already contained in the set, which can be performed in $O(1)$ time using a hash table. This part therefore incurs a total cost of $O(kn)$. Combining both parts, the overall time complexity of the GRR selection algorithm is $O(kn \log n)$.

Beyond computational complexity, the number of two-hop neighbors k also has a direct impact on the selection performance of GRR. As k increases, which is common in dense UAV swarm networks, each one-hop neighbor tends to cover a smaller fraction of the two-hop neighborhood. This reduces the marginal contribution of each candidate node and makes the greedy selection less effective in approximating the optimal solution. Consequently, more forwarding nodes may be required to satisfy the QoS constraint, leading to a gradual degradation of the approximation performance of GRR as k grows.

6. Implementation on low resource agents

In this section, we employ a swarm of resource-constrained micro-UAVs to form a multi-agent system. By employing the proposed GRR selection algorithm as the implementation of the MPR mechanism, we developed an improved OLSR protocol. In conjunction with the swarm ranging protocol, this further enhanced the accuracy of link-state estimation. Then we implemented a prototype system on micro UAVs.

6.1. External communication

In our previous work [27,28], we developed a UWB swarm ranging protocol to accurately measure the relative distances among UAVs in dynamic and dense swarms. In this subsection, we extend the ranging unit of this protocol to carry additional information alongside the distance measurements, as illustrated in Fig. 7. Specifically, the ranging unit is extended with two additional fields: (i) the *Packet Loss Rate* field, which records the PLR of one-hop neighbors; (ii) and the binary *Flags* field, which indicates the status of the communication links. To enable the mutual notification of relay node selections for populating subsequent topology control (TC) messages, the protocol specifies that

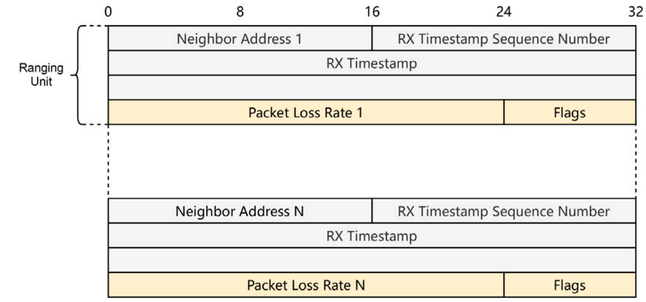


Fig. 7. Extended format of the ranging unit in the UWB swarm ranging protocol.

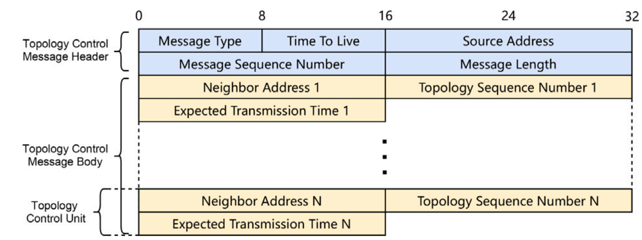


Fig. 8. Format of the topology control message.

when the *Flags* field is set to 1, it indicates that the sending node has chosen the corresponding neighbor as a relay node for forwarding TC messages.

Using the GRR selection algorithm, each node selects an appropriate subset of its one-hop neighbors to forward the topology information it maintains. This information is included in the TC message. As shown in Fig. 8, the TC message consists of two parts: the message header and the message body. The header includes metadata of TC message: *Message Type*, *Time to Live*, *Source Address*, *Message Sequence Number*, and *Message Length*. The message body is composed of multiple topology control units with identical structures. Each unit corresponds to a one-hop neighbor that has selected the message sender (*i.e.,* the current node) as its relay node. It includes the *Neighbor Address* of the corresponding neighbor, the *Topology Sequence Number* used to distinguish the freshness of the topology information, and the *Expected Transmission Time*.

6.2. Internal communication

In this system, all software functional modules of the UAV platform operate as tasks within the embedded real-time multitasking operating system, such as FreeRTOS. All internal system tasks are uniformly scheduled by the preemptive scheduler provided by RTOS. The internal communication module is primarily responsible for UWB data transmission, reception, and scheduling. Considering that both the swarm ranging module and network communication module depend on the internal communication module, this work adopts the Actor concurrency model. The UWB sends/receives operations from each module are transformed into read/write operations on blocking queues, thus avoiding tight coupling between modules, as illustrated in Fig. 9.

The internal communication module consists of a data receiving task and a data sending task. The DW3000 extension board designed and implemented in this system writes received UWB packets into a global receiving queue in a non-blocking manner. The data receiving task reads UWB packets from the global receiving queue using blocking reads and dispatches the packets to the corresponding receive queues of the appropriate functional modules based on the packet type. The data sending task, in turn, reads UWB packets written by functional modules

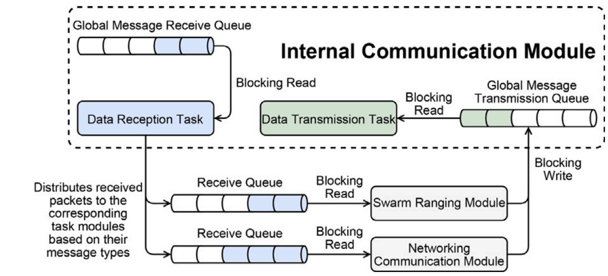


Fig. 9. Design of the internal communication module based on the Actor concurrency model.UWB send/receive operations from different modules are converted into read/write operations on blocking queues, enabling decoupled communication among tasks scheduled by FreeRTOS.

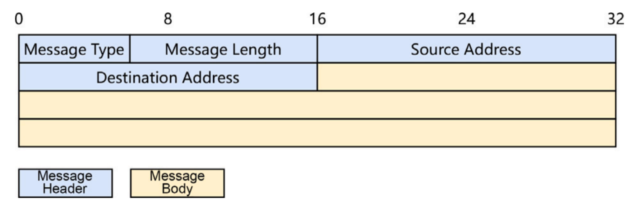


Fig. 10. Unified UWB packet format for the internal communication module.

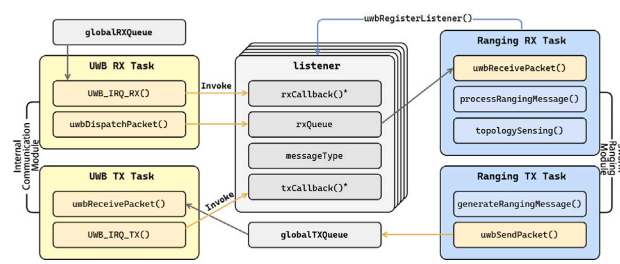


Fig. 11. Interaction between the swarm ranging module and the internal communication module. The ranging module registers a receive queue and callback functions for its message type, while the internal communication module relies on DW3000 interrupt interfaces to handle packet transmission and reception, triggering the corresponding callbacks to enable UWB-based interactions among UAV modules.

from the global sending queue via blocking reads and sends them one by one through the DW3000 board.

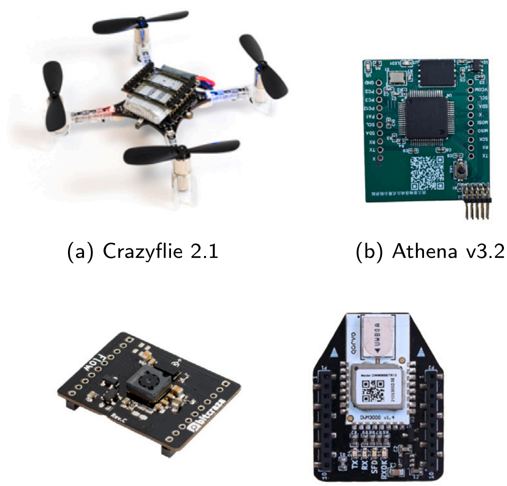
To implement this packet dispatching mechanism, a unified UWB packet format is defined for the internal communication module, as shown in Fig. 10. In this format, the *Message Type* field enables packet dispatching within the system; the *Message Length* field supports parsing of variable-length payloads; and the *Source Address* and *Destination Address* fields decouple the upper-layer MAC frame addresses from the internal UAV addressing scheme. The payload size can dynamically vary depending on the upper MAC frame format and the DW3000 configuration, with a maximum capacity of 1018 bytes in this system.

Based on the above design, this system defines a set of interaction interfaces for the internal communication module, as shown in Table 2. Any functional module in the system only needs to register a reception queue for its corresponding message type via the `Register` interface. Subsequently, it can send and receive messages using the `Send` and `Receive` interfaces. When communication is no longer required, the module can deregister its queue using the `Deregister` interface to free associated resources.

Taking the interaction between the ranging module and the internal communication module as an example, shown in Fig. 11, the ranging module registers a receive queue for its specific message type using the `uwbRegisterListener` interface. It also provides callback

Table 2
Interface definitions and descriptions of the internal communication module.

Interface	Definition
Register (`type`, `rxQueue`)	Registers `rxQueue` as the receive queue for messages of type `type`
Deregister (`type`)	Deregisters the receive queue associated with message type `type`
Send (`message`, `timeout`)	Sends the message `message` to the global send queue using blocking writes; `timeout` specifies the blocking duration
Receive (`type`, `timeout`)	Retrieves messages of type `type` from the corresponding receive queue using blocking reads; `timeout` specifies the blocking duration



(a) Crazyflie 2.1 (b) Athena v3.2

(c) Flow deck (d) DW3000 deck

Fig. 12. Hardware setup of the UAV platform.

functions for sending and receiving events related to ranging packets, allowing extension of ranging functionality. The internal communication module uses external interrupt interfaces `UWB_IRQ_TX` and `UWB_IRQ_RX` from the DW3000 board to transmit and receive packets and triggers the corresponding registered callback functions upon interrupts. In this way, UWB-based packet interactions between various functional modules within the micro-UAV swarm are realized.

6.3. Hardware implementation

The hardware composition of the UAV swarm used in this study mainly includes the micro-UAV platform and the monitoring platform. The micro-UAV platform is based on the Crazyflie 2.1 developed by Bitcraze and supports both officially provided and custom-developed extension decks. In this study, we integrate the Athena computing deck along with other extension decks to form a versatile hardware development platform. Developers can embed different types of extension decks into the Crazyflie 2.1 in a hardware-pin-compatible manner, allowing flexible extension and combination of functionalities. This highly customizable and open micro-UAV ecosystem is widely adopted in various applications for building UAV systems. The UAVs communicate with each other via UWB signals using the UWB extension decks, and each UAV is equipped with the ranging protocol described in Section 6.1 and the MPR selection algorithm presented in Section 5.1. The monitoring platform mainly consists of a computer and peripheral auxiliary equipment. The computer interfaces with the UAVs through USB serial ports or 2.4 GHz signals via auxiliary devices to obtain real-time UAV status, logs, and position information from external positioning systems.

The hardware devices used in the implementation and deployment of the proposed system were presented in Fig. 12, including the Crazyflie and its various extension decks.

Table 3
Configuration parameters of Crazyflie 2.1.

Hardware configuration	Parameter
Weight	27 g
Max Payload	15 g
Processor	STM32F405
Frequency	168 MHz
Memory	192 KB
Compatible Buses	UART/SPI/USB
Battery Capacity	250 mAh

Fig. 12(a) shows the main UAV Crazyflie 2.1 used in this paper, with body dimensions of $92 \times 92 \times 29$ mm, a net take-off weight of only 27 g, and a maximum payload of 15 g. It is powered by a 250 mAh battery. Crazyflie 2.1 is equipped with an STM32F405 microprocessor as its core computing unit, operating at 168 MHz with 192 KBytes of SRAM and 1 MB of Flash. Additionally, it includes an nRF51822 microprocessor as the communication core, operating at 32 MHz with 16 KBytes of SRAM and 128 KBytes of Flash, mainly used for communication with the external base station via 2.4 GHz signals. Detailed configurations of Crazyflie 2.1 are presented in Table 3.

Fig. 12(b) shows the Athena v3.2 computing deck mounted on the Crazyflie 2.1 UAV, which consists of a computing core, a communication unit, and a storage unit. The computing core is a low-power computing unit using the STM32L496RGT6 microcontroller, operating at 80 MHz with 320 KBytes of SRAM and 1 MB of Flash. It supports multiple bus protocols such as I2C, SPI, and UART, and consumes about 91 μA/MHz in run mode.

Fig. 12(c) shows the Flow Deck mounted at the bottom of the Crazyflie 2.1, equipped with a PMW3901 optical flow sensor that can perceive the distance to the ground and relative planar displacement, providing relative altitude and motion information to support base station flight control of the UAV.

Fig. 12(d) shows the DW3000 UWB deck mounted on Crazyflie 2.1 and driven by the Athena low-power computing core. The DW3000 deck integrates the DWM3000 UWB module produced by Decawave, which complies with the IEEE 802.15.4z standard. In this system, the Athena computing deck communicates with the DW3000 chip via SPI by using a self-implemented UWB driver to configure the chip and send/receive UWB signals through register read/write.

7. Experiments

7.1. Simulation settings and baseline algorithms

Extensive experiments were conducted to evaluate the performance of the proposed method. In the simulations, the wireless communication radius of agents is set to $R_c = 2$ m, and the gain factor for PLR upper bound is set to $\gamma = 1.2$. For Eq. (10), we set $a = -1.61$, $b = 9.97$, and $c = 1.01$, which are fitted from the measurements. The UAVs are uniformly distributed in the simulation area, resulting in diverse network topologies. Since the location information of each node is known, the distance between any two nodes can be easily calculated, and the link quality can then be estimated using the PLR model defined in Eq. (11).

The proposed GRR selection algorithm is compared with the following four algorithms:

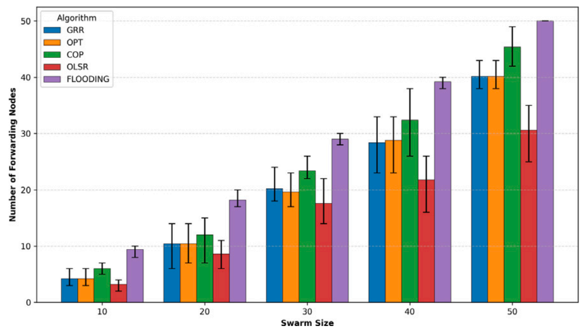


Fig. 13. Average number of forwarding nodes under different swarm sizes.

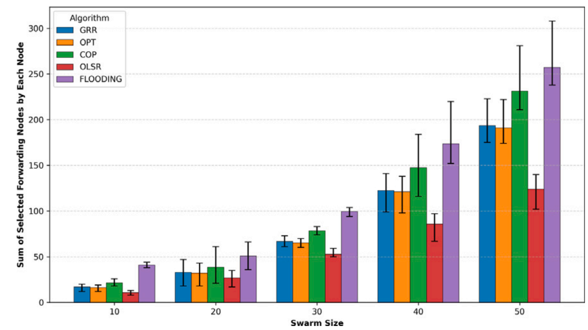


Fig. 14. Sum of the number of forwarding nodes selected by each node. If a node is selected by multiple neighbors, it is counted multiple times.

- **Depth-first search based selection algorithm**, referred to as DFS. For a given sender node, this algorithm explores all feasible combinations of one-hop relay node combinations and selects the one with the minimum size that satisfies Constraint (8b). The time complexity of this algorithm is $O(k2^n)$.
- **Link-quality-aware routing variant**, referred to as COP. This algorithm is implemented as a simplified version inspired by [47]. Specifically, it estimates link duration solely based on inter-node distance and selects forwarding nodes according to the predicted link duration.
- **MPR selection algorithm employed in OLSR protocol**, referred to as OLSR-MPR [23]. This algorithm selects an optimal forwarding set of one-hop neighbors with minimum size to cover all two-hop neighbors, without considering link quality. It is a greedy algorithm that aims to minimize the number of forwarding nodes.
- **Flooding broadcast algorithm**, referred to as FLOODING. This is a classical flooding algorithm where each node forwards the received message to all its one-hop neighbors, without any selection of forwarding nodes. This algorithm is used as a baseline for comparison.

7.2. Simulation results

We evaluate the performance of the algorithms based on the number of selected forwarding nodes, as this metric directly reflects the scale of UAV involvement in the packet forwarding process. Specifically, a node is counted if it is chosen by any neighbor as a forwarder. A smaller number of forwarding nodes indicates lower packet redundancy in the network.

For each specified swarm size, *i.e.,* the number of micro-UAVs, we randomly generated ten instances. Fig. 13 shows the average number of selected forwarding nodes across these instances among all nodes. As the swarm size increases from 10 to 50, FLOODING selects all nodes as forwarders, meaning every node except the source node itself rebroadcasts the packet. In contrast, OLSR-MPR consistently yields the smallest forwarding set because it greedily minimizes the number of forwarding nodes without considering the link quality. COP selects a number of forwarding nodes that lies between FLOODING and GRR, mainly because it adopts a simplified design and considers only the effect of distance on link quality. As a heuristic, GRR selects slightly more forwarders than DFS, which is consistent with expectations. It should be noted, however, that DFS only ensures minimality within each node's one-hop neighborhood, but it does not necessarily minimize the global forwarding count across the entire swarm. For example, as a swarm size of 40, GRR slightly outperforms DFS on average. Overall, both GRR and DFS, which account for link quality, select more nodes than OLSR-MPR in terms of the number of forwarding nodes. However, they still choose fewer forwarders than COP, another link-quality-aware method, and achieve a reduction in forwarding compared to FLOODING.

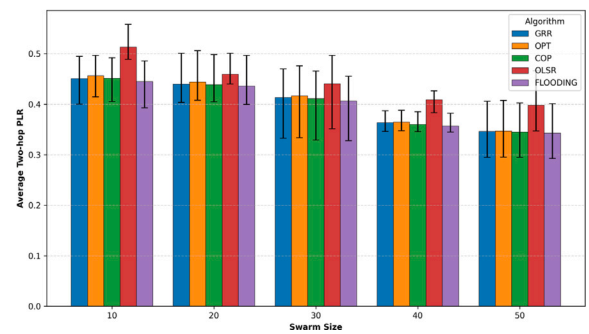


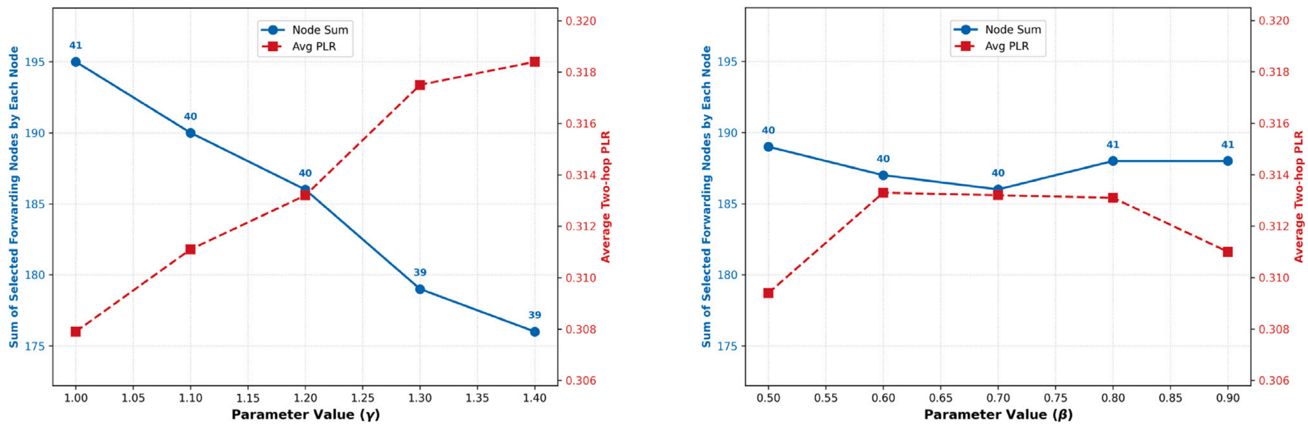
Fig. 15. Average two-hop PLR under different swarm sizes.

The first experiment measures the number of forwarding nodes selected by a single node on average. However, since each node independently selects forwarding nodes independently, a node may be selected by multiple neighbors. Therefore, we calculate the total number of forwarding selections across all nodes, where a node selected by multiple neighbors will be counted multiple times.

The average result over ten instances is presented in Fig. 14. At a swarm size of 50, the total number of selections for FLOODING is around 250, indicating that each node is selected by about five neighbors on average. In contrast, the corresponding values for DFS, COP, OLSR-MPR, and GRR are markedly lower. Among these, OLSR-MPR performs the best, followed by DFS and then GRR, while COP performs the worst. Unlike in the first experiment, DFS consistently outperforms GRR in this metric. This is because the total selections count directly reflects redundant selections, that is, how frequently nodes are selected, whereas the global forwarding count from the first experiment depends on interactions among local selections.

Although OLSR-MPR minimizes the number of forwarding nodes, it ignores link quality, leading to potential degradation in end-to-end reliability. To assess broadcast reliability, we therefore evaluate the average PLR experienced by two-hop neighbors under each algorithm. The results are presented in Fig. 15. As shown in the figure, FLOODING achieves the lowest PLR due to its maximal redundancy, whereas OLSR-MPR yields the highest PLR as a direct consequence of its disregard for link quality. Both GRR, DFS and COP, which incorporate link quality information into the selection process, improve delivery reliability over OLSR-MPR and achieve performance comparable to FLOODING. Moreover, the average PLR decreases with increasing swarm size, since a larger swarm provides more one-hop neighbors and alternative paths, increasing the probability of high-quality routes.

In summary, the proposed GRR selection algorithm selects a forwarding set whose size is close to that of the optimal set obtained by DFS, while reducing redundant transmissions compared with FLOODING and COP. By explicitly accounting for link quality, GRR also achieves markedly better broadcast reliability than the OLSR-MPR heuristic.



(a) Varying γ (b) Varying β

Fig. 16. Performance under different parameters. The number of forwarding nodes is annotated above each Node Sum data point, while average two-hop PLR is presented in dash lines.

7.3. Parameter sensitivity analysis

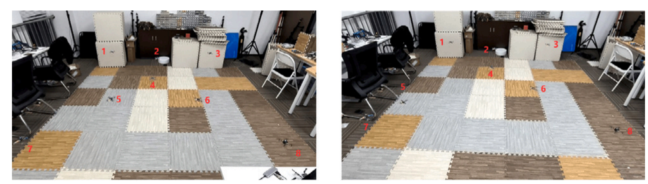
To evaluate the robustness of the proposed algorithm with respect to parameter variations, we conduct a sensitivity analysis under the same simulation settings as in the previous Section 7.2, with a swarm size of 50. Specifically, while keeping all other parameters fixed, we vary the gain factor γ and the correction coefficient β over five values each to examine their impact on performance. The parameter γ ranges from 1.0 to 1.4 with a step size of 0.1, while β ranges from 0.5 to 0.9 with the same step size. The results are shown in Fig. 16. The number of forwarding nodes is annotated above each Node Sum data point, while the other two metrics are represented using different colors.

As shown in Fig. 16(a), with the increase of the gain factor γ, both the number of forwarding nodes and the sum of selected forwarding nodes by each node decrease, while the average two-hop PLR increases. This is because γ controls the lower bound of acceptable PLR in the greedy forwarding node selection process. A larger γ allows a higher PLR threshold, enabling some previously selected forwarding nodes to be excluded, which leads to fewer forwarding nodes but higher PLR—consistent with the observed results.

In contrast, Fig. 16(b) shows the correction coefficient β does not exhibit a clear positive or negative correlation with the performance metrics, and its overall impact is relatively limited. This is because β determines the relative weighting between the distance-based predicted PLR and the EWMA-based estimated PLR, thereby affecting the accuracy of PLR estimation and, consequently, the algorithm's performance. The optimal value of β typically depends on the specific environment. For example, in our experimental setting, $\beta = 0.7$ yields the most accurate PLR estimation. Nevertheless, the results also show that even when β deviates from its optimal value, the performance degradation remains moderate, demonstrating the robustness of the proposed method.

7.4. Real world experiments

To evaluate the performance of the proposed algorithm under realistic conditions, we conducted real-world experiments with a swarm of micro-UAVs. In these experiments, each UAV performed random movements within a 1 m × 1 m planar grid, as well as collective formation maneuvers at speeds ranging from 0.2 m/s to 1.0 m/s, while maintaining a fixed altitude of 0.5 m. The experiments took place in a 6.5 m × 6.5 m × 3 m flight arena, as shown in Fig. 17. This configuration was designed to simulate network topology dynamics in collaborative swarm scenarios. Our protocol was based on the OLSR protocol, with the proposed GRR selection algorithm as the implementation of the MPR mechanism.



(a) Link probe interval = 200 ms (b) Link probe interval = 500 ms

Fig. 17. Experimental scenario for real-world validation. UAVs performed random 1 m × 1 m grid movements and formation maneuvers at speeds of 0.2 m/s to 1.0 m/s within a 6.5 m × 6.5 m × 3 m flight area.

We first evaluated the Round-Trip Time (RTT) of application-layer packets in a five-UAV swarm. Each UAV periodically sent 100-byte application-layer packets to randomly selected destinations at 50 ms intervals. The transmission and reception timestamps of these packets were recorded to compute the RTT between UAVs. The average RTT across the swarm is presented in Fig. 18. In comparison to AODV, whose routing performance degraded significantly as the swarm topology became more dynamic, our protocol demonstrated stronger adaptability to highly dynamic topological changes. On average, it reduced the RTT by approximately 54.7% compared to AODV, benefiting from the inherent advantages of proactive routing in maintaining up-to-date routes and reducing route discovery delays.

We also measured the PDR of the swarm. In this experiment, each UAV was configured to periodically transmit a 100-byte application-layer packet at 20 ms intervals to a randomly selected UAV (excluding itself) within the swarm. Each packet contained information on historical transmissions and receptions, allowing the PDR to be calculated as the ratio of successfully received packets to the total sent. The TC message interval was set to 500 ms for both our protocol and OLSR, while link probe message intervals were tested at 200 ms and 500 ms.

The results are presented in Fig. 19. For a fixed swarm size, the PDR generally decreased with increasing mobility. Conversely, at a given flight speed, shorter link probing intervals enabled nodes to detect local topology changes more rapidly, which improved the PDR. Notably, in Fig. 19(c), the network became heavily congested, making packet loss the dominant factor limiting delivery performance. Under these congested conditions, when the flying speed was 0.4 m/s, longer probe intervals yielded better performance by reducing probing overhead and alleviating channel contention. However, as the flying speed increased further and topology changes accelerated, timely topology awareness again became the primary determinant of routing performance.

Compared to OLSR, our protocol improves local topology awareness via a distance-based, real-time link-quality estimation model and

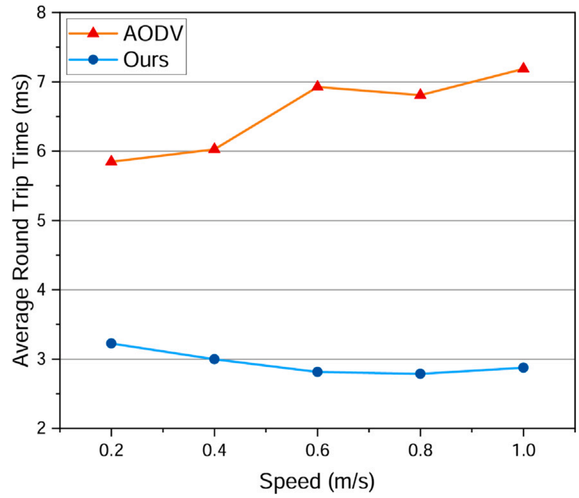


Fig. 18. Average RTT under different UAV mobility.

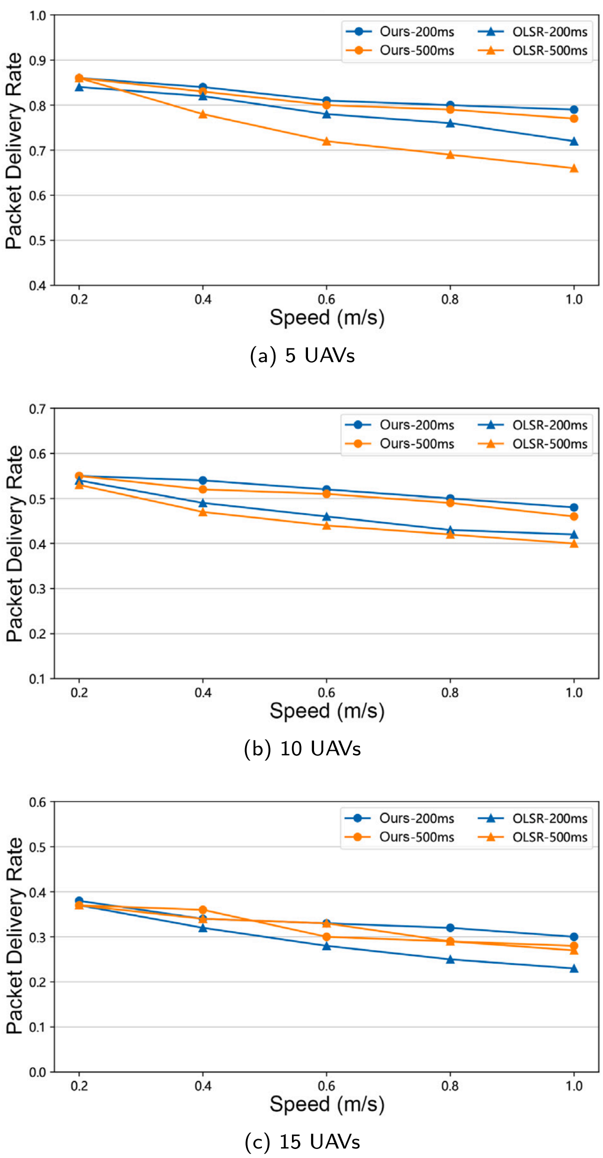


Fig. 19. PDR of the swarm under different UAV mobility and link probe intervals, comparing our protocol with typical OLSR.

reduces TC flooding overhead by leveraging link-quality information in forwarding decisions. As a result, under highly dynamic conditions, our protocol achieves an average PDR improvement of approximately 17.3% over OLSR, thereby providing more reliable routing paths within the swarm.

8. Conclusion

This paper investigates the problem of reliable communication in dynamic and dense embodied agents networks. The nature of dynamic network and high node density collectively cause rapid changes of network topology and increased packet collisions. To address these issues, we propose an MPR selection mechanism based on data link quality estimation and a greedy rotation algorithm, and analyze its approximation ratio with respect to the optimal solution. We further design a corresponding message format, implement a software prototype, and deploy it on micro-UAV hardware. Finally, both simulations and real-world UAV experiments are conducted to evaluate the proposed protocol, confirming its effectiveness and advantages in dynamic and dense networking scenarios.

CRediT authorship contribution statement

Fangyu Zhou: Writing – review & editing, Writing – original draft, Software, Data curation. **Yiqian Wang:** Writing – review & editing, Visualization. **Feng Shan:** Writing – review & editing, Project administration, Supervision. **Wenjia Wu:** Visualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Feng Shan reports was provided by National Natural Science Foundation of China. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China Grants 62132009, 62472090, 62232004, the National Natural Science Foundation of Jiangsu Grants BK20242026, Key Laboratory of Computer Network and Information Integration of the Ministry of Education of China Grant 93K-9, and the Jiangsu Provincial Key Laboratory of Network and Information Security Grant BM2003201.

Data availability

Data will be made available on request.

References

[1] A. Cangelosi, J. Bongard, M.H. Fischer, S. Nolfi, Embodied intelligence, in: Springer Handbook of Computational Intelligence, Springer, 2015, pp. 697–714.
[2] L. Ren, J. Dong, S. Liu, L. Zhang, L. Wang, Embodied intelligence toward future smart manufacturing in the era of AI foundation model, IEEE/ASME Trans. Mechatronics (2024).
[3] N. Roy, I. Posner, T. Barfoot, P. Beaudoin, Y. Bengio, J. Bohg, O. Brock, I. Depatie, D. Fox, D. Koditschek, et al., From machine learning to robotics: Challenges and opportunities for embodied intelligence, 2021, arXiv preprint arXiv:2110.15245.
[4] L. Smith, M. Gasser, The development of embodied cognition: Six lessons from babies, Artif. Life 11 (1–2) (2005) 13–29.
[5] D. Garikapati, S.S. Shetiya, Autonomous vehicles: Evolution of artificial intelligence and the current industry landscape, Big Data Cogn. Comput. 8 (4) (2024) 42.
[6] D. Parekh, N. Poddar, A. Rajpurkar, M. Chahal, N. Kumar, G.P. Joshi, W. Cho, A review on autonomous vehicles: Progress, methods and challenges, Electronics 11 (14) (2022) 2162.

[7] R. Bogue, Disaster relief, and search and rescue robots: the way forward, Ind. Robot.: Int. J. Robot. Res. Appl. 46 (2) (2019) 181–187.

[8] S. Pavaimalar, S. Ravi, N. Kumaran, S. Priya, et al., AI-driven autonomous robots for search and rescue operations in disaster zones, in: 2025 International Conference on Data Science, Agents & Artificial Intelligence, ICDSAAI, IEEE, 2025, pp. 1–6.

[9] F.A. Almalki, S.M. Salem, W.M. Fawzi, N.S. Alfeteis, S.A. Esaifan, A.S. Alharthi, R.Z. Alnefaiey, Q.H. Naith, Coupling an autonomous UAV with a ML framework for sustainable environmental monitoring and remote sensing, Int. J. Aerosp. Eng. 2024 (1) (2024) 4285475.

[10] V. Ninkovic, D. Vukobratovic, D. Miskovic, UAV-assisted distributed learning for environmental monitoring in rural environments, 2024, arXiv preprint arXiv:2407.02693.

[11] N. Ya'acob, M. Zolkapli, J. Johari, A.L. Yusof, S.S. Sarnin, A.Z. Asmadinar, UAV environment monitoring system, in: 2017 International Conference on Electrical, Electronics and System Engineering, ICEESE, IEEE, 2017, pp. 105–109.

[12] G. Li, X. Chen, F. Zhou, Y. Liang, Y. Xiao, X. Cao, Z. Zhang, M. Zhang, B. Wu, S. Yin, et al., Self-powered soft robot in the mariana trench, Nature 591 (7848) (2021) 66–71.

[13] O.O. Elijah, A.O. Barnabas, S. Ramamurthy, H. Alowaidi, Hybrid swarm robotics for autonomous deep-sea exploration and biodiversity mapping in uncharted marine zones, Synth.: A Multidiscip. Res. J. 2 (4) (2024) 41–54.

[14] N.T. Jafferis, E.F. Helbling, M. Karpelson, R.J. Wood, Untethered flight of an insect-sized flapping-wing microscale aerial vehicle, Nature 570 (7762) (2019) 491–495.

[15] K. Zhang, P. Chermprayong, F. Xiao, D. Tzoumanikas, B. Dams, S. Kay, B.B. Kocer, A. Burns, L. Orr, T. Alhinai, et al., Aerial additive manufacturing with multiple autonomous robots, Nature 609 (7928) (2022) 709–717.

[16] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Müller, V. Koltun, D. Scaramuzza, Champion-level drone racing using deep reinforcement learning, Nature 620 (7976) (2023) 982–987.

[17] T.F. Nygaard, C.P. Martin, J. Torresen, K. Glette, D. Howard, Real-world embodied AI through a morphologically adaptive quadruped robot, Nat. Mach. Intell. 3 (5) (2021) 410–419.

[18] R. Baines, S.K. Patiballa, J. Booth, L. Ramirez, T. Sipple, A. Garcia, F. Fish, R. Kramer-Bottiglio, Multi-environment robotic transitions through adaptive morphogenesis, Nature 610 (7931) (2022) 283–289.

[19] D. Gadasin, A. Shvedov, I. Vakurin, Designing WI-FI wireless networks with high density of subscriber service, in: 2020 Systems of Signals Generating and Processing in the Field of on Board Communications, IEEE, 2020, pp. 1–4.

[20] T.Q. Duong, X. Chu, H.A. Suraweera, Ultra-dense Networks for 5G and beyond: Modelling, Analysis, and Applications, John Wiley & Sons, 2019.

[21] X. Ba, Y. Wang, D. Zhang, Y. Chen, Z. Liu, Effective scheduling scheme for multi-connectivity in intra-frequency 5G ultra-dense networks, in: 2018 IEEE International Conference on Communications Workshops (ICC Workshops), IEEE, 2018, pp. 1–5.

[22] C. Perkins, E. Belding-Royer, S. Das, Ad Hoc On-Demand Distance Vector (AODV) Routing, Tech. Rep, 2003.

[23] T. Clausen, P. Jacquet, Optimized link state routing protocol (OLSR), 2003, Standards Track, RFC 3626.

[24] D. Nguyen, P. Minet, Analysis of MPR selection in the OLSR protocol, AINAW'07, in: 21st International Conference on Advanced Information Networking and Applications Workshops, vol. 2, IEEE, 2007, pp. 887–892.

[25] A. Ouacha, N. Lakki, J. El Abbadi, A. Habbani, B. Bouamoud, M. Elkoutbi, Reliable MPR selection based on link lifetime-prediction method, in: 2013 10th IEEE International Conference on Networking, Sensing and Control, ICNSC, IEEE, 2013, pp. 11–16.

[26] S. Mahfoudh, P. Minet, EOLSR: an energy efficient routing protocol in wireless ad hoc and sensor networks, J. Interconnect. Netw. 9 (04) (2008) 389–408.

[27] F. Shan, J. Zeng, Z. Li, J. Luo, W. Wu, Ultra-wideband swarm ranging, in: IEEE INFOCOM 2021 - IEEE Conference on Computer Communications, 2021, pp. 1–10.

[28] F. Shan, H. Huo, J. Zeng, Z. Li, W. Wu, J. Luo, Ultra-wideband swarm ranging protocol for dynamic and dense networks, IEEE/ACM Trans. Netw. 30 (6) (2022) 2834–2848.

[29] olsrd Link Quality Extensions. http://www.olsr.org/docs/README-LinkQuality.html. [EB/OL].

[30] T. Clausen, C. Dearlove, P. Jacquet, U. Herberg, The Optimized Link State Routing Protocol Version 2, Tech. Rep., 2014.

[31] S. Rosati, K. Krużelecki, G. Heitz, D. Floreano, B. Rimoldi, Dynamic routing for flying ad hoc networks, IEEE Trans. Veh. Technol. 65 (3) (2015) 1690–1700.

[32] S. Gangopadhyay, V.K. Jain, A position-based modified OLSR routing protocol for flying ad hoc networks, IEEE Trans. Veh. Technol. (2023).

[33] Y. Huang, R. Xie, B. Gao, J. Wang, Dynamic routing in flying ad-hoc networks using link duration based MPR selection, in: 2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall), IEEE, 2020, pp. 1–5.

[34] D.-g. Zhang, T. Zhang, Y. Dong, X.-h. Liu, Y.-y. Cui, D.-x. Zhao, Novel optimized link state routing protocol based on quantum genetic strategy for mobile learning, J. Netw. Comput. Appl. 122 (2018) 37–49.

[35] C. Zhou, C. Dong, X. Zhu, L. Zhang, Multi-antenna based fast routing recovery protocol for dynamic UAV networks, in: 2022 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), IEEE, 2022, pp. 507–514.

[36] L. Hong, H. Guo, J. Liu, Y. Zhang, Toward swarm coordination: Topology-aware inter-UAV routing optimization, IEEE Trans. Veh. Technol. 69 (9) (2020) 10177–10187.

[37] Y. Shen, M.Z. Win, Fundamental limits of wideband localization—Part I: A general framework, IEEE Trans. Inform. Theory 56 (10) (2010) 4956–4980.

[38] S. Gezici, H.V. Poor, Position estimation via ultra-wide-band signals, Proc. IEEE 97 (2) (2009) 386–403.

[39] E. Karapistoli, F.-N. Pavlidou, I. Gragopoulos, I. Tsetsinas, An overview of the IEEE 802.15. 4a standard, IEEE Commun. Mag. 48 (1) (2010) 47–53.

[40] D. Coppens, A. Shahid, S. Lemey, B. Van Herbruggen, C. Marshall, E. De Poorter, An overview of UWB standards and organizations (IEEE 802.15. 4, fira, apple): Interoperability aspects and future research directions, IEEE Access 10 (2022) 70219–70241.

[41] M. Cheraghinia, A. Shahid, S. Luchie, G.-J. Gordebeke, O. Caytan, J. Fontaine, B. Van Herbruggen, S. Lemey, E. De Poorter, A comprehensive overview on UWB radar: Applications, standards, signal processing techniques, datasets, radio chips, trends and future research directions, IEEE Commun. Surv. Tutorials (2024).

[42] Z. Sahinoglu, S. Gezici, Ranging in the IEEE 802.15. 4a standard, in: 2006 IEEE Annual Wireless and Microwave Technology Conference, IEEE, 2006, pp. 1–5.

[43] T. Clausen, C. Dearlove, P. Jacquet, U. Herberg, The optimized link state routing protocol version 2 (OLSRv2), 2014, Standards Track, RFC 7181.

[44] S. Patterson, B. Bamieh, A. El Abbadi, Convergence rates of distributed average consensus with stochastic link failures, IEEE Trans. Autom. Control 55 (4) (2010) 880–892, http://dx.doi.org/10.1109/TAC.2010.2041998.

[45] S. Kandeepan, K. Wang, T. Samarasinghe, C. Divarathne, Spatial correlation in shadow fading for wireless communication, in: 2024 3rd International Conference on Artificial Intelligence for Internet of Things (AIIoT), 2024, pp. 1–5, http://dx.doi.org/10.1109/AIIoT58432.2024.10574778.

[46] Z. Wang, M. Hassan, T. Moors, A study of spatial packet loss correlation in 802.11 wireless networks, in: IEEE Local Computer Network Conference, 2010, pp. 568–575, http://dx.doi.org/10.1109/LCN.2010.5735774.

[47] S. Gangopadhyay, V.K. Jain, A position-based modified OLSR routing protocol for flying ad hoc networks, IEEE Trans. Veh. Technol. 72 (9) (2023) 12087–12098, http://dx.doi.org/10.1109/TVT.2023.3265704.