International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 7297-7307

https://doi.org/10.70517/ijhsa464632

Dynamic Routing Strategies for UAV Ad Hoc Network Communications in Emergency Housing Support

Jing Liu[1,*]
[1] Hunan Post and Telecommunication College, Changsha, Hunan, 410000, China
Corresponding authors: (e-mail: jing_hnyd@163.com).

Abstract Post-disaster spectrum resource shortages and user mobility issues are the main challenges to the effective application of drone communications in emergency housing. This paper establishes a UAV-based D2D communication network system model to simulate post-disaster emergency housing scenarios. The communication link instability caused by frequent user mobility in emergency housing scenarios is transformed into the minimum perturbation problem in graph theory, and an interference model is established. To address load imbalance and link congestion in post-disaster emergency housing scenarios, a DDPG deep reinforcement learning method is developed by combining DQN and DPG methods, and a dynamic routing strategy based on DDPG is designed. This strategy can dynamically and adaptively adjust the routing overhead of wireless links based on the network status in the emergency housing scenario, thereby ensuring stable communication in the emergency housing scenario. Under the DDPG-based dynamic routing strategy, the average jitter of the routing algorithm remains below 3.0 ms, and the node death time is delayed (138 s), demonstrating superior stability and energy balancing capabilities.

Index Terms emergency housing support, DDPG, drone communication, dynamic routing strategy

I. Introduction

Emergency housing assistance refers to housing measures provided by the government in the event of major natural disasters or other emergencies. As a new, efficient, foldable housing solution designed specifically for emergencies, emergency housing assistance demonstrates excellent response effectiveness in crisis situations [1]-[4]. Emergency housing assistance offers numerous advantages, such as rapid assembly, simple overall structure, and ease of transportation and relocation—features not found in conventional housing [5]-[7]. The emergence of emergency housing support enables us to quickly and efficiently construct housing in the face of natural disasters, such as earthquakes, or during outdoor construction projects, providing temporary work and living spaces for people [8]-[11]. However, during emergency housing support operations, challenges such as infrastructure damage caused by disasters and complex rescue environments result in low efficiency and ineffective outcomes for manual rescue efforts [12]-[14].

With the development of drone hardware and software technology, multi-drone swarm-based self-organizing drone ad hoc networks (FANETs) have garnered increasing attention from both academia and industry. Their flexible deployment and rapid response capabilities enable them to efficiently complete various tasks under disaster conditions [15]-[18]. UAV self-organizing network routing protocols are one of the most important methods for improving service quality (QoS). In emergency housing support, dynamic routing strategies for UAV self-organizing network communications utilize intelligent algorithms to optimize data transmission paths between UAVs in real time, addressing issues such as node movement and energy constraints, thereby ensuring stable communications in disaster-affected housing areas [19]-[22].

This paper first conducts mathematical modeling of post-disaster drone-to-ground user communication scenarios and constructs a drone-based D2D model network system. To address the issue of unstable communication links in the communication scenario, an interference graph is constructed to form an interference model. Next, the DDPG algorithm is introduced to design a solution method for the network model and interference model based on DDPG, thereby proposing a dynamic routing strategy based on DDPG. Furthermore, under different conditions of node quantity and node movement speed, this paper conducts simulation experiments comparing the average end-to-end latency, throughput, and packet loss rate between the proposed routing protocol and traditional routing protocols. Finally, by evaluating the average jitter performance and energy balancing capability of the designed dynamic routing algorithm, the feasibility and reliability of the proposed dynamic routing strategy are validated.

II. Communication Network System Model and Problem Statement

II. A.Network Model

This paper considers a communication model for emergency housing response scenarios (a drone-based D2D communication network system model), as shown in Figure 1. The figure depicts an emergency communication network composed of UAVs and ground users, including fixed-position tethered drone base stations and mobile sensing drones, disaster-affected individuals, and rescue personnel. The set of sensing UAVs is denoted by $U=\{u_1,u_2,\cdots,u_n\}$ and the ground user set is denoted by $D=\{d_1,d_2,\cdots,d_m\}$ respectively. Communication links exist between sensing UAVs, between sensing UAVs and assisting/rescue personnel, and between assisting personnel and rescue personnel. Each device node in the network is equipped with a wireless interface. Additionally, communication services between devices include data transmission, voice calls, and video transmission. In this scenario, due to damage to some base stations, spectrum resources face significant challenges, and it is assumed that all communication links reuse POCs. The technology used in this paper is based on partially overlapping channels of the IEEE 802.11b/g standard, which supports 2.4 GHz links and up to 11 available channels, where channels 1, 6, and 11 are mutually orthogonal and do not interfere with each other.

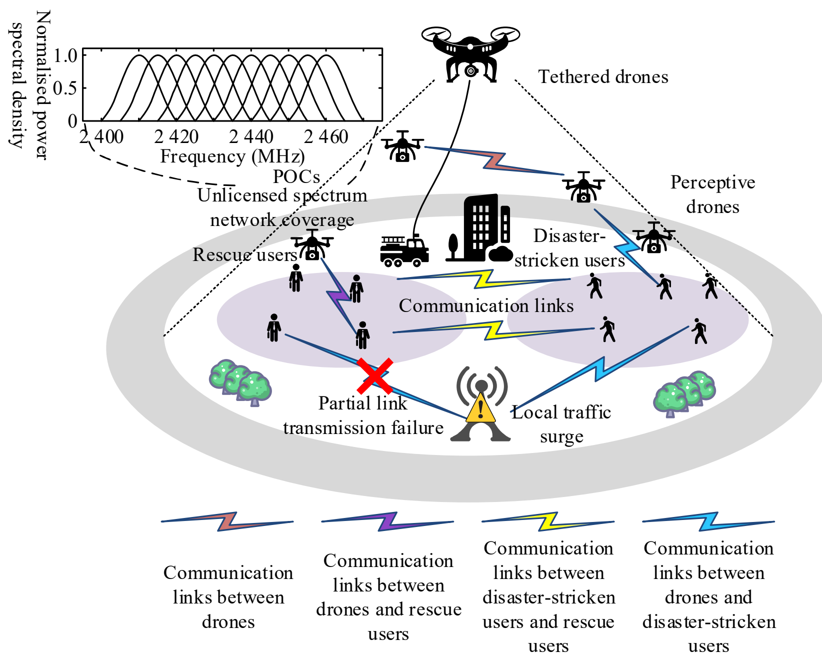

Figure 1: Unmanned aerial vehicle (UAV) joint D2D communication network model

II. B.Interference Model

Since the spectral overlap reduction of POCs decreases the occupied bandwidth, it is beneficial for improving spectral efficiency. However, it also introduces a new issue: if channel allocation is performed improperly, it may lead to more severe interference and cause network congestion. Therefore, effective allocation of POCs is one of the key challenges. Before allocation, the first consideration is how to characterize the interference of POCs in the network. To this end, this paper constructs an interference graph topology to characterize the interference of POCs in the network as follows.

Let p and q denote the channel numbers in POCs, and let $\delta(t)$ denote the absolute interval between channel p and channel q at time slot t , i.e., Equation (1):

$$\delta(t)=|p-q| \tag{1}$$

$IR(\delta(t))$ indicates the interference range when the channel p and the channel q are separated from each other at $\delta(t)$. Suppose p is the channel occupied by node i in the t time slot, and q is the channel allocated by node j in the t time slot. In addition, there is no communication link between node i and node j, and the distance between the two nodes in the t time slot is $d_{ij}(t)$, based on the interference range table, the interference edge can be established according to the channel spacing and distance between the two nodes, and

$I_{ij}(t)$ is used to represent whether there is interference between node i and node j in the t time slot, if node i and node j If the distance between them is less than the interference range $IR(\delta(t))$ when the channel interval is $\delta(t)$, then there is interference, otherwise there is no interference, which is expressed as Eq. (2):

$$I_{ij}\left(t\right)=\begin{cases}1, IR(\delta(t))\geq d_{ij}(t)\\ 0, IR(\delta(t))<d_{ij}(t)\end{cases} \tag{2}$$

III. Dynamic routing strategy based on DDPG

III. A. Routing Protocols

III. A. 1) Routing Concepts

Routing is the process of transmitting information from a source through a network to a destination, passing through at least one intermediate node during transmission. The main difference between routing and bridging is that bridging occurs at the second layer (link layer) of the OSI reference model, while routing occurs at the third layer (network layer). This distinction means that the two use different information during the transmission process, thereby accomplishing their tasks in different ways.

The concept of routing has been around in the computer industry for a long time, but it wasn't until the mid-1980s that it achieved commercial success. The main reason for this delay was that networks in the 1970s were relatively simple, and larger networks became more common later on.

III. A. 2) Routing Components

Routing involves two basic actions: determining the optimal path and transmitting information over the network. During the routing process, the latter is also referred to as (data) switching. Switching is relatively simple, while path selection is complex.

(1) Path Selection

A metric is a measurement standard used by routing algorithms to determine the optimal path to the destination, such as path length. To assist in path selection, routing algorithms initialize and maintain routing tables containing path information, which varies depending on the routing algorithm used. Routing algorithms use various pieces of information to populate the routing table. The destination/next-hop address pair informs the router that the best way to reach the destination address is to send the packet to the router representing the "next hop." When a router receives a packet, it checks its destination address and attempts to associate this address with its "next hop." The routing table may also include other information. Routing tables compare metrics to determine the best path, and these metrics vary depending on the routing algorithm used. Common metrics will be discussed below. Routers communicate with each other by exchanging routing information to maintain their routing tables. Routing update information typically includes all or part of the routing table. By analyzing routing update information from other routers, the router can construct a network topology map. Another example of information exchanged between routers is link-state broadcast information, which notifies other routers of the sender's link status. Link information is used to build a complete topology map, enabling routers to determine the optimal path.

(2) Switching

Switching algorithms are relatively simple and are the same for most routing protocols. In most cases, when a host decides to send data to another host, it obtains the router's address through certain methods and then sends a data packet to the physical address (MAC) of the router. The protocol address of the packet is directed to the destination host. The router examines the destination protocol address of the packet and determines whether it knows how to forward the packet. If the router does not know how to forward it, it typically discards the packet. If the router knows how to forward it, it changes the destination physical address to the next-hop physical address and sends it there. The next hop may be the final destination host, or it may be another router, which will perform the same steps.

III. B. Model solving based on DDPG

III. B. 1) Algorithm Principles

Since the state space and action space of the system are continuous in the network model and interference model mentioned above, this paper adopts deep reinforcement learning methods to solve the problem. The DDPG algorithm is considered an effective means for learning optimization strategies in high-dimensional continuous spaces, exhibiting good stability and convergence. The DDPG algorithm is based on the AC framework. Since the state and action spaces are continuous, neural networks are used to fit the policy function and the Q function. The Actor network represents the policy network, which generates actions, while the Critic network represents the Q network, which evaluates the quality of the policy and updates the parameters of the Actor network by calculating the error function. Compared to other AC algorithms, DDPG is based on deterministic policies, so the Actor network directly outputs actions rather than probability distributions of actions. DDPG stores historical data such as state transitions and immediate rewards from the interaction between the agent and the environment in an experience replay pool through experience replay. When updating parameters, random samples are drawn from the experience replay pool for use as training data. Due to the randomness of sample extraction, the correlation between training data is reduced, and high temporal correlation between data can cause the neural network to get stuck in a local optimum. Additionally, DDPG employs target network technology, which involves decoupling the network used to compute target values during algorithm training. A separate target network with the same structure as the training network is used to compute target values, and the target network's parameters are updated by copying from the training network's parameters, thereby stabilizing the algorithm's iterative training process. Therefore, this chapter employs the DDPG algorithm to solve the model. This algorithm is applicable to continuous state and action spaces, suitable for dynamic environments, and the routing algorithm is easily scalable, resulting in a more stable and effective routing algorithm model. In this study, the DDPG algorithm is considered for solving the MDP model, and a corresponding intelligent dynamic adjustment algorithm for link weights is designed.

The basic logical architecture of the DDPG algorithm is shown in Figure 2. Similar to the AC algorithm, the DDPG algorithm also consists of two modules: Actor and Critic. Both the Actor and Critic modules contain two neural networks, namely the online network and the target network, and they share the same network structure. The online network is used for real-time learning and training, while the target network is used to reduce the correlation between training data. The parameters of the target network are periodically updated based on the parameters of the online network. During training, an experience replay pool is used to store interaction data. The neural networks of the online Actor and online Critic are trained using sampled data, while the parameters of the target Actor and Critic networks are updated based on the parameters of the online Actor and Critic networks. The target network uses soft updating to update its parameters, meaning it copies a portion of the parameters from the online network rather than copying all of them.

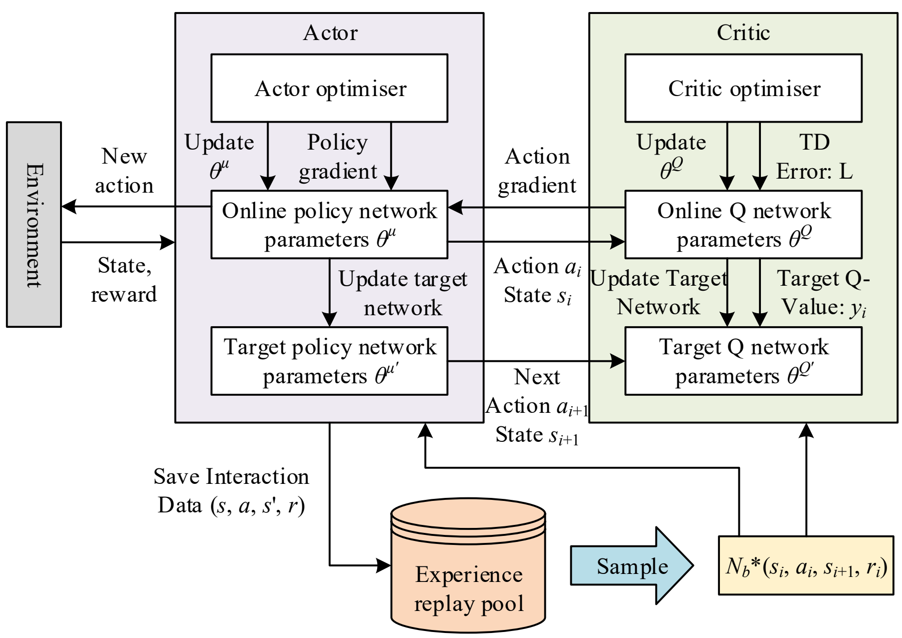

Figure 2: The logical architecture of DDPG

III. B. 2) Actor Network

As shown in Figure 2, the Actor network is used to output actions. There are two Actor networks in DDPG. The online Actor network is used to generate the current action a based on the current state s, and is responsible for updating the policy parameters θ^μ. The action interacts with the environment to produce the next state s' and obtain the reward value r. This interaction information (s, a, s', r) is stored in the experience replay pool. Based on the sampled next state s' from the experience replay pool, the target Actor network is input to obtain the optimal next action a'. The parameters of the online Actor network are updated using the action gradients obtained from the online Critic network, while the parameters of the target Actor network are updated using soft updating based on the parameters of the online Actor network. Since the action space is continuous, this study uses a neural network to learn and obtain actions from the system state. In this study, the input to the online Actor network is the state of all links, and the output is the weights of the links. A portion of data (s,a,s',r) is sampled from the experience cache pool. To update the parameters of the online Actor network, backpropagation of the policy gradient is required, with the policy gradient calculated as in Equation (3):

$$\begin{aligned}&\nabla_{\theta^\mu} Y\left(\mu_{\theta^\mu}\right) = grad[Q]^* grad[\mu] \\ &\approx \frac{1}{N_b}\sum_i \nabla_a Q\left(s,a\mid\theta^Q\right)\Big|_{s=s_i,a=\mu(s_i)} \nabla_{\theta^\mu}\mu\left(s\mid\theta^\mu\right)\Big|_{s_i}\end{aligned} \tag{3}$$

Among these, $grad[Q]$ represents the action gradient obtained from the online Critic network, which is used to characterize the adjustment direction of the online Actor network in order to achieve higher returns. Meanwhile, $grad[\mu]$ is the parameter gradient of the online Actor network, which is used to characterize how the online Actor's neural network adjusts parameters to select high-return actions. Here, N_b denotes the number of randomly sampled samples, and equation (4) applies:

$$a = \mu_{\theta^\mu}(s) + N_t \tag{4}$$

s denotes the agent's state, a denotes the action, μ_{θ^μ} denotes the policy function, and N_t denotes random noise. Since the action space is continuous, the policy function can be fitted using a neural network, with the neural network parameters denoted as θ^μ. The parameters of the online Actor network are updated using backpropagation, as shown in Equation (5):

$$\theta^\mu = \theta^\mu + \alpha\nabla_{\theta^\mu} Y\left(\mu_{\theta^\mu}\right) \tag{5}$$

Here, α denotes the parameter update step size of the Actor network. Due to the deterministic behavior generated by the policy function, random noise N_t is introduced to explore the environment. By introducing random noise on top of the deterministic behavior in the output, the agent can randomly explore the unknown action space, thereby enhancing the algorithm's learning performance. Therefore, the action strategy is transformed into a stochastic process, and actions are sampled from the stochastic process to obtain the corresponding actions, as shown in Figure 3. In this study, Gaussian noise is used as the introduced random noise to drive the agent to randomly explore unknown states. At the same time, since the state space studied in this paper is continuous, a neural network is used to fit the action value function, where θ^Q is the parameter of the online Q network.

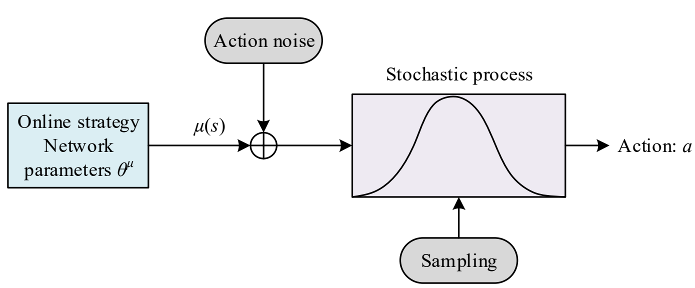

Figure 3: Action decision-making process

III. B. 3) Critic Network

The Critic network outputs the action value function. In DDPG, there are two Critic networks, one of which is the online Critic network responsible for updating the parameters of the value network. It calculates the Q value of the current state and current action sampled from the experience replay pool, and this Q value is used to update the parameters of the online Actor network. The target Critic network is used to calculate the Q value for the next state s' sampled from the experience replay pool and the next action a' selected from the target Actor network. The two Q values are used to calculate the loss for updating the online Critic network, while the target Critic network adopts soft updates based on the parameters of the online Critic network. Since the state space is continuous, this study fits the Q function using a neural network. The online Critic network is updated using mean squared error, which can be calculated using the TD bias. In DDPG, the TD bias can be obtained from both the online network and the target network as shown in Equation (6):

$$L = \frac{1}{N_b}\sum_i \left(y_i - Q\left(s_i, a_i \mid \theta^Q\right)\right)^2 = \frac{1}{N_b}\sum_i \delta_i^2 \tag{6}$$

Among these, N_b is the random sampling batch size, and y_i is the target Q value, representing the Q value obtained by taking action a_{i+1} in the target network when calculating the next state s_{i+1}. The next action is obtained from the target Actor network, with the next state as its input. δ_i denotes the TD error. The calculation of y_i is as shown in Equation (7):

$$y_i = r_i + \gamma Q'\left(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'}) \mid \theta^{Q'}\right) \tag{7}$$

Among them, r_i represents the reward value obtained by the agent when taking action a_i in state s_i. The parameters of the target Q network are $\theta^{Q'}$ and the parameters of the target Actor network are $\theta^{\mu'}$. The next action can be obtained according to the target strategy, that is, equation (8):

7

$\gamma \in [0,1]$ represents the discount factor. The parameters of the online Critic network are updated using the gradient descent method, as shown in equation (9):

$$\theta^Q = \theta^Q + \alpha\delta\nabla_a Q(s, a \mid \theta^Q) \tag{9}$$

where α represents the parameter update step size of the Critic network, and δ represents the TD bias.

In the DDPG algorithm, soft updates are used to update the parameters of the target Actor network and the target Critic network, resulting in equations (10)-(11):

$$\theta^{\mu'} = \left(1-\tau\right)\theta^{\mu'} + \tau\theta^{\mu} \tag{10}$$

$$\theta^{Q'} = \left(1-\tau\right)\theta^{Q} + \tau\theta^{Q'} \tag{11}$$

Here, τ denotes the target network update parameter, τ is a constant, and $0 < \tau < 1$.

IV. Evaluation of the effectiveness of dynamic routing strategies

IV. A. Testing of operational performance

To evaluate the performance of the dynamic routing strategy described in this paper, we compare the routing protocol based on this dynamic routing strategy (M3) with the classic AODV routing protocol (M1) and the ACO-AODV routing protocol (M2). We compare the performance of three metrics—average end-to-end delay, packet loss rate, and throughput—under different node numbers and node movement speeds.

IV. A. 1) Impact of Different Node Numbers

In the experiment, the node speed was set to 5 m/s, with the number of nodes ranging from 10 to 50. Each set of five drone nodes constituted an experimental point. Ten simulations were conducted under different node configurations, and the final results were obtained by averaging the data from the ten simulations.

The impact of different node counts on average end-to-end latency is shown in Figure 4. As the number of nodes increases, the average end-to-end latency of all routing protocols gradually increases. The (M3) routing protocol, which comprehensively considers network node signal reception strength, congestion, and hop count, reduces average end-to-end latency by 16–21 ms compared to the traditional AODV routing protocol, thereby reducing network latency.

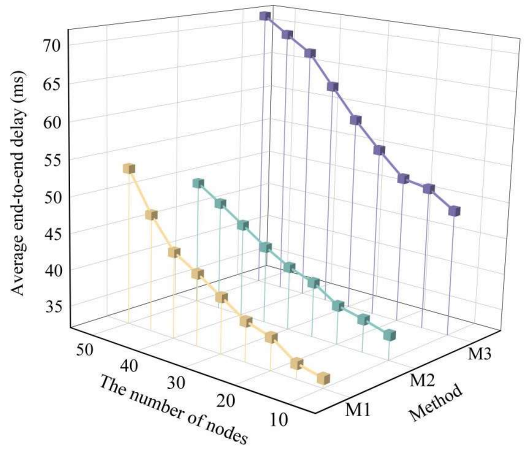

Figure 4: The influence of different node numbers on the average end-to-end delay

The impact of different numbers of nodes on throughput is shown in Figure 5. As the number of nodes increases, the throughput of the drone network gradually decreases. This is because an increase in the number of nodes leads to an increase in the number of routing messages transmitted by the drone network, resulting in network congestion. The (M3) routing protocol selects more stable routing links by considering factors such as node congestion, thereby transmitting more data packets. Compared to the traditional AODV routing protocol, throughput can be increased by up to 15.23%, improving the throughput performance of the routing protocol.

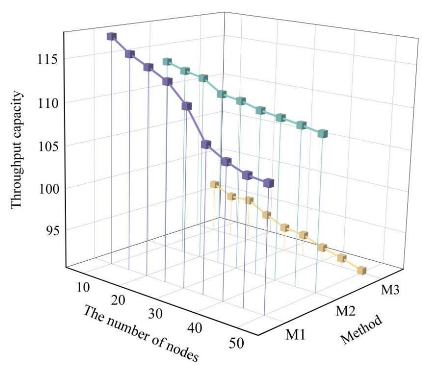

Figure 5: The influence of different node numbers on throughput

The impact of different node numbers on packet loss rate is shown in Figure 6. The packet loss rate of the drone network gradually increases. Only by discarding packets during data transmission can the packet loss rate be reduced. The (M3) routing protocol selects routing links with fewer link breaks by combining factors such as signal reception strength. Compared with the traditional AODV routing protocol, the packet loss rate is reduced by 5.23%, thereby reducing data loss during data transmission.

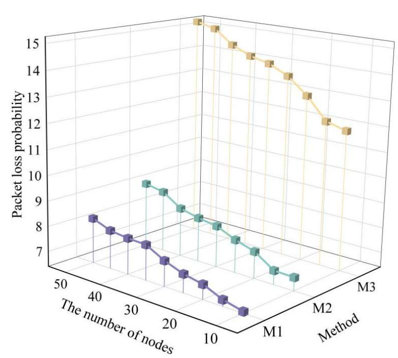

Figure 6: The influence of different node numbers on packet loss rate

IV. A. 2) Impact of different node movement speeds

The average end-to-end latency performance comparison of different protocol networks under varying node movement speeds is shown in Figure 7. As the node movement speed increases, the average end-to-end latency exhibits an upward trend. Compared to the (M1) classic AODV routing protocol and the (M2) ACO-AODV routing protocol, the (M3) routing protocol reduces the average end-to-end latency by at least 23.78% and up to 39.86%. The (M3) routing protocol further reduces the probability of data packets encountering routing holes, decreases the number of link switches, and lowers the probability of nodes going offline. Therefore, it not only improves data forwarding efficiency but also enhances communication reliability and efficiency.

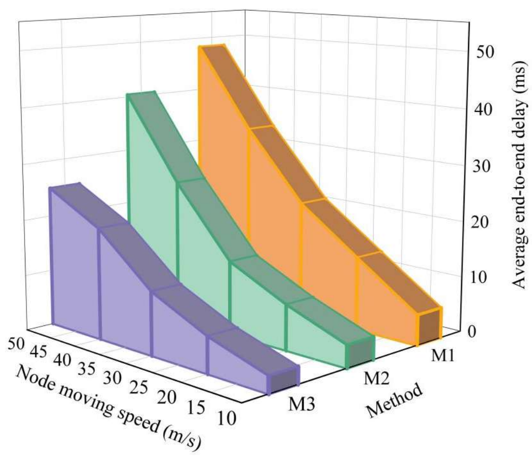

Figure 7: Average end-to-end delay

Figure 8 shows the packet loss rate comparison of different protocol networks under different node movement speeds. Although the packet loss rate of the three routing protocols increases with the increase in node speed, the packet loss rate of the (M3) routing protocol is still at least 15.79% lower than that of the other two traditional routing protocols and can be reduced by up to 27.13%, which can effectively improve the success rate of data forwarding and enhance the reliability of network transmission.

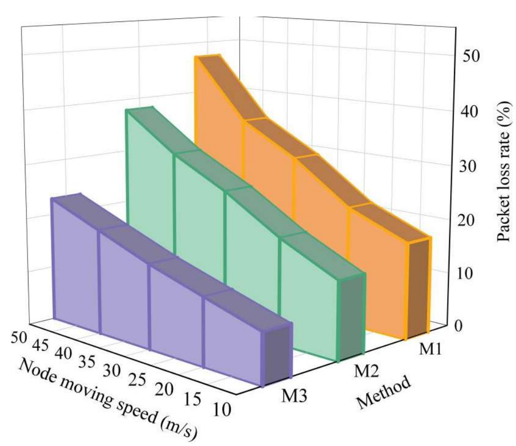

Figure 8: Packet loss rate

The throughput performance comparison of different protocol networks under varying node movement speeds is shown in Figure 9. Overall, as the node movement speed increases, the throughput of all three protocol networks decreases. However, the (M3) routing protocol still shows an improvement compared to the other two traditional routing protocols, with an increase of at least 6.93% and up to 23.95%. This is because the (M3) routing protocol in this paper takes longer to send data, resulting in larger data volumes and higher throughput in the network, which further enhances network reliability.

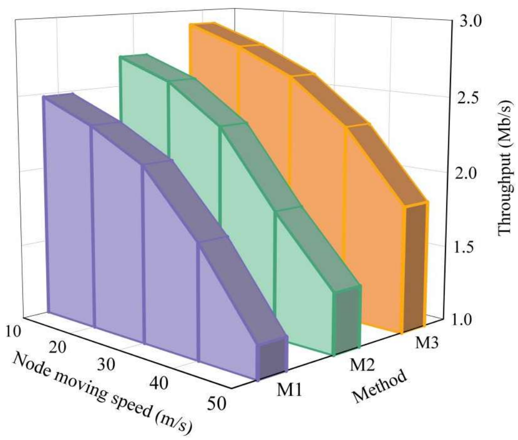

Figure 9: Throughout

IV. B. Verification of the effectiveness of dynamic routing strategies

IV. B. 1) Average jitter performance

The average jitter performance of different algorithms at various drone movement speeds is shown in Figure 10. Overall, the average jitter of all three algorithms decreases as the drone's movement speed increases. Among them, the (M3) routing protocol algorithm consistently exhibits lower average jitter than the other two traditional routing protocol algorithms (<3.0 ms). When the drone's movement speed is 30 m/s, the difference in average jitter performance between the (M3) routing protocol algorithm and the traditional routing protocol algorithms is the largest, reaching 1.428 ms.

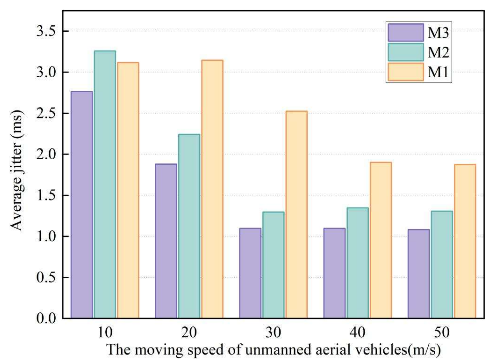

Figure 10: Average jitter of each algorithm in the random motion scenario of the drone

IV. B. 2) Energy balance capability

To analyze and compare the energy balancing capabilities of the routing algorithm designed in this paper (S1) and the traditional routing algorithm (S2) in dynamic scenarios, the node speed was set to 15 m/s. Figure 11 shows the changes in the number of node deaths for both algorithms over a simulation time of 300 s.

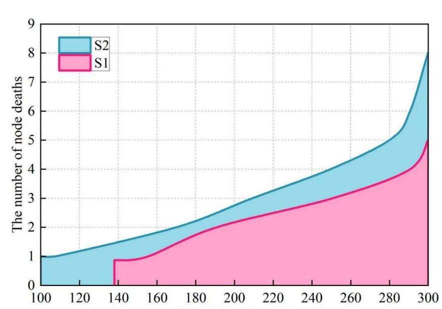

Figure 11: Change of node death number with time

The number of node deaths for both algorithms continues to increase as the simulation progresses, with the rate of increase accelerating. This is because the energy consumption of nodes gradually increases over time, leading to a continuous decrease in the overall remaining energy of the network nodes. Additionally, the increased overhead associated with route rediscovery caused by node deaths further accelerates the rate of node deaths. However, the first node death time in the routing algorithm designed in this paper (S1) is later (138 seconds) and the total number of deaths is lower (0.87). In contrast, the curve is flatter, and the rate of increase in the total number of deaths is slower. This validates the superior performance of the routing algorithm designed in this paper in terms of energy balancing compared to traditional algorithms.

V. Conclusion

Based on a mathematical model for unmanned aerial vehicle (UAV) communication scenarios in emergency housing support, this paper designs a DDPG deep reinforcement learning method for dynamic adjustment of wireless link routing costs. This dynamic routing strategy effectively balances network load, not only enhancing the adaptability of UAV ad hoc network communication routing protocols in rapidly changing network environments but also ensuring communication service quality and energy balance, thereby guaranteeing the rapid transmission of communication data within UAV ad hoc networks.

Under different node numbers, the routing protocol equipped with the dynamic routing strategy proposed in this paper reduces the average end-to-end delay by 16-21 ms and increases throughput by up to 15.23% compared to traditional routing protocols. Under different node movement speed conditions, the routing protocol equipped with the dynamic routing strategy described in this paper reduces average end-to-end latency by up to 39.86%, reduces packet loss rate by up to 27.13%, and increases throughput by up to 23.95% compared to traditional routing protocols. Additionally, thanks to the support of the dynamic routing strategy described in this paper, the average jitter performance of the routing protocol remains below that of traditional routing protocol algorithms (<3.0 m/s), with later node death times (138 s) and fewer total deaths (0.87).

Funding

This research was supported by the:

2023 Vocational Education Research Special Project of the 14th Five-Year Plan Project of Hunan Education Science: Research and practice of digital transformation of Hunan Vocational Education Curriculum under the background of "5G+ Smart Education" (No.: XJK23AZJ010);

2023 Research Project on Education and Teaching Reform of Vocational Colleges in Hunan Province: Research on the Construction and Application Path of digital Textbooks in Higher Vocational Colleges under the background of AI large Model (No.: ZJGB2023495);

2021 Outstanding Youth Project of Science Research Project of Hunan Provincial Department of Education: Visualization Research and Practice of multi-source Data Fusion in teaching process under the background of "5G+ Smart Education" (No.: 21B0913).

References

[1] Wang, Y., & Xu, Z. (2023). An emergency shelter location model based on the sense of security and the reliability level. Journal of Systems Science and Systems Engineering, 32(1), 100-127.
[2] Müller, C. C., & ElZomor, M. (2024). Origami Housing: An Innovative and Resilient Postdisaster Temporary Emergency Housing Solution. Journal of Architectural Engineering, 30(3), 04024025.
[3] Hanapi, N. L., Sh Ahmad, S., & Abd Razak, A. (2018). Emergency safety for multi-storey public housing in Kuala Lumpur. e-Academia Journal, 8, 64-70.
[4] Axelsson, F. R. E. D. R. I. K. (2012). The Emergency Housing Project. Chalmers University Of Technology.
[5] Peacock, W. G., Dash, N., Zhang, Y., & Van Zandt, S. (2017). Post-disaster sheltering, temporary housing and permanent housing recovery. Handbook of disaster research, 569-594.
[6] Ghanbarzadeh Ghomi, S., Wedawatta, G., Ginige, K., & Ingirige, B. (2021). Living-transforming disaster relief shelter: A conceptual approach for sustainable post-disaster housing. Built Environment Project and Asset Management, 11(4), 687-704.
[7] Lines, R., Walker, J. F., & Yore, R. (2022). Progression through emergency and temporary shelter, transitional housing and permanent housing: A longitudinal case study from the 2018 Lombok earthquake, Indonesia. International Journal of Disaster Risk Reduction, 75, 102959.
[8] Calle Müller, C., & Elzomor, M. (2024). Origami Housing: A Post-Disaster Temporary Emergency Housing Solution. In Construction Research Congress 2024 (pp. 346-355).
[9] Davlasheridze, M., & Miao, Q. (2021). Natural disasters, public housing, and the role of disaster aid. Journal of Regional Science, 61(5), 1113-1135.
[10] Najafi, E. (2023). Locating emergency and temporary housing after the earthquake in Damghan using fuzzy model. Disaster Prevention and Management Knowledge, 13(1), 62-79.
[11] Biswas, A. (2019). Exploring Indian post-disaster temporary housing strategy through a comparative review. International Journal of Disaster Resilience in the Built Environment, 10(1), 14-35.

[12] Choi, J., Deshmukh, A., & Hastak, M. (2016). Increase in stress on infrastructure facilities due to natural disasters. International journal of urban sciences, 20(sup1), 77-89.
[13] Sakai, J., Onodera, T., & Ikeuchi, K. (2022). Case Study on Damage and Recovery of Water Supply Related Facilities Damaged by Recent Flood Disasters. Journal of Disaster Research, 17(6), 1101-1109.
[14] Mao, X., Fung, O. W. M., Hu, X., & Loke, A. Y. (2018). Psychological impacts of disaster on rescue workers: A review of the literature. International journal of disaster risk reduction, 27, 602-617.
[15] Wang, H., Huo, D., & Alidaee, B. (2014). Position unmanned aerial vehicles in the mobile ad hoc network. Journal of Intelligent & Robotic Systems, 74, 455-464.
[16] Krichen, L., Fourati, M., & Fourati, L. C. (2018, August). Communication architecture for unmanned aerial vehicle system. In International conference on ad-hoc networks and wireless (pp. 213-225). Cham: Springer International Publishing.
[17] Jovičić, R., Gostimirović, L., & Milašinović, S. (2024). Use of New Technologies in the Field of Protection and Rescue During Disasters. International Journal of Disaster Risk Management, 6(1), 111-122.
[18] Agrawal, J., Kapoor, M., & Tomar, R. (2022). A novel unmanned aerial vehicle ‐ sink enabled mobility model for military operations in sparse flying ad ‐ hoc network. Transactions on Emerging Telecommunications Technologies, 33(5), e4466.
[19] Arafat, M. Y., & Moh, S. (2019). Routing protocols for unmanned aerial vehicle networks: A survey. IEEE access, 7, 99694-99720.
[20] Nazib, R. A., & Moh, S. (2020). Routing protocols for unmanned aerial vehicle-aided vehicular ad hoc networks: A survey. IEEE Access, 8, 77535-77560.
[21] Zeng, Y., Zhou, J., Liu, Y., Cao, T., Yang, D., Liu, Y., & Shi, X. (2023). Stable routing protocol for unmanned aerial vehicle ad ‐ hoc networks based on DQN ‐ OLSR. Iet Communications, 17(1), 73-85.
[22] Cambra, C., Sendra, S., Lloret, J., & Parra Boronat, L. (2015). Ad hoc network for emergency rescue system based on unmanned aerial vehicles. Network Protocols and Algorithms, 7(4), 72-89.