

SARNet: A Novel Routing Protocol for Search and Rescue Flying Ad-hoc Networks

Housseem E. Turki^{1†}, Hideaki Takanobu^{1,2}

¹Department of Mechanical Systems Engineering, Kogakuin University, Tokyo, Japan

²Stanford University, California, USA

Abstract: This paper introduces SARNet, a hybrid routing protocol for Search-and-Rescue (SAR) Flying Ad-hoc Networks that integrates event-driven OLSR broadcasts with SINR-triggered link adaptation and ant-inspired unicast reinforcement. It aims to achieve robust, low-latency communication for UAV swarms by combining deterministic broadcast with probabilistic multipath unicast. The protocol is lightweight and suited to dynamic, infrastructure-less environments. Simulations benchmark SARNet against AODV under varying mobility to validate delay stability and delivery performance.

Keywords: Swarm Intelligence (SI), Flying Ad-hoc Networks (FANETs), Search and Rescue (SAR), UAV Swarms

1. INTRODUCTION

In multi-robot and drone applications, Swarm Intelligence (SI) enables groups to coordinate tasks, adapt to dynamic environments, and maintain robust operation even when individual agents fail. A prominent example is the use of UAV swarms in search-and-rescue (SAR) missions. Here, drones work together to rapidly scan hazardous or expansive areas, share real-time data, and access locations unsafe for humans. Swarm coordination accelerates response times and improves reliability, making SAR efforts more effective.

Central to the success of UAV swarms is the ability to communicate without fixed infrastructure. This is achieved through ad hoc networks, which have evolved from ground-based Mobile Ad Hoc Networks (MANETs) to Vehicular Ad Hoc Networks (VANETs), and now to Flying Ad Hoc Networks (FANETs) designed specifically for aerial vehicles. In FANETs, UAVs act as both nodes and routers, forwarding data across the network in a peer-to-peer, multi-hop fashion. However, FANETs present unique challenges compared to their ground-based counterparts: UAVs move rapidly in three dimensions, causing frequent and unpredictable changes in network topology [1]; communication links are often line-of-sight but can be disrupted by distance, environmental factors, or urban obstructions; and drones are constrained by limited energy and bandwidth. Traditional MANET routing protocols struggle to meet these demands. Proactive protocols, which maintain up-to-date routes at all times, generate excessive control traffic and quickly drain the limited resources of UAVs. Reactive protocols, which establish routes only when needed, can suffer from delays and temporary communication losses when network links break. Geographic and cluster-based approaches, while reducing some overhead, depend on constant position updates or fixed structures that are ill-suited to the dynamic and unpredictable nature of FANETs. To overcome these challenges, there is a growing

recognition of the need for hybrid, adaptive, and distributed routing approaches for SAR-FANETs. Prior work on routing for SAR-FANETs has produced specialized delay-tolerant and adaptive protocols that use location and mission context to manage intermittent links. LADTR combines geographic forwarding with store-carry-forward, predicting UAV positions and using dedicated ferries to maintain connectivity [2], which achieves high delivery rates with low overhead in disaster scenarios. GeoSaW leverages UAV flight plans, spraying a single message copy to the node closest to the destination [3], resulting in high delivery with minimal overhead, though with increased latency due to store-and-forward. AZRP partitions the network into zones, exchanging proactive updates within zones and using reactive queries between them [4], dynamically switching modes based on mission context to optimize routing. These SAR protocols all aim to maintain connectivity in dynamic, obstacle-prone environments but differ in their approaches. Unlike LADTR and GeoSaW, which forward whole messages along predicted paths, or AZRP, which manages routes at zone-level aggregates, we propose SARNet, a protocol that operates per-packet, using OLSR for proactive link-state broadcasts and an AntHocNet-inspired reactive mechanism for unicast. It incorporates real-time link-quality triggers via Signal-to-Interference-plus-Noise Ratio (SINR) monitoring for immediate adaptation, unlike earlier methods that rely on pre-planned trajectories or coarse queries. SARNet also tunes routes based on quality-of-service (QoS) needs and enhances resilience through multiple candidate paths and redundancy, unlike single-copy schemes in LADTR and GeoSaW. Our approach blends proactive efficiency with swarm adaptability, allowing UAV networks to respond swiftly to changing conditions and maintain resilient, QoS-aware connectivity in SAR environments. This paper introduces the proposed framework and validates its preliminary implementation through simulation.

2. SARNET

[†] Housseem E. Turki is the presenter of this paper.

To meet SAR-FANET demands, our protocol integrates two complementary routing paradigms: OLSR's efficient, stable backbone and AntHocNet's adaptive intelligence.

2.1 Event-Based Topology Propagation

Optimized Link State Routing (OLSR) refreshes network state by flooding Topology-Control (TC) packets at fixed intervals [5], ensuring all nodes eventually share a consistent view of the topology but at the cost of continuous control overhead. In sparse or high-mobility scenarios typical of SAR-FANETs, OLSR's reliance on periodic TC messages can consume considerable bandwidth and energy. Moreover, the protocol's periodic update mechanism may delay adaptation to sudden topology changes, making it less resilient in rapidly evolving SAR environments. SARNet addresses this by replacing OLSR's periodic TC floods with a demand-driven approach. Nodes send lightweight HELLO beacons at short fixed intervals, and when a neighbor is lost or discovered, a TC- Δ message is immediately broadcast, encoding only the changed links and an incremented Advertised Neighbor Sequence Number (ANSN). Multipoint Relays (MPRs) forward this update across the swarm, ensuring rapid network-wide propagation without periodic timers. This event-driven method minimizes routine control traffic during stable flight, reduces convergence time after link disruptions, and maintains global consistency by using ANSN to purge outdated entries as soon as fresher information arrives.

2.2 Anticipatory Link Management with SINR

SARNet enhances reliable communication in SAR drone swarms by anticipating link degradation rather than simply reacting to failures. It supplements the traditional OLSR neighbor test with continuous monitoring of SINR on each active wireless link. Each HELLO updates a running SINR estimator; when SINR drops below a predefined threshold, the link is flagged as degrading and the routing layer is warned well before packet loss occurs. To prevent unnecessary route changes from normal signal fluctuations, our approach maintains a smoothed quality score for each neighbor, adjusting it based on SINR observations and applying dual hysteresis thresholds to govern state transitions. If the score falls below the lower threshold, the link is marked bad and routes are recomputed; recovery only occurs after the score exceeds the higher threshold, stabilizing routing in dynamic environments. When a link first shows signs of degradation, both endpoints issue a localized DEGRADATION_ALERT within two hops, allowing affected flows to reroute before the channel fails. If an alternate next-hop is available, traffic is shifted immediately; otherwise, the link remains active until it is truly unusable. This approach balances early detection with network stability, reduces unnecessary overhead, and ensures robust connectivity.

2.3 Ant-Inspired Multipath Resilience

A lightweight ant-colony-inspired mechanism is integrated into the OLSR backbone to enable continuous multipath exploration and reinforcement for unicast

routing. Its key strengths lie in path diversity, robustness, and flexibility, essential for uncertain SAR terrains where optimal paths may change due to obstructions, interference, or node failures. Artificial "ant" packets mimic the behavior of AntHocNet: forward ants (FANTs) discover routes, backward ants (BANTs) reinforce effective paths, and pheromone values decay over time to fade out obsolete routes. By restricting ant propagation to OLSR's MPR structure, overhead remains low while providing real-time multipath resilience, an advantage over classical OLSR and standard AntHocNet flooding. FANTs travel during unicast sessions along the highest-probability next hop at regular intervals to refresh link metrics and seek shortcuts. The probability $P_{ij}^d(t)$ of node i selecting neighbor j for destination d is given by:

$$P_{ij}^d(t) = \frac{[\tau_{ij}^d(t)]^\beta}{\sum_{k \in N_i} [\tau_{ik}^d(t)]^\beta} \quad (1)$$

Where $\tau_{ij}^d(t)$ is the pheromone value for link (i, j) toward destination d , N_i is the set of all neighbors of node i , and the exploitation/exploration parameter $\beta > 1$ biases selection toward better-ranked links [6]. Each ant carries metrics including hop count, queuing delay, congestion, and SINR averages, allowing route reinforcement to reflect both network topology and wireless link quality. When a BANT returns, the pheromone value for each traversed link is updated as:

$$\tau_{ij}^d(t+1) \leftarrow (1 - \alpha)\tau_{ij}^d(t) + \alpha Q \quad (2)$$

with Q proportional to the inverse of the observed path cost, and $0 < \alpha < 1$ as the learning rate. Traffic is probabilistically distributed over multiple promising routes, with pheromone levels updated based on observed path costs and periodically evaporated to remove stale information. This evaporation is captured by:

$$\tau_{ij}^d(t+1) \leftarrow (1 - \rho)\tau_{ij}^d(t) \quad (3)$$

where $0 < \rho < 1$ is the evaporation rate controlling how quickly old pheromone information fades. In each update cycle, Eq. (3) is applied first, and the freshly returned BANT then reinforces its path with Eq. (2). Overhead control is achieved by limiting reactive FANT unicasts to the MPR subtree, capping exploration depth, and piggybacking BANTs on data acknowledgments. When link quality degrades, pheromone values on affected links are penalized, shifting traffic toward healthier alternatives and enabling rapid route recovery. This design ensures path redundancy and auto load balancing combining the adaptability of ant-based routing with the deterministic broadcast plane of OLSR.

2.2 Deterministic Broadcast, Adaptive Unicast

Forwarding decisions are separated by flow type, keeping broadcast and unicast states distinct. Broadcast forwarding uses only the standard OLSR neighbor and MPR tables, while unicast forwarding builds on the same

OLSR-seeded routing table, extending each destination entry with desirability weights that are refined by ant-based path reinforcement. Broadcast packets, such as network-wide alerts, HELLOs, event-driven topology updates, and any mission-wide data that benefits from reliable MPR-based flooding (e.g., situational maps or global status reports), are forwarded strictly along the OLSR MPR backbone. This ensures deterministic reach and minimizes duplication in dense airspaces. Unicast mode handles high-rate payloads like video, LiDAR, and command acknowledgments using a probabilistic next-hop lookup. Initially, the routing table entry for a new destination contains only the OLSR-derived hop-count next hop. As ants gather metrics (count, queue congestion, SINR...) these entries are augmented with desirability weights, enabling adaptive load balancing and resilience. Forward ants periodically update these weights using multi-metric costs, and each transmission selects the next hop probabilistically according to current weights. On sending or receiving a packet, the node determines its class and applies the corresponding policy: rebroadcasting for broadcast flows if it is an MPR, or selecting the highest-probability next hop for unicast flows. This split design keeps broadcast forwarding lightweight and deterministic while granting unicast flows the adaptive multipath agility essential for bandwidth-intensive search-and-rescue operations.

3. BENCHMARKING SARNET AGAINST AODV

3.1 Simulation Setup and Parameters

The performance of SARNet (implemented here with only its event-driven TC updates and SINR-based link-quality handling) was evaluated against Ad hoc On-demand Distance Vector (AODV) in a simulated FANET search-and-rescue scenario. We used the OMNeT++ (v5.0) simulator [7] with INET 3.2.4, modeling 10 mobile UAV nodes and one fixed sink node at the center of a 500 m×500 m area. UAVs followed either a Random Waypoint (RWP) or a Gauss–Markov (GM) mobility model to capture different movement patterns. RWP is simple and widely used in MANET simulations, despite known artifacts (e.g. nonuniform node distribution); we include it for comparability with prior studies. The GM model (with $\alpha=0.8$, variance=0.5, update interval=0.4 s, and 10 m boundary margin) provides smoother, correlated motion suitable for UAVs. Key simulation parameters were chosen to reflect realistic SAR UAV operations and QoS demands. UAV speeds were swept over {4, 8, 12, 16, 20} m/s, covering the range from slow multirotor (~4 m/s) to fast fixed-wing (~20 m/s) flight. Each mobile node generated UDP traffic (one 1472-byte packet every 0.0059 s, ~2 Mb/s) toward the sink, starting at 10 s and running for 1200 s. Wireless settings used a two-ray ground path-loss model and IEEE 802.11g MAC/PHY (2.4 GHz, 54 Mbps), as commonly adopted in UAV network studies. The radio range was set to 200 m with fixed transmit power. SARNet-specific parameters were fixed (SNIR_threshold=1200, link-quality low=0.2, high=0.6, penalty=0.02, reward=0.01,

HELLO interval=1 s), while AODV ran with INET’s default settings. To ensure statistical validity, each scenario (combination of mobility model and speed) was repeated 5 times with independent random seeds. We averaged performance over these runs. Specifically, end-to-end delay was computed as the average one-way travel time of successfully delivered packets (per run, then averaged across runs). Packet delivery ratio (PDR) was computed as the fraction of packets sent by all mobile nodes that were received by the sink (averaged per run across nodes, then across runs). These two metrics are standard QoS measures for routing effectiveness: PDR indicates the delivery success rate, and end-to-end delay measures the time for a packet to traverse from source to destination. With the above configuration, we obtained statistically reliable comparisons of AODV and SARNet. The results of these experiments are presented next.

3.2 Results and Discussion

Fig. 1 shows packet delivery ratio (PDR) against node speed (4–20 m/s) for SARNet and AODV (aggregating Random Waypoint and Gauss–Markov mobility). Overall, both protocols achieve moderate delivery ratios (roughly 0.38–0.47).

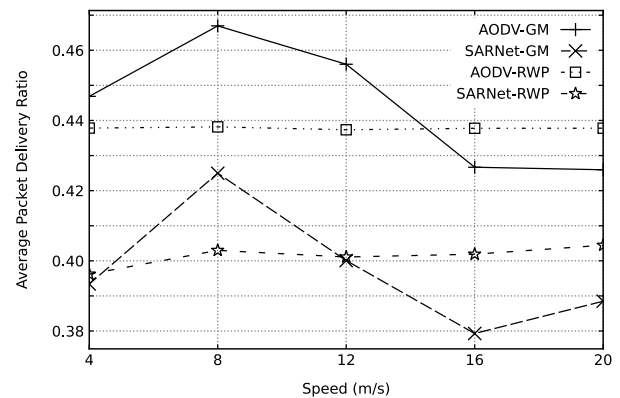


Fig.1 Average Packet Delivery Ratio vs. Node Speed

AODV under RWP maintains a nearly constant PDR of ≈ 0.44 across all speeds, while AODV under Gauss–Markov shows a decline from ≈ 0.47 at 8 m/s to ≈ 0.43 at 20 m/s. SARNet exhibits similar trends as AODV with lower PDR: it remains around 0.40 under RWP with very minimal variation as speed increases, while under GM PDR decreases linearly from a peak of about 0.42 at 8 m/s to a low of 0.38 at 16 m/s, then rises slightly to around 0.39 at 20 m/s. In other words, both SARNet and AODV maintain relatively stable packet delivery with speed under RWP, while both experience some reduction at higher speeds under GM. Fig. 2 shows the average end-to-end delay versus node speed for the same protocols. SARNet exhibits nearly constant delay (around 0.42–0.43 seconds) across all speeds and mobility models. In contrast, AODV’s delay is lower than SARNet’s at low speeds but rises sharply as speed increases. For AODV under both mobility models, delay drops to a minimum at moderate speeds (≈ 0.34 – 0.36 s around 8–12 m/s) and then climbs to ≈ 0.42 – 0.43 s by 16–20 m/s, matching SARNet’s delay at the highest speeds.

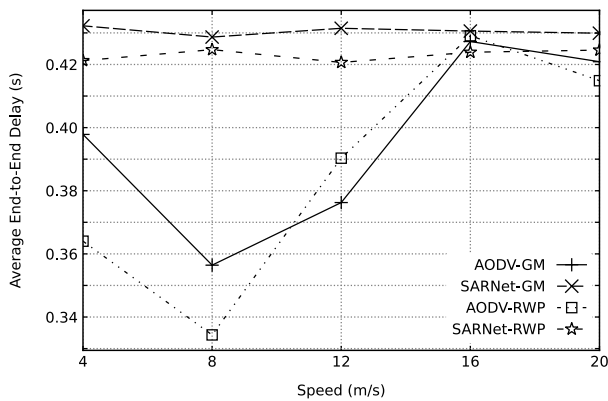


Fig.2 Average End-to-End Delay vs. Node Speed

Thus, at low mobility AODV yields smaller latency than SARNet, but at high mobility AODV incurs much larger delays (up to ≈ 0.43 s). These patterns are evident in Fig. 2: SARNet’s delay curves are flat, while AODV’s curves rise steeply at high speed.

These results reflect the competing design trade-offs embodied by the two protocols. SARNet’s proactive OLSR backbone, coupled with event-driven TC updates, keeps forwarding information continuously available, so packets rarely queue for route discovery; consequently, its end-to-end delay stays essentially flat at ≈ 0.42 s across the entire speed sweep (Fig. 2). At the same time, the SINR-triggered link-degradation handler is deliberately conservative: once a neighbour is flagged as weak, traffic is redirected or temporarily buffered even if the substitute path is longer or momentarily congested. In sparse topologies this early-warning behaviour can cost deliveries, which is why SARNet’s PDR sits a few percentage points below AODV’s, about four under RWP and up to nine under Gauss–Markov at the highest speeds (Fig. 1). (Ant-based reinforcement and flow-aware weighting, not yet implemented, would be expected to recoup some of this loss in future work.) AODV tells the opposite story. At low mobility the network changes slowly; route requests resolve quickly and the protocol enjoys both higher PDR (≈ 0.44 – 0.47) and lower delay (≈ 0.34 – 0.36 s). As speed rises, however, link breaks accumulate. Each break triggers a fresh route-discovery flood, during which packets queue or are dropped, so end-to-end delay climbs sharply to ≈ 0.43 s by 20 m/s (Fig. 2) and PDR declines modestly, particularly under Gauss–Markov where motion is more correlated (Fig. 1). The curve thus mirrors AODV’s stop-and-repair cycle, whereas SARNet’s flat curve reflects continuous connectivity maintained by its pre-emptive SINR alerts and instantaneous TC updates.

In short, the current partial implementation of SARNet trades a modest reduction in delivery ratio for highly stable latency, a balance that aligns with search-and-rescue priorities.

4. CONCLUSIONS AND FUTURE WORK

This paper introduced SARNet, a hybrid routing framework that merges the deterministic OLSR backbone with event-driven topology-control updates

and a proactive SINR-based link-degradation handler. While some adaptive layers are still in development, the current implementation already achieves one main search-and-rescue (SAR) goal: maintaining near-constant end-to-end delay as mobility increases. Simulations with ten UAVs show SARNet sustains a flat latency (~ 0.42 s) from 4 to 20 m/s, while AODV’s delay rises sharply beyond 12 m/s. This stability comes at the cost of a modest 4–9% reduction in packet-delivery ratio compared to AODV.

Looking ahead, SARNet development will concentrate on completing the ant-based path-reinforcement and flow-aware weighting modules to boost packet-delivery ratios without increasing latency. We will also introduce a Data Distribution Service (DDS) gateway for ROS 2, exposing SARNet routes with QoS guarantees and streamlining deployment in heterogeneous multi-robot teams. Planned evaluations include larger UAV swarms, richer traffic mixes, and urban-canyon propagation scenarios, while hardware-in-the-loop field trials will validate the protocol under real-world radio and mobility conditions..

REFERENCES

- [1] A. Guillen-Perez and M.-D. Cano, “Flying Ad Hoc Networks: A New Domain for Network Communications,” *Sensors*, Vol. 18, No. 10, Article 3571, 2018.
- [2] M. Y. Arafat and S. Moh, “Location-aided delay tolerant routing protocol in UAV networks for post-disaster operation,” *IEEE Access*, Vol. 6, pp. 59891–59906, 2018.
- [3] A. Bujari, C. T. Calafate, J.-C. Cano, P. Manzoni, C. E. Palazzi, and D. Ronzani, “A location-aware waypoint-based routing protocol for airborne DTNs in search and rescue scenarios,” *Sensors*, Vol. 18, No. 11, Article 3758, 2018.
- [4] S. A. Mostafa, A. A. Ramli, M. A. Jubair, S. S. Gunasekaran, A. Mustapha, and M. H. Hassan, “Integrating human survival factor in optimizing the routing of flying ad-hoc networks in search and rescue tasks,” in *Proc. AHFE Int. Conf. on Human Factors in Software and Systems Engineering, AHFE Open Access*, Vol. 61, pp. 50–56, 2022.
- [5] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot, “Optimized link state routing protocol for ad hoc networks,” *Proceedings of the IEEE International Multi Topic Conference (INMIC 2001), Technology for the 21st Century*, Lahore, Pakistan, pp. 62–68, 2001.
- [6] G. Di Caro, F. Ducatelle, and L. M. Gambardella, “AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks,” *European Transactions on Telecommunications*, Vol. 16, No. 5, pp. 443–455, 2005.
- [7] OMNeT++ Team, “OMNeT++ Discrete Event Simulator,” OMNeT++ Official Website, [Online]. Available: <https://omnetpp.org>, Accessed: July 14, 2025.