Comparative Analysis of KNN, RNG and K-RNG for Inter-Robot Communication

Hazem Chaabi
Inria, France
hazem.chaabi@inria.fr

Nathalie Mitton
Inria, France
nathalie.mitton@inria.fr

***Abstract*—In distributed multi-robot exploration, effective communication among mobile robots significantly impacts the efficiency of area coverage, mapping accuracy, and collaborative decision-making. Typically, robots share their information map based on dynamically established network topologies to balance communication overhead and information accuracy. This paper comparatively analyzes three dynamic topology construction strategies for mobile robots exploring unknown environments: the Relative Neighborhood Graph (RNG), the k-Relative Neighborhood Graph (K-RNG), and the K-Nearest Neighbors (KNN). RNG is a graph-based approach that maintains adaptive links based on geometric proximity, efficiently managing connectivity while reducing redundant data exchanges. The K-RNG generalizes RNG by allowing a tunable number of points within the geometric neighborhood, providing flexible control over network density. In contrast, KNN selects neighbors by prioritizing robots in immediate proximity, quickly adapting to local density changes. We conduct extensive simulations to evaluate these strategies against critical Quality-of-Service (QoS) metrics, including packet delivery ratio and latency. Results demonstrate RNG's effectiveness in reducing unnecessary data transmissions while maintaining stable connections, K-RNG's performance tunability based on the parameter k, and the strong dependency of KNN's performance on the choice of k.**

***Index Terms*—KNN, K-RNG, MRS, NS3**

I. Introduction

Multi-Robot Systems (MRS) offer significant advantages over single robots for complex tasks involving distributed objectives, redundancy, robustness, and notably increased exploration speed [1]–[3]. Their growing importance is evident in applications like search and rescue, environmental monitoring, and logistics [1], [4]. A critical MRS capability is the exploration of unknown environments for tasks such as mapping hazardous areas or post-disaster assessment, where efficient coverage and map accuracy are paramount [1], [5].

Effective distributed multi-robot exploration hinges on inter-robot communication during exploration for coordination and collective intelligence. Communication enables map sharing, coordinated movement, and collaborative decision-making, directly impacting exploration speed and map quality [1], [6]. Designing suitable communication strategies for use during exploration is crucial, especially in unknown environments where the communication medium itself is uncertain, necessitating adaptive and localized approaches rather than static infrastructures [1].

Maintaining robust communication in dynamic settings presents several challenges. Robot mobility causes changing topologies and requires dynamic links [1], [7]. There exists a fundamental trade-off between information sharing and communication overhead (bandwidth, energy, computation) [1]. Communication-restricted environments (e.g., tunnels, urban canyons) introduce intermittent connectivity issues due to obstacles and interference [1]. Additionally, motion and sensing uncertainties can disrupt links [4]. These factors demand adaptive communication strategies.

Graph-based approaches are used to manage dynamic inter-robot communication, aiming for connectivity and performance optimization [4], [7]. This paper compares three such strategies: the Relative Neighborhood Graph (RNG), the k-Relative Neighborhood Graph (K-RNG), and K-Nearest Neighbors (KNN).

A quantitative comparison of RNG, K-RNG, and KNN for dynamic topology control in multi-robot exploration using Quality-of-Service (QoS) metrics is needed. Critical QoS metrics include the packet delivery ratio (PDR) for reliability and latency for timely data exchange [1], [8]. While RNG is parameter-free, the performance of K-RNG depends on its parameter k, and the performance of KNN depends on its parameter k, both requiring careful tuning for optimal operation in MRS scenarios.

This paper presents a comparative analysis of RNG, K-RNG, and KNN for dynamic communication in distributed multi-robot exploration through NS-3 simulations. We evaluated these strategies against QoS metrics to demonstrate how each approach handles the trade-offs between connectivity, overhead, and performance.

The remainder of this paper is organized as follows. Section II reviews related work in multi-robot communication strategies. Section III details the background and implementation of RNG, K-RNG, and KNN communication models. Section IV describes the experimental setup and evaluation metrics. Section V presents and discusses the simulation results. Finally, Section VI concludes the paper and suggests future research directions.

II. Related Work

A. Challenges in Multi-Robot Communication

Maintaining reliable inter-robot communication in complex environments remains a core challenge for distributed Multi-Robot Systems (MRS). In subterranean or obstructed settings, signal attenuation and non-line-of-sight (NLoS) paths frequently cause intermittent connectivity, leading to potential data loss or inconsistent worldviews among robots [1]. Shetty *et al.* demonstrate that, under realistic motion and sensing uncertainties, traditional connectivity controllers degrade substantially when communication delays are introduced, motivating adaptive schemes that account for time-varying link quality and delays [4]. To address sporadic connectivity, systems like ACHORD expose low-level networking metrics (e.g., RSSI, packet delivery ratio) to planning layers, enabling robots to autonomously deploy or reposition droppable radios as relays to restore network links [1]. Overall, limited bandwidth, variable link quality, and dynamic disconnections necessitate communication-aware planning and continual network self-reconfiguration to preserve coordination in real-world MRS deployments [4].

B. Graph-Based Topology Control in MRS

Graph-based topology control has become a prevalent approach for preserving connectivity while minimizing communication overhead. A foundational method uses the *Relative Neighborhood Graph* (RNG), which connects two robots p and q only if no other robot lies within the closed disc intersection (the lune) defined by p and q [9]. This yields a sparse planar graph that provably preserves connectivity when starting from an all-to-all connected network, significantly reducing average node degree compared to fully connected topologies. Capelli *et al.* implement a decentralized Control Barrier Function (CBF) that enforces RNG-based connectivity constraints under time-delays, experimentally showing that accounting for communication latency is critical to maintain network integrity in practice [9].

Beyond RNG, the *k-Relative Neighborhood Graph* (K-RNG) generalizes the RNG criterion by allowing up to k other robots within the lune, tuning graph density for enhanced redundancy [10]. Yang *et al.* leverage probabilistic Line-of-Sight Control Barrier Certificates (PrLOS-CBC) to maintain k-RNG constraints under Gaussian localization noise, ensuring high-probability LoS connectivity for multi-robot teams while minimizing interference with nominal tasks [10]. Alternatively, the *K-Nearest Neighbors* (KNN) approach connects each robot to its k closest peers by Euclidean distance, dynamically adapting to local density changes but potentially forming non-planar graphs that incur higher communication overhead [11]. Empirical studies show that augmenting KNN with LoS or signal-strength awareness outperforms purely distance-based KNN in obstructed environments [11].

Recent work also integrates topology control into motion planning: Mazloomi *et al.* formulate a multi-objective QoS optimization for swarm robotics, linking control inputs to QoS metrics (e.g., latency, PDR) and employing evolutionary algorithms to jointly optimize motion and communication configuration for large robot teams [12]. Similarly, Luo *et al.* apply federated deep reinforcement learning to select *Reconfigurable Intelligent Surface* (RIS) phase shifts and robot trajectories, improving indoor multi-robot communication in NLoS scenarios by endowing robots with distributed RIS-assisted channel estimation and path planning [13]. These approaches illustrate the trend toward tightly coupling connectivity maintenance with high-level planning, beyond static graph selection.

C. Quality-of-Service (QoS) for Evaluating MRS Communication

To quantify and improve communication reliability, researchers measure standard network QoS metrics such as *Packet Delivery Ratio* (PDR) and end-to-end latency. Low PDR indicates frequent packet loss, which can lead to inconsistent mapping or delayed consensus; Jalil *et al.* show that optimizing local cache mechanisms and ROS2 QoS policies (e.g., `DEPTH`, `DEADLINE`) reduces latency and packet loss in Aggregated Robot Processing (ARP) architectures, thereby improving MRS cooperation [8]. Indeed, Jalil *et al.* report up to 35% reduction in average latency and a corresponding increase in PDR when balancing QoS settings across ROS2 nodes under realistic network congestion.

Abu-Aisheh *et al.* introduce *CARA*, a dynamic relay-placement algorithm that employs real-time link-quality estimation (e.g., RSSI, PDR) to position minimal relays, demonstrating a 10× faster mission completion by halving relay usage compared to distance-based algorithms [11]. Their simulations in the MDPI *Sensors* testbed show sustained PDR above 0.95 during multi-robot exploration in simulated mines. In parallel, Capelli *et al.* validate that Control Barrier Functions accounting for network delays maintain connectivity without sacrificing throughput, reporting PDR improvements of 12–18% under 100ms link latency compared to naive CBF implementations [9].

Recent field experiments by Saboia *et al.* further confirm the importance of QoS prioritization: ACHORD's multi-layer design stratifies data (e.g., critical telemetry vs. low-priority mapping) and uses droppable radios to maintain PDR ≈ 0.9 in DARPA SubT Finals trials, even when robots traverse deep NLoS sections [1]. In summary, optimizing PDR and latency via adaptive routing, relay placement, and QoS-aware middleware emerges as a key enabler for robust, scalable MRS communication under realistic conditions.

Our contribution differs from existing work by providing a direct comparative analysis of RNG, K-RNG, and KNN strategies using standardized QoS metrics in realistic multi-robot exploration scenarios through systematic NS-3 simulations.

III. Background and System Models

A. Communication Strategy Background

1) Relative Neighborhood Graph (RNG): The RNG was originally proposed by Toussaint for finite planar point sets [14]. In the context of multi-robot systems, an edge exists between two robots p and q if no third robot is closer to both. Formally, robots p and q are connected if their lune, defined as the intersection of two open disks of radius $d(p, q)$ centered at p and q respectively, contains no other robot from the set. This creates a sparse network based on an empty "lune" criterion. RNG is a subgraph of the Delaunay triangulation and a supergraph of the Euclidean Minimum Spanning Tree, ensuring connectivity [14], [15].

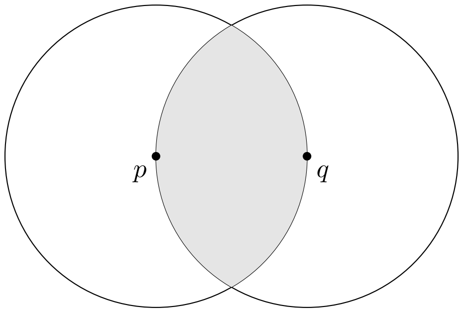

Fig. 1: Lune region for RNG: the shaded intersection of disks of radius $d(p, q)$.

2) k-Relative Neighborhood Graph (K-RNG): The K-RNG is a generalization of the RNG that offers greater flexibility in defining communication links [15]–[17]. An edge exists between two robots p and q in a K-RNG if and only if their lune $L(p, q)$ contains at most k other robots from the set [16], [17]. The parameter k, a non-negative integer, allows for a tunable definition of neighborhood: when $k = 0$, the K-RNG reduces to the standard RNG [14], [16]. As k increases, the K-RNG tends to become denser, incorporating more edges and potentially capturing broader neighborhood relationships relevant for robust communication. A K-RNG on n robots has $O(kn)$ edges [16].

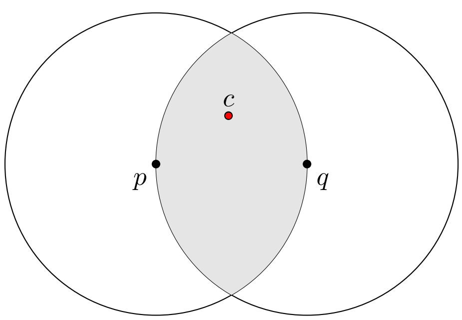

Fig. 2: K-RNG lune with $k = 1$: the single interior point c does not block the edge.

3) K-Nearest Neighbors (KNN): KNN involves each robot communicating with its k closest neighbors, adapting to local density changes [15], [18]. Unlike RNG and K-RNG, which use a lune-based geometric criterion, KNN uses a simple distance criterion. While KNN identifies the k metrically closest robots irrespective of their spatial configuration relative to each other, K-RNG (like RNG) uses a lune-based criterion that considers the local density within a geometrically defined region influenced by both candidate robots [14], [16], [17].

B. Implementation Models

We analyze three dynamic topology construction strategies implemented within a NS-3 application designed for multi-robot communication. Each robot independently maintains a neighbor table based on received periodic "Hello" messages, which contain the sender's current position and IP address. The neighbor table is updated with timestamps, and entries are removed if a neighbor is not seen for a specified timeout period.

1) RNG Implementation: The RNG strategy connects two robots if no third robot is closer to both. Our implementation calculates RNG neighbors by checking if for any other neighbor c, the maximum of the distances $d(p, c)$ and $d(q, c)$ is less than the distance $d(p, q)$, where $d(p, q)$ represents the Euclidean distance between robots p and q. If such a neighbor c exists, q is not an RNG neighbor of p.

2) K-RNG Implementation: Our implementation calculates K-RNG neighbors by counting how many neighbors c satisfy the lune condition with respect to the pair (p, q), i.e., $d(p, c) < d(p, q)$ and $d(q, c) < d(p, q)$. An edge is formed if this count is less than or equal to the strategy's configured k value.

3) KNN Implementation: Our implementation calculates KNN neighbors by computing the distance to all known neighbors, sorting them by distance, and selecting the k closest ones.

C. Robot Communication Application

The core communication logic is implemented in the `RobotCommunicationApp` class running on each robot node. Its primary function is to manage the dynamic communication topology based on the selected strategy and periodically send sensor data packets to determined neighbors. The application uses UDP sockets for communication on a fixed port. Data packets are assigned sequence numbers and timestamp tags upon creation. "Hello" messages are broadcast periodically at specified intervals to facilitate neighbor discovery.

D. Network Stack and Wireless Model

Each robot node is equipped with the NS-3 Internet Stack, which provides IPv4 capabilities. For wireless communication, we use the NS-3 WiFi module configured for IEEE 802.11n standard. The physical layer uses a YansWifiChannel with constant speed propagation delay model and LogDistance propagation loss model. The WiFi MAC is configured as AdhocWifiMac, with communication managed using ConstantRateWifiManager at 11 Mbps PHY rate.

E. Mobility Model

Robot mobility is modeled using the NS-3 Random Walk Mobility Model operating in "Time" mode, with movement segments lasting 2 seconds. Robot speed is drawn from UniformRandom distribution around a configured average of 1.0 m/s. Robots are restricted to move within a 100m × 100m rectangular simulation area, with initial positions randomly assigned.

IV. Experimental Setup

A. Simulation Framework

The experimental evaluation was conducted using NS-3 discrete-event network simulator with IEEE 802.11n wireless communication in ad-hoc mode. The Yans WiFi channel model was configured with Log Distance propagation loss (path loss exponent 3.0) and transmission power of 16.0206 dB.

Three geometric graph-based strategies were evaluated:

- K-Nearest Neighbors (KNN)
- Relative Neighborhood Graph (RNG)
- K-Relative Neighborhood Graph (K-RNG)

B. Experimental Scenarios

1) Scenario 1: Local Communication Evaluation: This scenario assessed the efficiency of direct peer-to-peer communication using a controlled environment with 5 to 60 robot nodes in a 100×100 meter area. The nodes used RandomWalk2D mobility (1-3 m/s) in 10-second simulations. The simulation application implemented neighbor discovery through periodic Hello messages (1-second intervals) and data transmission to neighbors determined by topology.

2) Scenario 2: Multi-hop Routing Performance: This scenario evaluated scalability and routing efficiency in data collection applications. Key parameters included:

- **Network scale**: 10-60 robot nodes
- **Field dimensions**: 200×200, 300×300, 400×400 meters
- **Mobility**: RandomWalk2D (1.0-3.0 m/s) over 300 seconds
- **Base station**: Fixed at field center
- **Transmission range**: 80 meters
- **Data generation**: 1 packet/second/robot (512 bytes)

We implemented greedy geographic forwarding method, which selects the neighbor with minimum Euclidean distance to the base station. Packets are dropped when no progress towards the destination is possible.

C. Performance Metrics

Both experimental scenarios employed identical core performance metrics to enable direct comparison between local communication and multi-hop routing efficiency:

- **Packet Delivery Ratio (PDR)**: Percentage of successfully received packets at the intended destination relative to total packets transmitted by the source.
- **End-to-end delay**: Time elapsed from packet generation to successful reception at the final destination through multiple hops.
- **Peer-to-peer delay**: Time elapsed from packet generation to successful reception by the immediate neighbor in direct communication.

Each configuration was executed across 5 independent runs with different random seeds for statistical significance, systematically varying node density size and topology parameters ($K \in \{1, 2, 3, 4\}$).

D. Implementation Details

Both scenarios utilize custom packet tagging with sequence numbers and timestamps for end-to-end tracking. Dynamic topology construction was performed based on current node positions, with neighbor relationships updated according to the selected geometric strategy. The experimental framework provides comprehensive trace collection enabling detailed performance analysis and result validation.

V. Results and Discussion

A. Scenario 1: Peer-to-Peer Communication Results

1) Packet Delivery Ratio Performance: The PDR analysis reveals distinct performance characteristics across all evaluated strategies, as shown in Figure 3. RNG demonstrates superior packet delivery performance, starting at approximately 95% for 5 robots and declining to 33% with 60 robots. KNN strategies ($k = 2, 3, 4$) exhibit comparable initial performance around 85-90% but converge to similar end-point values of 27-35%. K-RNG strategies consistently underperform, with initial PDR values ranging from 70-80% and declining to 22-28% at maximum network density. This underperformance can be attributed to the increased network density from allowing k points within lunes, leading to higher collision rates.

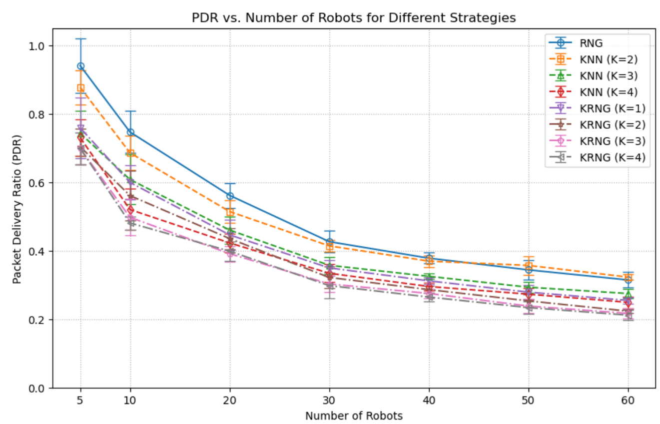

Fig. 3: Packet Delivery Ratio vs. Number of Robots for Different Strategies

The universal decline in PDR across all strategies reflects the fundamental challenge of maintaining communication quality as network density increases. However, RNG's consistently higher PDR throughout the evaluation range demonstrates its effectiveness in creating sparse, efficient communication topologies that minimize packet collisions and interference.

2) Peer-to-Peer Delay Analysis: Delay characteristics exhibit contrasting patterns between strategies, as illustrated in Figure 4. RNG shows the steepest relative delay increase, starting from approximately 0.05 seconds with 5 robots and reaching 0.37 seconds with 60 robots. However, despite this dramatic rate of change, RNG maintains the lowest absolute delay values throughout the evaluation range and appears to approach an asymptotic limit. This suggests that while RNG experiences the most significant relative scaling challenges, it still provides the best overall delay performance compared to other strategies.

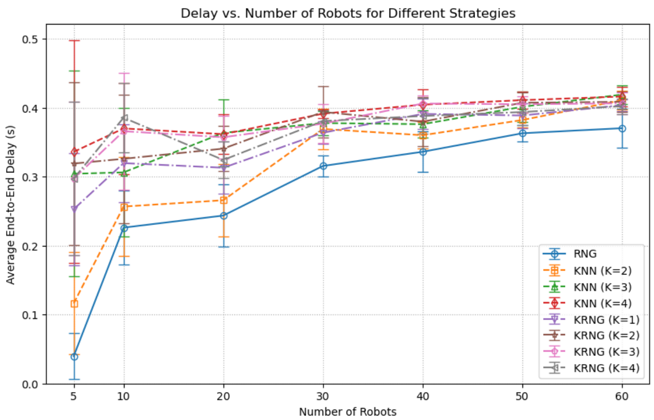

Fig. 4: Average Peer-to-Peer Delay vs. Number of Robots for Different Strategies

KNN strategies maintain moderate delay characteristics, beginning around 0.12-0.18 seconds and increasing to 0.37-0.42 seconds. The K-RNG family demonstrates more stable delay behavior, with higher baseline delays (0.25-0.50 seconds) that remain relatively consistent around 0.37-0.43 seconds across varying network sizes.

3) Strategy-Specific Performance Trade-offs: RNG emerges as the most balanced approach for peer-to-peer communication, offering superior PDR performance while being parameter-free, eliminating the need for connectivity parameter tuning. This characteristic makes RNG particularly attractive for deployment scenarios where optimal parameter selection is challenging or where adaptive behavior is required across varying network conditions.

KNN strategies provide moderate performance across all metrics but suffer from a fundamental limitation: there is no guarantee that KNN graphs can be fully connected, particularly in sparse network configurations or irregular robot distributions. This connectivity uncertainty can lead to network partitioning, severely impacting communication reliability. The k parameter in KNN shows less dramatic impact on performance compared to K-RNG, but requires careful tuning to balance connectivity and overhead.

K-RNG strategies exhibit lower PDR performance despite their geometric foundation. While the lune-based criterion provides theoretical advantages for creating planar graphs, K-RNG is computationally more extensive in calculation than RNG, requiring evaluation of potential interior points within geometric lunes. This computational overhead, combined with the parameter sensitivity of the k value, makes K-RNG less practical for real-time applications where processing efficiency is critical. The higher baseline delays suggest that increased connectivity does not compensate for the additional computational and network overhead in dense deployments.

B. Scenario 2: Data Logging to Base Station Results

1) Packet Delivery Ratio Distribution Analysis: The PDR distribution analysis for multi-hop data collection reveals different performance characteristics than peer-to-peer scenarios, as illustrated in Figure 5. The box plot analysis demonstrates that K-RNG strategies achieve the highest median PDR performance, with K-RNG ($k = 3$) showing the best performance at approximately 32-33%, followed closely by other K-RNG variants and RNG at around 31-32%.

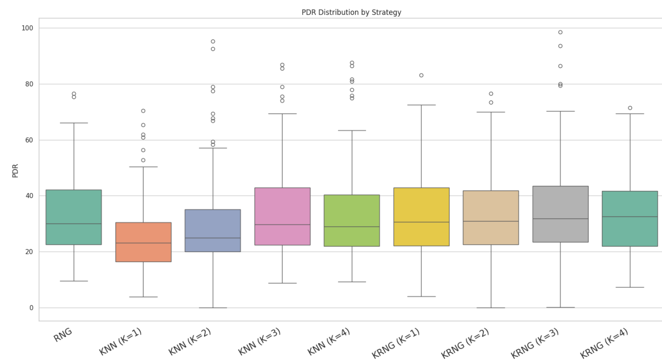

Fig. 5: PDR Distribution by Strategy for Data Collection Scenario

RNG demonstrates consistent performance with a median PDR of approximately 31-32% and relatively compact distribution, indicating stable routing behavior across different network conditions. The parameter-free nature of RNG proves advantageous in multi-hop scenarios where network topology changes dynamically, eliminating the need for real-time parameter adaptation.

KNN strategies exhibit lower median performance compared to K-RNG and RNG, with KNN ($k = 1$) shows the lowest median at approximately 24-25%, KNN ($k = 2$) at 27-28%, KNN ($k = 3$) at 30-31%, and KNN ($k = 4$) at 29-30%. The connectivity challenges inherent in KNN graphs become more pronounced in multi-hop scenarios, where disconnected components can severely impact end-to-end reachability to the base station.

K-RNG strategies demonstrate the most consistent high performance across all k values (31-33%), with K-RNG ($k = 3$) achieving the highest median PDR. Despite the computational overhead compared to RNG, the geometric lune criterion with moderate tolerance for interior points appears to create more robust multi-hop paths in this specific scenario.

2) Multi-hop Routing Performance Characteristics: The geographic forwarding implementation reveals that K-RNG's geometric constraints provide better routing foundation than KNN's purely distance-based approach. The lune-based selection criterion appears to create more robust multi-hop paths, as evidenced by the consistently higher PDR values across all K-RNG variants compared to their KNN counterparts.

The considerable variance observed across all strategies in Figure 5, evidenced by extensive whiskers and outliers reaching up to 70% PDR in optimal conditions, reflects the inherent challenges of multi-hop routing in mobile ad-hoc networks. However, K-RNG strategies show more consistent upper quartile performance, suggesting better resilience to network dynamics despite their computational complexity.

3) Strategy-Specific Multi-hop Analysis: **K-RNG Superior Performance:** K-RNG strategies demonstrate clear advantages in multi-hop scenarios, with all variants achieving median PDR values above 31%. The geometric lune criterion with controlled interior point tolerance appears to create more stable routing topologies compared to pure distance-based selection. K-RNG ($k = 3$) represents the optimal balance, achieving the highest median performance, though at the cost of increased computational overhead compared to parameter-free alternatives.

RNG Competitive Performance: RNG achieves comparable performance to K-RNG strategies (31-32% median PDR) while maintaining its parameter-free advantage and computational efficiency. The sparse but geometrically sound connectivity proves effective for multi-hop routing, making RNG an attractive option when computational resources are limited or parameter tuning is impractical.

KNN Limitations in Multi-hop: KNN strategies consistently underperform in multi-hop scenarios, with even the best variant KNN ($k = 3$) achieving only 30-31% median PDR. Pure distance-based neighbor selection, combined with the lack of connectivity guarantees, appears particularly unsuitable for geographic forwarding scenarios where reliable paths to the base station are essential for successful data collection.

VI. Conclusion

This paper presented a comparative analysis of three dynamic network topology construction strategies for maintaining connectivity while enabling unconstrained exploration in multi-robot systems: RNG, K-RNG, and KNN. Through extensive NS-3 simulations, we evaluated their performance using packet delivery ratio and end-to-end delay metrics across peer-to-peer and multi-hop routing scenarios.

Our results show that RNG provides the best balance of performance and simplicity, achieving superior PDR in peer-to-peer communication (95 to 33% across network densities) while maintaining competitive performance in multi-hop scenarios (31-32% median PDR). Its parameter-free design eliminates the need for tuning, making it particularly suitable for dynamic environments.

K-RNG demonstrates superior performance in multi-hop routing, with K-RNG ($k = 3$) achieving the highest median PDR (32-33%), although at increased computational cost. KNN consistently underperforms across both scenarios due to connectivity limitations and lacks the geometric foundations that make RNG and K-RNG more robust.

For practitioners, RNG offers the optimal choice for most multi-robot applications, while K-RNG should be considered when maximum routing performance justifies the computational overhead. Future work should investigate these strategies under realistic environmental conditions and explore adaptive parameter selection mechanisms.

Acknowledgment

This work has been partially supported by the European Union's Horizon Europe research and innovation project under grant agreement No. 101070487 (NEPHELE).

References

[1] M. Saboia, L. Clark, V. Thangavelu, J. A. Edlund, K. Otsu, G. J. Correa, V. S. Varadharajan, A. Santamaria-Navarro, T. Touma, A. Bouman, H. Melikyan, T. Pailevanian, S.-K. Kim, A. Archanian, T. S. Vaquero, G. Beltrame, N. Napp, G. Pessin, and A.-a. Agha-mohammadi, "Achord: Communication-aware multi-robot coordination with intermittent connectivity," *IEEE Robotics and Automation Letters*, 2022.

[2] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, "Cooperative mobile robotics: antecedents and directions," *Autonomous Robots*, vol. 4, no. 1, pp. 7–27, 1997.

[3] H. Chaabi and N. Mitton, "Distributed Multi-Robot Exploration Approach With Connectivity Maintenance," in *DCOSS-IoT 2025 - 21st Annual International Conference on Distributed Computing in Smart Systems and the Internet of Things*, Lucca, Italy, 2025.

[4] A. Shetty, T. Hussain, and G. Gao, "Decentralized connectivity maintenance for multi-robot systems under motion and sensing uncertainties," *Navigation, Journal of the Institute of Navigation*, 2023.

[5] S. Kloder and S. Hutchinson, "Path planning for fixed-wing UAVs using L+/L- path segments," *IEEE Transactions on Robotics*, 2006.

[6] Z. Yan, N. Jouandeau, and A. Chérif, "A survey and analysis of multi-robot coordination," *International Journal of Advanced Robotic Systems*, vol. 10, no. 12, p. 399, 2013.

[7] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009.

[8] A. Jalil, J. Kobayashi, and T. Saitoh, "Performance improvement of multi-robot data transmission in aggregated robot processing architecture with caches and QoS balancing optimization," *Robotics*, 2023.

[9] B. Capelli, H. Fouad, G. Beltrame, and L. Sabattini, "Decentralized connectivity maintenance with time delays using control barrier functions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 1586–1592.

[10] Y. Yang, Y. Lyu, Y. Zhang, S. Yi, and W. Luo, "Decentralized multi-robot line-of-sight connectivity maintenance under uncertainty," *Robotics: Science and Systems (RSS)*, 2024.

[11] R. Abu-Aisheh, F. Bronzino, L. Salaün, and T. Watteyne, "Cara: Connectivity-aware relay algorithm for multi-robot expeditions," *Sensors*, vol. 22, no. 23, p. 9042, 2022.

[12] N. Mazloomi, Z. Zandinejad, A. Zaretalab, and M. Gholipour, "Multi-objective qos optimization in swarm robotics," *Robotics and Autonomous Systems*, vol. 182, p. 104796, 2024.

[13] R. Luo, W. Ni, H. Tian, and J. Cheng, "Federated deep reinforcement learning for ris-assisted indoor multi-robot communication systems," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 12 321–12 326, 2022.

[14] G. T. Toussaint, "The relative neighbourhood graph of a finite planar set," *Pattern Recognition*, vol. 12, no. 4, pp. 261–268, 1980.

[15] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.

[16] T.-H. Su and R.-C. Chang, "Computing the k-relative neighborhood graphs in euclidean plane," *Pattern Recognition*, 1991.

[17] M.-S. Chang, C. Y. Tang, and R. C. T. Lee, "Solving the euclidean bottleneck matching problem by k-relative neighborhood graphs," *Algorithmica*, 1992.

[18] A. Atias, K. Solovey, O. Salzman, and D. Halperin, "Effective metrics for multi-robot motion-planning," in *Robotics: Science and Systems XIII*, 2017.