International Journal on Mechanical Engineering and Robotics

ISSN: 2321-5747

Volume 15 Issue 1 (2026) | Pages: 15-22

Connectivity-Aware Reactive Path Planning for Small Multi-UAV Teams: A Reproducible Simulation Study with Fair Baselines and Link-Level Evaluation

Kamal Khokhar[1], Sandeep Kumar Tiwari[2]

Peer Review Information

Type: *Article*
Received: *17 Aug 2026*
Revised: *16 Sept2026*
Accepted: *20 Oct 2026*
Published: *19 Nov 2026*

Abstract

Small UAV teams must reach individual goals while keeping their radio network connected. This paper asks how much a lightweight, reactive correction that protects the algebraic connectivity λ_2 of the communication graph can add to a standard potential-field controller. Five controllers are compared on identical random scenarios: a geometric potential-field controller, a rigid-formation controller, an always-on cohesion controller, a Fiedler-vector gradient controller (Grad-λ_2), and the communication-aware (CA) controller, which blends goal and neighbour-centroid directions when predicted λ_2 falls below a threshold. All controllers share the same obstacle and inter-UAV repulsion, so differences are attributable to the connectivity logic alone. Each cell uses 100 paired Monte-Carlo runs, with Wilson and t-based confidence intervals and Holm-corrected paired tests. With a 40 m communication threshold, connectivity is never the binding constraint and the controllers are statistically indistinguishable. In a stressed regime (22 m threshold, dispersed goals, velocity noise) the CA controller raises connectivity uptime from 0.634 to 0.727 (paired $p < 0.001$) and the minimum λ_2 from 0.05 to 0.07,
at a cost of 0.5 m extra path, but its success-rate gain (0.55 vs 0.48) is not significant after multiplicity correction, and it does not reduce inter-UAV proximity events (4.9 vs 5.0). The Grad-λ_2 baseline outperforms CA (uptime 0.815, success 0.69). An earlier reported 45-fold reduction in proximity events is traced to an asymmetry in inter-UAV repulsion, not to the connectivity term: removing repulsion raises proximity events about tenfold for every controller. A post-hoc SNR-based link evaluation, measured runtimes, and an explicit list of limitations are provided. Code and raw per-run data are released.

Keywords: Multi-UAV Systems; Algebraic Connectivity; Reactive Path Planning; Potential Fields; Connectivity Maintenance; Reproducible Simulation

How to Cite This Article

Khokhar, K., & Tiwari, S. K. (2026). Connectivity-aware reactive path planning for small multi-UAV teams: A reproducible simulation study with fair baselines and link-level evaluation. *International Journal on Mechanical Engineering and Robotics,* 15(1), 15–22.

2321-5747/ © 2026 The Authors. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0).

Introduction

UAV teams used for search and rescue, surveillance, mapping or ad-hoc aerial relays must reach spatial objectives and keep an inter-vehicle network that carries coordination traffic and, often, a back-haul link. The two requirements conflict: goal-seeking motion stretches the team, whereas rigid formations preserve links at the price of progress. Recent work addresses this with joint energy and connectivity-radius optimisation validated on three real UAVs [1] and communication-aware distributed trajectory optimisation with asynchronous ADMM [2]. A long line of control work maintains algebraic connectivity with decentralised estimators and gradient or barrier methods [3, 4, 5, 7].

This paper does not compete with those frameworks. It isolates a narrower question: when, and how much, does a simple local connectivity-protection rule help a reactive controller, once all baselines are given the same safety layers and the same random scenarios? The study was revised after review found that the earlier draft (a) compared against a baseline lacking inter-UAV repulsion, (b) reported inconsistent run counts, (c) lacked statistical analysis and (d) overstated both novelty and networking content.

Contributions.

1. A fair, paired experimental protocol: identical safety layers for all controllers, common random numbers, 100 runs per cell, confidence intervals and Holm-corrected tests.
2. Two regimes, nominal and stressed, showing that the benefit of connectivity protection appears only when connectivity is binding.
3. Two additional baselines (always-on cohesion; Fiedler-gradient connectivity control) and ablations that separate the effect of the connectivity term from that of inter-UAV repulsion.
4. Sensitivity, noise-robustness, scalability (N = 3 to 20) and measured computation-time results.
5. A post-hoc SNR-based link evaluation with multi-hop delivery ratio, clearly separated from the planning model.
6. Release of code, raw per-run results and figures.

Related Work

Communication-aware multi-UAV planning. Samshad and Rajawat [1] jointly minimise energy and the inter-UAV connectivity radius for coverage; three-UAV flight tests reported close agreement between predicted and observed communication range. Yu, Zheng and He [2] combine parameterised differential dynamic programming with asynchronous ADMM under limited, delayed communication. These are optimisation-based and substantially more capable than the reactive controller studied here; no performance comparison with them is made or implied.

Connectivity maintenance. Algebraic connectivity, the second-smallest Laplacian eigenvalue [4], is a standard robustness measure. Decentralised estimation and gradient control of λ_2 [5], graph-theoretic connectivity control [6], and ADMM-based planning under uncertainty [3] are established. The Grad-λ_2 baseline here follows the Fiedler-vector gradient idea of [5] in a simplified, centralised form. Control barrier functions [7] provide formal safety guarantees and are an important baseline left for future work.

Reactive control. Artificial potential fields [8] and consensus/cohesion rules [9] form the geometric and cohesion baselines.

Novelty statement. Blending goal and neighbour-centroid directions under a λ_2 trigger is a simple instance of known ideas. The contribution is empirical: a transparent, fair quantification of when it helps and when it does not.

System and Communication Models

Dynamics

N = 6 UAVs (varied in Section 7.4) move in a 100 m × 100 m plane with single-integrator kinematics $p_i(k+1) = p_i(k) + v_i(k)\Delta t$, $\|v_i\| \le v_{max} = 6$ m/s, $\Delta t = 0.4$ s. A zero-mean Gaussian velocity disturbance of standard deviation σ per axis is added ($\sigma = 1$ m/s in the stressed regime), followed by re-saturation.

Planning Communication Model

Links exist when $d_{ij} \le d_{th}$, with weight $w_{ij} = \max(0.05, 1 - (d_{ij}/d_{th})^{1.2})$. The weighted Laplacian L has algebraic connectivity $\lambda_2(L)$; $\lambda_2 > 0$ if and only if the graph is connected [4]. Thresholds: protection $\lambda_{min} = 0.20$, uptime 0.22, success 0.10. These are design constants; Section 7.3 sweeps λ_{min}.

Obstacles And Safety

Circular obstacles of radius 6 m. Open: (50,50). Default: (38,38), (55,40), (40,58). Dense: default plus (60,60) and (45,45). A proximity event is one UAV pair closer than 5 m at one time step; reported as the mean count per episode. An obstacle contact (a step that would enter an obstacle and is cancelled) is counted separately.

Link-Level Evaluation Model (Post-Hoc Only)

Controllers never see this model. Per-link SNR = $S_{40} - 10 \cdot n \cdot \log_{10}(d/40) - P_NLoS \cdot$ [obstacle blocks the segment] + X, with S_{40} = 12 dB at 40 m, n = 2.5, P_NLoS = 12 dB, X ~ N(0, 4^2 dB) fixed per link per run. Packet delivery probability is a logistic function of SNR centred at 6 dB with slope 1.5 dB. The network delivery ratio (NDR) is the mean, over ordered UAV pairs and time steps, of the best multi-hop delivery probability (product of hop probabilities, no retransmissions). NDR is a simple analytical proxy, not a packet-level simulation.

Controllers

Every controller adds the same obstacle repulsion (influence radius 14 m from centre, with a tangential swirl component to avoid symmetric deadlocks) and the same inter-UAV repulsion (influence 7 m, full strength at 5 m), then saturates at v_max.

- Geometric (G): attraction $v = \min(v_{max}, \|g - p\|)$ toward its own goal, plus repulsions.
- Rigid (R): UAV 0 is leader moving to the centroid of the goals at 0.85 v_max; others track the leader plus their initial offset.
- Cohesion (C): G plus an always-on weighted pull toward connected neighbours (gain 0.25).
- Grad-λ_2 ($\nabla\lambda$): G plus gain 120 times the gradient of λ_2 with respect to the UAV position (Fiedler-vector formula), scaled by $\max(0, (0.4 - \lambda_2)/0.4)$.
- Communication-aware (CA): compute the nominal velocity v_nom of G; form the trial position $p + v_nom \cdot \Delta t$ with other UAVs held at their current positions; if trial $\lambda_2 < \lambda_min$ and current $\lambda_2 \geq 0.5 \cdot \lambda_min$, replace the goal direction by a blend $0.65 \cdot \hat{u}_goal + 0.35 \cdot \hat{u}_centroid$ (unit vectors; centroid of currently linked neighbours) and re-apply the repulsions.

Algorithm 1 (CA step for UAV i). (1) v_nom ← G command. (2) p_cand ← $p_i + v_nom \cdot \Delta t$. (3) $\lambda_trial \leftarrow \lambda_2$ with p_i replaced by p_cand. (4) If $\lambda_trial < \lambda_min$ and $\lambda_2(current) \geq 0.5 \cdot \lambda_min$ and i has neighbours: $b \leftarrow 0.65 \cdot \hat{u}_goal + 0.35 \cdot \hat{u}_centroid$; $v \leftarrow b/\|b\| \cdot \max(1, \min(v_{max}, \|v_{nom}\|))$; add obstacle and UAV repulsion; saturate. (5) Otherwise v ← v_nom. (6) Any step entering an obstacle is cancelled.

Properties and cost. The rule gives no formal connectivity guarantee: the lookahead moves one UAV while the others are frozen, so simultaneous moves can still break the graph; and because the rule is disabled once $\lambda_2 < 0.5 \cdot \lambda_min$ it cannot recover a collapsed network. Each check needs a dense symmetric eigenvalue computation, which is $O(N^3)$ in a centralised implementation (the earlier claim of cost linear in the number of neighbours was incorrect). A decentralised λ_2 estimator [5] would be needed for a truly local implementation. Measured times are in Section 7.5.

Experimental Design

Table 1. *Simulation parameters.*

Parameter	**Value**
UAVs N	6 (3, 10, 20 in scalability)
Workspace	100 m × 100 m
v_max, Δt, horizon	6 m/s, 0.4 s, 220 steps
Start region	square of side $\max(16, 6.5\sqrt{N})$ m around (20,20), min. separation 7.5 m
Goals	square of side $\max(24, 9\sqrt{N}) \cdot s_g$ m around (80,80), min. separation 8 m, random one-to-one assignment
Nominal regime	$d_th = 40$ m, $s_g = 1.0$, $\sigma = 0$
Stressed regime	$d_th = 22$ m, $s_g = 1.8$, $\sigma = 1$ m/s
Goal tolerance	10 m
Safety distance	5 m
Runs per cell	100 (ablation 100; sensitivity, noise 60; scalability 30)

Termination and metrics. An episode ends when every UAV is within 10 m of its own goal or after 220 steps. Success requires all UAVs within tolerance and $\lambda_2 > 0.10$ at that moment. Connectivity uptime is the fraction of executed steps with $\lambda_2 \geq 0.22$. Also reported: mean and minimum λ_2, path length per UAV, proximity events per episode, NDR, and time to goal.

Design decisions made before the final runs. The two regimes and the obstacle layouts were fixed after a small exploratory pilot (12 runs per layout) whose only purpose was to find settings where the problem is non-trivial (the nominal regime was found to be trivial). No controller parameters were tuned on the evaluation seeds; Grad-λ_2 and cohesion gains received one coarse sanity setting each and are

therefore probably not optimal.

Statistics. Scenario seed = 10000 + run index, identical across controllers (paired design). Success: Wilson 95% intervals and exact McNemar tests. Continuous metrics: t-based 95% intervals and paired Wilcoxon signed-rank tests. Holm correction over 72 CA-vs-baseline comparisons (2 regimes × 3 environments × 2 comparators × 6 metrics).

Main Results

Nominal regime (d_th = 40 m)

Table 2. *Nominal regime, 100 runs per cell. Rigid achieves 0.00 success in every environment because a formation that follows one leader cannot place each UAV within 10 m of its own, randomly assigned goal. Dense: all reactive controllers become trapped between obstacles (no UAV reaches its goal).*

Env.	Controller	Success (95% CI)	Uptime	Min λ_2	Path (m)	Prox. events
Open	G / ∇λ / CA	1.00 (0.96–1.00)	1.000	2.80	91	4.5
Open	Rigid	0.00	1.000	3.50	100	4.6
Open	Cohesion	0.87 (0.79–0.92)	0.992	0.49	205	186.1
Default	G	0.96 (0.90–0.98)	0.965	2.29	94	6.1
Default	CA	0.97 (0.92–0.99)	0.973	2.29	94	6.1
Default	∇λ	0.96 (0.90–0.98)	0.965	2.29	94	6.0
Default	Rigid / Cohesion	0.00 / 0.90	0.953 / 0.957	2.89 / 0.33	112 / 185	11.2 / 168.0
Dense	G / ∇λ / CA	0.00 for all	0.18 / 0.18 / 0.19	≈ 0.01	71–78	≈ 99

With a 40 m threshold connectivity is not binding: G, ∇λ and CA are identical in open space and differ by less than one run in 100 in the default environment. In this regime the protection term has nothing to protect, and the earlier draft's gains cannot be reproduced.

Stressed regime (d_th = 22 m, dispersed goals, σ = 1 m/s)

Table 3. *Stressed regime, 100 paired runs per cell (Open/Dense: rigid and cohesion omitted for space; rigid success 0.00, cohesion 0.65 open / 0.00 dense, with proximity counts of 62.9 and 155.3).*

Env.	Controller	Success (95% CI)	Uptime ± CI	Mean λ_2	Min λ_2	Path (m)	Prox.	NDR
Default	Geometric	0.48 (0.38–0.58)	0.634 ± 0.044	0.71	0.05	97	5.0	0.997
Default	Rigid	0.00 (0.00–0.04)	0.993	1.97	0.85	222	10.8	1.000
Default	Cohesion	0.66 (0.56–0.75)	0.882	1.92	0.25	118	55.6	0.998
Default	Grad-λ_2	0.69 (0.59–0.77)	0.815	0.77	0.12	97	4.5	0.997
Default	**Comm-Aware**	0.55 (0.45–0.64)	0.727 ± 0.040	0.74	0.07	97	4.9	0.997
Open	Geometric	0.53	0.749	0.92	0.07	94	2.9	1.000
Open	Grad-λ_2	0.72	0.887	0.97	0.14	94	2.7	1.000
Open	**Comm-Aware**	0.58	0.816	0.93	0.09	94	3.2	1.000
Dense	Geometric	0.06	0.113	0.15	0.00	160	39.0	0.772
Dense	Grad-λ_2	0.02	0.108	0.13	0.00	163	55.5	0.750
Dense	**Comm-Aware**	0.07	0.113	0.15	0.00	161	46.6	0.762

Statistical findings (CA versus Geometric, default environment). Uptime +0.092 (paired Wilcoxon, $p < 0.001$, survives Holm); minimum λ_2 +0.025 ($p < 0.001$); success +0.07 (exact McNemar raw $p = 0.016$, Holm-adjusted $p = 0.69$, not significant); path length +0.5 m; proximity events −0.14 ($p = 0.61$, no effect). In the open environment the uptime gain is +0.067 ($p < 0.001$).

CA versus Grad-λ_2. Grad-λ_2 is better: uptime +0.088 and success +0.14 in the default environment (Holm-adjusted $p = 0.0065$ for success), at the same path length and proximity count. A simple always-on cohesion rule raises uptime and success even more than CA, but at roughly tenfold more proximity events and 20% longer paths.

Dense environment. No reactive controller reaches its goals reliably (success ≤ 0.07). The failure is geometric trapping (hundreds of cancelled obstacle contacts per run), not connectivity: all connectivity-aware variants are statistically equal to the baseline. This confirms that cluttered scenes need a global or receding-horizon planner.

Link-level view. Because the link model is calibrated to 12 dB SNR at 40 m, the effective radio range is much longer than the 22 m planning threshold, so low planning-model λ_2 rarely caused delivery loss in the open and default environments (NDR ≥ 0.997 for all controllers except trapped dense runs, where NDR falls to 0.75–0.77 for G, ∇λ and CA against 0.89 for Rigid). Hence the planning-model uptime gains should not be read as delivery-ratio gains. Verifying that the planning threshold is a safe proxy for a real radio requires a calibrated channel model and hardware data.

Figures 1 to 4 show representative trajectories (scenario seed 10000, default environment, stressed regime; the run was fixed in advance, not selected), the corresponding λ_2 traces, the main metrics with confidence intervals, and the scalability, noise and runtime results.

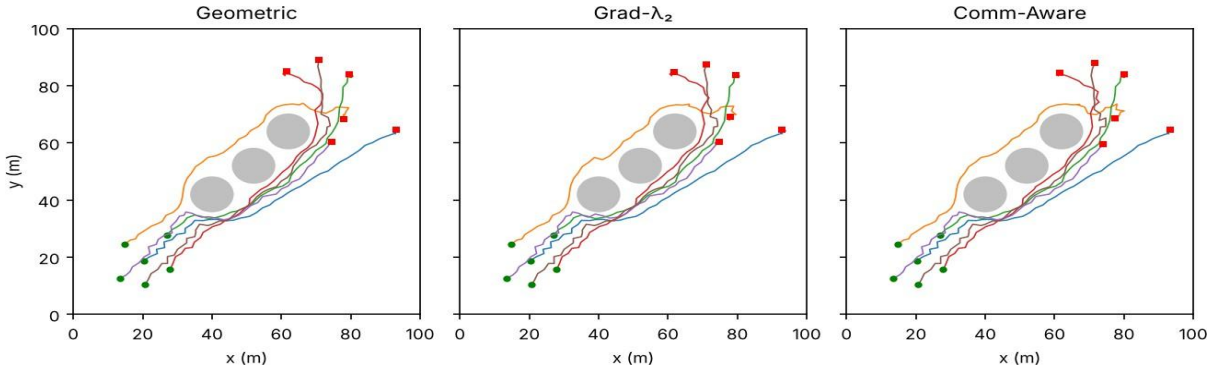


Fig. 1. *Trajectories of the Geometric, Grad-λ_2 and Comm-Aware controllers (green = start, red = final position, grey = obstacles); stressed regime, default environment, seed 10000.*

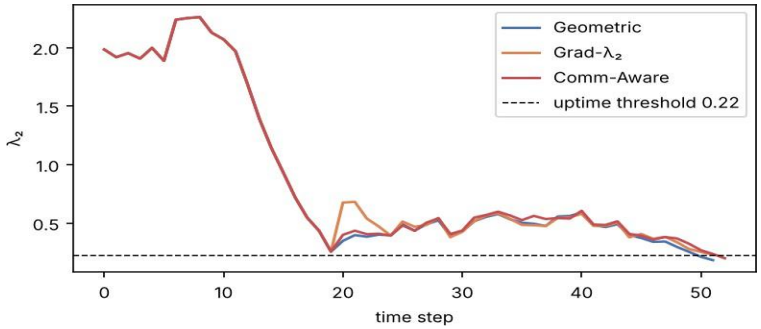


Fig. 2. *Algebraic connectivity λ_2 over time for the same run; dashed line = uptime threshold 0.22.*

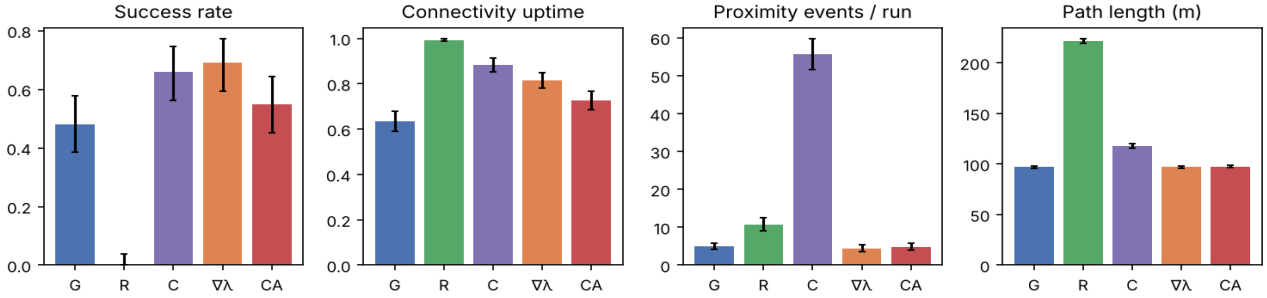


Fig. 3. *Stressed regime, default environment, 100 runs: success rate (Wilson 95% CI), uptime, proximity events per run and path length (t-based 95% CI). G = Geometric, R = Rigid, C = Cohesion, ∇λ = Grad-λ_2, CA = Comm-Aware.*

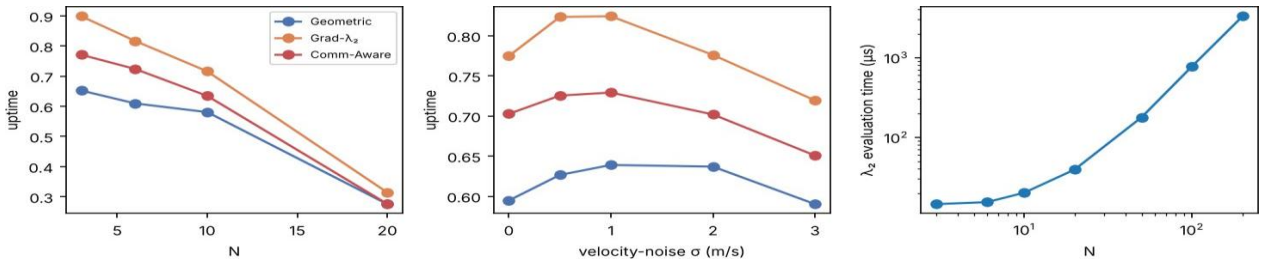


Fig. 4. *Left: uptime versus team size N. Centre: uptime versus velocity-noise level (60 runs). Right: runtime of one λ_2 evaluation versus N (log-log).*

Ablation, Sensitivity, Robustness, Scalability and Cost

Ablation (stressed, default environment, 100 runs)

Table 4. *Ablation. Setting λ_min = 0 disables the trigger and reproduces the geometric controller exactly, so it is not listed separately.*

Variant	**Success**	**Uptime**	**Prox. events**	**Min λ_2**
Geometric (with repulsion)	0.48	0.634	5.0	0.05
CA full (with repulsion)	0.55	0.727	4.9	0.07
CA, goal-weight 1.0 (blend neutralised)	0.48	0.631	4.8	0.05
Geometric, **no** inter-UAV repulsion	0.53	0.612	**48.0**	0.04
CA, **no** inter-UAV repulsion	0.56	0.724	**49.5**	0.08

The connectivity term accounts for the uptime gain (+0.09, $p < 0.001$, both with and without repulsion) and a modest success gain. Removing inter-UAV repulsion multiplies proximity events by about ten for both controllers ($p < 10^{-17}$). The earlier headline (423.9 versus 9.3 proximity events) is therefore consistent with an unequal-safety-layer comparison and is withdrawn; the original code was unavailable, so this attribution is an inference from the ablation, not a re-run.

Blend-weight sensitivity (stressed, default, 60 runs)

Goal weight	**1.0**	**0.8**	**0.65**	**0.5**	**0.3**
Uptime	0.635	0.687	0.729	0.784	0.830
Success	0.52	0.52	0.57	0.73	0.78
Path (m)	97	97	97	108	133

Stronger centroid pull monotonically improves uptime and success but lengthens paths (up to +37% at weight 0.3). The default 0.65/0.35 is a conservative point, not an optimum.

Threshold, range and speed sensitivity

λ_min	**0.05**	**0.1**	**0.2**	**0.4**	**0.8**
Uptime	0.646	0.665	0.729	0.786	0.804
Min λ_2	0.050	0.056	0.074	0.089	0.120

d_th (m)	**18**	**22**	**30**	**40**
Success: CA / G	0.23 / 0.15	0.57 / 0.50	0.98 / 0.98	1.00 / 1.00
Uptime: CA / G	0.45 / 0.38	0.73 / 0.64	0.98 / 0.97	1.00 / 1.00

The benefit of protection shrinks to zero as the radio range grows, matching Section 6.1. Varying v_max (4, 6, 8 m/s) changes uptime by less than 0.01 and proximity events by about 2 (fewer at lower speed).

Noise robustness and scalability (60 and 30 runs)

σ (m/s)	**0**	**0.5**	**1**	**2**	**3**
Uptime G	0.595	0.627	0.639	0.637	0.590

Uptime ∇λ	0.775	0.823	0.824	0.776	0.719
Uptime CA	0.703	0.725	0.729	0.702	0.651

N	**3**	**6**	**10**	**20**
Uptime G / ∇λ / CA	0.65 / 0.90 / 0.77	0.61 / 0.82 / 0.72	0.58 / 0.72 / 0.63	0.28 / 0.31 / 0.28
Success G / ∇λ / CA	0.53 / 0.80 / 0.60	0.50 / 0.70 / 0.60	0.47 / 0.63 / 0.47	0.00 / 0.03 / 0.00
Prox. events (CA)	0.7	3.7	16.9	59.1

The ranking G < CA < ∇λ holds for noise up to 3 m/s and for N ≤ 10; the gain of CA shrinks with N. At N = 20 only half the UAVs reach their goals within the horizon for every controller (crowding at the goal region), so no connectivity benefit can appear. Scalability beyond 20 UAVs, wind, UAV failure and communication dropout were not tested.

Computation time

One λ_2 evaluation (NumPy symmetric eigensolver, single CPU core) takes 15 µs (N = 6), 39 µs (N = 20), 178 µs (N = 50), 0.78 ms (N = 100) and 3.3 ms (N = 200). CA needs one evaluation per UAV per step, so a centralised step costs about N such calls. This is negligible for N ≤ 20 but grows roughly cubically per call at large N, which supports using a distributed estimator for large swarms.

Discussion

1. Connectivity protection matters only when connectivity binds. With a generous radio range it has no effect; with a short range it measurably raises uptime.
2. The effect is real but modest, and not a safety technique. CA adds roughly 0.07 to 0.09 uptime and a non-significant success gain; inter-UAV safety comes from repulsion, not from the connectivity term.
3. A better-founded baseline wins. Gradient ascent on λ_2 outperforms centroid blending at no extra path or proximity cost, as expected from established connectivity-control theory [5]. The practical value of the CA rule is its simplicity.
4. Reactive methods fail in clutter, independent of connectivity handling.
5. Planning-model connectivity is not network performance. In these scenarios the distance-threshold model was conservative relative to the SNR model; calibration against measured links (such as the three-UAV flight data in [1]) is required before any claim about real networks.

Limitations. 2-D single-integrator kinematics; distance-threshold planning model; static circular obstacles; one start/goal geometry; centralised λ_2 evaluation; one coarse gain setting for each baseline; centroid blend has no formal guarantee; NDR is analytical, not a packet-level (e.g. NS-3) simulation; no routing, MAC or interference modelling; no hardware or software-in-the-loop validation; obstacle layouts were selected in a small pilot. The study makes no claim of superiority over MPC, ADMM or CBF methods.

Conclusion and Future Work

Under a fair, paired protocol, centroid-blending connectivity protection raises connectivity uptime when the radio range is short, has no effect when it is long, does not reduce proximity events, and is outperformed by a Fiedler-gradient controller. Future work: (i) CBF-based and distributed-MPC baselines;

(ii) a calibrated SNR/line-of-sight model inside the planner and a packet-level NS-3 evaluation with routing; (iii) a global or receding-horizon layer for dense scenes; (iv) 3-D quadrotor dynamics, dynamic obstacles, UAV failures and dropouts; (v) a PX4/Gazebo software-in-the-loop test and flight validation; (vi) public urban multi-UAV datasets and FANET traces for benchmarking.

Reproducibility. Released files: uav_sim.py (simulator), run_all.py (campaign), analysis.py (statistics and figures), results.json (all per-run raw data), stats.json (tables and tests), four figures. Python 3.12, NumPy 2.5, SciPy 1.18, Matplotlib 3.11; about 11 minutes on one CPU core. Seeds are 10000 + run index. Repository/DOI: to be added by the authors before submission.

References

1. Samshad, M., & Rajawat, K. (2024). Communication and Energy-Aware Multi-UAV Coverage Path Planning for Networked Operations. arXiv:2411.02772.
2. Yu, Y., Zheng, X., & He, S. (2025). Communication-Aware Asynchronous Distributed Trajectory Optimization for UAV Swarm. arXiv:2511.14994.
3. Shetty, A., Knowles, D., & Gao, G. X. (2021). Connectivity Maintenance for Multi-Robot Systems Under Motion and Sensing Uncertainties Using Distributed ADMM-based Trajectory Planning. arXiv:2012.09808.

4. Fiedler, M. (1973). Algebraic connectivity of graphs. Czechoslovak Mathematical Journal, 23(2), 298–305.
5. Yang, P., Freeman, R. A., Gordon, G. J., Lynch, K. M., Srinivasa, S. S., & Sukthankar, R. (2010). Decentralized estimation and control of graph connectivity for mobile sensor networks. Automatica, 46(2), 390–396.
6. Zavlanos, M. M., Egerstedt, M. B., & Pappas, G. J. (2011). Graph-theoretic connectivity control of mobile robot networks. Proceedings of the IEEE, 99(9), 1525–1540.
7. Ames, A. D., Xu, X., Grizzle, J. W., & Tabuada, P. (2017). Control barrier function based quadratic programs for safety critical systems. IEEE Transactions on Automatic Control, 62(8), 3861–3876.
8. Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots. International Journal of Robotics Research, 5(1), 90–98.
9. Olfati-Saber, R., & Murray, R. M. (2004). Consensus problems in networks of agents with switching topology and time-delays. IEEE Transactions on Automatic Control, 49(9), 1520–1533.