Information, Computing and Intelligent Systems, 2026, No. 8, 67 – 80
UDC 004.896:681.51 https://doi.org/10.20535/2786-8729.8.2026.357485

DECENTRALIZED LEADER ELECTION PROTOCOL FOR UNMANNED GROUND VEHICLE SWARMS IN DYNAMIC ENVIRONMENTS

Myroslav Rudnytskyi*
https://orcid.org/0000-0002-6166-1078

Iryna Klymenko
https://orcid.org/0000-0001-5345-8806

National Technical University of Ukraine
“Igor Sikorsky Kyiv Polytechnic Institute”, Kyiv, Ukraine

*Corresponding author: m.rudnytskyi@kpi.ua

Received: 14 April 2026 / Accepted: 11 May 2026 / Published: 28 May 2026

Unmanned ground vehicle swarms are gaining relevance as the foundation of autonomous systems capable of effectively operating in dynamic conditions. However, reliability issues constrain their widespread adoption. Traditional navigation algorithms based on a static leader model create a single point of failure, increasing formation vulnerability to hardware failures, depletion of energy resources, and unpredictable terrain changes. This paper investigates the process of autonomous swarm navigation based on a two-dimensional environment simulation model. The research aims to improve efficiency and fault tolerance by developing a decentralized protocol for swarm leader election. The proposed protocol is integrated into a hybrid navigation method, described in the authors' previous works. It combines planning (using A* algorithm and artificial potential fields) and stabilization (hysteresis and artificial vortex field). The scientific novelty is the multi-parameter utility function for the protocol, which considers the vehicles' energy, terrain traversal cost, proximity to the goal, and a safe distance to obstacles. Integrating the protocol into the specified hybrid method ensures an adaptive role distribution in the swarm within a dynamic environment. The simulation findings confirmed that the developed protocol's integration into the hybrid navigation method increases fault tolerance. It enables the swarm to overcome local minima during leader failures under moderate communication interference (up to 60% packets lost). Experiments indicate that transferring control to followers in better conditions increases efficiency by reducing the total time for passing high-cost traversal zones. Comparative analysis demonstrates its advantage over a fixed-leader approach: for a 5-vehicle swarm, the total traversal time is reduced by 3% and the formation mean squared error by 8%.

Keywords: unmanned ground vehicles, decentralized leader election protocol, energy efficiency, dynamic environment, autonomous swarm navigation.

1. Introduction

Unmanned ground vehicles (UGVs) swarms are groups of autonomous ground robots that perform a common task, exchanging data and coordinating their trajectories and actions, similar to a flock in nature [1]. As established a prior study [2], the core benefit of decentralized swarms is their capacity to form sophisticated global patterns through basic, individual-level interactions. Consequently, a central control node is not required for system operation [2].

A review of the literature indicates that UGV swarms are used for large, complex, or dangerous tasks. These include search and rescue, mine clearance, toxic spill cleanup, and environmental monitoring with various sensor sets. Simultaneous spatial coverage, fault tolerance, and decentralized decision-making are important traits for such types of applications [3].

As a result, the operation of UGV swarms requires a leader to set common course and formation. The leader coordinates individual robots actions to ensure effective area coverage and obstacle avoidance by the entire swarm as a single, integrated system [4].

Despite considerable progress in recent years, many existing approaches to swarm leader elections rely on static methods. Nevertheless, their application potential is constrained in real-world situations that are very dynamic due to unexpected situations during mission execution. This clearly demonstrates the need to improve the current algorithms that are not only efficient but also fault-tolerant and adaptable.

Therefore, the importance of this study is outlined by the necessity to connect the gap between the theoretical swarm intelligence concepts and the actual demands of the dynamic environments. This research focuses on the enhancement of existing approaches to make them capable of operating under constantly changing conditions that improve the efficiency and reliability of current assignments.

2. Literature review and problem statement

Unmanned ground vehicles swarm is usually controlled using two main approaches: centralized and distributed [5]. For centralized architectures control is originated from single central point. While this approach provides global optimization and high-quality decisions, it suffers from inflexibility, large computational and communicational costs and tends to be vulnerable for single point of failure [6]. In contrast, distributed architecture provides approach, where every vehicle operates autonomously and makes independent decisions, that reduces computational costs especially when the swarm contains large number of vehicles [5].

Leader-follower approach takes special place in this field. While it is still structurally centralized (with a designated leader), but many systems perform leader election or replacement using distributed protocols, making it de facto hybrid. For example, unmanned aerial vehicles (UAVs) swarm, described in work [5] dynamically identifies leaders during operation and switch to leader-tracking navigation autonomously. Similar ideas appear in dynamic leader election for collective tracking and formation maintenance, where the leader identity is treated as a controllable variable over time [7]. In Cooperative Autonomous Distributed Robotic Exploration (CADRE's) lunar rovers, leader election is continuous and resilient to network changes, again blurring centralization with distributed decision making [8].

Many existing papers confirm leader-follower pattern as highly effective formation-control architecture. A comprehensive UGV review highlights this approach as a major application class, alongside lane tracking and obstacle avoidance [5]. Formation-centric coverage planning with deformable virtual leader-follower formations shows how this approach reduces complexity compared with allocating independent regions, while preserving robustness and communication advantages in GPS-challenged environments [9].

Common practice includes the leader's trajectory is first planned (both on global and local levels) and followers only track relative positions [2]. A hybrid UGV swarm planner uses A* for global path planning and artificial potential fields (APF) for local one, supporting linear and V-shaped formations. Results show that this combination both respects formation and avoids APF local minima. However, the study also highlights that dynamic obstacles could stress formation scalability alongside with method limitation in case of narrow passages [2]. As alternative to APF, article [10] utilizes an improved Dynamic Window Approach (DWA). This method incorporates formation-keeping and direction-coordination terms to penalize any deviation from desired formation geometry. Simulations demonstrate smaller formation-position deviation near obstacles and faster, more accurate avoidance than classical APF formation control [10].

Several works couple formation maintenance and collision avoidance. For example, a Robot Operating System (ROS)-based formation controller uses a *Lyapunov* backstepping law for formation plus a limit-cycle obstacle avoidance module to ensure safe reconfiguration during leader failure and role switching [11]. Work on formation morphing further integrates avoidance and morphing so that formations deform around obstacles and then are reconstructed, including energy-efficient recovery via thin-plate splines [12]. In this paper, leadership is effectively zonal: a global leader coexists with secondary leaders. These leaders temporarily guide local subgroups when communication or obstacle layout demands it, providing local decision centers in different parts of the workspace [12].

Several papers note the vulnerability of static leaders. For example, this work stresses that most prior studies “have not even considered the possibility of leader failure” and therefore proposes explicit leader replacement when a failure condition is detected [11]. Other study explicitly points out that loss of the dedicated leader or its communication causes failure of the rest of the swarm, motivating temporary and secondary leaders [12]. But such mechanisms are largely reactive: leader change occurs only after failure or disconnection, which naturally incurs time for regrouping and transient formation loss. In the CADRE lunar mission, periodic re-election is introduced mainly as a fail-safe against hardware or network issues, functioning rather than as a predictive, topology-aware reconfiguration [8].

Some works move beyond simple leader election in case of previous one failure toward persistent, context-aware leadership. Leader election for collective tracking defines a performance metric depending on the chosen leader and the communication graph’s properties. As a result, swarm then periodically changes leader to maximize convergence speed to the reference velocity and formation [7]. The algorithm is explicitly decentralized and operates continuously.

CADRE’s architecture separates leader election from leader appointment with a unique *“appointer”*, that collects health or resource metrics and then elects the leader, with repeating this process in background [8]. This resembles role-based leadership along a mission timeline, where suitability evolves with position and resource status.

Several approaches use simple heuristics or structural assumptions to decide who becomes leader. In some swarms, a ranking of nodes is defined and any robot can be a leader: when it fails the next suitable node in the formation graph becomes leader [11]. Another formation-maintenance framework introduces temporary leaders for subsets of the swarm, using tree structure and neighborhood relations rather than a global optimization [12]. A communication-centric UAV swarm scheme election leaders using a greedy method based on channel gain, data rate and delay, while coordinators are chosen by simple distance to a referencee [13]. This improves throughput and delay but ignores physical health and environment beyond communication quality.

More sophisticated papers formulate leader election as a graph-theoretic optimization. Leader election framework effectively defines a cost that trades off current tracking error and graph convergence rate, and then chooses the leader minimizing this cost at each step. This might be called multi-criteria decision over robot states and topology, though not yet explicitly extended to energy or sensing [7]. For lunar rovers, the leader election algorithm is built on *Gallager-Humblet-Spira* to create spanning trees. The root *“appointer”* then elects leaders based on aggregated resource information while handling communication limits [8]. Other communication-oriented leader election aggregates channel gain, rate, and delay in a greedy framework, providing an example of utility construction over several metrics, though still limited to networking aspects [13].

The completed literature review brings out a dichotomy in current swarm robotics research. On the one hand, traditional leader-follower architectures provide highly effective and structured formation control, but they remain vulnerable to a single point of failure. However, existing dynamic election mechanisms are frequently overly specialized, relying on isolated metrics such as network connectivity, or sensor coverage. While this review confirms, that aspects of energy, computation, sensing, and local repulsion are well studied, but rarely aggregated into a single, explicit, multi-objective utility for leader election. Furthermore, these existing methods often fail to take into account situational complexity of the environment.

While ongoing studies have made significant progress, comprehensive solutions that unify cost-aware path planning with dynamic role assignment remain largely unexplored. This situation highlights the urgent need for a resilient, scalable method for unmanned ground vehicle swarms that incorporates a multi-parameter utility function for leader election. Such approach must simultaneously account for the vehicle's physical states, terrain constraints, and local repulsive forces, ensuring a reliable, fault-tolerant, and real-time response to dynamic environmental changes.

This highlights a significant research gap, as no method for dynamic leader election currently integrates a comprehensive multi-parameter utility function. Specifically, existing approaches do not simultaneously consider global routing, energy level, communication quality and local environment factors while ensuring real-time performance.

3. The aim and objectives of the study

The aim of this research is to enhance efficiency and fault tolerance of the autonomous navigation for UGV swarms by investigating navigation processes based on a two-dimensional environment simulation model. To achieve the stated aim, the following objectives have been defined:

– to develop a decentralized protocol for dynamic swarm leader election based on a multi-parameter utility function, combining terrain traversal cost, vehicle energy level, proximity to the goal, and safety from obstacles;

– to integrate the developed protocol into a hybrid method for autonomous navigation, together with stability mechanisms, including hysteresis to eliminate chaotic role re-assignment at the terrain boundaries and an artificial vortex field (AVF) to ensure smooth bypassing of failed vehicles;

– to conduct experiments using a complex two-dimensional environment simulation model with both static and dynamic obstacles and network interference to evaluate the increased efficiency and improved fault tolerance for UGV swarms.

4. The study materials and methods for leader election protocol

4.1. The object, subject and hypothesis of the study

The object of this study is the process of leader election for an unmanned ground vehicle swarm within a complex and dynamic two-dimensional simulated environment. The multi-vehicle swarm behaves as a synchronized unit. Its primary goal involves navigating from an initial position to a target destination, simultaneously preserving the assigned formation and preventing impacts with obstacles.

The subject of this research is a decentralized leader election protocol, integrated with a hybrid path planning algorithm for UGV swarm, described in work [2]. The study investigates how the incorporation of energy-level based consensus mechanism, coupled with leadership hysteresis enhances fault tolerance and stability of the swarm in dynamic environments with unreliable communication networks.

The main hypothesis of this research states that replacing static leader with decentralized leader election protocol, based on energy levels, improves swarm performance. The approach should increase overall mission success rate compared to static single point of failure architecture. It is expected that hybrid method will maintain the benefits of global path lookup and local real-time collision avoidance. This enables adaptive behavior in different situations, including: hardware failures, battery degradation and challenging terrain.

This main hypothesis can be broken down into three hypotheses focusing on specific aspects.

The first hypothesis (H1) admits that the integration of AVF alongside the distributed consensus protocol prevents the swarm from deadlocking when the current leader suffers a hardware failure. Disabled leader acts as a strong repulsive obstacle, creating a local minimum on the global path trajectory. It is expected that vortex component will allow the newly elected leader to smoothly bypass the immobilized vehicle, ensuring continuous mission execution.

The second hypothesis (H2) addresses the issue of network instability. It is expected that distributed election algorithm, enhanced with a dynamic timeouts and leadership hysteresis, maintains consensus and prevents continuous role-switching even under significant packet loss conditions. The hysteresis acts as a stabilizing filter, ensuring that brief fluctuations in utility scores do not trigger chaotic swarm reconfigurations.

The third hypothesis (H3) focuses on environmental adaptability. It is expected that incorporating terrain traversal costs into the utility function allows the swarm to reduce overall traversal time. When the current leader enters a high-cost area (mud), a follower situated on a more favorable surface (grass or asphalt) will dynamically assume leadership. This proactive behavior should enrich swarm with abilities to pass through complex terrains with reduced overall traversal time.

4.2. Decentralized leader election protocol for unmanned ground vehicles swarm

The proposed in [2] hybrid navigation approach relies on a leader-follower pattern. Rather than independently routing towards the target, the remaining vehicles function as followers, focusing entirely on maintaining a specific formation anchored to the leader's coordinates. At every simulation step, each follower's assigned waypoint is computed in real time as a vector offset anchored to the leader's current position [2].

Different formations deliver unique tactical benefits, directly aligning with the specific mission profiles allocated to the swarm. This study primarily focuses on the V-shaped formation, the geometry and basic properties of which are detailed in the previous study [2].

While this approach is well suited for environmental exploration, it creates a single point of failure for the entire swarm. In the event of a hardware failure of the leader, a critical discharge of its battery, or entering difficult terrain areas, the movement of the formation can be completely blocked. To solve this problem, a method of dynamic leader election has been developed, which operates in a decentralized swarm network with probabilistic packet loss. The initiation of the election process occurs through three main triggers:

- loss of connection to the current leader, which is recorded by followers due to exceeding heartbeat timeout thresholds;
- critical decrease in the leader's battery charge level;
- leader entering into a local minimum of the APF algorithm or an area with an extremely high traversal cost, when another vehicle has significantly better movement conditions.

To determine best candidate for the role of the new leader, a multi-parameter utility function $U\left(i\right)$ is introduced for each vehicle i. Its value is calculated locally at each vehicle and broadcasted to other ones during the election

$$U_i = W_g \times C_g + W_s \times C_s + W_t \times C_t + W_e \times C_e, \tag{1}$$

where C_g is the normalized approximation coefficient to the next global waypoint of the path; C_s is the safety score, which is inversely proportional to the total repulsion force from obstacles $\|F_{obs}\|$; C_t is the traversal cost coefficient $cost\left(p_i\right)$ under the vehicle, which stimulates the choice of a leader on optimal surface; C_e is the current battery charge level of the vehicle. The weighting coefficients W_g, W_s, W_t, W_e allow for flexible adjustment of swarm priorities depending on the mission.

To avoid the effect of constant reassignment of the leader due to insignificant fluctuations of metrics (for example, when passing a dynamic obstacle), a hysteresis mechanism is integrated into the algorithm. The newly elected leader receives temporary “immunity” for the role change for a certain number of simulation cycles. Leadership transfer due to traversal cost difference is initiated only when the utility of the follower $max\left(U_{followers}\right)$ exceeds the utility of the leader U_{leader} by a given sensitivity threshold $\Delta U_{threshold}$. This ensures the stability of the swarm behavior on the border of environments with different traversal cost.

The consensus process is based on a modified message exchange algorithm. When the election is initiated, all vehicles enter voting mode and broadcast their values of function U_i. After the election time window ends, the robot with the maximum utility value automatically assumes the leadership authority. This vehicle takes the next global point p_{goal}generated by the A* algorithm as its goal, and sends status confirmations to other agents, clearing their local role cache.

4.3. Modified method for obstacle and collision avoidance in the swarm

Leadership transfer creates two significant problems at the level of the local APF planner: blocking the path by failed leader and rapid changes in the attraction forces in the formation. If the previous leader fails, it turns into a static obstacle. Since the target vector of the new leader often passes through the coordinates of the failed vehicle, a symmetric local minimum arises. To overcome this problem, inter-vehicle repulsion force, F_{UGV} is enriched with AVF component

$$F_{vortex} = \gamma \times \|F_{rep\_ij}\| \times \frac{1}{\|d\|} \begin{pmatrix} -d_y \\ d_x \end{pmatrix}, \tag{2}$$

where γ is the vortex intensity coefficient, and $\|d\|$ is the distance vector between vehicles. This allows the new leader to smoothly bypass the obstacle instead of trying to pass through it.

The second problem is that an instant change of the leader causes a sharp change in the ideal positions of the formation p_{form} for all followers. To prevent excessive accelerations and internal collisions, a smooth reconfiguration method is introduced. Instead of instantly switching the target vector, the local target of the follower $p_{form}(t)$ is linearly interpolated during the transition period $T_{transition}$

$$p_{form}(t) = p_{new\_leader}(t) + \left((1-\tau) \times d_{old\_offset} + \tau \times d_{ideal\_offset}\right), \tag{3}$$

where $\tau \in [0,\ 1]$ is an interpolation parameter that increases uniformly with each simulation cycle from 0 to 1, d_{old_offset} is the relative vector of the vehicle position to the new leader at the moment of role change and d_{ideal_offset} , is the target offset vector according to the given formation geometry (V-shaped). Due to this approach, the change in the swarm member's positions occurs without the chaotic movements, guaranteeing the preservation of the integrity and safety of unmanned ground vehicles.

To guarantee operational safety, the system relies on a collision-avoidance mechanism, detailed in our previous work [2]. Prior to executing a move, each agent's planned path is evaluated for conflicts with neighboring vehicles. In the event of a predicted collision, the vehicle bearing the higher identity is restricted from advancing by zeroing its calculated displacement. Consequently, this approach effectively mitigates intra-swarm collisions during navigation.

4.4. Environment simulation model

To practically verify hypotheses H1–H3, an updated version of the simulation model, initially described in [2], was deployed. This setup validated the proposed leader election protocol and evaluated its fault tolerance and efficiency. The developed simulator's architecture is based on a custom script, written in *Python 3*, utilizing several libraries that provide auxiliary functionality. First, the `pathfinding` library offers the core logic for extending the A* algorithm to support terrain traversal costs. Second, `NumPy` provides the vector and matrix operations necessary to accurately calculate forces and positions. Finally, `Matplotlib` is utilized for generating plots and visualizing paths.

To ensure experimental control and setup flexibility, terrain distributions are parsed from an external `JSON` configuration. This file initializes a 256 × 256 occupancy grid, where navigation costs vary per coordinate. The layout features four terrain profiles: basic solid ground (low traversal cost), grassy regions (medium cost), muddy segments causing significant deceleration (maximum finite cost), and solid barriers (infinite cost). The primary test map was adapted from the architectural geometry of an urban district in Kyiv, Ukraine.

To emulate dynamic real-world conditions, moving obstacles were incorporated into the simulation environment. Each obstacle was assigned starting coordinates, a constant velocity, and a path composed of a sequence of waypoints. During every execution cycle, these objects advance through their planned routes at uniform rates, effectively transforming the static grid into a time-variant testing arena.

Throughout the trial, the swarm preserves the baseline inter-vehicle spacing and maintains a predefined V-shaped formation. Each vehicle is modeled with specific physical constraints: a collision radius, an upper limit on forward momentum, and a bounded detection range. The total number of UGVs in the formation is dictated by the initialization parameters of the chosen scenario configuration.

The developed simulator was deployed locally to the standard hardware, executing under Windows 10 Pro (version 19044.6396) 64-bit operating system. The setup is featuring Intel i7-10870H processor, 16 GB RAM, and Nvidia GTX 1650 graphics card (4 GB of GDDR6 memory).

5. Results of the research on decentralized leader election protocol

The primary outcome of this research is a decentralized leader election protocol designed for unmanned ground vehicle swarms operating in dynamic environments. The scientific novelty of this study lies in the development of a multi-parameter utility function for leader election protocol and its integration into the hybrid navigation method developed in [2]. Unlike traditional models that rely on a static leader, which makes the entire swarm vulnerable to a single point of failure, the proposed approach allows the swarm to autonomously reassign leadership. By evaluating the utility function locally at each vehicle, protocol ensures swarm stability under conditions of hardware failure, battery depletion, and varying terrain traversability.

Furthermore, in contrast to existing solutions, this study incorporates the concept of terrain traversal cost into the leader election. This enables an emergent adaptive behavior where leadership is transferred to vehicles located on more favorable terrain, optimizing efficiency of entire swarm.

To systematically evaluate the proposed approach and align with the defined research objectives, the results are presented in three subsections. All results have been obtained using a simulator, ensuring that all testing scenarios remained strictly repeatable. To test each of the hypotheses (H1–H3), a baseline scenario was initialized, followed by the execution of multiple experimental iterations. This approach allowed the researchers to obtain reliable data for comparison since the influence of other factors on the simulator's behavior was isolated.

Conditions for the baseline scenario were initialized using a specific set of variables: the evaluation was performed on the standard map with a swarm of 5 vehicles, maintaining a V-shaped formation. The decentralized leader election protocol was activated alongside with dynamic obstacles. To increase the statistical strength of the findings, each scenario was executed three times, and the resulting metrics were averaged.

5.1. Decentralized leader election protocol with utility function verification

The first objective was to develop and test the core protocol driven by the multi-parameter utility function. In contrast to existing solutions, this study incorporates the concept of terrain traversal cost into the leader election protocol. This enables emergent adaptive behavior where leadership is transferred to a vehicle located on more favorable terrain, optimizing the efficiency of the entire swarm.

To validate hypothesis H3, an experiment focused on swarm behavior when moving to a zone with a high traversal cost (mud) was conducted. As the current leader entered the mud zone and lost speed, its utility score dropped. The protocol transferred control to a follower, which was still on the hard surface and had a higher utility score, as shown in Figs. $1a$–$1b$.

This confirms that the utility-driven local response effectively reassigns roles based on environmental conditions.

5.2. Decentralized leader election protocol integration with stability mechanisms

The second objective focused on integrating the protocol into the hybrid method alongside stability mechanisms to handle critical failures safely. A novel modification to the local APF planner was introduced to address the issue of symmetrical local minima. These minima typically occur when a

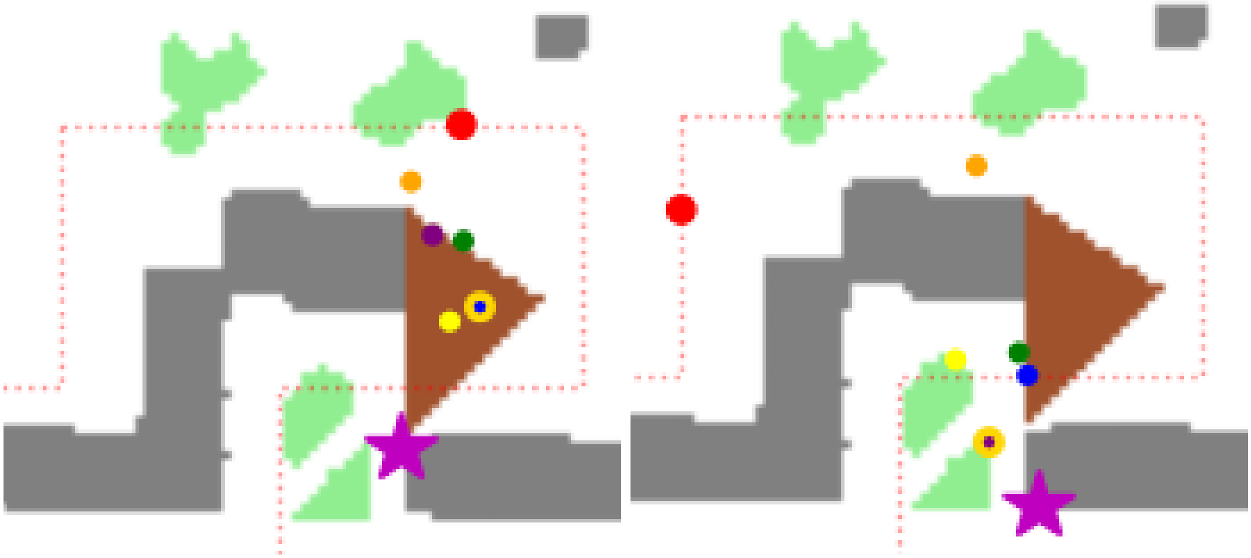


Fig. 1. Swarm handles high cost traversal zone: *a* – leader entered mud zone; *b* – new leader for zone traversal was elected

failed leader becomes a static obstacle blocking the path for the swarm. The integration of a smooth reconfiguration interpolation method and an AVF enhances the inter-vehicle repulsive forces.

Hypothesis H1 was experimentally validated using the recorded telemetry combined with path visualizations. In the first series of experiments, the hardware failure of the leader on the path was simulated. In the case of using the static approach, the global planner continued to generate the optimal path, but the local APF could not cope with the failure. The failed leader turned into a static obstacle, which led to a complete failure of the mission, as illustrated in Fig. 2*a*. In contrast, the decentralized leader election protocol successfully solved this problem by instantly electing a new leader. This leader bypassed the faulty vehicle, continuing to move towards the target, as depicted in Fig. 2*b*.

The experiment also aimed to test the ability of the swarm to avoid falling into a symmetric local minimum of the APF algorithm. The newly elected leader, using an integrated AVF, demonstrated the ability to bypass a failed vehicle, heading to the intermediate waypoint (indicated by a star). Simultaneously, linear interpolation allowed followers to rebuild the formation into a proper V-shape without internal collisions, as demonstrated in Figs. 3*a*–3*b*.

These experimental results demonstrate that an architectural framework combining strategic global cost-aware planning with a dynamic, utility-driven local response is highly resilient for swarm navigation in initially unknown or unpredictable environments. The newly elected leader bypasses unresponsive vehicles without causing internal collisions or destabilizing the formation geometry, thereby ensuring continuous and autonomous mission execution under hardware disruptions.

5.3. Decentralized leader election protocol efficiency and fault tolerance

The third objective involved conducting experiments to evaluate the efficiency and fault tolerance of the hybrid approach with the decentralized leader election protocol compared to a static leader replacement baseline, where election triggers were restricted to critical battery levels.

A separate aspect of fault tolerance was the analysis of the protocol's resilience to network interference (hypothesis H2). During a series of simulations, where the number of packet loss varied from 0 to 80%, the relationship between mission completion time and packet loss was evaluated for

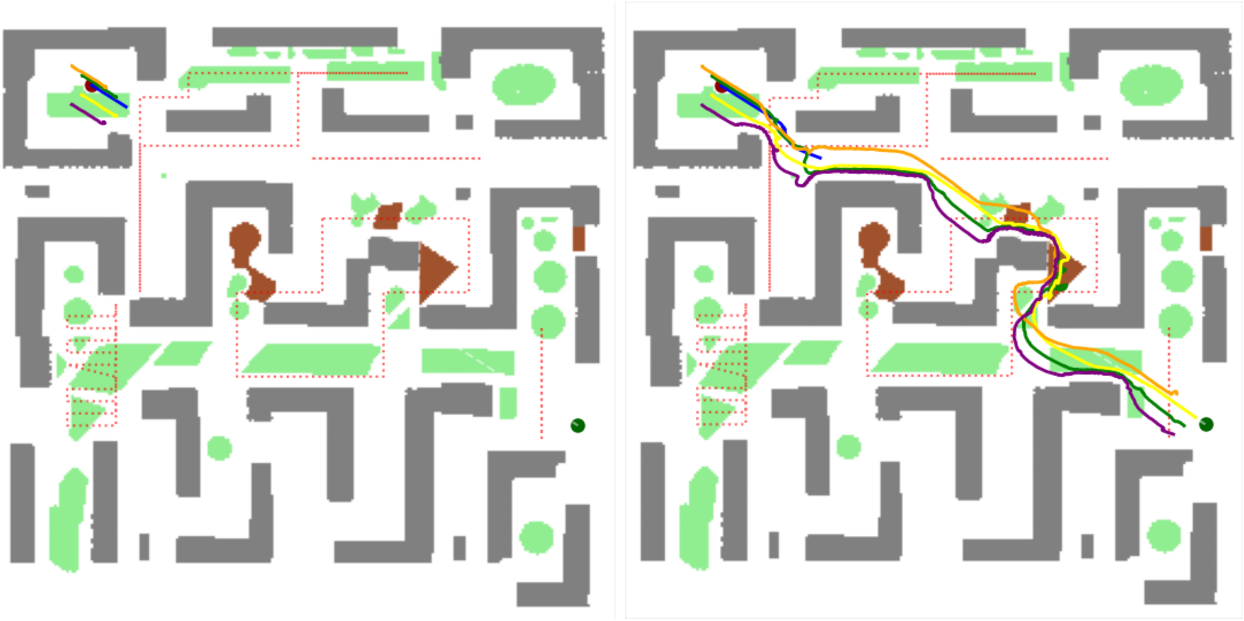

a *b*

Fig. 2. Path comparison: *a* – static leader approach failed mission because of leader (blue) failure; *b* – proposed approach elected new leader (yellow) and finished mission

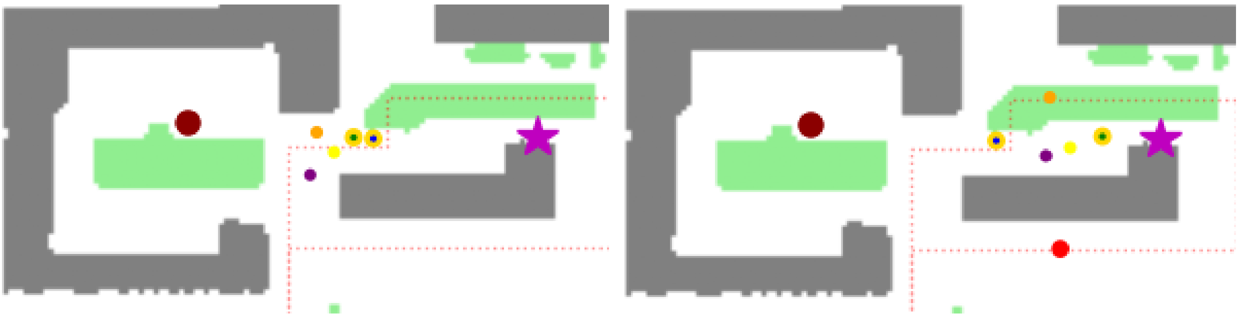

a *b*

Fig. 3. Swarm handles leader failure: *a* – new leader was elected, both have yellow edge; *b* – new leader passed by the failed vehicle with proper V-shaped formation

both approaches. The results indicate that while the protocol handles moderate interference, it remains vulnerable to severe communication disruptions, as depicted in Fig. 4.

To analyze the efficiency (hypothesis H3), experiments were conducted in high-cost terrain conditions (mud), which could not be avoided across different swarm sizes (3, 5, and 7 vehicles). Several key metrics were collected during the simulations, including the mission completion time and the mean squared error (MSE) of the formation. Figure 5*a* illustrates mission completion time, while Fig. 5*b* shows its correlation with the average MSE.

The experiment compared the proposed protocol against a baseline, forcing the initial leader to traverse the high-cost zone without handing off control. At each simulation step, the MSE was calculated as the squared *Euclidean distance* between the actual position of each follower and its ideal target coordinate. These ideal coordinates are defined within the V-shaped formation relative to the current leader.

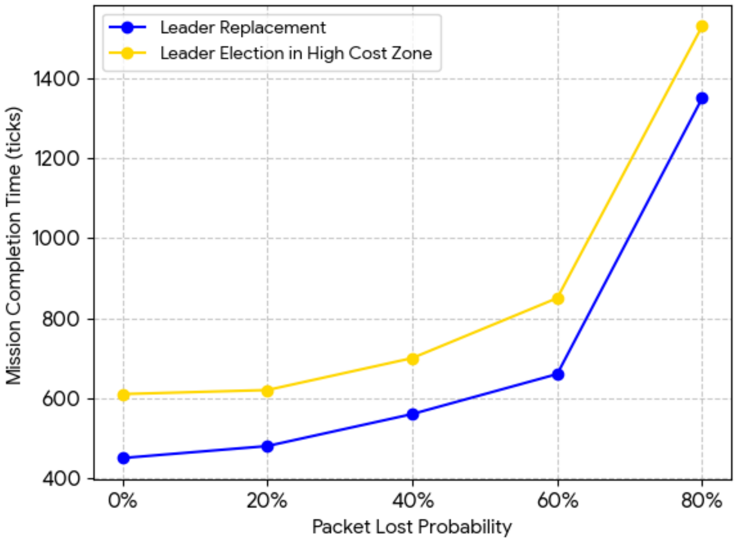


Fig. 4. Mission completion time as a function of packet loss probability

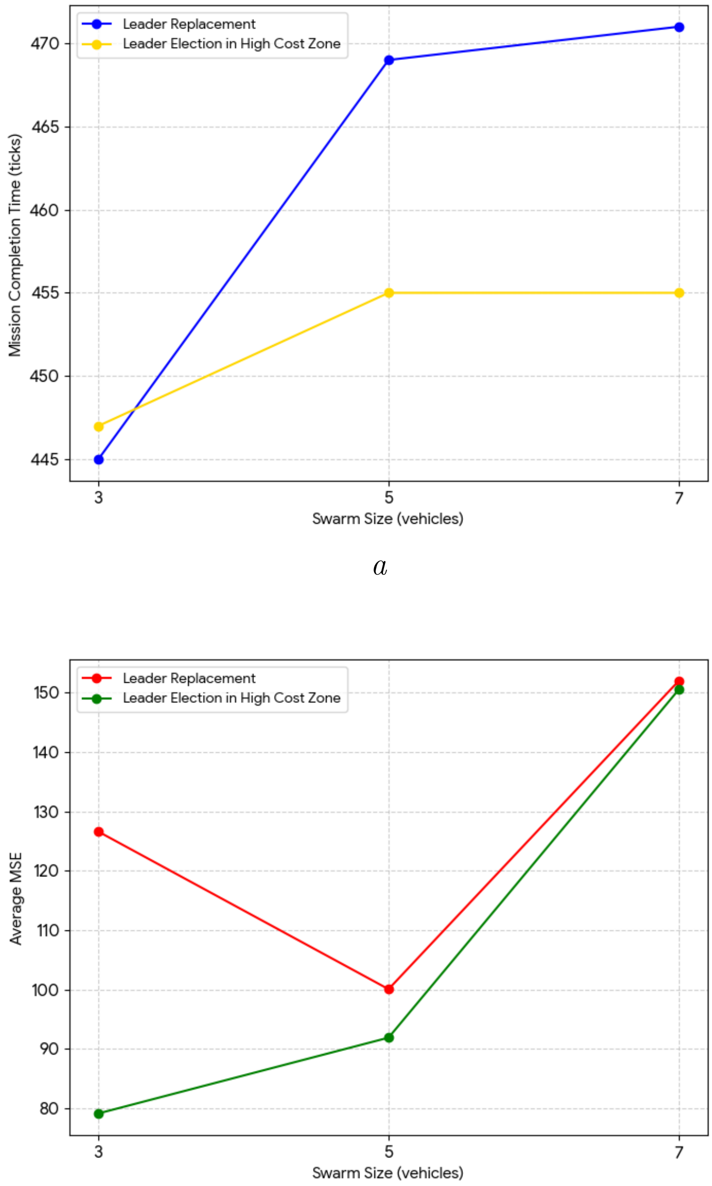
The comparative analysis confirms the high efficiency of the proposed approach. By transferring leadership to a follower situated on lower-cost terrain, the swarm maintains formation geometry while simultaneously reducing the total traversal time through the environment.

6. Discussion of results for decentralized leader election protocol

This discussion is dedicated to interpreting data from the previous section within the context of stated hypotheses. Hypothesis H1 stated that integrating an AVF with a distributed consensus protocol prevents the swarm from deadlocking when the current leader suffers a hardware failure. The experimental data fully supported this assumption. In the static leader architecture, a failed vehicle acted as a static obstacle in the formation's path, creating a local minimum that halted the entire swarm. Since the dynamic approach allows the swarm to reassign leadership, it enables the newly elected leader to bypass the obstacle and continue the mission. Thus, H1 is confirmed, demonstrating that the proposed approach successfully eliminates the single point of failure.

The second hypothesis concerning network instability and the prevention of continuous role-switching (H2) yielded more complex results than expected. It was hypothesized that dynamic timeouts and leadership hysteresis would maintain consensus and stabilize the swarm even under significant packet loss. However, the simulations revealed that the proposed decentralized election algorithm is still vulnerable to severe communication disruptions. While the proposed protocol handles moderate interference, extreme network degradation (up to 80% packet loss) disrupts the peer state updates required for accurate utility calculations. This leads to delayed communication and negatively impacts the overall mission completion time. Therefore, hypothesis H2 is only partially supported, indicating a limitation in the algorithm's resilience under extreme network stress.

The third hypothesis (H3) focused on approach efficiency, stating that incorporating the terrain traversal costs into the utility function would allow the swarm to reduce overall traversal time. The experimental results validated this expectation. When the swarm encountered high-cost terrain (mud), the proposed approach provided proactive behavior by transferring leadership to the follower situated on the more favorable terrain. The static baseline forced the entire formation to slow down, waiting for the leader. However, the experimentally obtained data indicate that the actual difference in performance between approaches is relatively small.



a

b

Fig. 5. Swarm size influence: *a* – on mission completion time; *b* – on average mean squared error

Total mission time analysis shows minimal deviations, with the decentralized leader election protocol reducing time by approximately 3% (from 469 to 455 simulation ticks) for a 5-vehicle swarm. This difference was slightly more than 1% (455 vs. 461 simulation ticks) for a 7-vehicle swarm. Moreover, in the case of the smallest swarm (3 vehicles), the static approach proved to be faster by 2 ticks (445 vs. 447). A similar trend is observed for the average mean square error. Although for a small swarm, the proposed approach significantly improved the geometry stability, reducing the error from 126.60 to 79.15, as the swarm size increase this advantage was quickly leveled off. Thus, for 5 vehicles the difference in MSE decreased approximately to 8% (91.91 vs. 100.07), and for 7 vehicles the metrics turned out to be almost identical (150.50 vs. 151.97). These results show that the static approach remains competitive without hardware failures and proposed method is effective in case of small network disruptions, so H3 is partially supported.

The constraints encountered during this research provide several focus areas for future work. Future research should focus on improving the resilience of the decentralized protocol under severe network degradation, addressing the vulnerabilities uncovered during the H2 evaluation. Potential solutions could involve integrating a gossip-based communication protocol to reduce reliance on continuous broadcasts. Furthermore, transitioning from software-based modeling to physical hardware deployment will offer empirical validation of the protocol's applicability in unpredictable real-world environments.

Conclusion

This research focused on improving the efficiency and fault tolerance for the autonomous navigation of UGV swarms in a dynamic two-dimensional environment simulation model. The literature review identified that traditional static-leader architectures create a single point of failure, making the formation vulnerable to hardware failures, depletion of energy resources, and unpredictable terrain changes. To address these problems and achieve the overall aim of the study, the following three objectives were successfully fulfilled.

The first objective was achieved by developing a decentralized protocol for dynamic swarm leader election. The scientific novelty of the protocol lies in its multi-parameter utility function, combining terrain traversal cost, vehicle energy level, proximity to the goal, and safety from obstacles. The developed protocol allows the swarm to autonomously and dynamically reassign leadership without relying on a central node, thereby eliminating the single point of failure.

The second objective was fulfilled by integrating the developed decentralized protocol into the existing hybrid autonomous navigation method, previously detailed by the authors in [2]. The proposed integration allowed the swarm to perform autonomous leader reassignment based on battery and terrain metrics. Furthermore, it incorporated stability mechanisms such as leadership hysteresis, to eliminate chaotic role re-assignments at terrain boundaries, and AVF to ensure the smooth bypassing of failed vehicles while maintaining formation geometry.

To implement the third objective and conduct the experimental study, a custom two-dimensional dynamic environment simulator was developed. This simulator allows for complex modeling of swarm behavior under the influence of both static and dynamic obstacles, as well as probabilistic network interference. The effectiveness and fault tolerance of the proposed protocol were demonstrated through a series of controlled experiments using this simulator.

The experimental results validated the formulated hypotheses and verified the protocol's efficiency. Specifically, hypothesis H1 was confirmed, demonstrating that the AVF prevents deadlocks during leader failures. The evaluation of hypotheses H2 and H3 revealed both the strengths and limitations of the approach. Regarding H2, while effective under moderate interference, experiments showed that extreme packet loss (up to 80%) still poses a vulnerability to the protocol. Finally, in support of H3, transferring control to followers on more favorable terrain increased overall efficiency. Compared to a static leader baseline, the proposed protocol reduced the total traversal time by 3% and improved the formation MSE by 8% for a 5-vehicle swarm.

In conclusion, this investigation contributes to the advancement of fault-tolerant multi-agent systems. The practical value of these findings lies in their applicability to real-world scenarios where coordinated, autonomous collective action of UGV swarms is required. The developed protocol can be directly utilized in robotic systems designed for search-and-rescue operations, environmental monitoring, and automated logistics, ensuring mission continuity despite individual vehicle failures or challenging terrain conditions.

Acknowledgements

Declaration on the use of Artificial Intelligence. The authors used the generative AI tool Gemini 3 during the preparation of this manuscript for language editing. All AI-generated content was carefully

reviewed and edited by the authors. The authors take full responsibility for the accuracy, integrity, and originality of the article.

References

[1] H. Duan, M. Huo, and Y. Fan, “From animal collective behaviors to swarm robotic cooperation,” *National Science Review*, vol. 10, no. 5, May 2023. https://doi.org/10.1093/nsr/nwad040.

[2] M. Rudnytskyi and I. Klymenko, “Hybrid path planning method for unmanned ground vehicles swarm in dynamic environments,” *Inf. Comput. and Intell. syst. j.*, no. 6, pp. 87–99, Sep. 2025. https://doi.org/10.20535/2786-8729.6.2025.333730.

[3] F. Refis, N. Mahammedi, C. Kerrache, and S. Dhelim, “From network sensors to intelligent systems: A decade-long review of swarm robotics technologies,” *Sensors*, vol. 25, no. 19, pp. 6115–6147, Oct. 2025. https://doi.org/10.3390/s25196115.

[4] M. Zhou, Z. Wang, J. Wang, and Z. Dong, “A hybrid path planning and formation control strategy of multi-robots in a dynamic environment,” *J. Adv. Comput. Intell. Intell. Informatics*, vol. 26, no. 3, pp. 342–354, May 2022. https://doi.org/10.20965/jaciii.2022.p0342.

[5] A. Bekmez and K. Aram, “A communication system for dynamic leader selection in distributed UAV swarm architecture,” *European Journal of Technic*, vol. 14, no. 1, Jun. 2024. https://doi.org/10.36222/ejt.1434751.

[6] G. Aguzzi *et al.*, “A field-based approach for runtime replanning in swarm robotics missions,” in *Proc. 2025 IEEE International Conference on Autonomic Computing and Self-Organizing Systems*, Tokyo, Japan, Nov. 2025, pp. 1–10. [Online]. Available: https://doi.org/10.1109/ACSOS66086.2025.00017

[7] A. Franchi and P. Giordano, “Online leader selection for improved collective tracking and formation maintenance,” *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, Jan. 2016. https://doi.org/10.48550/arXiv.1305.5719.

[8] J.-P. de la Croix *et al.*, “Multi-agent autonomy for space exploration on the CADRE lunar technology demonstration,” in *Proc. 2024 IEEE Aerospace Conference*, Big Sky, MT, USA, May 2024, pp. 1–14. [Online]. Available: https://doi.org/10.1109/AERO58975.2024.10521425

[9] V. Tran, A. Perera, M. Garratt, K. Kasmarik, and S. Anavatti, “Coverage path planning with budget constraints for multiple unmanned ground vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 12 506–12 522, Nov. 2023. https://doi.org/10.1109/tits.2023.3285624.

[10] L. Zhaobo and S. Shuanglei, “Multi-vehicle formation and obstacle avoidance control based on pigeon-inspired optimization and dynamic window approach,” *Chinese Journal of Engineering*, vol. 46, no. 7, pp. 1279–1285, Nov. 2024. https://doi.org/10.13374/j.issn2095-9389.2023.10.11.003.

[11] Y. Abdul-Razaq, “Leader-follower formation control of ROS enabled mobile robots subject to robots failure,” Master’s thesis, University of Alberta, Edmonton, AB, Canada, Jun. 2022. https://doi.org/10.7939/r3-p0ex-ej31.

[12] J. Yasin, M.-H. Haghbayan, M. Yasin, and J. Plosila, “Swarm formation morphing for congestion-aware collision avoidance,” *Heliyon*, vol. 7, no. 8, Aug. 2021. https://doi.org/10.1016/j.heliyon.2021.e07840.

[13] C. Tesfay, Z. Xiang, L. Yang, M. Berwo, J. Mahmood, and S. Chaudhry, “An effective and efficient UAV leader selection scheme in swarm of UAVs,” *Physical Communication*, vol. 67, Dec. 2024. https://doi.org/10.1016/j.phycom.2024.102513.

УДК 004.896:681.51

ПРОТОКОЛ ДЕЦЕНТРАЛІЗОВАНОГО ВИБОРУ ЛІДЕРА ДЛЯ РОЮ БЕЗПІЛОТНИХ НАЗЕМНИХ РОБОТІВ В ДИНАМІЧНИХ СЕРЕДОВИЩАХ

Мирослав Рудницький
https://orcid.org/0000-0002-6166-1078

Ірина Клименко
https://orcid.org/0000-0001-5345-8806

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна

Отримано: 14.04.2026р. / Прийнято: 11.05.2026р. / Опубліковано: 28.05.2026р.

Рої наземних роботизованих комплексів набувають актуальності як основа автономних систем, здатних ефективно функціонувати в динамічних умовах. Проте проблема надійності управління стоїть на шляху їх широкого впровадження. Традиційні алгоритми навігації, що базуються на моделі статичного лідера, формують єдину точку відмови, яка підвищує вразливість формації до апаратних збоїв, виснаження енергетичних ресурсів та непередбачуваних змін рельєфу. В роботі досліджуються процеси автономної навігації рою на основі імітаційної моделі двовимірного середовища. Метою дослідження є підвищення ефективності та відмовостійкості шляхом розробки децентралізованого протоколу вибору лідера рою. Запропонований протокол інтегровано до гібридного методу навігації, описаного в попередніх роботах авторів, який поєднує планування (алгоритм A* і штучне потенційне поле) та механізми стабілізації (гістерезис і штучне вихрове поле). Наукова новизна полягає у розробці протоколу на основі багатопараметричної функції корисності, яка враховує енергетичні ресурси агентів, вартість проходження рельєфу, наближення до цілі та безпечну відстань до перешкод. Інтеграція протоколу у зазначений гібридний метод забезпечує адаптивний розподіл ролей у рої в умовах змінного середовища. Результати моделювання підтвердили, що інтеграція розробленого протоколу в гібридний метод навігації підвищує відмовостійкість, дозволяючи рою долати локальні мінімуми при відмовах лідера за помірних перешкод зв'язку (до 60% втрачених пакетів). Експерименти показали, що передача керування послідовникам у кращих умовах підвищує ефективність за рахунок зменшення часу проходження ділянок зі складною прохідністю. Порівняльний аналіз показує перевагу запропонованого підходу над методом з фіксованим лідером: для рою з 5 агентів загальний час проходження зменшується на 3%, а середня квадратична похибка утримання формації – на 8%.

Ключові слова: наземний роботизований комплекс, децентралізований протокол вибору лідера, енергоефективність, динамічне середовище, автономна навігація рою.