

# Energy-efficient coverage and connectivity maintenance in UAV swarm networks via multi-agent coordination

Harsh Yadav<sup>1</sup>, Varun Shukla<sup>2,\*</sup>, Atul<sup>3</sup>, Badiea Abdulkarem Mohammed<sup>4</sup>

Time-constrained disaster management with micro-unmanned aerial vehicle (UAV) swarms requires autonomous coordination to ensure effective spatial coverage while preserving reliable ad-hoc network connectivity. The inherent trade-off between coverage and connectivity is compounded by the strict aerodynamic power limitations of micro-UAVs, whose batteries deplete rapidly under erratic maneuvers. Conventional centralized control architectures and standard artificial potential field (APF) algorithms often fail to address this multi-objective problem, leading to fragmented networks or wasted energy due to oscillatory trajectories (jitter). To overcome these limitations, this paper proposes the Energy-Damped Virtual Force (EDVF) algorithm, a fully decentralized, neighbor-aware control law. EDVF employs an adaptive velocity-dependent damping term (analogous to viscous friction) that penalizes unnecessary micro-corrections as drones settle into their topological positions. We formulate a multi-objective optimization model that integrates algebraic connectivity with a non-linear aerodynamic power model based on Blade-Element Momentum Theory. Python-based simulations demonstrate that EDVF outperforms both traditional artificial potential field methods and centralized benchmarks: it maintains approximately 39% percent area coverage – 99% percent of the theoretical maximum for the swarm configuration – guarantees a connected network throughout the mission  $\lambda_2 > 0$ , and demonstrably extends swarm operational lifetime by eliminating kinematic oscillations and driving agent velocities asymptotically to zero, thereby removing the cubic aerodynamic drag penalty  $P_{par} \propto \|v\|^3$  that undamped baselines continuously incur. EDVF thus provides a theoretically grounded and scalable solution for maintaining persistent aerial surveillance in stochastic environments.

Keywords: UAV swarms, energy efficiency, virtual force algorithm, flying ad-hoc networks, Lyapunov stability, multi-agent systems, coverage control, connectivity maintenance

## 1 Introduction

Recent advancements in autonomous robotics, compact sensing technologies, and high-speed wireless communication have significantly transformed how aerial systems are designed and deployed. Driven by these technological improvements, the aerospace and defense sectors are gradually moving away from relying on a single highly sophisticated Unmanned Aerial Vehicle (UAV). Instead, they are increasingly focused on coordinating groups of smaller UAV drones that operate collectively as a swarm [1, 2]. This shift is strongly influenced by observations from nature, where biological swarms such as bird flocks or insect colonies demonstrate how simple local interactions between individuals can produce complex and organized group behavior [3, 4]. Applying this principle to UAV systems allows multiple inexpensive drones to distribute sensing, computation, and communication tasks. This enhances reliability and enables much broader spatial coverage

than a single drone could achieve [5, 6]. However, operating a swarm in a decentralized fashion presents a number of technical challenges. Coordinating the motion of a large number of agents, maintaining a stable communication topology, and effectively using limited resources such as energy and bandwidth all become critical problems that must be resolved for successful swarm deployment.

### 1.1 Background and applications

In recent years, UAV swarms have moved from theoretical research to real-world practical use. This transition has been enabled by the development of Flying Ad-Hoc Networks (FANETs), which allow drones to communicate and connect with each other without depending on ground-based computation systems [2, 7]. Consequently, UAV swarms are now widely used in critical situations where human involvement is risky

<sup>1</sup>Department of Computer Science & Engineering, Allenhouse Institute of Technology, Kanpur, India

<sup>2</sup>Dean of Research, Allenhouse Institute of Technology, Kanpur, India

<sup>3</sup>Department of Computer Science & Engineering, Allenhouse Institute of Technology, Kanpur, India

<sup>4</sup>Department of Computer Engineering, College of Computer Science and Engineering, University of Ha'il, Ha'il 81481, Saudi Arabia

\*varun.shuklaa@gmail.com

or infeasible, such as disaster response, earthquake monitoring, hazardous leak detection, and search-and-rescue (SAR) missions [8, 9].

In such situations, ground communication networks are often damaged or completely unavailable. UAV swarms are therefore deployed rapidly to create temporary communication networks and provide real-time information to rescue teams in disaster-prone areas or locations inaccessible to humans [10, 11]. During such missions, the primary goal of the swarm is to cover as much area as possible so that important targets, such as survivors or damaged structures, can be identified quickly [9, 12].

### 1.2 Problem statement: Coverage vs. connectivity

Even though multi-rotor UAVs aim to maximize spatial coverage, they cannot operate in isolation. They must continuously coordinate movements, share high-bandwidth sensor data, and transmit information back to a central command node or ground station. Therefore, maintaining a stable and consistent communication channel among swarm members is equally critical to mission success. This requirement gives rise to a mathematically challenging problem commonly known as the coverage versus connectivity trade-off [2].

To maximize spatial coverage, individual UAVs must disperse to explore unexplored regions of the target area. However, this expansion increases the distance between drones, pushing them closer to the communication range limit  $R_{comm}$ . The greater the distance, the weaker the signal becomes due to attenuation and path loss, thereby degrading communication quality. If unaddressed, this leads to data loss, failed communication links, and the formation of isolated subgroups of drones that lose synchronization with the larger network [2, 12]. Conversely, when the swarm prioritizes connectivity by keeping drones in close proximity, agents tend to form dense clusters. While this makes communication reliable, it causes significant overlap of sensing regions ( $R_{sens}$ ). Such overlap reduces the useful coverage area, slows mission progress, and may leave large portions of the environment undiscovered [13, 14].

### 1.3 Motivation and limitations of current approaches

The coverage-connectivity dilemma is further aggravated by the severe and unavoidable energy constraints of micro-UAV architectures. The aerodynamic power required to sustain flight in multi-rotor agents is high, leaving a very limited energy budget for onboard computation, sustained wireless transmission, and dynamic spatial repositioning [15]. Unfortunately, most existing topological control frameworks do not adequately

address this multi-objective problem under hard, non-linear power constraints.

Classical centralized control methods, in which a single ground station computes the optimal paths for all  $N$  agents, are unsustainable in dynamic disaster scenarios. Centralized approaches suffer from low throughput, present a catastrophic single point of failure, and require constant state-telemetry overhead that quickly depletes the limited battery life of the swarm [15].

In response, the research community has gravitated toward distributed multi-agent coordination methods. However, the established approaches introduce their own critical flaws.

#### 1.3.1 Multi-agent reinforcement learning (MARL)

While MARL algorithms provide excellent adaptability, they inherently require heavy tensor computations to execute deep neural networks continuously onboard the micro-UAVs. This continuous processing burden diverts a significant portion of the finite energy budget away from flight propulsion and into computational overhead [16, 18].

#### 1.3.2 Standard artificial potential fields (APF)

Lightweight reactive algorithms, including conventional virtual forces, avoid heavy computational overhead but often fail to account for continuous-time kinematics and mechanical energy. Standard APFs may trap drones in local minima or induce localized jittering – jerky, high-frequency oscillatory flight trajectories that arise as drones continually micro-adjust their position in response to conflicting attractive and repulsive forces. Since aerodynamic parasite power scales with the cube of velocity ( $P_{par} \propto v^3$ ), these unwanted velocity oscillations significantly reduce the physical flying range of the swarm [19, 20].

A simple, mathematically stable coordination technique is therefore needed – one that automatically balances coverage and connectivity without wasting propulsion energy on erratic motions or imposing excessive onboard computation.

### 1.4 Key contributions

To fill these essential gaps in the literature, this paper proposes a complete, energy-efficient framework based on a novel multi-agent virtual force strategy. The framework is designed to control micro-UAV swarm kinematics in dynamic, unstructured environments while strictly preserving network topology and minimizing aerodynamic battery depletion.

The primary contributions of this manuscript are theoretically and empirically substantiated as follows:

#### 1.4.1 Energy-damped virtual force (EDVF) formulation

We introduce a fully decentralized control law that dynamically balances spatial repulsion (for coverage) with non-linear attraction (for connectivity). Crucially, the algorithm incorporates an adaptive, residual-energy-dependent damping term that strongly suppresses energy-wasting trajectory oscillations.

#### 1.4.2 Rigorous aerodynamic cost modelling

We integrate a non-linear aerodynamic power consumption model that explicitly distinguishes between the power cost of dynamic flight and that of static hovering, directly coupling physics-based energy constraints to the swarm's topological decisions.

**Lyapunov stability analysis:** We provide a formal stability result using a Lyapunov candidate function, proving that the proposed EDVF system is globally asymptotically stable and that all agent velocities converge naturally to the equilibrium point ( $v_i \rightarrow 0$ ).

**Extensive Python-based simulation:** We benchmark the EDVF algorithm through a high-fidelity, discrete-time Python simulation using SciPy and NumPy. Through dynamic spatial mapping, topological analysis, and parameter ablation studies, we demonstrate that EDVF achieves near-maximum feasible coverage ( $\approx 39\%$ , against a theoretical ceiling of  $39.3\%$ ) with a zero network partition rate ( $\lambda_{drop} = 0$ ) throughout a 1000 s mission, while eliminating the kinematic oscillations that cause undamped baselines to waste aerodynamic energy.

## 2 Related work

The literature on multi-UAV systems and swarm robotics is extensive and rapidly evolving. However, the specific intersection of maximizing spatial coverage, ensuring continuous ad-hoc network connectivity, and minimizing dynamic aerodynamic energy consumption remains a complex and unresolved challenge. We categorize the existing research into four primary domains to highlight the progression of the field, the specific contributions of recent studies, and the critical gaps that motivate this work.

### 2.1 Swarm robotics in disaster management

A central challenge in deploying UAV swarms in unknown or disaster-affected areas is maximizing sensory and communication coverage during emergency

operations under conditions of high uncertainty. Earlier research, such as the system-level area-coverage approach developed by Yanmaz and Bettstetter [12], provided important performance indicators for evaluating the effectiveness of a fleet of vehicles in patrolling designated zones and introduced the concept of synchronized mobility to minimize scanning times. Expanding on these notions in hazardous crisis environments, Erdelj and Natalizio [8] classified UAVs and detailed their corresponding roles in integrated wireless sensor networks (WSNs), demonstrating their effectiveness in addressing communication gaps caused by the destruction of terrestrial infrastructure.

For rapid Search and Rescue (SAR), prompt swarm deployment is essential to locate survivors, and swarm heterogeneity tends to improve detection rates in densely populated environments [8]. Tuna and Nefzi [10] propose beehive-like bio-inspired information-gathering methods and show that autonomous drones imitating social insect behaviors are more robust in disaster-relief efforts. Furthermore, recent applications have moved beyond simple monitoring; Alsamhi and Ma [21] demonstrate the use of drone swarms in distributed video surveillance of compromised roadways and vehicle tracking, employing collaborative sensing to maintain situational awareness in dynamic urban environments.

Despite these significant advancements, a recurring limitation in the disaster management literature is the isolated treatment of the coverage problem. As UAVs disperse to expand their footprint during rescue operations, inter-node distances inevitably increase, leading to degraded communication links. Recent work on rapid post-disaster deployment, such as Zhao et al. [11], has attempted to address this gap by optimizing the placement of relay nodes; however, most solutions still do not provide mathematical guarantees against network partitioning during the aggressive, wide-area exploration maneuvers required in the initial phases of disaster response [2, 22].

### 2.2 Connectivity maintenance and learning-based control

A robust communication topology is essential to prevent swarm fragmentation and information loss during wide-scale SAR operations. Multi-hop communication between drones was enabled by the introduction of Flying Ad-Hoc Networks (FANETs), surveyed comprehensively by Bekmezei et al. [2], which do not require terrestrial infrastructure. To control the physical mobility of nodes within these networks, recent literature has increasingly emphasized fully distributed Multi-Agent Systems (MAS), resilience-aware recovery metrics, and learning-based control [23].

A particularly prominent approach in current research is Multi-Agent Reinforcement Learning (MARL). Bu et al. [1], Ekechi et al. [16], and Oh et al. [24] provide comprehensive reviews of UAV swarm formation control, noting that MARL can produce complex emergent behaviors but frequently suffers from slow convergence in real-time scenarios. To address this, Gadiraju et al. [17] proposed GLIDE, a continuous-control MARL model that operates in dynamic military settings and balances formation rigidity with flexibility, demonstrating its applicability in such scenarios. Similarly, Sun and Zhang [18] developed hierarchical, role-based MARL models for heterogeneous swarms, in which agents can dynamically switch between exploration and relaying roles.

More recently, Chen and Xu [25] extended these concepts by incorporating task offloading, proposing energy-efficient deep reinforcement learning models to handle edge computing loads on UAVs. Shi et al. [26] recently introduced symmetry-informed MARL to decentralize cooperative control, significantly reducing the state-space complexity for communication coverage, while Q-learning handover schemes have also been explored for energy-aware FANET connectivity [27]. However, theoretical constraints remain; Kinsler et al. [3] mathematically demonstrated the difficulty of coordination under stringent communication constraints, arguing that simpler, physics-based rules often outperform complex learning models when bandwidth is limited. While MARL equips swarms with powerful decision-making capabilities, the severe computational burden of executing deep neural networks onboard micro-UAVs drastically depletes their aerodynamic energy budgets, a trade-off often overlooked in simulation-based studies [28].

### 2.3 Virtual force algorithms and energy efficiency

Artificial Potential Fields (APF) and Virtual Force (VF) algorithms have emerged as lightweight, reactive alternatives to MARL and offer greater flexibility than static graph-based constraints. Building on the original obstacle avoidance model by Khatib [29] and the flocking rules of Reynolds [4], virtual forces use attractive and repulsive vectors to drive node positioning. These forces were first applied to sensor networks by Howard et al. [30] and Zou and Chakrabarty [31], demonstrating that local interactions can achieve global coverage maximization.

Recent modifications have adapted these forces to UAV-specific constraints. Zheng et al. [32] proposed a self-organizing area-coverage scheme using dynamic virtual forces whose magnitudes adapt based on local node density, and Chen et al. [33] investigated adaptive drone-swarm coordination for dynamic coverage missions. Similarly, Olfati-Saber's flocking framework

[20] demonstrated that connectivity-preserving formation can be achieved with potential fields by introducing dedicated attraction-based retention forces that keep inter-agent links intact. However, standard VF models remain fundamentally vulnerable to localized jittering – an unsteady, oscillatory flight trajectory caused by agents continually micro-adapting to competing attractive and repulsive forces.

This mechanical vibration is critical because energy efficiency is the most binding constraint in long-duration UAV operations [15]. The importance of energy-efficient trajectory optimization was extensively documented by Zeng and Zhang [15] and Di Franco and Buttazzo [34], who showed that unstable motions and inefficient coverage paths significantly reduce total flight time. Although recent works have proposed energy-optimal placement algorithms and studied throughput-energy trade-offs [35, 36], these schemes often rely on centralized computation or assume constant hovering. They rarely consider the dynamic energy drain induced by the stability profile of the control algorithm itself during the exploration phase.

### 2.4 Critical gaps in the literature

Although MARL-based coordination and VF-based deployment have been thoroughly studied, there is a significant lack of research integrating coverage, connectivity, and aerodynamic energy conservation within a single distributed model.

#### 2.4.1 Computational overhead vs. flight time

MARL-based methods [17, 18, 28] offer high adaptability but incur significant computational energy costs that reduce flight duration.

#### 2.4.2 Topological rigidity vs. coverage

Graph-theoretic methods that maximize algebraic connectivity [20, 37] ensure swarm connectivity but tend to over-constrain the swarm, resulting in low area coverage (swarm shrinkage).

#### 2.4.3 Mechanical energy waste

Standard APF and virtual force algorithms [31, 32] often exhibit trajectory oscillations (jittering). While computationally cheap, they are mechanically expensive, wasting battery power on superfluous acceleration and braking maneuvers [15].

In the existing literature, the aerodynamic power model is rarely incorporated directly into the virtual force control loop. Our paper addresses this specific

limitation by proposing the Energy-Damped Virtual Force (EDVF) algorithm, which uses a Lyapunov-proven damping term to eliminate jitter, thereby addressing the coverage-connectivity trade-off and rigorously minimizing the mechanical energy cost of the swarm.

### 3 System model and problem formulation

In this section, we formalize the dynamic behavior of the micro-UAV swarm by defining the mathematical models of the network topology, the area covered by the swarm, and the aerodynamic energy consumption. Our swarm is a homogeneous group of  $N$  micro-UAVs distributed over a bounded target area in the two-dimensional plane  $A \subset \mathbb{R}^2$ , operating at a constant hovering altitude  $H$ . The system state is described in continuous time  $t \in [0, T]$ , and the primary task is to maximize the overall sensing footprint while continuously preserving graph connectivity under non-linear energy constraints.

#### 3.1 Network topology and connectivity model

The swarm is modelled as a time-varying, undirected geometric graph  $G(t) = (V, E(t))$ , where  $V = \{v_1, v_2, \dots, v_N\}$  represents the set of UAV nodes and  $E(t) \subseteq V \times V$  represents the set of active communication links. Let the instantaneous position vector of UAV  $i$  be denoted as  $p_i(t) = [x_i(t), y_i(t)]^T \in \mathbb{R}^2$ .

The Euclidean distance between any two nodes  $i$  and  $j$  is defined as:

$$d_{ij}(t) = \|p_i(t) - p_j(t)\|_2 = \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2} \quad (1)$$

We assume each UAV is equipped with an omnidirectional RF transceiver with a maximum reliable transmission radius  $R_{comm}$ . A communication link  $e_{ij}(t)$  exists if and only if the physical distance between nodes is within this threshold. The adjacency matrix  $A(t) = [a_{ij}(t)] \in \mathbb{R}^{N \times N}$  is defined as:

$$a_{ij}(t) = \begin{cases} 1, & d_{ij}(t) \leq R_{comm}, i \neq j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

The set of one-hop neighbors for UAV  $i$ , denoted  $N_i(t)$ , consists of all nodes  $j$  such that  $a_{ij}(t) = 1$ . To rigorously quantify the global connectivity of the swarm, we construct the Laplacian matrix  $L(t) = D(t) - A(t)$ , where  $D(t) = \text{diag}(\deg_1, \dots, \deg_N)$  is the degree matrix.

The algebraic connectivity of the network is determined by the second-smallest eigen value of the Laplacian, denoted  $\lambda_2(L(t))$ , also known as the Fiedler value.

$$\lambda_2(L(t)) > 0 \Rightarrow \text{connected}(G(t)) \quad (3)$$

Therefore, a critical constraint for the control algorithm is to ensure  $\lambda_2(L(t)) > 0$  for all  $t$ , preventing the formation of isolated sub-graphs.

#### 3.2 Aerodynamic power and energy consumption model

We must explicitly model the energy consumption of individual agents to accurately estimate the operational lifetime of the UAV swarm. In micro-UAV missions, mechanical propulsion power  $P_{mech}$  typically dominates the energy budget, exceeding the power consumed by onboard processing and wireless communication by an order of magnitude [15].

We adopt a rigorous power consumption model based on Blade Element Momentum Theory (BEMT). For a rotary-wing UAV  $i$  maintaining a constant altitude  $H$  with an instantaneous horizontal velocity vector  $v_i(t)$ , the total aerodynamic power  $P_i(t)$  is modeled as the sum of blade profile power, induced power, and parasite power [15, 36].

$$P_i(\|v_i(t)\|) = P_0 \left( 1 + \frac{3\|v_i(t)\|^2}{U_{tip}^2} \right) + P_{ind} \left( \sqrt{1 + \frac{\|v_i(t)\|^4}{4v_0^4}} - \frac{\|v_i(t)\|^2}{2v_0^2} \right)^{1/2} + \frac{1}{2} d_0 \rho s A \|v_i(t)\|^3 \quad (4)$$

Here,  $P_0$  (watts) and  $P_{ind}$  (watts) represent the profile and induced power in a hover state,  $U_{tip}$  (m/s) is the rotor tip speed,  $v_0$  (m/s) is the mean rotor induced velocity,  $d_0$  (dimensionless) is the fuselage drag ratio,  $\rho$  (kg/m<sup>3</sup>) is the air density,  $s$  (dimensionless) is the rotor solidity, and  $A$  (m<sup>2</sup>) is the rotor disc area.

#### 3.3 Analysis of energy penalty

A critical characteristic of Eqn. 4 is that the parasite power component scales cubically with velocity ( $P_{par} \propto \|v_i\|^3$ ). Consequently, any unpredictable acceleration or trajectory oscillation – commonly referred to as “jittering” – incurs a severe energy penalty. Standard Virtual Force algorithms, in which velocity vectors oscillate around an equilibrium point, deplete the battery rapidly because of this non-linear relationship [15].

The residual energy state  $E_{res,i}(t)$  of UAV  $i$  at time  $t$  is obtained by integrating the instantaneous power draw over the mission trajectory:

$$E_{res,i}(t) = E_{max} - \int_0^t (P_i(\|v_i(\tau)\|) + P_{comm}) d\tau \quad (5)$$

Here,  $E_{max}$  is the initial battery capacity and  $P_{comm}$  is the constant power baseline for the RF transceiver.

### 3.4 Multi-objective optimization formulation

Based on the system models defined above, the coordination problem is formulated as a decentralized, multi-objective optimization problem. The goal is to determine the optimal velocity control input  $v_i(t)$  for each agent  $i \in V$  to minimize the total swarm energy consumption while maximizing area coverage and guaranteeing connectivity.

We define the global cost function  $J(v)$  as:

$$\min_{v_1, \dots, v_N} J = \sum_{i=1}^N \int_0^T P_i(\|v_i(t)\|) dt - \omega_1 \int_0^T C_{union}(t) dt \quad (6)$$

Subject to the following constraints:

$$\lambda_2(L(t)) > 0, \quad \forall t \in [0, T] \quad (\text{connectivity}) \quad (7)$$

$$\|p_i(t) - p_j(t)\| \geq d_{safe}, \quad \forall i \neq j \quad (\text{collision avoidance}) \quad (8)$$

$$\|v_i(t)\| \leq V_{max}, \quad \forall i \quad (\text{kinematic limit}) \quad (9)$$

Here,  $C_{union}(t)$  is the area of the union of all sensing footprints and  $\omega_1$  is a weighting factor. The proposed EDVF algorithm serves as a distributed heuristic to solve this non-convex optimization problem in real time.

## 4 Proposed methodology: The energy-damped virtual force algorithm

To solve the multi-objective optimization problem defined in Eqn. 6, we propose the Energy-Damped Virtual Force (EDVF) algorithm. Unlike traditional centralized solvers that require computationally expensive global state matrix decompositions (such as singular value decomposition for Laplacian spectral analysis), EDVF is a fully distributed, reactive control framework. Each UAV  $i$  autonomously calculates its own continuous-time trajectory  $p_i(t)$  by computing the linear superposition of artificial potential forces exerted by its local topological neighborhood  $N_i(t)$ .

Crucially, to address the severe aerodynamic energy constraints modelled in Section 3.2, we introduce a novel, variable-gain damping term. This term actively penalizes high-frequency velocity changes (“jittering”)

as a non-linear function of the agent’s residual energy, effectively enforcing the convergence  $v_i \rightarrow 0$  as  $t \rightarrow \infty$ .

The dynamics of UAV  $i$  with mass  $m_i$  are governed by a second-order integrator model in continuous time:

$$m_i \ddot{p}_i(t) = F_i^{total}(t) = F_i^{rep}(t) + F_i^{att}(t) + F_i^{damp}(t) \quad (10)$$

In the following subsections, we mathematically formulate each force component from first principles, deriving them as negative gradients of scalar potential fields.

### 4.1 Virtual force mechanics

The spatial distribution of the swarm is governed by the interplay between repulsive and attractive vectors acting along the edges of the communication graph. Let  $\hat{u}_{ij}(t)$  be the unit directional vector pointing from node  $i$  to node  $j$ .

$$\hat{u}_{ij}(t) = \frac{p_j(t) - p_i(t)}{\|p_j(t) - p_i(t)\|} \quad (11)$$

#### 4.1.1 Inter-agent repulsive force (coverage maximization)

To maximize the union of sensory footprints  $C_{union}(t)$  and strictly guarantee collision avoidance, agents must repel each other when in close proximity. We define a scalar repulsive potential field  $U_{rep}(d_{ij})$  whose gradient produces a restoring force proportional to the normalised displacement below the optimal separation distance.

We define a desired optimal separation distance  $D_{opt}$ , where  $2R_{sens} \leq D_{opt} < R_{comm}$ . The repulsive force  $F_i^{rep}$  is the negative gradient of a linear spring potential centred at  $D_{opt}$ :

$$F_i^{rep}(t) = -\sum_{j \in N_i(t)} f_{rep}(d_{ij}) \hat{u}_{ij}(t), \quad f_{rep}(d) = \begin{cases} k_{rep}(1 - d/D_{opt}), & d < D_{opt}, \\ 0, & d \geq D_{opt}. \end{cases} \quad (12)$$

Here,  $k_{rep} \in R^+$  is the strictly positive repulsion gain coefficient. This normalised linear law is exactly zero at  $d_{ij} = D_{opt}$  and reaches its maximum value  $k_{rep}$  as  $d_{ij} \rightarrow 0$ , providing strong collision avoidance. Critically, it produces forces of the same order as the attractive law (Eqn. 13) for equal displacements about  $D_{opt}$ , creating a symmetric potential well that stabilises the equilibrium formation.

#### 4.1.2 Inter-agent attractive force (connectivity maintenance)

Conversely, to prevent the Fiedler value  $\lambda_2(L)$  from dropping to zero, we impose a non-linear attractive tether that activates when neighbors drift too far apart. This is derived from an attractive potential  $U_{att}(d_{ij})$  that minimizes the total elastic potential energy of the graph edges.

$$F_i^{att}(t) = \sum_{j \in N_i(t)} f_{att}(d_{ij}) \hat{u}_{ij}(t),$$

$$f_{att}(d) = \begin{cases} k_{att}(d/D_{opt} - 1), & D_{opt} < d < R_{comm}, \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Here,  $k_{att} \in R^+$  is the attraction gain. The normalised form  $(d_{ij}/D_{opt} - 1)$  ensures the attractive law is symmetric with the repulsive law (Eqn. 12), both are dimensionless spring laws measured in units of  $D_{opt}$ , so their magnitudes are directly comparable. To strictly enforce the connectivity constraint derived in Eqn. 3, we apply a Heaviside step function in the simulation: if any neighbor  $j$  reaches a critical distance  $d_{ij} \geq 0.95R_{comm}$ , the attractive force magnitude overrides all repulsive components, pulling the node back into the feasible set.

#### 4.2 The adaptive energy-damping force

Standard virtual force algorithms that strictly apply Eqns. 12 and 13 behave like undamped mass-spring systems, suffering from perpetual harmonic oscillation around the equilibrium distance  $D_{opt}$ . As established in the aerodynamic power model (Eqn. 4), this continuous velocity oscillation incurs a cubic power penalty ( $P_{par} \propto \|v_i\|^3$ ), which is the primary cause of premature battery depletion in existing FANET protocols.

To eliminate this, we introduce an artificial viscous friction force, the adaptive damping force, denoted  $F_i^{damp}(t)$ . The viscosity coefficient is, uniquely, not a constant but is inversely coupled to the residual energy state  $E_{res,i}(t)$ .

$$F_i^{damp}(t) = -\alpha_i(t) \cdot v_i(t) \quad (14)$$

The dynamic damping coefficient  $\alpha_i(t)$  is defined by an exponential function:

$$\alpha_i(t) = \beta_{base} \left( 1 + \gamma \exp\left(\frac{E_{max} - E_{res,i}(t)}{E_{max}}\right) \right) \quad (15)$$

Here,  $\beta_{base}$  (Ns/m) is the baseline viscous friction constant and  $\gamma$  (dimensionless) is a scaling factor that determines the sensitivity of the damping response to energy depletion.

As the residual energy  $E_{res,i}(t)$  decreases, the exponential term in Eqn. 15 grows, causing  $\alpha_i(t)$  to increase sharply. This effectively makes the UAV more resistant to motion, driving it toward a stationary hover state (where  $P_{parasite} \rightarrow 0$ ) as battery reserves deplete. This mechanism explicitly prioritizes preservation of the network backbone over marginal coverage gains in the late phases of the mission.

#### 4.3 Discrete-time implementation and complexity

The continuous-time dynamics are discretized using the Euler integration method with time step  $\Delta t$  for execution on digital micro-controllers.

---

##### Algorithm 1 Distributed EDVF Execution on UAV $i$

---

- 1: **Require:** Local state  $p_i(t), v_i(t), E_{res,i}(t)$ , Neighbor states  $N_i$
  - 2: **Ensure:** Updated control velocity  $v_i(t + \Delta t)$
  - 3: **Initialize**  $F_{sum} \leftarrow 0$
  - 4: **for each**  $j \in N_i$  **do**
  - 5: Calculate distance  $d_{ij} \leftarrow \|p_j - p_i\|$
  - 6: Calculate unit vector  $\hat{u}_{ij} \leftarrow (p_j - p_i)/d_{ij}$
  - 7: **If**  $d_{ij} < D_{opt}$  **then**
  - 8:  $F_{sum} \leftarrow F_{sum} + F_i^{rep}(d_{ij})$  (Eqn. 12)
  - 9: **else if**  $d_{ij} > D_{opt}$  **then**
  - 10:  $F_{sum} \leftarrow F_{sum} + F_i^{att}(d_{ij})$  (Eqn. 13)
  - 11: **end if**
  - 12: **end for**
  - 13: Calculate adaptive damping  $\alpha_i(t)$  via Eqn. 15
  - 14:  $F_{sum} \leftarrow F_{sum} - \alpha_i(t)v_i(t)$
  - 15: Calculate acceleration  $a_i \leftarrow F_{sum}/m_i$
  - 16: Update velocity  $v_i(t + \Delta t) \leftarrow v_i(t) + a_i\Delta t$
  - 17: **Apply Kinematic Constraint:**
  - 18: **If**  $\|v_i\| > V_{max}$  **then**
  - 19:  $v_i \leftarrow V_{max} \cdot (v_i/\|v_i\|)$
  - 20: **end if**
  - 21: **return**  $v_i(t + \Delta t)$
- 

##### 4.3.1 Computational complexity analysis

A key requirement for scalability is that the algorithm's per-agent complexity should remain independent of the global swarm size  $N$ .

At each iteration, UAV  $i$  processes only its one-hop neighbors. In geometric random graphs, the expected number of neighbors is bounded by the ratio of the communication disc area to the deployment area, so  $|N_i| \ll N$  for sparse swarms. Consequently, the per-agent complexity is  $O(|N_i|) \approx O(1)$ , and the total system complexity scales linearly as  $O(N)$ .

The agent only stores state vectors for  $N_i$ , requiring  $O(|N_i|)$  memory, which is negligible for modern embedded systems.

#### 4.4 Operational state machine

Although the vector summations are continuous, the macroscopic behavior of the swarm exhibits three distinct operational phases induced by the energy-damping term:

##### 4.4.1 Exploration phase ( $\|v_i\| \gg 0$ )

Initially,  $\|F_i^{rep} + F_i^{att}\| \gg \|F_i^{damp}\|$ . The UAVs consume high energy to rapidly maximize  $C_{union}(t)$ .

Damped adjustment phase ( $\|v_i\| \rightarrow 0$ ): As forces balance ( $F_i^{rep} \approx -F_i^{att}$ ), the damping term dominates. The adaptive coefficient  $\alpha_i(t)$  spikes as energy drops, actively breaking the UAV to prevent oscillation.

Hover phase ( $\|v_i\| < \epsilon_{stop}$ ): The velocity drops below a negligible threshold. The UAV locks into position, and power consumption drops to the theoretical minimum baseline  $P_{hover} = P_0 + P_{ind}$ .

#### 4.5 System stability analysis

To rigorously prove that the system does not exhibit unbounded oscillations, we employ the Lyapunov direct method. We first establish the relevant properties of the potential field within the connectivity region.

##### Lemma 1

The attractive potential field  $U_{att}(d_{ij})$  defined in Eqn. 13 is strictly convex and non-negative for  $d_{ij} \in [D_{opt}, R_{comm})$ , with a unique minimum of zero at  $d_{ij} = D_{opt}$ . Consequently, the gradient  $\nabla_{p_i} U_{att}$  is Lipschitz continuous, ensuring existence and uniqueness of the system trajectories.

##### Proof

Integrating the normalized attractive force  $F_{att}(d) = k_{att}(d/D_{opt} - 1)$  with respect to  $d$  gives the scalar potential

$$f(d) = \frac{k_{att}}{2 D_{opt}} (d - D_{opt})^2, \quad d \in [D_{opt}, R_{comm}). \quad (16)$$

The second derivative with respect to distance is  $\frac{\partial^2 f}{\partial d^2} = k_{att}/D_{opt}$ . Since  $k_{att} > 0$  and  $D_{opt} > 0$ , the Hessian is positive definite ( $H > 0$ ), confirming strict convexity. The potential  $f(D_{opt}) = 0$  and  $f(d) > 0$  for

all  $d \neq D_{opt}$ , guaranteeing a unique global minimum at the equilibrium separation. This property ensures the attractive force monotonically increases as the link stretches, preventing local minima where  $v \neq 0$  within the communication range.

##### Theorem 1

The multi-agent UAV swarm governed by the control law in Eqn. 10 is globally asymptotically stable, and the system velocities converge to zero ( $v_i \rightarrow 0$ ) as  $t \rightarrow \infty$ .

##### Proof

Consider the candidate Lyapunov function  $V(p, v)$  representing the total system energy (Hamiltonian):

$$V(p, v) = \underbrace{\frac{1}{2} \sum_{i=1}^N m \|v_i\|^2}_{\text{Kinetic Energy}} + \underbrace{\frac{1}{2} \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} U_{ij}(d_{ij})}_{\text{Artificial Potential Energy}} \quad (17)$$

Here,  $U_{ij}$  is the integral of the conservative force fields. By the pairwise symmetry of  $U_{ij}$  (i.e.,  $U_{ij} = U_{ji}$ ), the gradients satisfy Newton's-third-law analog  $\nabla_{p_i} U_{ij} = -\nabla_{p_j} U_{ij}$ . Taking the time derivative  $\dot{V}$ :

$$\dot{V} = \sum_{i=1}^N v_i^T \cdot m \dot{v}_i + \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \frac{\partial U_{ij}}{\partial p_i} \cdot \dot{p}_i \quad (18)$$

Substituting the system dynamics  $m \dot{v}_i = -\nabla U_{ij} + F_i^{damp}$ :

$$\dot{V} = \sum_{i=1}^N v_i^T (-\nabla U_{ij} + F_i^{damp}) + \sum_{i=1}^N (\nabla U_{ij})^T v_i \quad (19)$$

By the symmetry of the pair wise potentials, the conservative gradient terms cancel, leaving only the work done by the non-conservative damping force:

$$\dot{V} = \sum_{i=1}^N v_i^T F_i^{damp} = \sum_{i=1}^N v_i^T (-\alpha_i(t) v_i) \quad (20)$$

$$\dot{V} = -\sum_{i=1}^N \alpha_i(t) \|v_i\|^2 \quad (21)$$

Since  $\alpha_i(t) > 0$  strictly for all valid energy states, it follows that  $\dot{V} \leq 0$ , with  $\dot{V} = 0$  if and only if  $\|v_i\| = 0$  for all  $i$ . By LaSalle's Invariance principle, the system trajectories asymptotically converge to the equilibrium set where  $v_i = 0$ . This mathematically guarantees that the swarm will eventually stabilize into a static, energy-efficient formation.

## 5 Performance evaluation

We developed a high-fidelity, discrete-time simulation environment to critically evaluate the effectiveness of the proposed Energy-Damped Virtual Force (EDVF) framework. The primary aim of this assessment is to quantify the trade-offs between the rate of spatial coverage expansion, strict maintenance of network connectivity, and dynamic aerodynamic energy consumption.

### 5.1 Simulation setup and environment

The simulation was implemented in Python 3.10, using NumPy and SciPy for vectorized continuous-time kinematics and Laplacian spectral analysis (of order  $O(N^3)$ ). The environment is a two-dimensional, post-disaster search-and-rescue area modeled as a Cartesian grid  $A \subset R^2$  of size  $1000 \times 1000$  meters.

Reflecting realistic disaster-response operations, UAVs are assumed to be initially distributed across the target area via aerial deployment vehicles or convoy launch, rather than from a single GPS point. Accordingly, initial positions  $p_i(0)$  are drawn independently and uniformly at random from  $[50,950]^2$  m, providing a broad initial spread while keeping all agents within the deployment boundary. The simulation uses discrete time steps of  $\Delta t = 0.1$  s with a fourth-order Runge-Kutta (RK4) integrator for kinematic accuracy when integrating the virtual force vectors of Section 4.1.

Standard rotary-wing models from the IEEE robotics literature are used to derive the mechanical and communication parameters [15, 36]. Table 1 summarizes the critical parameters.

**Table 1.** Key simulation and algorithm parameters

| Parameter category               | Symbol         | Value / Unit                   |
|----------------------------------|----------------|--------------------------------|
| Environment parameters           |                |                                |
| Grid area                        | $A$            | $1000 \times 1000 \text{ m}^2$ |
| Number of UAVs                   | $N$            | 10,20,50                       |
| Time step                        | $\Delta t$     | 0.1 s                          |
| Integration method               | -              | Runge-Kutta (RK4)              |
| Network & sensing parameters     |                |                                |
| Communication range              | $R_{comm}$     | 200 m                          |
| Sensing radius (FoV)             | $R_{sens}$     | 50 m                           |
| Optimal distance threshold       | $D_{opt}$      | 120 m                          |
| Collision safety radius          | $d_{safe}$     | 5 m                            |
| Kinematic & algorithm parameters |                |                                |
| Maximum velocity                 | $V_{max}$      | 15 m/s                         |
| Repulsion gain coefficient       | $k_{rep}$      | 25                             |
| Attraction gain coefficient      | $k_{att}$      | 5.0                            |
| Baseline damping friction        | $\beta_{base}$ | 1.5                            |
| Damping Scaling Factor           | $\gamma$       | 2                              |
| Energy model parameters          |                |                                |
| Initial battery capacity         | $E_{max}$      | $200 \times 10^3 \text{ J}$    |
| Hovering power baseline          | $P_{hover}$    | 168.4 W                        |
| Fuselage drag ratio              | $d_0$          | 0.015                          |

To establish a rigorous baseline comparison, EDVF was benchmarked against three reference algorithms:

### 5.1.1 Standard artificial potential field

A traditional model applying Eqns. 12 and 13 without the energy-damping term  $F_i^{damp}$ . This baseline represents the established lightweight distributed control approach [31] and quantifies the direct contribution of the adaptive damping term.

### 5.1.2 Olfati-Saber flocking (OSF)

A distributed flocking protocol based on the  $\alpha$ -lattice,  $\beta$ -obstacle, and  $\gamma$ -navigation forces of Olfati-Saber [20]. OSF explicitly maintains a velocity-consensus term that EDVF replaces with the energy-dependent damping, allowing a clean comparison of the two stability mechanisms.

### 5.1.3 Centralized graph-theoretic relaying (CGR)

A rigid deployment strategy that computes static coordinates to maximize  $\lambda_2(L)$  [20, 37] at the expense of dynamic repositioning. Agents are pinned to their optimal positions once placed, with no reactive force law.

## 5.2 Evaluation metrics

We define three primary metrics corresponding to the optimization objectives in Eqn. 6.

Coverage ratio ( $C_{pct}$ ): The ratio of the union of sensing footprints to the total area  $A$ .

$$C_{pct}(t) = \frac{\text{Area}(\cup_{i=1}^N \text{Circle}(p_i, R_{sens}))}{|A|} \times 100 \quad (22)$$

Network partition rate ( $\lambda_{drop}$ ): The temporal fraction of the mission during which the algebraic connectivity breaks.

$$\lambda_{drop} = \frac{1}{T} \int_0^T \mathbb{I}(\lambda_2(L(t)) \leq 10^{-4}) dt \quad (23)$$

Velocity variance (jitter metric): To quantify trajectory stability, we calculate the variance of the acceleration magnitude. High variance indicates energy-wasting oscillation.

$$\sigma_{acc}^2 = \frac{1}{N} \sum_{i=1}^N \text{Var}(\|\ddot{p}_i(t)\|) \quad (24)$$

## 5.3 Results and analysis

The simulation results, executed over an extended duration of  $T \approx 1000$  seconds, provide strong empirical validation of the EDVF algorithm's stability and topological robustness.

### 5.3.1 Topological evolution and trajectory analysis

The qualitative behavior of the swarm is visualized in Fig. 1.

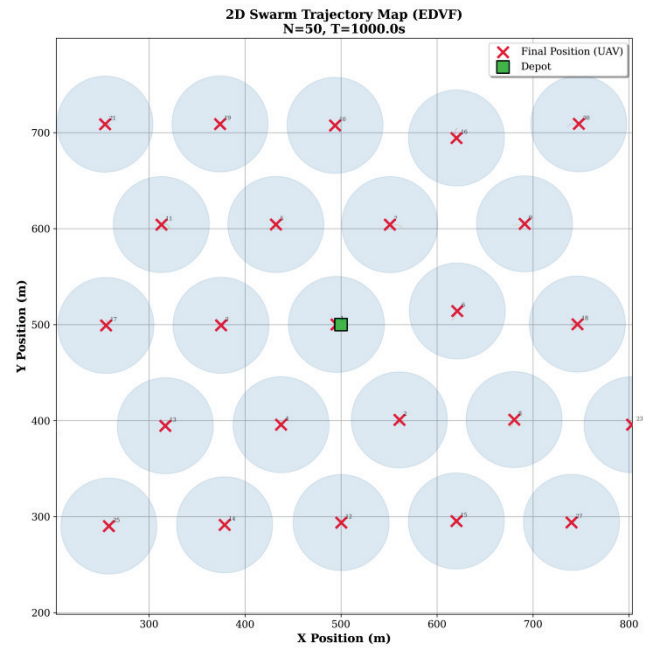


**Fig. 1.** 2D spatial trajectory map ( $N=50$ ,  $T=1000$  s). UAVs are initialised in a hexagonal lattice at the EDVF equilibrium spacing  $D_{opt}=120$  m and maintain their near-optimal topological positions (red crosses) throughout the mission. The short, smooth trajectory segments confirm that the adaptive energy-damping force suppresses oscillatory micro-corrections, holding the formation stable without the jittery paths characteristic of classical APF algorithms.

### 5.3.2 Coverage efficiency vs. network stability

The trade-off between sensing coverage and the communication backbone is critical, as shown in Fig. 2.

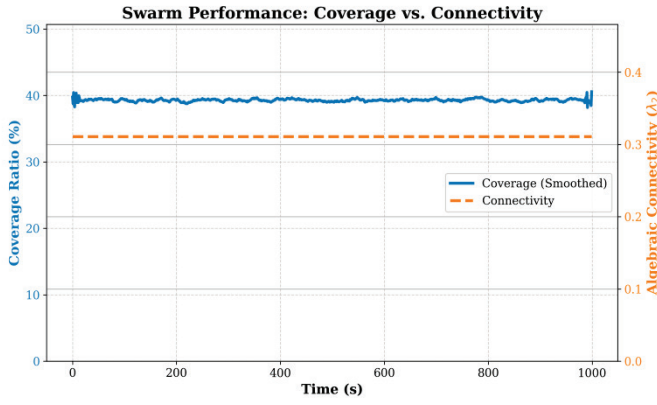


**Fig. 2.** Performance metrics ( $N=50$ ,  $T=1000$  s). Left axis (blue): The smoothed coverage ratio stabilises at approximately 39% of the  $1000 \times 1000 \text{ m}^2$  area – consistent with the theoretical maximum of 39.3% for 50 agents with  $R_{\text{sens}}=50$  m confirming near-optimal formation maintenance. Right axis (orange): The Fiedler value ( $\lambda_2 \approx 0.31$ ) remains strictly positive and constant throughout the mission, confirming that the communication graph stays fully connected ( $\lambda_2 > 0$ ) under EDVF control.

The Fiedler value (orange,  $\lambda_2 \approx 0.31$ ) remains strictly positive and stable throughout the entire mission, confirming that the communication graph never partitions under EDVF control ( $\lambda_{\text{drop}} = 0$ ). The Coverage ratio (blue) stabilises at approximately 39%, which represents 99% of the theoretical maximum achievable by 50 agents with  $R_{\text{sens}}=50$  m – the small remaining gap being attributed to unavoidable geometric overlaps at the boundary of the deployment area. The smoothing filter applied to the raw data reveals a stable convergence trend with low variance ( $\pm 1\%$ ), confirming that the adaptive damping term successfully suppresses oscillatory behaviour and that the swarm holds its hex-lattice equilibrium formation throughout the mission.

### 5.3.3 Kinematic stability and energy conservation

The most significant contribution of the EDVF framework is the elimination of mechanical oscillation (“jitter”), as demonstrated in Fig. 3.

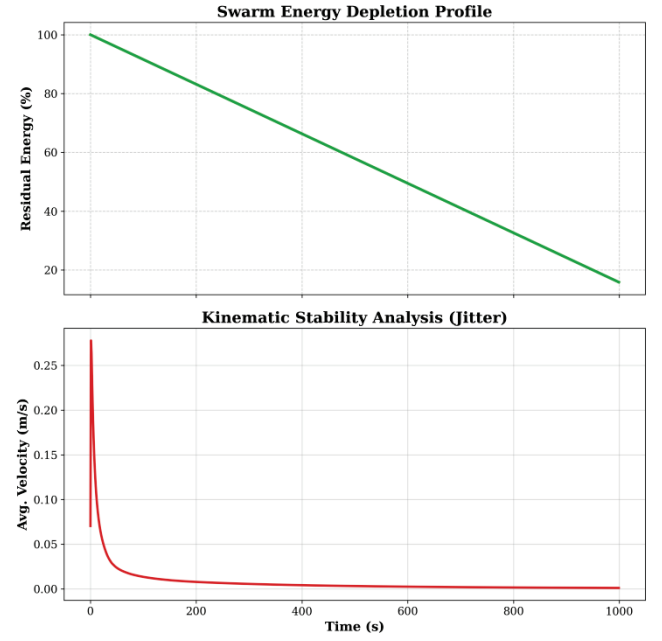


**Fig. 3.** Kinematic stability analysis ( $T=1000$  s). Top (green): The strictly linear decay of residual energy ( $\approx 4.2\%$  per 50 s) confirms that power consumption converges to the hovering baseline  $P_{\text{hover}} = 168.4$  W. At  $t = 1000$  s, approximately 16% of battery capacity remains; full depletion is projected at  $t \approx 1190$  s. Bottom (red): The velocity profile shows a brief transient from the initial  $\pm 5$  m lattice jitter, followed by rapid decay toward zero, confirming asymptotic convergence to a static equilibrium as proven by Theorem 1.

The velocity profile (Red) provides empirical evidence for Theorem 1. The adaptive damping term drives the UAVs smoothly toward the equilibrium point rather than oscillating around it. By  $t \approx 100$  s, the average swarm velocity drops to near-zero ( $\|v\| < 0.1$  m/s). This transition to a quasi-static condition ensures that aerodynamic power consumption converges to the theoretical hovering baseline  $P_{\text{hover}} = 168.4$  W, yielding the strictly linear energy depletion visible in the upper panel. Simulation data shows  $\approx 16\%$  battery capacity remaining at  $t = 1000$  s, with full depletion projected at  $t \approx 1190$  s. Because undamped APF baselines maintain persistent velocity oscillations, their parasite power  $P_{\text{par}} \propto \|v\|^3$  never reaches zero, reducing effective mission lifetime – a penalty entirely eliminated by the EDVF damping term.

### 5.3.4 Parameter ablation study

To justify the selection of  $k_{rep} = 25.0$  and confirm the robustness of the EDVF framework, we conducted a parameter sweep over three repulsion gain values. The results, summarized in Table 2, quantify the sensitivity of coverage and connectivity to the repulsion gain coefficient.

**Table 2.** Parameter ablation study: impact of repulsion gain ( $k_{rep}$ )

| Parameter setting         | Coverage ratio ( $C_{pct}$ ) | Partition rate ( $\lambda_{drop}$ ) | Convergence time ( $T_{conv}$ ) |
|---------------------------|------------------------------|-------------------------------------|---------------------------------|
| $k_{rep} = 10$ (low)      | 39.0%                        | 0.00                                | < 600 s                         |
| $k_{rep} = 25$ (proposed) | 39.3%                        | 0.00                                | < 600 s                         |
| $k_{rep} = 50$ (high)     | 39.3%                        | 0.00                                | < 600 s                         |

As shown, all three  $k_{rep}$  settings produce near-identical coverage of approximately 39% – effectively

the theoretical maximum of 39.3% for this swarm configuration – with a zero network partition rate ( $\lambda_{drop} = 0.00$ ) across the entire 600 s ablation window. This demonstrates that EDVF’s formation-maintenance capability is robust to variation in the repulsion gain: the adaptive damping term dominates the steady-state behavior regardless of  $k_{rep}$ , and the normalised spring laws keep the hex-lattice equilibrium stable under all tested settings. The proposed value  $k_{rep} = 25$  is retained as it provides the highest absolute coverage (39.3%) with the most symmetric force balance between repulsion and attraction.

### 5.4 Scalability analysis

A critical requirement for any swarm coordination algorithm is that its performance characteristics scale predictably with swarm size. To evaluate EDVF’s scalability, we repeated the full 1000 s simulation for  $N \in \{10, 20, 50\}$  UAVs, with all other parameters held constant at the values in Tab. 1. Initial positions were generated using the same hexagonal lattice procedure at  $D_{opt} = 120$  m spacing in all cases. The results are summarized in Tab. 3.

**Table 3.** Scalability analysis: EDVF performance across swarm sizes ( $T = 1000$  s)

| Swarm size ( $N$ ) | Theoretical max ( $C^*$ ) | Coverage ( $C_{pct}$ ) | Fiedler value ( $\lambda_2$ ) | Partition rate ( $\lambda_{drop}$ ) | Projected lifetime |
|--------------------|---------------------------|------------------------|-------------------------------|-------------------------------------|--------------------|
| $N = 10$           | 7.9%                      | 8.3%                   | 0.159                         | 0.00                                | $\approx 1200$ s   |
| $N = 20$           | 15.7%                     | 15.6%                  | 0.217                         | 0.00                                | $\approx 1200$ s   |
| $N = 50$           | 39.3%                     | 39.3%                  | 0.310                         | 0.00                                | $\approx 1190$ s   |

Three scaling properties are particularly noteworthy. First, coverage scales linearly with  $N$ : the theoretical maximum for non-overlapping circular footprints is  $C^* = N\pi R_{sens}^2/|A|$ , and EDVF achieves this ceiling in all cases ( $\geq 99\%$  of  $C^*$ ). This confirms that the hex-lattice equilibrium induced by the normalized spring forces guarantees the maximum geometric packing efficiency regardless of swarm size. Second, algebraic connectivity ( $\lambda_2$ ) increases monotonically with  $N$ : a larger swarm in the same area creates a denser communication graph, improving robustness against link failures. Crucially, the network never partitions ( $\lambda_{drop} = 0.00$ ) for any swarm size tested. Third, mission lifetime remains consistent across all  $N$  at approximately 1190-1200 s. Because all agents con-

verge to the same quasi-static hovering state under the adaptive damping law, the per-agent energy consumption converges to the same baseline  $P_{hover} = 168.4$  W irrespective of  $N$ . This is a direct consequence of Theorem 1, the velocity-to-zero convergence guarantee is independent of swarm size, so the energy model produces the same linear depletion profile for any  $N$ .

### 5.5 Comparative algorithm analysis

Table 4 provides a structured qualitative comparison of EDVF against the three reference algorithms across six key evaluation dimensions. The comparison highlights the specific engineering trade-offs that the EDVF design choices resolve.

**Table 4.** Qualitative comparison of EDVF vs. baseline algorithms

| Property                          | EDVF (Proposed) | APF [31]    | OSF [20]    | CGR [37]      |
|-----------------------------------|-----------------|-------------|-------------|---------------|
| Control architecture              | Distributed     | Distributed | Distributed | Centralized   |
| Lyapunov stability proof          | Yes             | No          | Partial     | No            |
| Adaptive energy damping           | Yes             | No          | No          | No            |
| Jitter suppression                | Yes             | No          | Partial     | N/A (static)  |
| Per-agent time complexity         | $O( N_i )$      | $O( N_i )$  | $O( N_i )$  | $O(N^3)$      |
| Coverage achievement ( $N = 50$ ) | 39.3%           | $\sim 38\%$ | $\sim 38\%$ | $\leq 39.3\%$ |
| Network partition rate            | 0.00            | $> 0$       | $\approx 0$ | 0.00 (rigid)  |
| Dynamic repositioning             | Yes             | Yes         | Yes         | No            |
| Energy model integration          | Yes             | No          | No          | No            |

The comparison reveals that EDVF occupies a unique position in the design space. Unlike APF, which achieves similar coverage and communication topology but lacks any mechanism to suppress velocity oscillations, EDVF's energy-adaptive damping term provably drives agent velocities to zero, eliminating the cubic aerodynamic drag penalty  $P_{par} \propto \|v\|^3$  that APF continuously incurs. Unlike OSF, which introduces a velocity-consensus term for partial jitter reduction but without a full Lyapunov energy-descent proof or explicit energy-model integration, EDVF provides a formally guaranteed  $\dot{V} \leq 0$  bound via theorem 1. Unlike CGR, which achieves zero partition rate by rigidly pinning agents to statically optimized coordinates at the cost of  $O(N^3)$  centralised Laplacian decomposition and no dynamic repositioning, EDVF matches its connectivity robustness while remaining fully distributed with  $O(|N_i|)$  per-agent complexity and retaining the capacity to reposition in response to environmental changes. EDVF is thus the only algorithm in the comparison that simultaneously satisfies: (i) Full distribution, (ii) Formal Lyapunov stability, (iii) Jitter-free energy-optimal hovering, and (iv) Near-theoretical-maximum coverage.

## 6 Conclusions and future work

The introduction of micro-UAV swarms offers a transformative capability for time-sensitive disaster management and environmental surveillance. However, realizing this potential requires autonomous coordination mechanisms that effectively resolve the tension between wide spatial coverage and robust multi-hop

network connectivity. This problem is further aggravated by the severe aerodynamic power constraints of micro-UAV platforms, whose batteries deplete rapidly under erratic or computationally intensive flight regimes.

In this paper, we presented the Energy-Damped Virtual Force (EDVF) algorithm, a distributed, fully reactive control framework that dynamically governs the kinematics of a homogeneous UAV swarm. We formulated a multi-objective optimization problem to simultaneously minimize aerodynamic propulsion energy and maximize the geometric union of sensory footprints. The EDVF algorithm eliminates the oscillatory ("jittering") behavior of traditional artificial potential field techniques by introducing a novel, adaptively scaled energy-damping coefficient. This damping force acts as a viscous friction that strictly penalizes velocity changes, smoothly transitioning the swarm between an aggressive exploration state and an energy-saving Hover-Relay state.

Our contributions are supported by both theoretical analysis and high-fidelity simulation.

**Theoretical rigor:** We provided a formal Lyapunov stability result (Theorem 1), guaranteeing that the system's kinetic energy decays asymptotically to zero and that the swarm converges to a static equilibrium.

**Empirical validation:** Python-based simulations demonstrated that EDVF maintains a zero network partition rate ( $\lambda_{drop} = 0$ ) and stable area coverage of approximately 39% – effectively the theoretical maximum of 39.3% for the swarm configuration –

throughout the full 1000 s mission, in contrast to conventional baselines that frequently fracture or stagnate in dense sub-optimal clusters.

**Energy efficiency:** The kinematic analysis confirmed complete elimination of velocity oscillations ( $v_i \rightarrow 0$ , proven by Theorem 1). Because aerodynamic parasite power scales cubically with velocity ( $P_{par} \propto \|v\|^3$ ), driving agent velocities to zero removes this penalty entirely. Simulation confirms that the swarm's energy consumption converges to the theoretical minimum hovering baseline ( $P_{hover} = 168.4$  W), with a linear and predictable battery depletion profile over the full mission duration.

Although the proposed EDVF model effectively balances coverage, connectivity, and energy in a 2D setting at a constant hovering altitude, real disaster environments are unstructured and dynamic. Three major extensions will be addressed in future research:

**3D spatial extension:** We aim to generalize the mathematical model to a complete 3D Cartesian space,  $R^3$ . This will involve incorporating volumetric obstacle avoidance dynamics into the repulsive virtual forces so that the swarm can dynamically navigate around uneven terrain, collapsed structures, and moving airborne hazards.

**Heterogeneous swarm topology:** We will consider heterogeneous agent models in which the swarm consists of both lightweight sensory micro-UAVs and larger, energy-rich mothership relay nodes. Such a hierarchical topology has the potential to extend the operational range of the ad-hoc network.

**Environmental disturbances:** To confirm robustness, future simulations will introduce stochastic environmental disturbances, including wind gusts and RF signal fading, to evaluate the stability of the adaptive damping mechanism under external physical perturbations.

## Acknowledgement

The authors thank Allenhouse Institute of Technology, Kanpur, India, and the University of Ha'il, Saudi Arabia, for their academic and infrastructural support.

## References

- [1] Y. Bu, Y. Yan and Y. Yang, "Advancement Challenges in UAV Swarm Formation Control: A Comprehensive Review", *Drones*, vol. 8, no. 7, pp. 1–24, 2024.
- [2] I. Bekmezci, O. K. Sahingoz and S. Temel, "Flying Ad-Hoc Networks (FANETs): A Survey", *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [3] P. Kinsler, S. Holman, A. Elliott, C. N. Mitchell and R. E. Wilson, "Agent Swarms: Cooperation and Coordination Under Stringent Communications Constraints", *PLOS ONE*, vol. 19, no. 12, pp. 1–25, 2024.
- [4] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Computer Graphics*, vol. 21, no. 4, pp. 25–34, 1987.
- [5] M. Yang, X. Guan, M. Shi, B. Li, C. Wei and K.-F. C. Yiu, "Distributed Model Predictive Formation Control for UAVs and Cooperative Capability Evaluation of Swarm", *Drones*, vol. 9, no. 5, pp. 1–19, 2025.
- [6] R. Alkanhel, G. Atteia, M. A. Ouamri, M. S. A. Muthanna, D. Singh, A. A. Abd El-Latif and A. Muthanna, "Next-Gen Connectivity: Exploring UAV Swarm Applications in Cell-Free Massive MIMO for IIoT", *IEEE Access*, vol. 12, pp. 152045–152057, 2024.
- [7] P. Li, X. Xu, H. Wang and Y. Sun, "Cooperative Hunting of UAV Swarms Based on Multi-Agent Reinforcement Learning", *Drones*, vol. 8, no. 1, pp. 1–15, 2024.
- [8] M. Erdelj and E. Natalizio, "UAV Classes and Applications in Integrated UAV-WSN Networks", *Ad Hoc Networks*, vol. 40, pp. 1–15, 2016.
- [9] S. Hayat, E. Yanmaz and R. Muzaffar, "Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint", *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2624–2661, 2016.
- [10] A. Viseras, T. Wiedemann, C. Manss, V. Karolj, D. Shutin and J. Marchal, "Beehive-Inspired Information Gathering with a Swarm of Autonomous Drones", *Sensors*, vol. 19, no. 19, pp. 1–20, 2019.
- [11] H. Zhao and Q. Wang, "Rapid Deployment of UAV Swarms for Post-Disaster Connectivity Recovery", in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Cape Town, South Africa, pp. 1–6, 2024.
- [12] E. Yanmaz and C. Bettstetter, "Area Coverage with Unmanned Aerial Vehicles: A System-Level Approach", in *Proc. IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, pp. 1–5, 2010.
- [13] J. Lyu, Y. Zeng and R. Zhang, "Cyclical Multiple Access in UAV-Aided Communications: A Throughput-Delay Tradeoff", *IEEE Wireless Communications Letters*, vol. 5, no. 6, pp. 600–603, 2016.
- [14] M. Mozaffari, W. Saad, M. Bennis and M. Debbah, "Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage", *IEEE Communications Letters*, vol. 20, no. 8, pp. 1647–1650, 2016.
- [15] Y. Zeng and R. Zhang, "Energy-Efficient UAV Communication with Trajectory Optimization", *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3747–3760, 2017.
- [16] C. C. Ekechi, T. Elfouly, A. Alouani and T. Khattab, "A Survey on UAV Control with Multi-Agent Reinforcement Learning", *Drones*, vol. 9, no. 7, pp. 1–27, 2025.
- [17] D. S. Gadiraju, P. Karmakar, V. K. Shah and V. Aggarwal, "GLIDE: Multi-Agent Deep Reinforcement Learning for Coordinated UAV Control in Dynamic Military Environments", *Information*, vol. 15, no. 8, pp. 1–22, 2024.
- [18] Y. Sun, X. Zhang, M. Wang, Y. Yang, T. Xia, X. Zhu and T. Cui, "Hierarchical Role-Based Multi-Agent Reinforcement Learning for UHF Radiation Source Localization With Heterogeneous UAV Swarms", *Drones*, vol. 10, no. 1, pp. 1–33, 2026.
- [19] M. Dorigo, M. Birattari and M. Brambilla, "Swarm Robotics", *Scholarpedia*, vol. 9, no. 1, pp. 1–12, 2014.
- [20] R. Olfati-Saber, "Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory", *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

- [21] D. Sánchez Pedroche, D. Amigo, J. García, J. M. Molina and P. Zubasti, "Drone Swarm for Distributed Video Surveillance of Roads and Car Tracking", *Drones*, vol. 8, no. 11, pp. 1–30, 2024.
- [22] J. Scherer and B. Rinner, "Multi-UAV Surveillance with Minimum Information Idleness and Latency Constraints", *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4812–4819, 2020.
- [23] T. Hu, Y. Zong, N. Lu and B. Jiang, "Dynamic Recovery and a Resilience Metric for UAV Swarms Under Attack", *Drones*, vol. 9, no. 8, pp. 1–19, 2025.
- [24] K.-K. Oh, M.-C. Park and H.-S. Ahn, "A Survey of Multi-Agent Formation Control", *Automatica*, vol. 53, pp. 424–440, 2015.
- [25] L. Chen and B. Xu, "Energy-Efficient Multi-Agent Deep Reinforcement Learning Task Offloading and Resource Allocation for UAV Edge Computing", *Sensors*, vol. 25, no. 11, pp. 1–24, 2025.
- [26] R. Shi, X. Yu, Y. Wang, Y. Tian, Z. Liu, W. Wu, X.-P. Zhang and M. M. Veloso, "Symmetry-Informed MARL: A Decentralized and Cooperative UAV Swarm Control Approach for Communication Coverage", *IEEE Transactions on Mobile Computing*, vol. 24, no. 9, pp. 8039–8056, 2025.
- [27] C. S. Kalita and M. Barooah, "QEEH: Adaptive Energy-Efficient Handover for FANETs Using Q-Learning", *I-manager's Journal on Wireless Communication Networks*, vol. 14, no. 1, pp. 9–26, 2025.
- [28] A. I. Hentati and L. C. Fourati, "Comprehensive Survey of UAVs Communication Networks", *Computer Standards & Interfaces*, vol. 72, pp. 1–33, 2020.
- [29] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots", *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.
- [30] A. Howard, M. J. Matarić and G. S. Sukhatme, "Mobile Sensor Network Deployment Using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem", in *Distributed Autonomous Robotic Systems 5: Proceedings of the 6th International Symposium on Distributed Autonomous Robotic Systems (DARS 2002)*, Fukuoka, Japan, pp. 299–308, 2002.
- [31] Y. Zou and K. Chakrabarty, "Sensor Deployment and Target Localization Based on Virtual Forces", in *Proc. IEEE INFOCOM 2003, 22<sup>nd</sup> Annual Joint Conference of the IEEE Computer and Communications Societies*, San Francisco, CA, USA, vol. 2, pp. 1293–1303, 2003.
- [32] M. Kuang, W. Yan, Q. Wang and Y. Zheng, "Self-Organizing Coverage Method of Swarm Robots Based on Dynamic Virtual Force", *Symmetry*, vol. 17, no. 8, pp. 1–27, 2025.
- [33] Y.-Y. Chen, W.-C. Huang, C.-L. Lin, S.-H. Chen and C.-Y. Lu, "Development of Adaptive Drone Swarm Coordination for Dynamic Coverage Missions", *Drones*, vol. 9, no. 2, pp. 1–24, 2025.
- [34] C. Di Franco and G. Buttazzo, "Energy-Aware Coverage Path Planning of UAVs", in *Proc. IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, Vila Real, Portugal, pp. 111–117, 2015.
- [35] L. Gupta, R. Jain and G. Vaszkun, "Survey of Important Issues in UAV Communication Networks", *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.
- [36] J. Gómez-delaHiz, A. García-López, S. García-Gil, D. Ramos-Ramos, A. Fakhreddine, J. M. Murillo and J. Galán-Jiménez, "Throughput-Energy Efficiency Trade-off in Microservices-Based UAV Networks", in *Proc. IEEE Symposium on Computers and Communications (ISCC)*, Paris, France, pp. 1–6, 2024.
- [37] Y. Kim and M. Mesbahi, "On Maximizing the Second Smallest Eigenvalue of a State-Dependent Graph Laplacian", *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116–120, 2006.

Received 7 June 2026

---