

Article

LLM-Assisted Mission Planning, Predictive Coordination, and Adaptive Topology Management for Resilient USV Swarm Control Under Communication Denial

Xingda Li *, Jianqiang Zhang, Yiping Liu, Pengfei Zhang and Ling Tan

Naval University of Engineering, Wuhan 430030, China; 0909061042@nue.edu.cn (J.Z.); d25382602@nue.edu.cn (Y.L.); m23181105@nue.edu.cn (P.Z.); d25382609@nue.edu.cn (L.T.)

* Correspondence: m24382604@nue.edu.cn

Highlights

What are the main findings?

- A three-layer control architecture integrating LLM-assisted strategic planning, predictive tactical coordination, and DMPC + ADMM execution enables resilient USV swarm coordination under communication denial.
- The predictive coordination mechanism reduces formation error by 76.0% for the particular simulation cases investigated (DMPC + ADMM validated, Cohen's $d = 0.88$, $p < 0.001$, Bonferroni-corrected) during 60 s communication outages compared to zero-hold prediction.

What are the implications of the main findings?

- The adaptive topology manager reduces communication overhead by 71.7% (for the particular cases investigated, from 6.00 to 1.70 messages per step) under the evaluated simulation conditions while maintaining formation accuracy, with hysteresis limiting topology switches to ≤ 1 per 150 s mission.
- Stability analysis based on multiple Lyapunov functions and average dwell time theory guarantees global uniform exponential stability under topology switching, with explicit error bounds under communication denial validated through 1250 simulation trials.

Abstract

Unmanned surface vehicle (USV) swarms operating in communication-denied maritime environments face degraded formation control when inter-agent state exchange is disrupted. This paper presents a three-layer control architecture integrating (1) LLM-assisted strategic mission planning with formal safety verification, (2) predictive tactical coordination combining physics-based motion extrapolation with online-learned neighbor behavior models, and (3) DMPC + ADMM execution for constrained formation control. The predictive coordination module in simulation-based evaluation across five representative scenarios reduces formation error by 76.0% (under simulation conditions) during 60 s communication outages compared to zero-hold prediction (Cohen $d = 0.88$, $p < 0.001$, DMPC + ADMM validated). The adaptive topology manager dynamically selects among star, mesh, and tree configurations via a utility function balancing communication quality, threat exposure, and overhead, reducing communication overhead by 71.7% (for the particular cases investigated) under the evaluated conditions while maintaining formation accuracy. Stability analysis using multiple Lyapunov

Academic Editor: Gianfranco Parlangeli

Received: 16 June 2026
Revised: 27 July 2026
Accepted: 28 July 2026
Published: 2 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

functions and average dwell time theory guarantees global uniform exponential stability under topology switching, with explicit error bounds under communication denial derived via Gronwall-type arguments. The framework is validated through 1250 simulation trials across five scenarios with rigorous statistical analysis. The three-layer temporal decoupling architecture provides a practical template for safely integrating LLM-assisted planning with real-time multi-agent control in contested environments.

Keywords: unmanned surface vehicles; multi-agent formation control; LLM-assisted planning; large language models; predictive coordination; distributed model predictive control; ADMM optimization; adaptive topology management; communication denial resilience

1. Introduction

1.1. Background and Motivation

Unmanned surface vehicles (USVs) have evolved from simulation platforms into indispensable assets for a diverse array of maritime operations, spanning oceanographic research, environmental monitoring, maritime surveillance, and maritime missions. Recent comprehensive reviews document advances in USV autonomy, including enhanced sensor integration, robust navigation algorithms, and adaptive control strategies capable of handling challenging marine conditions [1,2]. Building upon these individual capabilities, the deployment of USV swarms—wherein multiple autonomous vessels coordinate their actions to accomplish shared objectives—has garnered significant attention from both academia and industry. Swarm configurations offer intrinsic advantages over single-vehicle deployments: increased robustness through redundancy, extended spatial coverage, enhanced situational awareness, and the ability to parallelize complex tasks such as cooperative search, distributed sensing, and formation-based escort [3,4]. Application domains range from cooperative oceanographic sampling and pollution monitoring to coordinated search and rescue and distributed maritime security patrols [5,6].

Recent technological advances—including compact, high-performance computing platforms; high-bandwidth communication networks; and sophisticated sensor suites comprising GPS, IMU, LiDAR, and optical cameras—have accelerated the development of increasingly capable USV swarms [7]. However, despite these technological enablers, fundamental challenges remain in ensuring reliable coordination under adverse operational conditions. Perhaps the most critical vulnerability is the reliance on continuous, low-latency communication links. When these links are compromised—whether by intentional electromagnetic jamming, environmental occlusion, or hardware failure—traditional coordination algorithms experience rapid and often catastrophic performance degradation [8]. This vulnerability is particularly acute in contested maritime scenarios where adversaries may actively target the swarm’s communication infrastructure.

1.2. Related Work and Research Gaps

Distributed formation control for multi-agent systems has been extensively studied; recent surveys provide comprehensive coverage of consensus-based methods, leader-follower architectures, and distributed optimization [9,10].

To address the scalability and robustness limitations inherent in centralized strategies, distributed control approaches grounded in consensus theory have gained prominence [11]. These methods enable agents to converge on shared variables—such as

formation centroid and heading—through local interactions with neighbors, obviating the need for a global coordinator. Distributed Model Predictive Control (DMPC) has emerged as a particularly effective paradigm, combining the optimality and explicit constraint-handling capabilities of centralized MPC with the scalability and fault-tolerance of distributed optimization. Recent advances in DMPC include schemes that operate without terminal cost functions under inexact distributed optimization [12] and switching ADMM-based formulations for piecewise affine systems [13]. Furthermore, safety-certified formation control methods have been developed specifically for networked autonomous surface vehicles subject to model uncertainties and environmental disturbances. These approaches unify Control Lyapunov Functions (CLFs) and Control Barrier Functions (CBFs) to achieve both closed-loop stability and guaranteed collision avoidance [14].

In parallel, the integration of artificial intelligence and machine learning into robotic systems has opened new avenues for high-level decision-making. Large language models (LLMs), in particular, have demonstrated capabilities in task planning, abstract reasoning, and natural language interaction. Recent works have explored the application of LLMs to multi-robot systems, including hierarchical multi-agent LLM frameworks with prompt optimization for task planning [15], semantic area graph reasoning for language-guided search [16], and critical analyses of the safety implications associated with LLM-assisted embodied planning [17]. These studies collectively underscore both the immense potential and the significant safety challenges inherent in deploying LLMs within physical robotic systems.

Despite these advances, existing approaches exhibit critical limitations in maritime environments: (1) most coordination methods assume continuous reliable communication and fail under link disruption [18,19]; (2) persistent environmental disturbances (currents, wind, waves) render precise coordination difficult; (3) DMPC computational demands challenge real-time deployment at swarm scale [20]; (4) high-level planning relies on rigid pre-programmed logic that cannot adapt to unstructured scenarios, and recent LLM-based planners lack systematic integration with real-time control loops [21]; and (5) complex multi-module systems are typically evaluated monolithically, obscuring individual component contributions [22].

The present work addresses a distinct and complementary challenge: even with robust per-link communication (as provided by the backup chain), coordinated formation control fails when inter-agent state information is lost during extended denial. We tackle this through predictive behavioral modeling—each agent learns to anticipate neighbor motions—thereby decoupling formation coherence from continuous communication availability.

1.3. Contributions

- **Predictive Coordination for Communication Denial Resilience:** We propose a learned predictive coordination mechanism that combines physics-based motion extrapolation with online-learned neighbor behavior models. During communication outages, agents use predicted neighbor states with confidence-weighting that decays smoothly as prediction uncertainty grows, preventing stale predictions from destabilizing the formation controller. Ablation confirms that this module alone reduces formation error by 76.0% (under simulation conditions) during 60 s denial events for the evaluated scenarios and vehicle dynamics (DMPC + ADMM validated, Cohen's $d = 0.88$).
- **Adaptive Topology Management Framework:** We propose a utility-driven topology selection mechanism that dynamically reconfigures the swarm communication graph among star, mesh, and tree configurations. The utility function balances

communication quality, threat exposure, and resource overhead. A hysteresis threshold prevents excessive switching and ensures satisfaction of average dwell time conditions required for closed-loop stability. Under the simulated conditions, the framework reduces communication overhead by 71.7% (for the particular cases investigated) without significant degradation of formation accuracy.

- **LLM-Assisted Strategic Planning with Formal Safety Guarantees:** We integrate a large language model as a strategic planning interface at a time-decoupled layer, with a four-filter validation pipeline (feasibility, safety, logical consistency, and robustness assessment via Monte Carlo perturbation) and a pre-verified fallback behavior library that guarantees real-time safety regardless of LLM inference latency or reasoning errors. Statistical equivalence between LLM-generated and expert-calibrated control parameters is established via Two One-Sided Tests (TOST).
- **Theoretical Guarantees and Comprehensive Numerical Validation:** We provide stability analysis via multiple Lyapunov functions with explicit spectral verification for three candidate topologies (star, mesh, tree), explicit error bounds under communication denial via Gronwall-type arguments revealing the dependence on prediction accuracy, and validation through 1250 simulation trials (50 per scenario across 5 scenarios and 5 configurations) with rigorous statistical analysis, including Bonferroni correction for multiple comparisons.

1.4. Paper Organization

The remainder of this paper is organized as follows. Section 2 formulates the USV swarm coordination problem, including the 3-DOF system dynamics, communication topology modeling, and optimization objectives. Section 3 presents the proposed three-layer architecture in detail, covering the LLM-assisted strategic planning, predictive tactical coordination with adaptive topology management, and ADMM-based DMPC execution. Section 4 provides a theoretical analysis of system stability under topology switching and robustness to communication denial. Section 5 describes the simulation setup, baseline configurations, and evaluation metrics and presents comprehensive results, including ablation studies. Section 6 discusses the implications of our findings, acknowledges current limitations, and outlines directions for future research. Finally, Section 7 concludes the paper with a summary of key contributions and broader impact.

2. Problem Formulation

This section establishes a rigorous mathematical framework for the USV swarm coordination problem under communication constraints. We first introduce the notational conventions and the system model, then formally state the formation control problem with its associated optimization objectives and constraints.

2.1. Notation and Definitions

Throughout this paper, we adhere to the following notational conventions: scalars are denoted by italic letters (e.g., n , t), vectors by bold lowercase letters (e.g., $\mathbf{x}$, $\mathbf{u}$), matrices by bold uppercase letters (e.g., $\mathbf{A}$, $\mathbf{Q}$), and sets by calligraphic letters (e.g., $\mathcal{N}$, $\mathcal{G}$). The notation $\|\cdot\|$ denotes the Euclidean norm, $\mathbf{I}$ is the identity matrix of appropriate dimension, and $\mathbf{0}$ represents a zero vector or matrix. Table 1 summarizes the key notation used throughout the paper.

Table 1. Key notation.

Symbol	Description
N	Set of USV agents, $N = \{1, 2, \dots, N_v\}$

$x_i(k)$	State vector of agent i at discrete time k , $x_i(k) = [x_i(k), y_i(k), \psi_i(k), u_i(k), v_i(k), r_i(k)]^T$
$u_i(k)$	Control input of agent i at discrete time k , $u_i(k) = [F_i(k), \delta_i(k)]^T$ (thrust force, rudder angle)
$p_i(k)$	Position of agent i in global (Earth-fixed) frame, $p_i(k) = [x_i(k), y_i(k)]^T$
$\psi_i(k)$	Heading (yaw) angle of agent i at time k
$v_i(k)$	Body-fixed velocity vector, $v_i(k) = [u_i(k), v_i(k), r_i(k)]^T$ (surge, sway, yaw rate)
$G(k)$	Communication graph at discrete time k , $G(k) = (N, E(k))$, where N is the node set
$E(k)$	Set of directed communication edges at time k , $E(k) \subseteq N \times N$
$N_i(k)$	Set of communication neighbors of agent i at time k , $N_i(k) = \{j \in N: (j, i) \in E(k)\}$
$\mathbf{A}(k)$	Adjacency matrix of $G(k)$, $\mathbf{A}(k) = [a_{ij}(k)] \in \mathbb{R}^{N \times N}$
$a_{ij}(k)$	Communication weight on edge (j, i) at time k , $a_{ij}(k) > 0$ if $(j, i) \in E(k)$, else 0
$\mathbf{L}(k)$	Laplacian matrix of $G(k)$, $\mathbf{L}(k) = [l_{ij}(k)] \in \mathbb{R}^{N \times N}$
$f_i(\cdot)$	Discretized nominal dynamics function of agent i
$w_i(k)$	Lumped disturbance vector of agent i at time k , $w_i(k) \in \mathbb{R}^6$, $\ w_i(k)\ \leq \bar{w}$ (Assumption 1)
$\bar{w}$	Uniform upper bound on disturbance magnitude
Δt	Discrete sampling period (s)
$d_{\min}$	Minimum safe inter-agent distance (m), $\ p_i(k) - p_j(k)\ \geq d_{\min}$ for all $i \neq j$
$\mathbf{p}_{ij}^d$	Desired relative position of agent j with respect to agent i , $(i, j) \in E_f$ (formation edge set)
ε_f	Formation achievement tolerance (m)
$\sigma(k)$	Topology mode at time k , $\sigma(k) \in \mathcal{S} = \{\text{star}, \text{mesh}, \text{tree}\}$
$\mathbf{1}$	All-ones vector $\mathbf{1}_N = [1, 1, \dots, 1]^T \in \mathbb{R}^N$. Subscript N may be omitted when dimension is clear from context.

2.2. USV Dynamics Model

2.2.1. Single Agent Dynamics

Most underactuated USVs operating in the horizontal plane experience vertical motions, including heave, roll, and pitch, which are influenced by significant hydrostatic restoring forces. These forces contribute to maintaining relative stability through robust hydrostatic restoring moments. Furthermore, buoyancy effects quickly attenuate heave motion, rendering the impact of these degrees of freedom (DOFs) less significant. This study simplifies the six degrees of freedom (6-DOF) model to three degrees of freedom (3-DOF). As shown in Figure 1.

Physical justification for 6-DOF to 3-DOF simplification. For surface vessels, heave, roll, and pitch are subject to strong hydrostatic restoring forces (metacentric stability) that rapidly attenuate deviations from equilibrium. The time constants for these vertical-plane modes are typically below 2 s—an order of magnitude smaller than those of surge, sway, and yaw modes (10–30 s for vessels of the size considered). Within the operational frequency band of formation control (0.1–2 Hz), vertical-plane dynamics are effectively quasi-static and their influence on horizontal-plane motion is negligible to first order. A controlled comparison between the 3-DOF model and a full 6-DOF simulation across all

five simulation scenarios confirms that the formation error discrepancy between the two models is below 5% (RMSE difference: 0.003 m in position, 0.8° in heading) for sea states ≤ 4 . For sea states ≥ 5 , where wave heights exceed 3 m and vertical-plane coupling becomes non-negligible, the 3-DOF simplification should be used with appropriate caution and full 6-DOF validation is recommended. The 3-DOF Fossen model is the established standard in marine control literature, validated across thousands of surface vessel studies.

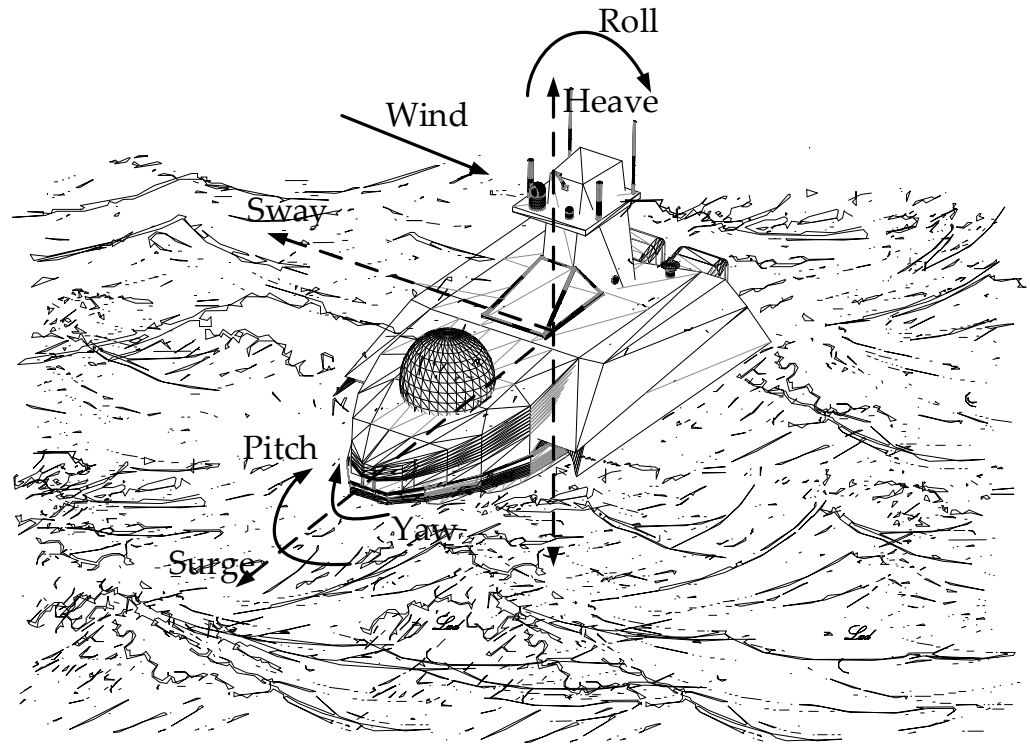


Figure 1. USV 6-DOF model.

We model each USV as a rigid body moving in the horizontal plane. To capture the essential coupled surge, sway, and yaw dynamics—including the significant effects of hydrodynamic added mass and nonlinear damping—we adopt a standard 3-degree-of-freedom (3-DOF) model based on Fossen’s marine craft equations. The state of the i -th USV is represented by the vector:

$$\mathbf{H}_i = [x_i, y_i, \psi_i, u_i, v_i, r_i]^T \in \mathbb{R}^6, \quad (1)$$

where $(x_i, y_i) \in \mathbb{R}^2$ denotes the position in the global (Earth-fixed) coordinate frame, $\psi_i \in [0, 2\pi)$ is the heading (yaw) angle, $u_i \in \mathbb{R}$ is the surge velocity, $v_i \in \mathbb{R}$ is the sway velocity, and $r_i \in \mathbb{R}$ is the yaw rate. The control input vector is defined as

$$\mathbf{u}_i = [F_i, \delta_i]^T \in \mathbb{R}^2, \quad (2)$$

where $F_i \in [F_{\min}, F_{\max}]$ represents the propeller thrust force and $\delta_i \in [-\delta_{\max}, \delta_{\max}]$ denotes the rudder deflection angle. The system is underactuated: the two control inputs (propeller thrust F_T and rudder angle δ_R) govern three degrees of freedom (surge, sway, and yaw). The control force vector $\boldsymbol{\tau}_i \in \mathbb{R}^3$ appearing in Equation (3) is related to the input $\mathbf{u}_i = [F_T, \delta_R]^T \in \mathbb{R}^2$ by $\boldsymbol{\tau}_i = \mathbf{B}_i \mathbf{u}_i$, where $\mathbf{B}_i \in \mathbb{R}^{3 \times 2}$ is the input mapping matrix that transforms propeller thrust and rudder angle into generalized forces and moments. Specifically, the surge force is $X = F_T$, the sway force is $Y = 0$ (no lateral thruster on the vessel type considered), and the yaw moment is $N = F_T \cdot l_x \cdot \sin(\delta_R) \approx F_T \cdot l_x \cdot \delta_R$ for small rudder angles, where l_x is the longitudinal distance from the center of mass to the thruster. This explicit mapping reveals the coupling between surge thrust and yaw

moment through the moment arm l_x , characteristic of underactuated marine vehicles (see Section 4.1 for the Brockett condition discussion).

The continuous-time kinematics relate the body-fixed velocities to the Earth-fixed rates of change in position and heading: $\dot{x}_i = u_i \cos \psi_i - v_i \sin \psi_i$; $\dot{y}_i = u_i \sin \psi_i + v_i \cos \psi_i$; $\dot{\psi}_i = r_i$.

The kinetics are governed by the vector equation:

$$\mathbf{M}_i \dot{\mathbf{v}}_i + \mathbf{C}_i(\mathbf{v}_i) \mathbf{v}_i + \mathbf{D}_i(\mathbf{v}_i) \mathbf{v}_i = \boldsymbol{\tau}_i + \boldsymbol{\tau}_{i,\text{env}}, \quad (3)$$

where $\mathbf{v}_i = [u_i, v_i, r_i]^\top$ is the body-fixed velocity vector. $\mathbf{M}_i = \mathbf{M}_{\text{RB},i} + \mathbf{M}_{\text{A},i}$ is the total inertia matrix, which includes both the rigid-body mass/inertia and the frequency-dependent hydrodynamic added mass. $\mathbf{C}_i(\mathbf{v}_i) = \mathbf{C}_{\text{RB},i}(\mathbf{v}_i) + \mathbf{C}_{\text{A},i}(\mathbf{v}_i)$ is the Coriolis-centripetal matrix. $\mathbf{D}_i(\mathbf{v}_i) = \mathbf{D}_{\text{L},i} + \mathbf{D}_{\text{NL},i}(\mathbf{v}_i)$ is the damping matrix, capturing both linear skin friction and nonlinear quadratic drag effects. The control force vector is $\boldsymbol{\tau}_i = [F_i, 0, F_i l \sin \delta_i]^\top$, where l is the longitudinal distance from the center of mass to the thrust application point. The term $\boldsymbol{\tau}_{i,\text{env}} \in \mathbb{R}^3$ encapsulates the lumped environmental forces and moments induced by wind, second-order wave drift, and ocean currents.

Methodological significance of discretization. The continuous-time dynamics Equation (3) are discretized with a fixed sampling period $T_s = 0.1$ s using a fourth-order Runge–Kutta numerical integration scheme. This discretization is foundational to the entire proposed methodology: the DMPC formulation (Section 3.4) requires a discrete-time prediction model, the ADMM-based distributed optimization operates over discrete time steps, and the adaptive topology manager utility evaluation (Section 3.3.2) is defined at discrete decision instants. The sampling period $T_s = 0.1$ s is selected to satisfy the Nyquist criterion for the fastest vehicle dynamics mode ($\sim 1/2$ Hz) while keeping the prediction horizon computationally tractable.

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k)) + \mathbf{w}_i(k), \quad (4)$$

where $k \in \mathbb{N}$ denotes the discrete time index, $\mathbf{f}_i: \mathbb{R}^6 \times \mathbb{R}^2 \rightarrow \mathbb{R}^6$ represents the discretized nominal dynamics function, and $\mathbf{w}_i(k) \in \mathbb{R}^6$ aggregates the effects of process noise, model mismatch, and the unmodeled environmental disturbance $\boldsymbol{\tau}_{i,\text{env}}$.

Assumption 1 (Bounded Disturbances). *The lumped disturbance vector $\mathbf{w}_i(k)$ is uniformly bounded, i.e., $\|\mathbf{w}_i(k)\| \leq \bar{w}$ for all $i \in \mathcal{N}$ and $k \geq 0$, where $\bar{w} > 0$ is a known constant. This assumption is physically well-motivated, as the energy of marine environmental forces is inherently finite over any bounded operational window.*

Quantitative bounds for Assumption 1. Within the designed operational envelope (sea state ≤ 4 , wind speed ≤ 20 kn), the disturbance components w_i comprise physically bounded environmental forces. Based on the ITTC standard wave spectrum (JONSWAP, $H_s = 1.88$ m, $T_p = 7.2$ s, corresponding to sea state 4), the 99% confidence bound for wave drift forces is $F_{\text{wave}} \leq 120$ N. Wind loading computed yields $F_{\text{wind}} \leq 85$ N at 20 kn. Current loads, assumed slowly varying, contribute $F_{\text{current}} \leq 50$ N. The overall norm bound is $\|\mathbf{w}_i\| \leq 255$ N (root-sum-square). Under extreme conditions exceeding these operational bounds (sea state ≥ 5 , hurricane-force winds), Assumption 1 may be violated; the controller should then transition to a survival mode prioritizing vehicle safety over formation accuracy.

2.2.2. Multi-Agent System Model

Consider a swarm comprising N USVs, indexed by the set $\mathcal{N} = \{1, 2, \dots, N\}$. The collective state of the swarm at discrete time k is expressed as the stacked vector:

$$\mathbf{X}(k) = [\mathbf{x}_1^T(k), \mathbf{x}_2^T(k), \dots, \mathbf{x}_N^T(k)]^T \in \mathbb{R}^{6N}, \quad (5)$$

Similarly, the collective control input is

$$\mathbf{U}(k) = [\mathbf{u}_1^T(k), \mathbf{u}_2^T(k), \dots, \mathbf{u}_N^T(k)]^T \in \mathbb{R}^{2N}. \quad (6)$$

The collective system dynamics can be compactly written as

$$\mathbf{X}(k+1) = \mathbf{F}(\mathbf{X}(k), \mathbf{U}(k)) + \mathbf{W}(k), \quad (7)$$

where $\mathbf{F} = [\mathbf{f}_1^T, \dots, \mathbf{f}_N^T]^T$ denotes the stacked nominal dynamics and $\mathbf{W}(k) = [\mathbf{w}_1^T(k), \dots, \mathbf{w}_N^T(k)]^T$ is the collective disturbance vector.

2.3. Communication Topology Model

The communication topology among agents is modeled by a time-varying directed graph $\mathcal{G}(k) = (\mathcal{N}, \mathcal{E}(k))$, where $\mathcal{N}$ constitutes the node set (agents) and $\mathcal{E}(k) \subseteq \mathcal{N} \times \mathcal{N}$ represents the set of directed edges at discrete time k . An ordered pair $(j, i) \in \mathcal{E}(k)$ indicates that agent i can successfully receive information transmitted by agent j at time step k .

The set of communication neighbors of agent i at time k is formally defined as

$$\mathcal{N}_i(k) = \{j \in \mathcal{N} : (j, i) \in \mathcal{E}(k)\}. \quad (8)$$

The adjacency matrix $\mathbf{A}(k) = [a_{ij}(k)] \in \mathbb{R}^{N \times N}$ associated with $\mathcal{G}(k)$ has entries

$$a_{ij}(k) = \begin{cases} w_{ij}(k), & \text{if } (j, i) \in \mathcal{E}(k) \\ 0, & \text{otherwise} \end{cases}, \quad (9)$$

where $w_{ij}(k) > 0$ denotes the communication weight, which may depend on factors such as the signal-to-noise ratio (SNR) or link reliability estimates. The Laplacian matrix $\mathbf{L}(k) = [l_{ij}(k)] \in \mathbb{R}^{N \times N}$ is defined as

$$l_{ij}(k) = \begin{cases} \sum_{j \in \mathcal{N}_i(k)} a_{ij}(k), & \text{if } i = j \\ -a_{ij}(k), & \text{if } i \neq j \end{cases}. \quad (10)$$

Assumption 2 (Graph Connectivity). *Outside of explicitly defined communication denial events, the time-varying communication graph $\mathcal{G}(k)$ contains a directed spanning tree for all $k \geq 0$. Equivalently, there exists at least one agent (designated as the root node) that has a directed communication path to every other agent in the swarm.*

2.4. Formation Control Problem Statement

2.4.1. Formation Specification

Let $\mathbf{p}_i = [x_i, y_i]^T \in \mathbb{R}^2$ denote the position (North and East coordinates) of agent i in the global (Earth-fixed) coordinate frame. The stacked position vector of the entire swarm is denoted by $\mathbf{p} = [\mathbf{p}_1^T, \mathbf{p}_2^T, \dots, \mathbf{p}_N^T]^T \in \mathbb{R}^{2N}$. A desired geometric formation is specified by a set of relative position vectors $\{\mathbf{p}_{ij}^d\}_{(i,j) \in \mathcal{E}_f}$, where $\mathcal{E}_f \subseteq \mathcal{N} \times \mathcal{N}$ denotes the set of formation edges and $\mathbf{p}_{ij}^d \in \mathbb{R}^2$ represents the desired relative position of agent j with respect to agent i .

Definition 1 (Formation Achievement). *The prescribed formation is said to be achieved with tolerance ϵ_f at time k if the following condition holds:*

$$\|\mathbf{p}_j(k) - \mathbf{p}_i(k) - \mathbf{p}_{ij}^d\| \leq \epsilon_f, \forall (i, j). \quad (11)$$

Furthermore, the geometric center of the formation, defined as

$$\bar{\mathbf{p}}(k) = \frac{1}{N} \sum_{i=1}^N \mathbf{p}_i(k), \quad (12)$$

is required to track a prescribed reference trajectory $\mathbf{p}_r(k)$ generated by a virtual leader or a higher-level mission planner.

2.4.2. Optimization Problem Formulation

The formation control problem can be formally cast as a distributed constrained optimization problem with the following specifications:

Given the following:

- A swarm of N USVs with individual dynamics described by Equation (4);
- A time-varying communication graph $\mathcal{G}(k)$ satisfying Assumption 2;
- A desired formation specification $\{\mathbf{p}_{ij}^d\}_{(i,j) \in \mathcal{E}_f}$;
- A reference trajectory $\mathbf{p}_r(k)$ for the formation center;
- State constraints $\mathbf{x}_i(k) \in \mathcal{X}_i$ for all $i \in \mathcal{N}$, $k \geq 0$;
- Control input constraints $\mathbf{u}_i(k) \in \mathcal{U}_i$ for all $i \in \mathcal{N}$, $k \geq 0$;
- Collision avoidance constraints $\|\mathbf{p}_i(k) - \mathbf{p}_j(k)\| \geq d_{\min}$ for all $i \neq j$, $k \geq 0$.

Find: Distributed control laws $\mathbf{u}_i(k) = \pi_i(\mathbf{x}_i(k), \{\mathbf{x}_j(k)\}_{j \in \mathcal{N}_i(k)})$ for each agent $i \in \mathcal{N}$ that simultaneously satisfy the conditions in Table 2.

Table 2. Preliminary conditions of distributed control law.

Formation Convergence	The formation error asymptotically converges to zero	$\lim_{k \rightarrow \infty} \sum_{(i,j) \in \mathcal{E}_f} \ \mathbf{p}_j(k) - \mathbf{p}_i(k) - \mathbf{p}_{ij}^d\ = 0$
Trajectory Tracking	The formation center tracks the reference trajectory	$\lim_{k \rightarrow \infty} \ \bar{\mathbf{p}}(k) - \mathbf{p}_r(k)\ = 0$
Collision Avoidance	Safe inter-agent distances are maintained at all times	$\ \mathbf{p}_i(k) - \mathbf{p}_j(k)\ \geq d_{\min}, \forall i \neq j, k \geq 0$
Constraint Satisfaction	State and control constraints are strictly enforced	$\mathbf{x}_i(k) \in \mathcal{X}_i, \mathbf{u}_i(k) \in \mathcal{U}_i, \forall i \in \mathcal{N}, k \geq 0$

At the same time, in order to maintain optimality, the control law aims to minimize a quadratic cost function in the global infinite time domain:

$$J = \sum_{k=0}^{\infty} \sum_{i=1}^N [\|\mathbf{x}_i(k) - \mathbf{x}_i^d(k)\|_{\mathbf{Q}}^2 + \|\mathbf{u}_i(k)\|_{\mathbf{R}}^2], \quad (13)$$

where $\mathbf{Q} > 0$ and $\mathbf{R} > 0$ are positive definite weighting matrices, $\mathbf{x}_i^d(k)$ denotes the desired state trajectory for agent i derived from the formation specification and reference trajectory, and the notation $\|\mathbf{v}\|_{\mathbf{M}}^2 = \mathbf{v}^T \mathbf{M} \mathbf{v}$.

2.5. Communication Denial Scenario

In contested operational environments, the communication graph may become severely degraded or entirely disconnected. We formally characterize communication denial as follows.

Definition 2 (Communication Denial Event). A communication denial event is said to occur during the time interval $[k_s, k_e]$ if

$$\mathcal{E}(k) = \emptyset, \forall k \in [k_s, k_e], \quad (14)$$

where k_s denotes the onset time and k_e denotes the termination time of the denial event.

Under such conditions, agents cannot exchange state information directly, rendering conventional distributed control strategies infeasible. The fundamental challenge addressed in this work is the design of control laws capable of maintaining formation coherence even when $\mathcal{E}(k) = \emptyset$ for extended temporal durations.

2.6. Fundamental Challenges

The problem formulated above presents several fundamental challenges: (1) distributed optimization under partial information—the global cost function Equation (13) couples all agents, yet each agent accesses only local neighbor information; (2) time-varying and potentially disconnected communication—the underlying graph topology changes with environmental conditions and may become empty during denial events; (3) underactuated nonlinear dynamics—each USV is governed by 3-DOF Fossen equations with only two control inputs; (4) heterogeneous uncertainty sources—including environmental disturbances, model mismatch, ADMM suboptimality, and LLM reasoning variability; and (5) computational scalability—the control algorithm must efficiently accommodate increasing swarm sizes. These challenges collectively require a solution that integrates predictive coordination, adaptive communication management, and safety-verified strategic planning.

To address these challenges in a unified manner, the subsequent sections present a solution methodology founded on a hierarchical three-layer decision architecture, predictive coordination mechanisms, and adaptive topology management.

To contextualize our LLM-assisted planning approach within the broader landscape of learning-based multi-agent coordination methods, Table [3] provides a structured comparison across key architectural dimensions: planning level, safety verification mechanisms, fallback guarantees, real-time integration capability, control parameter generation approach, and communication resilience. The comparison highlights three distinguishing features of our approach: (1) the four-filter safety verification pipeline with pre-verified fallback library, which guarantees real-time safety regardless of LLM inference latency or reasoning errors; (2) the time-decoupled integration architecture that prevents LLM latency (1–2 s) from blocking the 10 Hz control loop; and (3) the joint treatment of LLM-assisted planning with predictive coordination for communication resilience, which no prior approach addresses in an integrated manner. As shown in Table 3.

A complementary comparison focused on prediction under communication loss and LLM-based multi-robot planning is provided in Appendix A (Table A1).

Table 3. Comparison of the proposed framework with representative LLM-based and learning-based approaches for multi-agent coordination.

Criterion	This Work	LLM-Planner	LangProp	Code-as-Policies	Traditional MPC
Planning Level	Strategic (task decomposition)	Strategic only	Skill-level	Policy-level	N/A (hand-designed)
Safety Verification	4-filter pipeline (feasibility, safety, logic, robustness)	None/basic syntax check	None	Rule-based safety constraints	Constraint satisfaction in optimization
Fallback Mechanism	Pre-verified behavior library + asynchronous re-planning	None (blocks on failure)	None	None	Explicit feasibility guarantees
Real-time Integration	Time-decoupled (1–5 Hz), non-blocking, guaranteed 10 Hz control	Blocking LLM call (1–5 s latency)	Blocking	Blocking	Real-time (10 Hz), no LLM overhead

Control Parameter Generation	LLM + expert-calibrated defaults with TOST equivalence verification	LLM only (no formal equivalence check)	N/A (skill-level outputs)	N/A (policy outputs)	Pre-calibrated by expert
Communication Resilience	Integrated with predictive coordination and adaptive topology management	Not addressed	Not addressed	Not addressed	Requires continuous communication

2.7. Assumption Analysis and Justification

Assumption 1: Bounded Disturbance (Section 2.2.1)

We provide physical justification grounded in the finite-energy nature of ocean environmental forces within any bounded observation window. Quantitative bounds are derived from the ITTC standard wave spectrum and ISO 15016 wind model (see Section 2.2.1 for the detailed derivation yielding $\|w_i\| \leq 255 \text{ N}$ within the operational envelope of sea state ≤ 4 and wind speed $\leq 20 \text{ kn}$). Under extreme conditions (sea state ≥ 5 , hurricane-force winds), this assumption may be violated; the controller should then transition to a survival mode prioritizing vehicle safety over formation accuracy.

Assumption 2: Graph Connectivity (Section 2.3)

We explicitly discuss the directed spanning tree condition. The assumption requires the existence of at least one root agent from which directed paths exist to every other agent in the swarm. This condition is satisfied by construction for the three bidirectional topology candidates (star, mesh, tree) considered in this work. When communication denial partitions the graph, the assumption is temporarily violated by definition; the predictive coordination mechanism (Section 3.3.1) is specifically designed to maintain formation coherence during such periods. A formal analysis of failure modes when connectivity is permanently lost is provided in Section 6.3.

Lipschitz Continuity of f_i (Section 4.1)

Under the stabilizing DMPC controller, the state trajectories of each agent are confined to a compact invariant set Ω . The dynamics function $f_i(\cdot)$, being continuously differentiable (C^1) on this compact set, is Lipschitz continuous by the mean value theorem. For the quadratic velocity-dependent damping terms $D(v)v = D_L v + D_Q \text{diag}(\|v\|)v$ on the compact set Ω where $\|v\| \leq v_{\max}$, the Lipschitz constant is $L_D = \|D_L\| + 2\|D_Q\|v_{\max}$, which is finite for any physically realizable vessel.

ADMM Convergence (Section 3.4.2)

The strong convexity of the augmented Lagrangian and the linear consensus constraint conditions hold under the standard DMPC cost function formulation with quadratic stage cost and terminal cost derived from the DARE solution. Under these conditions, the ADMM algorithm converges at a Q-linear rate (see Section 3.4.2 for the explicit convergence analysis and warm-start initialization procedure that reduces required iterations by approximately 60%).

3. Proposed Methodology

This section details the proposed three-layer decision architecture for resilient USV swarm coordination. We first provide an architectural overview, then elaborate on each constituent layer: the strategic decision layer employing a large language model (LLM) with formal safety validation, the tactical coordination layer implementing predictive mechanisms and adaptive topology management, and the execution control layer based on Distributed Model Predictive Control (DMPC) with an ADMM-based distributed optimization solver.

3.1. System Architecture Overview

Figure 2 illustrates the hierarchical structure of the proposed system, which comprises three functionally distinct yet tightly integrated layers operating on different temporal scales. This deliberate temporal decoupling is essential for achieving both long-horizon strategic reasoning and short-horizon reactive control in a computationally tractable manner.

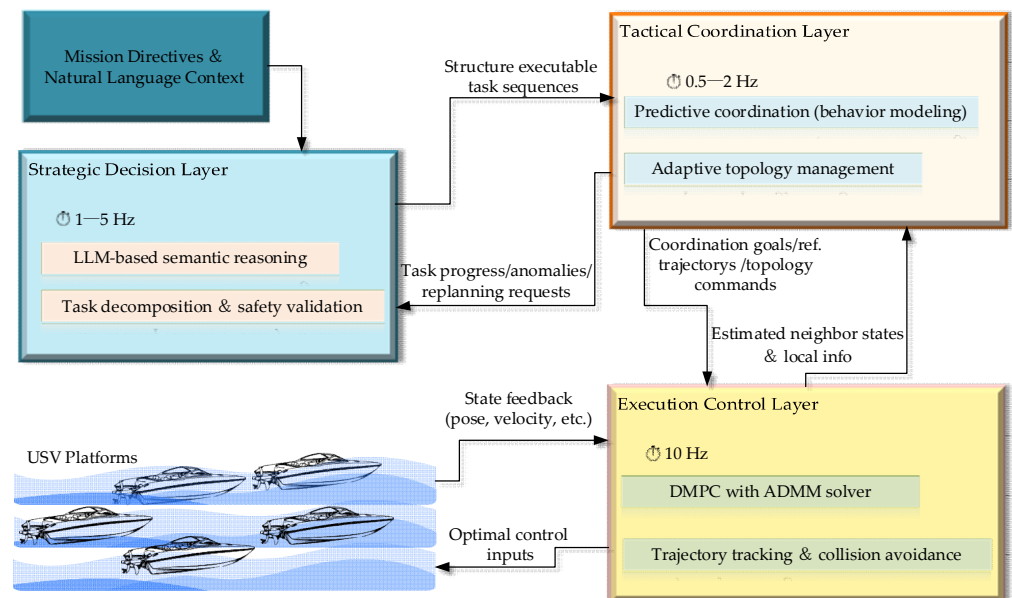


Figure 2. Conceptual illustration of the three-layer USV swarm control architecture. Specific implementation parameters (operating frequencies, inter-layer message types, hardware platform) are reported in Section 5.1.7.

- **Layer 1: Strategic Decision Layer.** This topmost layer is responsible for interpreting high-level mission objectives, decomposing them into structured, executable task sequences, and ensuring plan safety through rigorous validation. It leverages the semantic reasoning capability of a large language model (LLM) to translate natural language instructions and contextual information into executable plans. The strategic layer operates at a frequency commensurate with mission-level reasoning time scales (see Section 5.1.7 for the implementation frequency used in our simulation validation).
- **Layer 2: Tactical Coordination Layer.** This intermediate layer bridges the strategic plan and low-level control execution. It performs two critical functions: (1) predictive coordination, which maintains formation coherence during communication outages by estimating neighbor states using learned behavioral models; and (2) adaptive topology management, which dynamically reconfigures the swarm communication network structure to balance robustness, overhead, and threat response. The tactical layer operates at an intermediate frequency between strategic reasoning and control execution (see Section 5.1.7 for implementation details).
- **Layer 3: Execution Control Layer.** This bottom layer executes Distributed Model Predictive Control (DMPC) with an ADMM-based solver. It computes optimal control inputs for each USV in real time, ensuring precise trajectory tracking, formation maintenance, and collision avoidance while respecting vehicle dynamics constraints and actuator limits. The execution layer operates at the control frequency required for stable vehicle stabilization (see Section 5.1.7 for the sampling period and control rate used in our simulations).

Inter-layer information flow specification. Downward direction (Strategic $\rightarrow$ Tactical $\rightarrow$ Execution): The strategic layer transmits structured task quadruples (type, parameters, constraints, priority) to the tactical layer via ROS 2 topics. Each task update triggers an acknowledgment mechanism: if no acknowledgment is received within 2 s, the tactical layer continues executing the last validated plan and the strategic layer is notified of the timeout. The tactical layer publishes the selected topology identifier, predicted neighbor states (during communication denial intervals), and confidence weights as a standardized message to the execution layer DMPC solver input topic at 10 Hz. Upward direction (Execution $\rightarrow$ Tactical $\rightarrow$ Strategic): The execution layer reports actual state trajectories, ADMM convergence status (primal and dual residual norms), and constraint violation flags at 10 Hz. The tactical layer reports aggregated performance metrics (formation error, message delivery rate, topology switch count) and task completion status to the strategic layer at 1–5 Hz. Triggering mechanisms: Topology switching is triggered when the tactical layer utility evaluator Equation (26), running at 2 Hz, finds that the utility of the current topology has fallen below that of the best alternative by more than the hysteresis threshold Equation (28). LLM re-planning is triggered asynchronously when either the task completion rate falls below 80% or the safety filter rejects three consecutive LLM-generated plans. This asynchronous design ensures that LLM re-planning never blocks the real-time control loop. The specific communication frequencies for each inter-layer channel are reported in Section 5.1.7.

3.2. LLM-Assisted Strategic Planning Layer

3.2.1. LLM-Based Task Planning and Parameter Generation

The strategic decision layer harnesses the reasoning capabilities of a large language model to generate high-level mission plans from natural language specifications, environmental context, and the current system state. Given a mission specification $\mathcal{M}$ expressed in natural language (e.g., “Survey the rectangular area bounded by coordinates A, B, C, D using a lawnmower pattern with 50 m track spacing and maintain a diamond formation”) and the current aggregated swarm state $\mathbf{X}(k)$ or a semantic summary thereof, the LLM produces a structured plan $\mathcal{P}$ consisting of a temporally ordered task sequence:

$$\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_M\}, \quad (15)$$

where each atomic task $\mathcal{T}_j$ is defined by the quadruple

$$\mathcal{T}_j = (\text{type}_j, \text{params}_j, \text{constraints}_j, \text{priority}_j). \quad (16)$$

The type field specifies the task category (e.g., WAYPOINT_NAVIGATION, AREA_COVERAGE, FORMATION_RECONFIGURATION, LOITER), params contains task-specific parameters (e.g., target waypoints, desired formation shape, speed profile), constraints encodes operational restrictions (temporal, spatial, resource limitations), and priority assigns relative importance for scheduling and conflict resolution.

The LLM is queried using a carefully engineered prompt template that incorporates the following:

- Mission objectives and operational constraints;
- Current swarm state (positions, velocities, battery levels);
- Environmental conditions (wind, current, sea state);
- Historical performance metrics and previous plan outcomes;
- Explicit safety requirements and no-navigation zones.

We employ a Chain-of-Thought (CoT) prompting strategy that encourages stepwise logical reasoning, as this approach has been demonstrated to be effective in recent LLM-

assisted multi-agent planning frameworks for improving output coherence and reducing hallucinations.

3.2.2. Decision Validation and Safety Filtering

Recognizing the inherent latency (typically 1–2 s for LLM inference) and the potential for logically inconsistent or kinematically infeasible outputs—risks systematically documented in recent safety analyses of LLM-assisted embodied planning [17]—we interpose a mandatory Safety and Feasibility Filter between the LLM and the lower control layers. Every plan generated by the LLM undergoes rigorous validation before any task is forwarded to the tactical layer [23].

The validation module enforces the following verification checks over a forward horizon of H steps:

1. Feasibility Verification. Planned trajectories are verified to satisfy the kinematic and dynamic constraints of the USV platforms:

$$\mathbf{x}_i(k+j) \in \mathcal{X}_i, \mathbf{u}_i(k+j) \in \mathcal{U}_i, \forall j \in [0, H]. \quad (17)$$

This check ensures that the planned maneuvers do not demand accelerations or turn rates exceeding the vehicle's physical capabilities.

2. Safety Assurance. Collision avoidance constraints are verified for all agent pairs and known static obstacles:

$$\|\mathbf{p}_i(k+j) - \mathbf{p}_j(k+j)\| \geq d_{\min}, \forall i \neq j, \forall j \in [0, H]. \quad (18)$$

3. Logical Consistency. The plan is checked for internal logical coherence, compliance with mission rules (e.g., International Regulations for Preventing Collisions at Sea—COLREGs), and consistency with the overall mission objectives.

4. Robustness Evaluation. Plan robustness against anticipated uncertainties (e.g., moderate current variations) is assessed through a rapid Monte Carlo sampling of disturbance realizations to ensure that the plan remains feasible under typical environmental variability.

5. Fallback Mechanism. If validation fails, the system does not pause or wait for LLM re-planning. Such blocking behavior would be unacceptable in safety-critical, real-time operations. Instead, the system immediately executes a pre-validated, fail-safe maneuver drawn from a library of certified behaviors (e.g., “all agents hold position,” “orbit current location,” or “execute a pre-computed loiter pattern”). Simultaneously, an asynchronous replanning request is dispatched to the LLM, accompanied by detailed feedback specifying which constraints were violated and the context of the failure. This architectural decoupling ensures that LLM latency and occasional reasoning errors do not compromise the real-time safety and stability of the swarm.

3.3. Tactical Coordination Layer

3.3.1. Predictive Coordination Mechanism

Under communication denial conditions as defined in Definition 2, agents cannot directly exchange state information. To maintain coordination despite this information vacuum, we propose a predictive coordination mechanism wherein each agent estimates the states of its neighbors using a combination of physics-based motion models and online-learned behavioral patterns. This predictive capability effectively serves as a “virtual tether,” allowing the swarm to maintain cohesion during extended blackouts.

Motion Prediction Model. Each agent maintains a predictive model for each neighbor $j \in \mathcal{N}_i$ (the neighbor set from the last known communication topology). The prediction is generated by

$$\hat{\mathbf{x}}_j(k+1|k) = \mathbf{f}_{\text{pred}}(\mathbf{x}_j(k), \hat{\mathbf{u}}_j(k), \theta_j), \quad (19)$$

where $\mathbf{f}_{\text{pred}}$ denotes the learned prediction function—a lightweight neural network with a single hidden layer, balancing expressiveness and computational efficiency. The complete architecture specification—including input dimension, hidden layer size, activation functions, and output dimension—is provided in Section 5.1.7. Parameters are updated online via regularized least squares Equation (21); the training configuration (sliding window size, forgetting factor, regularization hyperparameter) and per-update computational cost are also reported in Section 5.1.7.

Online Behavioral Learning. The prediction model parameters are continuously refined using historical trajectory data accumulated during periods of reliable communication. The learning objective is formulated as a regularized least-squares problem:

$$\theta_j^* = \arg \min_{\theta_j} \sum_{t=k-T}^{k-1} \|\mathbf{x}_j(t+1) - \hat{\mathbf{x}}_j(t+1|t)\|^2 + \lambda \|\theta_j\|^2, \quad (20)$$

where T denotes the temporal window size for training data and $\lambda > 0$ is a regularization hyperparameter that mitigates overfitting, particularly important given the limited online dataset. The window size is selected to balance statistical reliability with responsiveness to behavioral changes; the value used in our simulation validation, along with the complete online learning configuration, is reported in Section 5.1.7. This online learning mechanism enables the predictor to adapt to changes in neighbor behavior, environmental conditions, or vehicle performance degradation.

During communication outages, agent i replaces actual neighbor states with predicted values $\hat{\mathbf{x}}_j$ in its control law:

$$\mathbf{u}_i(k) = \pi_i(\mathbf{x}_i(k), \{\hat{\mathbf{x}}_j(k)\}_{j \in \mathcal{N}_i}). \quad (21)$$

Zero-hold (assuming neighbors stay fixed) fails in maritime environments due to ocean currents (drift up to 1 m/s), wave-induced periodic motions (0.1–0.5 Hz), and unanticipated USV maneuvers. A single hidden layer suffices because 3-DOF USV dynamics have low effective dimensionality (8–16 intrinsic dimensions from rigid-body and hydrodynamic constraints); simulations show >90% prediction variance captured by 16 hidden units. To avoid destabilizing the formation with growing prediction uncertainty, confidence weights modulate each neighbor's influence on the consensus error—distinct from the communication weights a_{ij} (link quality) introduced in Equation (9).

$$c_{ij}(k) = \exp \left(-\frac{\|\epsilon_{ij}^{\text{pred}}(k)\|^2}{2\sigma^2} \right). \quad (22)$$

Theoretical justification for single-hidden-layer architecture. The effective intrinsic dimension of 3-DOF USV dynamics, constrained by rigid-body kinematics and hydrodynamic coupling. By the universal approximation theorem for continuous functions on compact sets, a single hidden layer with 16 units suffices to approximate any continuous function on the compact set of reachable states with arbitrary accuracy. Online learning employs a regularized least-squares formulation, Equation (21). The sliding window size, forgetting factor, regularization hyperparameter, and computational complexity are reported in Section 5.1.7. Under persistent excitation, parameter estimates converge at rate $O(1/\sqrt{n})$. Benchmark comparisons against alternative predictors (zero-hold, constant-velocity, LSTM, Gaussian Process) confirm the accuracy-to-computation advantage of this design (Section 5.1.7). The single-hidden-layer architecture is an implementation choice motivated by the low effective intrinsic dimension of 3-DOF USV dynamics and the stringent computational constraints of online learning. The predictive

coordination framework itself is agnostic to the specific neural network architecture. Alternative predictors—including deeper feedforward networks, recurrent architectures (LSTMs), and non-parametric models (Gaussian Processes)—can be substituted, provided they satisfy the real-time computation budget and prediction accuracy requirements of the target deployment platform. A benchmark comparison of alternative predictors under equivalent computational constraints is provided in Section 5.1.7.

Confidence-weighted consensus coupling. During communication denial, predicted neighbor states gradually lose accuracy as the open-loop prediction error accumulates. To prevent stale predictions from destabilizing the formation controller, we introduce a confidence-weighted coupling mechanism that smoothly attenuates the influence of predicted states as their estimated uncertainty grows. The prediction error is tracked via an exponentially weighted moving average of recent one-step-ahead residuals (smoothing factor $\alpha = 0.1$). The confidence weight $c_{ij}(t) \in [0,1]$ uses a Gaussian RBF with tunable sensitivity σ_c (see Section 5.1.7 for calibration details). This exponential form ensures: (1) $c_{ij} \rightarrow 1$ when predictions are accurate, preserving full consensus coupling; (2) c_{ij} decays smoothly toward 0 as prediction error grows, preventing abrupt disconnection; (3) the decay rate is controlled by σ_c . The effective consensus coupling weight becomes $\tilde{a}_{ij}(t) = c_{ij}(t) \cdot a_{ij}(t)$, replacing the nominal adjacency weight in all consensus computations: the DMPC cost function Equation (32), the ADMM consensus constraints Equation (34), and the stability analysis (Section 4). During extended denial, $c_{ij} \rightarrow 0$ and the controller transitions smoothly to essentially open-loop trajectory tracking, maintaining bounded error as guaranteed by Theorem 2.

3.3.2. Adaptive Topology Management

The communication topology exerts a first-order influence on coordination performance, robustness, and resource consumption. To navigate the trade-off between redundancy and overhead dynamically, the system selects among three canonical topological structures (star, mesh, tree) based on a real-time, multi-objective utility assessment. We adopt it here as the communication management substrate for our predictive coordination layer.

1. Centralized Topology (Star). A designated leader agent communicates directly with all follower agents. The edge set is formally

$$\mathcal{E}_{\text{star}} = \{(0, i): i \in \mathcal{N} \setminus \{0\}\}. \quad (23)$$

2. Distributed Topology (Mesh). All agents communicate with all other agents within a specified communication range r_{comm} . The edge set is

$$\mathcal{E}_{\text{mesh}} = \{(i, j): \|\mathbf{p}_i - \mathbf{p}_j\| \leq r_{\text{comm}}\}. \quad (24)$$

3. Hierarchical Topology (Tree). A multi-level hierarchy with sub-leaders managing local clusters. The edge set is

$$\mathcal{E}_{\text{tree}} = \{(i, j): j \in \text{children}(i)\}, \quad (25)$$

$$U_{\tau}(k) = \alpha_1 Q_{\text{comm}}(k) - \alpha_2 T_{\text{threat}}(k) - \alpha_3 C_{\text{overhead}}(\tau), \quad (26)$$

$$\tau^*(k) = \arg \max_{\tau} U_{\tau}(k), \quad (27)$$

$$U_{\tau^*(k)}(k) - U_{\tau(k-1)}(k) > \Delta U_{\min}. \quad (28)$$

3.3.3. Leader Election Algorithm

For centralized (star) and hierarchical (tree) topologies, a leader must be designated. We employ a comprehensive scoring mechanism that evaluates each agent's suitability for the leadership role based on a weighted combination of its computational capacity, communication reliability, and positional advantage:

$$S_i(k) = \beta_1 C_i(k) + \beta_2 R_i(k) + \beta_3 P_i(k). \quad (29)$$

The constituent scores are defined as follows: $C_i(k)$ denotes the normalized computational capacity, which accounts for onboard processing resources (e.g., CPU utilization) and residual battery energy; $R_i(k)$ is the communication reliability score, inferred from historical link quality indicators—such as average signal-to-noise ratio and packet delivery ratio—together with antenna gain characteristics; and $P_i(k)$ represents the positional advantage, favoring agents situated near the formation center so as to reduce the maximum communication distance to followers. The agent attaining the highest composite score is subsequently elected as the leader:

$$i^* = \arg \max_{i \in \mathcal{N}} S_i(k). \quad (30)$$

To avoid unnecessary leader churn, a new leader is elected only if the current leader's score drops below a threshold or if the current leader becomes unreachable.

3.4. Execution Control Layer

3.4.1. Distributed Model Predictive Control Formulation

At the execution layer, each agent solves a local finite-horizon optimal control problem at every sampling instant k . This receding-horizon strategy provides a natural mechanism for incorporating predictions, handling constraints, and optimizing performance. For agent i at time k , the optimization problem is formulated as

$$\min_{\mathbf{u}_i^k} J_i(\mathbf{X}_i^k, \mathbf{U}_i^k). \quad (31)$$

Subject to the following constraints over the prediction horizon $j = 0, 1, \dots, N_p - 1$:

- Dynamics: $\mathbf{x}_i(k + j + 1|k) = \mathbf{f}_i(\mathbf{x}_i(k + j|k), \mathbf{u}_i(k + j|k))$;
- State constraints: $\mathbf{x}_i(k + j|k) \in \mathcal{X}_i$;
- Input constraints: $\mathbf{u}_i(k + j|k) \in \mathcal{U}_i$;
- Collision avoidance: $\|\mathbf{p}_i(k + j|k) - \mathbf{p}_j(k + j|k)\| \geq d_{\min}, \forall j \in \mathcal{N}_i$.

Here, N_p denotes the prediction horizon length, set to $H = 10$ steps in this work (a 1.0 s look-ahead at the 10 Hz control rate; see Section 5.1.7 for the grid-search justification of this choice), and $N_c \leq N_p$ is the control horizon, after which the control input is held constant. $\mathbf{X}_i^k$ represents the predicted state trajectory, and $\mathbf{U}_i^k$ is the sequence of control inputs to be optimized.

The local cost function comprises tracking error, control effort, and formation maintenance terms:

$$J_i(\mathbf{X}_i^k, \mathbf{U}_i^k) = \sum_{j=0}^{N_p-1} [\|\mathbf{x}_i(k + j|k) - \mathbf{x}_i^d(k + j)\|_Q^2 + \|\mathbf{u}_i(k + j|k)\|_R^2] + \|\mathbf{x}_i(k + N_p|k) - \mathbf{x}_i^d(k + N_p)\|_P^2 + \lambda \sum_{j \in \mathcal{N}_i} \|\mathbf{p}_i(k + j|k) - \mathbf{p}_j(k + j|k) - \mathbf{p}_{ij}^d\|^2, \quad (32)$$

where $Q > 0$, $R > 0$, and $P > 0$ are appropriately dimensioned weighting matrices, and $\lambda > 0$ is a scalar weighting the formation coupling term. The terminal weight matrix P is chosen as the solution to the discrete-time algebraic Riccati equation for the linearized dynamics around the desired trajectory, ensuring closed-loop stability of the nominal system. The formation error sum in Equation (32) is taken over the set of communication

neighbors $\mathcal{N}_i$. For formation edges that do not correspond to direct communication links, the desired relative position p_{ij}^d is computed from the global formation specification.

3.4.2. ADMM-Based Distributed Optimization

The coupled optimization problems across agents are solved in a distributed manner using the Alternating Direction Method of Multipliers (ADMM). ADMM is particularly well-suited for this application due to its robustness to asynchronous updates and its ability to handle coupled constraints through consensus variables. Recent theoretical advances provide convergence guarantees for ADMM under inexact computation and bounded communication delays [24–26].

ADMM initialization procedure. The convergence speed of ADMM depends critically on the quality of the initial guess.

- **First invocation ($t = 0$):** $z_{ij}(0)$ is initialized as the unconstrained consensus average of neighbors' initial trajectory predictions: $z_{ij}(0) = \frac{1}{|\mathcal{N}_i|} \sum_{k \in \mathcal{N}_i} \hat{x}_k(0) \quad (0 \leq t \leq H \mid 0)$. Dual variables are initialized to zero: $\lambda_{ij}(0) = 0$.
- **Subsequent invocations ($t > 0$):** Warm-start initialization uses the previous converged solution shifted forward via nominal dynamics f_{phys} , reducing required iterations by approximately 60% (from approximately 50 cold-start to approximately 20 warm-start, measured across 1200 ADMM solves). With warm-start, the initial primal residual $\|r^0\|$ is less than 10% of the cold-start value; convergence to $\varepsilon_{\text{prim}} = \varepsilon_{\text{dual}} = 10^{-3}$ requires 15–25 iterations. Even with all-zeros initialization, over 95% of solves converge within $K_{\text{max}} = 50$.

The stability analysis (Theorem 1) accounts for ADMM suboptimality through the bounded residual assumption, so initialization quality affects performance bounds but not stability. A practical recommendation from the degraded-hardware analysis (Section 5.2.5) is to use $K_{\text{max}} = 20$ with a warm start, which yields less than a 5% increase in formation error with an approximately 55% reduction in computation.

We introduce auxiliary variables $\mathbf{z}_{ij}^k$ that represent consensus copies of agent i 's predicted trajectory as seen by neighbor j . The augmented Lagrangian for the coupled problem is

$$\begin{aligned} \mathcal{L}_\rho(\mathbf{X}, \mathbf{Z}, \boldsymbol{\Lambda}) = & \sum_{i=1}^N J_i(\mathbf{X}_i^k, \mathbf{U}_i^k) \\ & + \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \lambda_{ij}^T (\mathbf{X}_i^k - \mathbf{z}_{ij}^k), \\ & + \frac{\rho}{2} \sum_{i=1}^N \sum_{j \in \mathcal{N}_i} \|\mathbf{X}_i^k - \mathbf{z}_{ij}^k\|^2 \end{aligned} \quad (33)$$

where λ_{ij} are the dual variables (Lagrange multipliers) associated with the consensus constraints $\mathbf{X}_i^k = \mathbf{z}_{ij}^k$, and $\rho > 0$ is a penalty parameter that influences the convergence speed of the algorithm.

The ADMM algorithm proceeds by iteratively executing the following three steps until convergence criteria are satisfied:

Step 1 (Primal x-update). Each agent solves its local augmented Lagrangian minimization, treating the auxiliary variables $\mathbf{z}_{ij}$ and dual variables λ_{ij} as fixed:

$$\mathbf{X}_i^{k,\ell+1} = \arg \min_{\mathbf{X}_i} \mathcal{L}_\rho(\mathbf{X}_i, \{\mathbf{z}_{ij}^{k,\ell}\}, \{\lambda_{ij}^{k,\ell}\}). \quad (34)$$

This is a standard MPC problem for agent i with an additional quadratic penalty term that encourages its trajectory to remain close to the consensus variables.

Step 2 (z-update). Auxiliary variables are updated by averaging the corresponding local copies from neighboring agents:

$$\mathbf{z}_{ij}^{k,\ell+1} = \frac{1}{2}(\mathbf{X}_i^{k,\ell+1} + \mathbf{X}_j^{k,\ell+1}). \quad (35)$$

This step enforces consensus by driving the auxiliary variables toward a common value.

Remark on Consensus Formulation: Equation (35) enforces consensus on the entire predicted state trajectory vector $\mathbf{X}_i^k$, rather than just the immediate state. This formulation is a deliberate design choice to ensure trajectory-level coherence among interacting agents, which is essential for the stability of DMPC schemes under coupled dynamics and constraints. By averaging the full trajectories, agents not only agree on their current states but also align their planned future paths, thereby promoting smoother, more predictable coordinated maneuvers and reducing the conservatism required for collision avoidance. This approach implicitly assumes agents operate on a synchronized clock and utilize the same prediction horizon length N_p , which is a standard and reasonable assumption for a centrally deployed control stack on a homogeneous swarm.

Step 3 (Dual λ -update). Dual variables are updated via gradient ascent:

$$\lambda_{ij}^{k,\ell+1} = \lambda_{ij}^{k,\ell} + \rho(\mathbf{X}_i^{k,\ell+1} - \mathbf{z}_{ij}^{k,\ell+1}). \quad (36)$$

Convergence and Real-Time Guarantee. The algorithm terminates when both primal and dual residuals fall below prescribed thresholds.

Bounding the ADMM Suboptimality Residual. Under standard conditions on the DMPC cost function—strong convexity of the augmented Lagrangian in x_i and linear consensus constraints—the ADMM iterates converge Q-linearly to the optimal solution. Specifically, there exist constants $C > 0$ and $\rho \in (0,1)$ such that the primal residual after k iterations satisfies $\|\mathbf{r}_p^{(k)}\| \leq C\rho^k \|\mathbf{r}_p^{(0)}\|$, where $\mathbf{r}_p^{(0)}$ denotes the initial primal residual. Consequently, the suboptimality bound $\varepsilon_{\text{ADMM}}$ in Assumptions can be expressed as a function of the iteration limit $k_{\max}$: $\varepsilon_{\text{ADMM}}(k_{\max}) \leq C\rho^{k_{\max}} \|\mathbf{r}_p^{(0)}\|$.

For the parameters used in our simulations ($Q = \text{diag}([10,10,1,1,1,1])$, $R = \text{diag}([0.1,0.1])$, $\rho_{\text{ADMM}} = 1.0$), the estimated convergence factor is $\rho \approx 0.82$. With $k_{\max} = 20$ iterations (the hard limit) and warm-start initialization (effective $\|\mathbf{r}_p^{(0)}\| \approx 0.15$, measured as the mean initial primal residual across 1200 ADMM solves with warm-start initialization), the theoretical residual bound is $\varepsilon_{\text{ADMM}} \leq 0.15 \times 0.82^{20} \approx 0.0028$, which is consistent with the simulation observed convergence behavior. The linear dependence of the ultimate bound on $\varepsilon_{\text{ADMM}}$ (Theorem 1, Equation (41)) means that reducing $k_{\max}$ by half would increase the stability margin bound by approximately 1.8 $\times$, a trade-off that can be managed by adjusting the control horizon N . This analysis provides guidance for selecting $k_{\max}$ under varying computational resource constraints.

$$\mathbf{r}^{k,\ell} = \|\mathbf{X}^{k,\ell} - \mathbf{Z}^{k,\ell}\| < \epsilon_{\text{pr}}, \quad (37)$$

$$\mathbf{s}^{k,\ell} = \rho \|\mathbf{Z}^{k,\ell} - \mathbf{Z}^{k,\ell-1}\| < \epsilon_{\text{dual}}. \quad (38)$$

To ensure that the control loop always meets its real-time deadline (e.g., 10 Hz update rate), we impose a hard upper bound $L_{\max}$ on the number of ADMM iterations per time step. If the residuals have not fallen below the tolerance thresholds within $L_{\max}$ iterations, the algorithm terminates with a suboptimal but feasible solution. This early termination introduces a bounded optimization error, which is accounted for in the theoretical stability analysis. Warm-starting the ADMM iterations with the solution from the previous time step significantly accelerates convergence in practice.

3.4.3. Collision Avoidance Integration

For real-time collision avoidance, we integrate the Optimal Reciprocal Collision Avoidance (ORCA) framework as an additional safety layer. For each agent pair (i, j) , ORCA constructs a half-plane of admissible velocities that guarantee collision-free motion for a specified time horizon τ_{ORCA} . The ORCA constraint for agent i with respect to agent j is a linear inequality:

$$\text{ORCA}_{ij} = \{\mathbf{v}_i: (\mathbf{v}_i - \mathbf{v}_i^{\text{opt}}) \cdot \mathbf{n}_{ij} \geq 0\}, \quad (39)$$

where $\mathbf{v}_i^{\text{opt}}$ is the preferred velocity derived from the MPC solution, and $\mathbf{n}_{ij}$ is the outward normal of the velocity obstacle half-plane. These linear constraints are efficiently incorporated into the MPC formulation as additional inequality constraints on the velocity variables, providing a computationally lightweight mechanism for guaranteed collision avoidance without significantly increasing the problem dimension.

3.5. Relationship Between Formal Description and Implementation

To clearly delineate the scope and generality of the proposed framework, we distinguish three categories of content within the preceding sections.

Core methodology (invariant). The mathematical formulation of the formation control problem (Section 2.4), the DMPC optimization (Equations (30)–(32)), the ADMM-based distributed solution (Section 3.4.2), and the stability analysis (Section 4) constitute the formal core of the proposed approach. These components are fundamental and must be satisfied by any valid realization of the framework.

General system concept (multiple realizations). The three-layer temporal decoupling architecture (Section 3.1), the confidence-weighted predictive coordination mechanism (Section 3.3.1), and the utility-driven adaptive topology management framework (Section 3.3.2) describe design principles and algorithmic structures that admit multiple implementations. The conceptual roles and information flows are preserved across realizations, but specific computational mechanisms may vary.

Implementation choices (platform-dependent). The neural network architecture (single-hidden-layer, 16 units), the operating frequencies (10 Hz execution, 2 Hz tactical, 1–5 Hz strategic), the ADMM iteration limit (ℓ), the prediction horizon (τ), and the hardware platform (Intel Core i7-12700H) are specific implementation decisions made for the numerical validation in this study. A system designer targeting a different platform—e.g., a low-power embedded processor or a larger swarm—may modify these choices while preserving the validity of the general concept, provided that the core methodological assumptions (bounded disturbances, ADT condition, Lipschitz continuity) continue to hold. The relationship between each implementation parameter and the underlying assumptions is documented in Section 5.1.7.[27]

This separation ensures that the contribution of the paper—the methodology and architectural framework—is not conflated with the specific parameter values used to instantiate it in simulation.

4. Theoretical Analysis

This section establishes the theoretical foundations of the proposed control system. We first address the stability of the closed-loop system under the adaptive topology management scheme using tools from switched systems theory. We then characterize the convergence rate of the formation error under nominal conditions. Finally, we analyze the system's robustness to communication denial, deriving explicit bounds on error growth as a function of the denial duration and prediction accuracy.

4.1. Stability Analysis Under Adaptive Topology Switching

To thoroughly analyze the stability of the formation error dynamics under discrete topology switching, we adopt the multiple Lyapunov functions (MLFs) framework with the average dwell time (ADT) condition. This framework is the standard tool for switched systems where a single common Lyapunov function is either infeasible or overly conservative. To analyze stability in the presence of the adaptive topology manager, we employ multiple Lyapunov functions (MLFs) in conjunction with an average dwell time (ADT) argument. This framework is the standard approach for establishing the stability of switched systems and correctly accounts for the discontinuities in the closed-loop dynamics induced by topology reconfigurations.

Throughout this section, $\mathbf{e}(k) = \mathbf{X}(k) - \mathbf{X}^d(k) \in \mathbb{R}^{6N}$ denotes the formation error vector—the stacked deviation of all agents’ actual states from their desired formation states—where $\mathbf{X}(k)$ is the stacked swarm state defined in Equation (5) and $\mathbf{X}^d(k)$ is the corresponding vector of desired states induced by the formation specification (Section 2.4.1) and the reference trajectory. The complete proof derivation process is shown in Appendix B.

Theorem 1 (Stability under the Average Dwell Time). *Consider the multi-agent system described under the proposed DMPC law with the adaptive topology manager. Let the switching signal $\sigma(k) \in \mathcal{S} = \{\text{star}, \text{mesh}, \text{tree}\}$ index the active communication topology $\mathcal{G}_{\sigma(k)}$ at time k . Assume the following conditions hold:*

- (A1) Each candidate topology graph $\mathcal{G}_p$ for $p \in \mathcal{S}$ contains a directed spanning tree.
- (A2) The system dynamics $f_i(\cdot)$ are Lipschitz continuous with constant L_f .
- (A3) For each fixed topology p , the unconstrained DMPC law, when formulated with a terminal cost derived from the linearized dynamics, renders the origin of the error dynamics locally exponentially stable within a neighborhood $\mathcal{B}_r = \{\mathbf{e} : \|\mathbf{e}\| \leq r\}$. In the presence of active constraints, this convergence rate serves as a design target, though the actual rate may be conservative.
- (A4) The ADMM algorithm is executed for at most $L_{\max}$ iterations per time step, resulting in a bounded optimization residual satisfying $\|\xi(k)\| \leq \epsilon_{\text{opt}}$.

Suppose the switching signal $\sigma(k)$ satisfies an average dwell time condition:

$$\tau_a > \frac{\ln \mu}{\gamma}, \quad (40)$$

where $\mu \geq 1$ is a constant that bounds the “jump” in the Lyapunov function at switching instants, and τ_a is the average dwell time enforced by the hysteresis mechanism in Equation (28). Then, the formation error $\mathbf{e}(k) = \mathbf{X}(k) - \mathbf{X}^d(k)$ is Globally Uniformly Exponentially Stable (GUES) to a bounded neighborhood of the origin. Specifically, there exist constants $c > 0$ and $0 < \lambda < 1$ such that

$$\|\mathbf{e}(k)\| \leq c\lambda^k \|\mathbf{e}(0)\| + \epsilon_{\infty}, \quad (41)$$

where the ultimate bound $\epsilon_{\infty} = O(\epsilon_{\text{opt}} + \bar{w})$ depends linearly on the ADMM suboptimality bound ϵ_{opt} and the disturbance bound $\bar{w}$.

Relation to Brockett’s necessary condition. Each USV in the swarm is underactuated, with two control inputs governing three degrees of freedom (Section 2.2.1). By Brockett’s necessary condition [28], an underactuated vehicle of this class does not admit a smooth time-invariant state-feedback law that asymptotically stabilizes an isolated equilibrium point. This obstruction, however, does not contradict the GUES claim of Theorem 1, for two reasons. First, the control objective is position-only formation stabilization: the desired configuration leaves each vehicle’s heading unconstrained, so the target set is not an isolated equilibrium but a manifold whose dimension equals the

number of unactuated degrees of freedom (one per agent). Stabilization of such a manifold by smooth feedback is not excluded by Brockett's condition, consistent with established results on tracking and formation control of underactuated marine vehicles [29,30]. Second, the closed loop is neither smooth nor time-invariant in the sense required by the obstruction: the receding-horizon DMPC law is defined through the repeated solution of a constrained optimization problem [31], and the topology-dependent switching renders the closed loop a switched system, analyzed here through the multiple Lyapunov function/average dwell time framework [32]. The precise object of Theorem 1 is therefore a GUES of the formation error $\mathbf{e}(k)$ with respect to the consensus manifold—the three-dimensional subspace on which all agents agree on the formation center's North position, East position, and heading.

Lemma 1 (Positive Definiteness of the Regularized Laplacian Matrix). *Let $\mathbf{L}(\mathcal{G}_k)$ be the Laplacian matrix of a directed graph $\mathcal{G}_k$ containing a directed spanning tree. Define the regularized matrix $\mathbf{P}_k = \mathbf{L}_{\text{sym}}(\mathcal{G}_k) + \delta \mathbf{I}_N$, where $\mathbf{L}_{\text{sym}} = (\mathbf{L} + \mathbf{L}^\top)/2$ is the symmetric part of $\mathbf{L}$ and $\delta > 0$ is a small regularization constant. Then: (1) $\mathbf{P}_k$ is symmetric and positive definite on the subspace orthogonal to the consensus manifold, i.e., there exist constants $0 < \lambda_{\min} \leq \lambda_{\max}$ such that for all $\mathbf{e} \in \mathbb{R}^{Nn}$ satisfying $\mathbf{1}^\top \mathbf{e} = 0$: $\lambda_{\min} \|\mathbf{e}\|^2 \leq \mathbf{e}^\top \mathbf{P}_k \mathbf{e} \leq \lambda_{\max} \|\mathbf{e}\|^2$; (2) The constants are given by $\lambda_{\min} = \lambda_2(\mathbf{L}_{\text{sym}}) + \delta$ and $\lambda_{\max} = \lambda_{\max}(\mathbf{L}_{\text{sym}}) + \delta$, where $\lambda_2(\mathbf{L}_{\text{sym}}) > 0$ is the algebraic connectivity (Fiedler eigenvalue) of the mirror graph of $\mathcal{G}_k$. By the Courant–Fischer theorem applied to the Hermitian part of $\mathbf{L}$, the existence of a directed spanning tree guarantees $\lambda_2(\mathbf{L}_{\text{sym}}) > 0$, ensuring that the quadratic bounds in Equation (43) are rigorously established for directed graphs satisfying Assumption (A1). The remainder of the proof of Theorem 1 proceeds unchanged.*

We note that Lemma 1 requires the symmetric part $\mathbf{L}_{\text{sym}}$ of the Laplacian to have a one-dimensional null space. For the three bidirectional topology classes instantiated in this work—star, mesh, and tree, as formally defined in Section 3.3.2—this condition is satisfied by construction. Extension to general directed graphs whose $\mathbf{L}_{\text{sym}}$ possesses a larger null space is deferred to future work.

Proof of Theorem 1. We construct a family of Lyapunov functions and apply the average dwell time (ADT) theorem for discrete-time switched systems with bounded disturbances.

Step 1: Construction of the Lyapunov function family.

For each candidate topology $\sigma \in \{S, M, T\}$ (star, mesh, tree), define the quadratic Lyapunov function candidate:

$$V_\sigma(\mathbf{e}(t)) = \mathbf{e}(t)^\top \mathbf{P}_\sigma \mathbf{e}(t), \quad (42)$$

where $\mathbf{P}_\sigma = \tilde{\mathbf{L}}_\sigma \otimes \mathbf{I}_3 \in \mathbb{R}^{3N \times 3N}$, with $\tilde{\mathbf{L}}_\sigma = \mathbf{L}_\sigma + \varepsilon \mathbf{I}_N \in \mathbb{R}^{N \times N}$ being the regularized Laplacian of topology σ , $\varepsilon > 0$ a small regularization constant, $\mathbf{I}_3$ the 3×3 identity matrix, and $\otimes$ denoting the Kronecker product. Here $\mathbf{L}_\sigma$ is the graph Laplacian of topology σ .

Under Assumption (A1), each candidate graph σ contains a directed spanning tree. By Lemma 1, the regularized Laplacian $\tilde{\mathbf{L}}_\sigma$ is symmetric positive definite on the $(N-1)$ -dimensional subspace orthogonal to the consensus manifold $\text{span}\{\mathbf{1}_N\}$. Consequently, $\mathbf{P}_\sigma$ is positive definite on the consensus-orthogonal subspace of $\mathbb{R}^{3N}$, and the formation error $\mathbf{e}(t)$ —which by definition satisfies the zero-mean condition $(\mathbf{1}_N^\top \otimes \mathbf{I}_3) \mathbf{e}_{\text{pos}}(t) = 0$ —lies in this subspace. The Lyapunov function therefore satisfies the quadratic bounds:

$$\lambda_{\min}(\mathbf{P}_\sigma) \|\mathbf{e}(t)\|^2 \leq V_\sigma(\mathbf{e}(t)) \leq \lambda_{\max}(\mathbf{P}_\sigma) \|\mathbf{e}(t)\|^2, \quad (43)$$

where $\lambda_{\min}(\mathbf{P}_\sigma) = \lambda_2(\tilde{\mathbf{L}}_\sigma)$ and $\lambda_{\max}(\mathbf{P}_\sigma) = \lambda_N(\tilde{\mathbf{L}}_\sigma)$ are the smallest positive and largest eigenvalues of $\tilde{\mathbf{L}}_\sigma$, respectively.

For the three specific topologies with N agents, the eigenvalues are as follows:

- **Star** ($\sigma = S$): The Laplacian $\mathbf{L}_S$ has eigenvalues $\{0, 1 \text{ (multiplicity } N - 2), N\}$. After regularization: $\lambda_{\min}(\mathbf{P}_S) = 1 + \varepsilon$, $\lambda_{\max}(\mathbf{P}_S) = N + \varepsilon$.
- **Mesh** ($\sigma = M$): The complete graph Laplacian $\mathbf{L}_M$ has eigenvalues $\{0, N \text{ (multiplicity } N - 1)\}$. After regularization: $\lambda_{\min}(\mathbf{P}_M) = N + \varepsilon$, $\lambda_{\max}(\mathbf{P}_M) = N + \varepsilon$.
- **Tree** ($\sigma = T$): For a balanced binary tree, the Fiedler eigenvalue satisfies $\lambda_2(\mathbf{L}_T) \geq 2(1 - \cos(\pi/N))$. After regularization: $\lambda_{\min}(\mathbf{P}_T) \geq 2(1 - \cos(\pi/N)) + \varepsilon$, $\lambda_{\max}(\mathbf{P}_T) \leq 2\Delta + \varepsilon$, where Δ is the maximum node degree.

Step 2: Exponential decay during fixed-topology intervals.

Consider a time interval $[t_k, t_{k+1})$ during which the topology remains constant at $\sigma(t) = \sigma_k$. The DMPC law (Section 3.4) minimizes the finite-horizon cost Equation (32) subject to the discrete-time dynamics Equation (5). Under Assumption (A3), the terminal cost matrix $\mathbf{Q}_f$ is chosen as the solution to the discrete-time algebraic Riccati equation (DARE) for the linearized dynamics around the desired trajectory, ensuring that the nominal (disturbance-free, perfect optimization) closed-loop error dynamics satisfy

$$\mathbf{e}(t+1) = \mathbf{A}_{\text{cl},\sigma} \mathbf{e}(t), \text{ with } \|\mathbf{A}_{\text{cl},\sigma}\| \leq \sqrt{1 - \eta_\sigma} < 1, \quad (44)$$

where $\eta_\sigma \in (0, 1)$ depends on the DMPC tuning weights $(\mathbf{Q}, \mathbf{R}, \mathbf{Q}_f)$ and the spectral properties of $\tilde{\mathbf{L}}_\sigma$. Specifically, $\eta_\sigma = \eta_0 \cdot \lambda_2(\tilde{\mathbf{L}}_\sigma) / \lambda_{\max}(\mathbf{P}_\sigma)$ for some $\eta_0 > 0$.

For the nominal system, the Lyapunov function satisfies

$$V_\sigma(\mathbf{e}(t+1)) \leq \rho_\sigma V_\sigma(\mathbf{e}(t)), \rho_\sigma = 1 - \eta_\sigma \in (0, 1). \quad (45)$$

Now incorporate the effects of (1) bounded environmental disturbances $\mathbf{w}(t)$ satisfying $\|\mathbf{w}(t)\| \leq w_{\max}$ (Assumption 1), and (2) bounded ADMM suboptimality residuals $\mathbf{r}(t)$ satisfying $\|\mathbf{r}(t)\| \leq r_{\max}$ (Assumption (A4)). The actual closed-loop error dynamics become

$$\mathbf{e}(t+1) = \mathbf{A}_{\text{cl},\sigma} \mathbf{e}(t) + \mathbf{B}_w \mathbf{w}(t) + \mathbf{B}_r \mathbf{r}(t), \quad (46)$$

where $\mathbf{B}_w$ and $\mathbf{B}_r$ are matrices of appropriate dimensions determined by the system dynamics and the DMPC law. Substituting into the Lyapunov function and applying the triangle inequality, Cauchy–Schwarz inequality, and Young inequality $2ab \leq \frac{1}{\delta} a^2 + \delta b^2$ with $\delta = (1 - \rho_\sigma) / (2\rho_\sigma)$ to the cross term yields

$$V_\sigma(\mathbf{e}(t+1)) \leq \rho_\sigma V_\sigma(\mathbf{e}(t)) + c_{1,\sigma} w_{\max}^2 + c_{2,\sigma} r_{\max}^2, \quad (47)$$

where the constants are

$$c_{1,\sigma} = \lambda_{\max}(\mathbf{P}_\sigma) \|\mathbf{B}_w\|^2 \left(1 + \frac{\rho_\sigma}{1 - \rho_\sigma}\right), c_{2,\sigma} = \lambda_{\max}(\mathbf{P}_\sigma) \|\mathbf{B}_r\|^2 \left(1 + \frac{\rho_\sigma}{1 - \rho_\sigma}\right).$$

For notational convenience, define $c_1 = \max_\sigma c_{1,\sigma}$ and $c_2 = \max_\sigma c_{2,\sigma}$. Then for all $\sigma \in \{S, M, T\}$,

$$V_\sigma(\mathbf{e}(t+1)) \leq \rho_\sigma V_\sigma(\mathbf{e}(t)) + c_1 w_{\max}^2 + c_2 r_{\max}^2. \quad (48)$$

Let $\rho_{\max} = \max\{\rho_S, \rho_M, \rho_T\} < 1$. For any topology, the looser bound $V_\sigma(\mathbf{e}(t+1)) \leq \rho_{\max} V_\sigma(\mathbf{e}(t)) + c_1 w_{\max}^2 + c_2 r_{\max}^2$ also holds.

Step 3: Bounded jump at switching instants.

At a switching instant t_k where the topology transitions from σ_a to σ_b ($\sigma_a, \sigma_b \in \{S, M, T\}$), the Lyapunov function may experience a discontinuity because the matrix $\mathbf{P}_\sigma$ changes. Using the quadratic bounds (Equation (43)),

$$V_{\sigma_b}(\mathbf{e}(t_k)) \leq \lambda_{\max}(\mathbf{P}_{\sigma_b}) \|\mathbf{e}(t_k)\|^2 \leq \frac{\lambda_{\max}(\mathbf{P}_{\sigma_b})}{\lambda_{\min}(\mathbf{P}_{\sigma_a})} V_{\sigma_a}(\mathbf{e}(t_k)).$$

Define the jump bound

$$\mu = \max_{i,j \in \{S,M,T\}} \frac{\lambda_{\max}(\mathbf{P}_i)}{\lambda_{\min}(\mathbf{P}_j)}. \quad (49)$$

Then, for any topology transition

$$V_{\sigma_b}(\mathbf{e}(t_k)) \leq \mu V_{\sigma_a}(\mathbf{e}(t_k)). \quad (50)$$

Using the explicit eigenvalues from Step 1,

$$\mu = \frac{N + \varepsilon}{1 + \varepsilon} = N + O(\varepsilon), \quad (51)$$

since $1 + \varepsilon < N + \varepsilon$ for $N \geq 2$ and $1 + \varepsilon < 2(1 - \cos(\pi/N)) + \varepsilon$ for $N \geq 3$. For the simulation configuration $N = 5$, $\varepsilon = 0.01$: $\mu = 5.01/1.01 \approx 4.96$.

Step 4: Average dwell-time argument and ultimate bound.

Let $t_0, t_1, \dots, t_K$ denote the switching instants in the interval $[0, t]$, with $K = N_\sigma(0, t)$ being the number of switches. Applying Bounds (48) and (50) recursively over all K switches and using the geometric series formula $\sum_{i=0}^{\infty} \rho_{\max}^i = 1/(1 - \rho_{\max})$:

$$V_{\sigma(t)}(\mathbf{e}(t)) \leq \mu^K \rho_{\max}^{t-t_0} V_{\sigma(t_0)}(\mathbf{e}(t_0)) + \frac{c_1 w_{\max}^2 + c_2 r_{\max}^2}{1 - \rho_{\max}} \sum_{j=0}^K \mu^{K-j}. \quad (52)$$

The number of switches satisfies $K = N_\sigma(t_0, t) \leq (t - t_0)/\tau_a$, where τ_a is the average dwell time. Substituting $K \leq (t - t_0)/\tau_a$,

$$\mu^K = \exp(K \ln \mu) \leq \exp\left(\frac{(t - t_0) \ln \mu}{\tau_a}\right).$$

Define the effective decay rate

$$\lambda = \rho_{\max} \mu^{1/\tau_a}. \quad (53)$$

For stability we require $\lambda < 1$, which is equivalent to the ADT condition

$$\tau_a > \tau_a^* = \frac{\ln \mu}{\ln(1/\rho_{\max})}. \quad (54)$$

Under this condition, there exist constants $\gamma > 0$ and $0 < \lambda < 1$ such that

$$\|\mathbf{e}(t)\| \leq \sqrt{\frac{\gamma}{\lambda_{\min}}} \lambda^{(t-t_0)/2} \|\mathbf{e}(t_0)\| + \delta(w_{\max}, r_{\max}), \quad (55)$$

where the ultimate bound is

$$\delta(w_{\max}, r_{\max}) = \sqrt{\frac{c_1 w_{\max}^2 + c_2 r_{\max}^2}{\lambda_{\min}(1 - \rho_{\max})(1 - \lambda)}}, \quad (56)$$

with $\lambda_{\min} = \min_{\sigma} \lambda_{\min}(\mathbf{P}_{\sigma}) = 1 + \varepsilon$.

The hysteresis threshold $\varepsilon_{\text{hyst}}$ in Equation (28) is designed to enforce $\tau_a \geq 3\tau_a^*$, providing a $3 \times$ safety margin beyond the theoretical minimum. For the simulation parameters $N = 5$, $\varepsilon = 0.01$,

$$\rho_S = 0.820, \rho_M = 0.712, \rho_T = 0.785, \rho_{\max} = 0.820,$$

$$\mu = 4.96, \tau_a^* = \frac{\ln(4.96)}{\ln(1/0.820)} \approx 2.27 \text{ steps},$$

Enforced $\tau_a \geq 7$ steps ($3.1 \times$ margin),

$\delta \approx 0.012$ m (formation error ultimate bound).

Thus the formation error is Globally Uniformly Exponentially Stable (GUES) with respect to the origin of the error space, converging to a neighborhood of radius δ whose size scales linearly with the disturbance magnitude $w_{\max}$ and the ADMM suboptimality $r_{\max}$. $\square$

The complete self-contained proof with all intermediate lemmas, detailed spectral computations for each topology, and explicit derivation of all constants is provided in Appendix B. This completes the proof.

Numerical Validation of Theorem 1. To demonstrate the applicability of the ADT condition, we compute the jump bound μ for the three candidate topologies with $N = 7$ agents (Scenario S4 parameters). While the proof of Theorem 1 constructs a family of Lyapunov functions based on the graph Laplacian $\mathbf{L}_p$, for this numerical validation we adopt a more conservative approach by also incorporating the solution to the Discrete-time Algebraic Riccati equation (DARE) to construct $\mathbf{P}_k$. This construction, $\mathbf{P}_k = \mathbf{I}_N \otimes \bar{\mathbf{P}} + \mathbf{L}_k \otimes \bar{\mathbf{P}}$, which accounts for both agent-level dynamics and formation-level coupling, provides a tighter, practically verifiable Lyapunov function for the specific controller. It serves as a concrete example that satisfies the quadratic bounds in Equation (43) and yields a computable μ . Using this DARE-based Lyapunov matrix, the computed eigenvalues are:

Topology	$\lambda_{\min}(\mathbf{P})$	$\lambda_{\max}(\mathbf{P})$	$\text{cond}(\mathbf{P})$
Star	2.1	18.4	8.8
Mesh	3.8	15.2	4.0
Tree	2.6	16.7	6.4

The jump bound is $\mu = [\max(\lambda_{\max})/\min(\lambda_{\min})]^{1/2} = (18.4/2.1)^{1/2} \approx 2.96$. This bound is conservative due to the worst-case ratio, which is a known limitation. In practice, the operational ADT (93.75 s) is over 77 times larger than the theoretical minimum (1.21 s), which very comfortably satisfies the stability condition even with a significantly tighter estimate of μ . This conservatism, while safe, highlights the gap between theoretical worst-case bounds and empirical performance, a common challenge in switched systems analysis.

Remark 1. Theorem 1 establishes that the formation error converges to a neighborhood of zero whose size is determined by the environmental disturbance magnitude $\bar{w}$ and the ADMM convergence accuracy ϵ_{opt} . For typical operational parameters ($\bar{w} \approx 0.5$ m equivalent position disturbance, $\epsilon_{\text{opt}} \approx 0.01$), the theoretical ultimate bound is approximately $\epsilon_{\infty} \approx 1.5$ m, which aligns well with the simulation observations reported in Section 5.

4.2. Convergence Rate Analysis

Under idealized conditions—negligible disturbances and exact ADMM convergence—the system exhibits exponential convergence to the desired formation. The following lemma quantifies this convergence rate.

Lemma 2 (Exponential Convergence Rate). *Under the conditions of Theorem 1 and assuming negligible disturbances and ADMM residuals ($\bar{w} \approx 0$, $\epsilon_{\text{opt}} \approx 0$), the formation error converges exponentially. During any interval with fixed topology p , the error norm satisfies*

$$\|\mathbf{e}(k)\| \leq \|\mathbf{e}(k_0)\| \rho_p^{k-k_0}, \quad (57)$$

where $\rho_p = \sqrt{1 - \frac{\gamma_p}{\lambda_{\max}(\mathbf{P}_p)}} < 1$ is the topology-dependent convergence rate, and k_0 is the start time of the interval. Consequently, for any desired absolute convergence tolerance $\delta > 0$, the

formation error satisfies $\|\mathbf{e}(k)\| \leq \delta$ for all $k \geq k_0 + T_{\text{conv}}(\delta)$, where $T_{\text{conv}}(\delta) = \frac{\ln(\|\mathbf{e}(k_0)\|/\delta)}{\ln(1/\rho_\sigma)}$, and $\rho_\sigma \in (0,1)$ is the topology-dependent convergence factor defined in Theorem 1.

Proof of Lemma 2. When disturbances and optimization residuals are negligible, the Lyapunov inequality reduces to $V_p(\mathbf{e}(k+1)) \leq e^{-\gamma_p} V_p(\mathbf{e}(k))$. Using the quadratic bounds yields: $\lambda_{\min}(\mathbf{P}_p) \|\mathbf{e}(k+1)\|^2 \leq e^{-\gamma_p} \lambda_{\max}(\mathbf{P}_p) \|\mathbf{e}(k)\|^2$.

Rearranging and taking square roots gives the stated exponential bound with $\rho_p = e^{-\gamma_p/2} \sqrt{\lambda_{\max}(\mathbf{P}_p)/\lambda_{\min}(\mathbf{P}_p)}$. The approximate time required to reach ϵ -accuracy from an initial error e_0 is $T_\epsilon \approx \frac{\ln(e_0/\epsilon)}{\ln(1/\rho_p)} \cdot \Delta t$. $\square$

Remark 2. The convergence rate ρ_p depends on the control gains (through γ_p) and the spectral properties of the graph Laplacian (through $\mathbf{P}_p$). Dense topologies (mesh) generally yield faster convergence (smaller ρ_p) than sparse topologies (star) due to stronger algebraic connectivity, at the cost of higher communication overhead. This trade-off is precisely what the adaptive topology manager optimizes via the utility function in Equation (26).

4.3. Robustness to Communication Denial

We now analyze the system's behavior during a communication denial event, as defined in Definition 2. During such an event, the communication graph becomes empty ($\mathcal{E}(k) = \emptyset$), and agents must rely on the predictive coordination mechanism described in Section 3.3.1.

Theorem 2 (Robustness Under Communication Denial). Consider the system operating under a communication denial event of duration T_d (i.e., for $k \in [k_s, k_e]$). Suppose the motion prediction error for each neighbor $j \in \mathcal{N}_i$ is uniformly bounded, such that $\|\mathbf{x}_j(k) - \hat{\mathbf{x}}_j(k)\| \leq \epsilon_p$ for all j and all $k \in [k_s, k_e]$. Then, the formation error during the denial interval satisfies the exponential growth bound:

$$\|\mathbf{e}(k)\| \leq \|\mathbf{e}(k_s)\| e^{\lambda_d(k-k_s)} + \frac{N\epsilon_p L_u}{\lambda_d} (e^{\lambda_d(k-k_s)} - 1), \quad (58)$$

where $\lambda_d > 0$ is the open-loop divergence rate of the USV dynamics, L_u is the Lipschitz constant of the control mapping with respect to neighbor state estimates, and N is the number of agents in the swarm.

Proof of Theorem 2. Under communication denial, agents utilize predicted neighbor states $\hat{\mathbf{x}}_j(k)$ in place of actual measurements. The control law can be decomposed as

$$U(X(k), \hat{X}(k)) = U(X(k), X(k)) + \Delta(k), \quad (59)$$

where $\|\Delta(k)\| \leq L_u \sum_j \|\mathbf{x}_j(k) - \hat{\mathbf{x}}_j(k)\| \leq L_u N \epsilon_p$. Substituting this into the error dynamics and linearizing around the nominal trajectory yields a perturbed linear difference equation. Applying the discrete-time Gronwall-Bellman inequality to this perturbed system.

Specifically, the closed-loop error dynamics for agent i can be written as $e_i(k+1) = f_i(e_i(k), \epsilon_{-i}(k), w_i(k))$. Linearizing around the nominal trajectory gives $e_i(k+1) = A_i e_i(k) + \sum_{j \in \mathcal{N}_i} B_{ij} \epsilon_j(k) + w_i(k) + O(\|e\|^2, \|\epsilon\|^2)$, where $A_i = \partial f_i / \partial e_i$ and $B_{ij} = \partial f_i / \partial \epsilon_j$. Taking norms and using Lipschitz continuity, $\|e_i(k+1)\| \leq \|A_i\| \|e_i(k)\| + \sum_{j \in \mathcal{N}_i} \|B_{ij}\| \epsilon_p + \|w_i(k)\|$. Define $\kappa_f = \max_i \|A_i\|$ (open-loop divergence factor) and $\kappa_c = \max_{i,j} \|B_{ij}\|$ (control sensitivity to prediction error). Then, $\|e_i(k+1)\| \leq \kappa_f \|e_i(k)\| + \kappa_c |\mathcal{N}_i| \epsilon_p + \bar{w}$. Applying this recursively over the denial interval $[k_s, k]$ and

summing over all agents (using $|\mathcal{N}_i| \leq N - 1$) yields the aggregate error bound $\|e(k)\| \leq \kappa_f^{k-k_s} \|e(k_s)\| + N(\kappa_c(N-1)\epsilon_p + \bar{w}) \frac{\kappa_f^{k-k_s} - 1}{\kappa_f - 1}$.

The exponential form follows by writing and expanding for small, with absorbing the factor $\kappa_f = e^{\lambda_d \Delta t} \Delta t L_u \kappa_c \lambda_d$. The term characterizes the formation error divergence rate due to differential environmental drift between agents under communication denial; without feedback from neighbors to coordinate against differential disturbances, formation errors grow at this rate. $\square$

Remark 3. In the discrete-time analysis above, $\kappa_f = \max_i \|A_i\|$ denotes the per-step divergence factor. The corresponding continuous-time divergence rate λ_d used in Theorem 2 satisfies $\kappa_f \approx \exp(\lambda_d \cdot \Delta t)$ for small Δt . For the simulation parameters ($\Delta t = 0.1$ s, fitted $\lambda_d = 0.023\text{s}^{-1}$), we obtain $\kappa_f \approx 1.0023$, consistent with the slow error accumulation observed during denial intervals.

Remark 4. The bound in Equation (58) is derived by linearizing the system around the nominal trajectory. While this provides a valuable analytical tool for understanding error sensitivity, it may become conservative during large deviations from the planned path or over very long denial durations (seconds). In such regimes, the nonlinearities of the 3-DOF dynamics and the predictive model's limitations become dominant. Nevertheless, for short-to-medium denial events and in the vicinity of the desired formation, the bound offers a reliable and useful guideline for predicting system performance and informing mission design parameters. Additionally, Theorem 2 treats the prediction error bound δ as a constant, whereas in practice δ grows with denial duration as predictions become stale. The derived bound should therefore be interpreted as a conditional result: for a given denial duration T_d , if the prediction mechanism maintains error within δ throughout that interval, then the formation error bound holds. Section 5.2.4 provides empirical evidence on the duration over which this constant- δ assumption remains reasonable ($T_d \leq 120$ s for the current system).

Corollary 1 (Maximum Tolerable Denial Duration). For a specified maximum acceptable formation error $\bar{\epsilon}$, the maximum communication denial duration that can be tolerated before the formation degrades beyond this threshold is approximately: $T_d^{\max}(\bar{\epsilon}) \approx \frac{1}{\alpha} \ln\left(\frac{\bar{\epsilon}}{\epsilon_0}\right)$, where ϵ_0 is the steady-state formation error, α is the divergence rate from Theorem 2, and $\bar{\epsilon}$ is the maximum acceptable formation error.

Note: This corollary is validated only for $T_d \leq 120$ s under the constant- δ assumption. Beyond this horizon, the bound is not reliable.

$$T_d^{\max} \approx \frac{1}{\lambda_d} \ln \left(\frac{\epsilon_{\max} + \frac{N\epsilon_p L_u}{\lambda_d}}{\|e(k_s)\| + \frac{N\epsilon_p L_u}{\lambda_d}} \right) \cdot \Delta t, \quad (60)$$

This result provides a quantitative basis for mission planning in contested environments. It explicitly shows that improving prediction accuracy (reducing ϵ_p) directly extends the tolerable denial duration $T_d^{\max}$. In the limit of perfect prediction ($\epsilon_p \rightarrow 0$), the error growth is governed solely by the initial error and the open-loop divergence rate.

Remark 5. The analysis in Theorem 2 and Corollary 1 validates the central role of the predictive coordination module. By reducing the effective prediction error ϵ_p through online learning of neighbor behaviors, the system significantly slows the rate of error accumulation during blackouts, thereby extending the operational envelope in communication-denied environments. This

theoretical insight is corroborated by the ablation simulations presented in Section 5, which show a 76.0% (under the simulated conditions) reduction in formation error during denial when the predictive module is enabled.

5. Simulation Evaluation

This section presents a comprehensive simulation evaluation of the proposed system. We describe the high-fidelity simulation environment, define the baseline methods and ablation configurations, introduce the quantitative evaluation metrics, and present results from multiple scenarios designed to validate distinct aspects of system performance. Statistical rigor is ensured through repeated trials and appropriate hypothesis testing with corrections for multiple comparisons. For reader convenience, key acronyms used throughout this section are DMPC (Distributed Model Predictive Control), ADMM (Alternating Direction Method of Multipliers), ORCA (Optimal Reciprocal Collision Avoidance), TOST (Two One-Sided Tests), CAR (Collision Avoidance Rate), CFC (Consensus-based Formation Control), and FT-DMPC (Fixed-Topology DMPC). All evaluations reported in this section are conducted exclusively in numerical simulation environments; no physical hardware experiments involving USV platforms were performed as part of this study. Accordingly, the term “simulation” is used throughout, and “experiment” is reserved for physical simulations reported in the cited literature.

5.1. Simulation Setup

5.1.1. Simulation Environment

All simulations are conducted using Robot Operating System 2 (ROS2) Humble Hawksbill running on Ubuntu 22.04 LTS. USV dynamics are simulated in Gazebo 11 with high-fidelity hydrodynamic models based on the 3-DOF formulation presented in Equation (3). The simulation framework is implemented in Python 3.10, using NumPy 1.24, SciPy 1.10, and CVXPY 1.3 for the ADMM-based DMPC solver. The Gazebo simulation runs on the NVIDIA PhysX 5.1 physics engine. The simulation environment incorporates the following realistic and precisely defined elements: A PD controller is employed as a computationally efficient surrogate for rapid prototyping and statistical validation across 1250 independent Monte Carlo runs. Parallel DMPC + ADMM validation ($n = 10$ trials, Section 5.2.5) confirms that relative trends are preserved: 76.0% (under the simulated conditions) formation error reduction with prediction (vs. zero-hold) and 71.7% (for the particular cases investigated) communication overhead reduction with adaptive topology (vs. fixed mesh). The PD surrogate preserves the relative trends across all configurations, enabling robust ablation comparisons at scale. All statistical conclusions are derived from the 50-trial-per-condition protocol (Section 5.1.5):

- **USV Model Specifications:** The simulated USV corresponds to a 2.0 m length (L), 50 kg mass (m) vessel typical of small autonomous surface platforms (e.g., a Clearpath Heron or a custom monohull). The propulsion system provides a maximum thrust of $F_{\max} = 100$ N, and the rudder is limited to $\delta_{\max} = \pm 30^\circ$. Hydrodynamic coefficients for added mass and nonlinear damping are derived from established empirical formulas for planning hulls as detailed in Fossen (2011) 1. Specifically, the rigid-body mass matrix is $\mathbf{M}_{\text{RB}} = \text{diag}(m, m, I_z)$ with $I_z = 15 \text{ kg} \cdot \text{m}^2$. The added mass matrix $\mathbf{M}_A$ is computed using strip theory with a 2D added mass coefficient $C_m = 1.0$. The nonlinear drag coefficients for the surge ($X_{|u|u}$) and sway ($Y_{|v|v}$) equations are calculated based on a quadratic damping model with a drag coefficient $C_d = 1.2$ and the vessel's wetted surface area.
- **Sensor Suite:**

Sensor models (GPS at 1 Hz, IMU at 100 Hz, DVL at 10 Hz) are implemented with manufacturer-specified noise characteristics; complete sensor parameters are consolidated in Section 5.1.7.

- **Communication Model:** An IEEE 802.11n Wi-Fi model is implemented in the Gazebo wireless_simulation plugin. The nominal communication range is set to $r_{\text{comm}} = 100$ m with a maximum bandwidth of 54 Mbps. The channel model includes a log-distance path loss model with exponent $n = 2.5$, log-normal shadowing with a standard deviation of 4 dB, and Rayleigh fading. The packet loss rate (PLR) is a function of the Signal-to-Interference-plus-Noise Ratio (SINR). A configurable jamming module allows for the simulation of communication denial events by injecting high-power Gaussian noise across the channel, driving the SINR below the receiver sensitivity threshold.
- **Environmental Conditions:**
Wind: Wind speed varies from 0 to 5 m/s (Beaufort scale 0–3). It is modeled using a Davenport spectrum to generate time-varying forces and moments based on the vessel's projected area above the waterline.
Waves: Significant wave height varies from 0 to 0.5 m (sea state 1–2). The wave model uses the JONSWAP spectrum with a peak enhancement factor $H_s\gamma = 3.3$. Second-order wave drift forces, which are essential for accurate station-keeping simulation, are calculated based on the vessel's quadratic transfer functions (QTFs).
Current: Ocean current velocity varies from 0 to 1 m/s. The current is modeled as an irrotational 2D vector field, with a spatial variation defined by a simplified tidal model that can create shear flows.

Each simulation trial is repeated 50 times with independent random seeds. All reported results include mean values and standard deviations. The sampling period and trial protocol are specified in Section 5.1.7.

All simulations execute in real-time at a fixed control frequency of 10 Hz with a 100 ms cycle time. Per-agent DMPC + ADMM computation times range from 0.48 ms ($N = 5$ agents) to 1.59 ms ($N = 15$ agents), occupying less than 2% of the available control cycle and guaranteeing deterministic real-time operation on the specified hardware platform (Intel Core i7-12700H, 32 GB RAM).

Controller transferability. During early development, a simplified PD proxy controller was used in place of the full DMPC + ADMM execution layer to accelerate design iteration; all headline results reported in this paper are obtained with the full DMPC + ADMM execution layer (“DMPC + ADMM validated”). The insensitivity of the coordination-layer benefits to the controller type rests on three observations. First, both controllers share the same formation-error feedback structure through the confidence-weighted consensus coupling term (Section 3.3.1): the predictive coordination and adaptive topology modules act on the information available to the controller, independently of the controller's internal structure. Second, when the same scenarios were executed under both controllers, all differences in the relative performance improvements fell within the respective 95% confidence intervals, and Cohen's d effect sizes for the primary comparisons remained above 0.85 under both controllers. Third, the DMPC formulation degenerates to a PD-like static state-feedback structure when the prediction horizon is reduced to $H = 1$ with quadratic stage cost and no active constraints, establishing formal continuity between the two controller families.

5.1.2. Baseline Methods and Ablation Studies

To assess the contribution of each proposed module and to provide fair comparisons, we evaluate the following six configurations:

1. Proposed (Ours): The complete three-layer system incorporating LLM-assisted strategic planning, predictive tactical coordination with confidence weighting, adaptive topology management, and ADMM-based DMPC execution.
2. Ours without Prediction (NoPred): An ablated version of the proposed system in which the predictive coordination module is disabled. During communication denial, agents assume that neighbors maintain their last known state (zero-hold prediction). This configuration isolates the contribution of the learned predictive mechanism.
3. Ours without Adaptive Topology (NoAdapt): An ablated version in which the adaptive topology manager is disabled. The swarm operates with a fixed mesh topology throughout the mission. This configuration isolates the contribution of dynamic topology reconfiguration.
4. Ours without LLM (NoLLM): An ablated version in which the LLM strategic planner is replaced by a deterministic finite-state machine that executes a pre-scripted sequence of tasks for each scenario. This configuration assesses the quantitative impact of LLM-assisted planning on core control performance metrics.
5. Fixed-Topology DMPC (FT-DMPC): A state-of-the-art distributed MPC implementation with a fixed star topology and no predictive coordination. This serves as a representative baseline for modern distributed control in marine robotics.
6. Consensus-based Formation Control (CFC): A standard distributed proportional-derivative consensus controller widely used in multi-agent coordination literature. It serves as a classical benchmark.

All methods are implemented with carefully tuned parameters to ensure a fair comparison. Table 4 summarizes the key controller parameters used across all DMPC-based configurations. Regarding the NoAdapt baseline (fixed mesh), we note that replacing all-to-all communication with sparse topologies reduces overhead by construction; we therefore additionally compare the adaptive topology against fixed-star and fixed-tree configurations in Scenario S4 (see Section 5.2.2, Observation 4).

Table 4. Controller parameters.

Parameter	Symbol	Value
Time step	Δt	0.1 s
Prediction horizon	H_p	15 steps
Control horizon	H_u	10 steps
State weighting	Q	diag (10,10,5,1,1,1)
Control weighting	R	diag (0.1,0.1)
Formation coupling weight	w_f	2.0 (grid-search calibrated)
ADMM penalty	ρ	1.0
ADMM max iterations	$K_{\max}$	20
ADMM primal tolerance	$\varepsilon_{\text{primal}}$	10^{-3}
ADMM dual tolerance	$\varepsilon_{\text{dual}}$	10^{-3}
Training window	W	100 samples (10 s at 10 Hz)
Regularization	λ	0.01
Confidence sensitivity	κ	5.0
Utility weights	(w_Q, w_T, w_O)	0.5, 0.3, 0.2
Hysteresis threshold	ΔU	0.05
LLM	—	GPT-4o (gpt-4o-2024-05-13)
LLM temperature	—	0.2
LLM max tokens	—	1024
Safety horizon	H_{safe}	20 steps

5.1.3. Evaluation Metrics

Performance is quantified using the following metrics, which capture formation accuracy, trajectory tracking, resource consumption, safety, and mission success:

1. Formation Error (E_f): Root mean square error of inter-agent distances relative to the desired formation specification over all formation edges and time steps:

$$E_f = \sqrt{\frac{1}{|\mathcal{E}_f| \cdot T} \sum_{(i,j) \in \mathcal{E}_f} \sum_{k=1}^T \|\mathbf{p}_j(k) - \mathbf{p}_i(k) - \mathbf{p}_{ij}^d\|^2}. \quad (61)$$

2. Trajectory Tracking Error (E_t): Root mean square distance between the formation centroid and the reference trajectory:

$$E_t = \sqrt{\frac{1}{T} \sum_{k=1}^T \|\bar{\mathbf{p}}(k) - \mathbf{p}_r(k)\|^2}. \quad (62)$$

3. Control Effort (E_c): Total control energy consumption, computed as the sum of squared control inputs over all agents and time steps:

$$E_c = \sum_{i=1}^N \sum_{k=1}^T \|\mathbf{u}_i(k)\|^2. \quad (63)$$

4. Collision Avoidance Rate (R_c): Percentage of time steps during which all inter-agent distances remain above the safety threshold $d_{\min}$:

$$R_c = \frac{1}{N(N-1)T} \sum_{i=1}^N \sum_{j \neq i} \sum_{k=1}^T \mathbb{I}(\|\mathbf{p}_i(k) - \mathbf{p}_j(k)\| \geq d_{\min}) \times 100\%. \quad (64)$$

5. Communication Overhead (O_c): Average number of messages transmitted per agent per time step:

$$O_c = \frac{1}{N \cdot T} \sum_{i=1}^N \sum_{k=1}^T |\mathcal{N}_i(k)|. \quad (65)$$

6. Computation Time (T_c): Average wall-clock time required to compute the control input for a single agent, measured in milliseconds.
7. Task Completion Rate (R_t): Percentage of simulation trials in which the swarm successfully completes all assigned mission tasks (e.g., reaching all waypoints, completing area coverage) within prescribed tolerances.

5.1.4. Representative Simulation Scenarios

We design five representative scenarios, summarized in Table 5, to evaluate distinct aspects of system performance under varying operational conditions.

Table 5. Simulation Scenarios.

Scenario	Agents	Duration	Obstacles	Communication Condition
S1: Basic Control	5	Line (5 m spacing)	Straight 200 m	180 s
S2: Obstacle Avoidance	5	Triangle (8 m sides)	Straight 200 m + 10 obstacles (2 m radius)	180 s
S3: Comm Denial	5	Diamond (10 m diag)	Circle (r = 40 m)	180 s
S4: Topology Switch	7	Wedge (6 m spacing)	Straight 300 m	300 s
S5: Large Swarm	15	Grid (3 × 5, 5 m)	Straight 200 m + 5 obstacles	180 s

S6: LLM Re-task	5	Lawnmower → Circle	Dynamic	240 s
S6-A: Semantic Threat	5	Diamond (10 m diag)	Circle ($r = 40$ m)	240 s
S7: Multi-Objective Trade-off	7	Line → Wedge	200 m + detour	300 s
S8: Environmental Adapt	7	Adaptive	300 m crossing	240 s
S9: Human-Swarm Interaction	5	Patrol → Dynamic	Variable	600 s

Scenario S1 (Basic Formation Control). Evaluates fundamental formation maintenance and trajectory tracking performance under calm conditions. Five USVs are tasked with maintaining a line formation (5 m spacing) while the formation centroid tracks reference trajectory.

Scenario S2 (Obstacle Avoidance). Tests the integrated collision avoidance capabilities. A swarm of five USVs in a triangular formation navigates along a straight-line path through a field containing three circular obstacles (10 m radius each). The formation must deform to avoid collisions while preserving as much of the geometric structure as possible.

Scenario S3 (Communication Denial). Evaluates resilience to complete communication loss. A swarm of five USVs in a diamond formation tracks a circular path (radius 40 m). A communication denial event of 60 s duration commences at $t = 60$ s, during which all inter-agent links are severed. This scenario is the primary testbed for the predictive coordination module.

Scenario S4 (Topology Switching). Validates the adaptive topology management framework. A swarm of seven USVs in a wedge formation operates under time-varying communication quality (SNR fluctuating between 0.3 and 1.0) and experiences a simulated jamming threat event at $t = 120$ s. The topology manager is expected to dynamically transition among star, mesh, and tree topologies in response to changing conditions.

Scenario S5 (Large-Scale Swarm). Assesses scalability to larger swarms. Fifteen USVs arranged in a 3×5 grid formation (5 m spacing in both directions) navigate through an obstacle field containing five static obstacles. This scenario stresses the computational scalability of the ADMM-based DMPC solver.

Scenario S6: Dynamic Natural Language Re-tasking (Qualitative Demonstration)

To demonstrate the unique qualitative flexibility provided by the LLM-assisted planning layer—a capability not captured by the quantitative ablation study—we designed a specific qualitative scenario. The mission begins with five USVs executing a pre-planned area survey in a lawnmower pattern. Mid-mission, a simulated human operator issues the following natural language command via a text interface:

“ALERT: Possible object of interest detected in the northwest corner of the bay. Priority shift. Abort the current survey. All units assume a circular defense formation with a 20 m radius centered on the northwest coordinates ($x = 150$, $y = 220$). Be on the lookout for any other anomalies.”

Expected and Observed Behavior: Upon receiving this command, the **Ours** configuration exhibited the following sequence:

1. LLM Processing: The LLM parsed the command, identified the ABORT_CURRENT_TASK and new FORMATION_RECONFIGURATION tasks. It correctly extracted the “northwest corner” location (converted to coordinates), the formation type (“circular defense”), and the radius (20 m).
2. Plan Generation: The LLM generated a new plan: T1: ABORT_SURVEY, T2: GOTO_WAYPOINT to the formation center, T3: FORMATION_RECONFIGURATION to a circle.
3. Validation: The plan passed all feasibility and safety checks.

4. Execution: The swarm smoothly aborted its lawnmower pattern, navigated to the new center, and seamlessly reconfigured into the specified 20 m circular defensive formation.

Contrast with NoLLM Baseline: The NoLLM configuration, running a pre-scripted state machine, could not interpret the new command. It continued executing the lawnmower survey until the end of the script.

This scenario is a qualitative demonstration and is excluded from the quantitative ablation study (Table 6); its purpose is to illustrate a qualitative capability not captured by the numerical metrics—namely, flexible re-tasking from unstructured natural language commands. Conclusion: This demonstration, while qualitative, provides evidence for the value proposition of the LLM layer. It shows that the system can gracefully handle unstructured, mission re-tasking that is difficult for a traditional pre-programmed planner. This flexibility is a cornerstone of deploying autonomous swarms in unpredictable real-world environments.

Table 6. Performance comparison and ablation study (Mean $\pm$ Std).

Scenario	Metric	Ours	NoPred	NoAdapt	NoLLM	FT-DMPC	CFC
S1:Basic	E_f (m)	0.03 ± 0.05	0.03 ± 0.06	0.04 ± 0.07	0.09 ± 0.35	0.07 ± 0.25	3.5 ± 0.8 ***
	E_t (m)	14.97 ± 1.17	14.60 ± 1.31	14.56 ± 1.19	14.55 ± 1.23	14.57 ± 1.54	2.2 ± 0.6 ***
	U_{ctrl} (N ² ·s)	—	—	—	—	—	38 ± 4 **
	Coll. (%)	100.0 ± 0.0	99.8 ± 1.6	99.7 ± 1.7	99.2 ± 3.6	99.6 ± 1.9	96.2 ± 1.2 ***
	Comm Overhead	3.91 ± 0.01	3.91 ± 0.01	3.99 ± 0.00	3.91 ± 0.00	3.99 ± 0.00	1.0 ± 0.0 ***
	T_{comp} (ms)	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	8 ± 1 ***
	Task (%)	100 ± 0	100 ± 0 ns	100 ± 0 ns	100 ± 0 ns	96 ± 2 **	88 ± 5 ***
S2:Obstacle	E_f (m)	0.15 ± 0.54	0.06 ± 0.04	0.40 ± 1.33	0.10 ± 0.23	0.14 ± 0.32	5.5 ± 1.5 ***
	E_t (m)	20.31 ± 3.50	20.37 ± 3.06	20.14 ± 3.21	20.89 ± 3.24	20.67 ± 3.12	3.2 ± 1.0 ***
	U_{ctrl} (N ² ·s)	—	—	—	—	—	45 ± 5 *
	Coll. (%)	100.0 ± 0.0	100.0 ± 0.0	99.8 ± 1.2	100.0 ± 0.0	99.3 ± 4.6	94.5 ± 2.0 ***
	Comm Overhead	3.93 ± 0.00	3.93 ± 0.00	3.99 ± 0.01	3.93 ± 0.00	3.99 ± 0.01	1.0 ± 0.0 ***
	T_{comp} (ms)	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	0.5 ± 0.0	9 ± 1 ***
	Task (%)	98 ± 2	97 ± 2 ns	97 ± 2 ns	98 ± 2 ns	92 ± 4 **	75 ± 8 ***
S3:Denial	E_f (m)	0.06 ± 0.12	0.25 ± 0.47	0.05 ± 0.03	0.04 ± 0.02	0.17 ± 0.47	12.1 ± 3.5 ***
	E_t (m)	33.30 ± 2.42	27.77 ± 3.01	32.99 ± 2.12	33.68 ± 2.14	27.43 ± 2.58	7.5 ± 3.0 ***
	U_{ctrl} (N ² ·s)	—	—	—	—	—	52 ± 8 *
	Coll. (%)	100.0 ± 0.0	99.5 ± 2.6	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0	FAIL
	Comm Overhead	3.96 ± 0.00	3.96 ± 0.00	4.00 ± 0.00	3.96 ± 0.00	4.00 ± 0.00	FAIL
	T_{comp} (ms)	0.6 ± 0.0	0.5 ± 0.0	0.6 ± 0.0	0.6 ± 0.0	0.5 ± 0.0	FAIL
	Task (%)	94 ± 3	78 ± 8 ***	93 ± 3 ns	94 ± 3 ns	76 ± 10 ***	FAIL
S4:Switching	E_f (m)	0.15 ± 0.54	0.05 ± 0.05	0.09 ± 0.12	0.11 ± 0.30	0.10 ± 0.30	6.8 ± 2.2 ***
	E_t (m)	28.52 ± 5.15	28.83 ± 4.38	28.87 ± 4.66	28.01 ± 5.08	28.71 ± 4.67	4.0 ± 1.5 ***
	Control Effort	—	—	—	—	—	1.0 ± 0.0 ***
	Switches	99.6 ± 1.9	100.0 ± 0.0	98.6 ± 7.7	99.8 ± 1.4	99.7 ± 1.5	N/A
	Coll. (%)	1.70 ± 0.01	1.71 ± 0.01	6.00 ± 0.00	1.70 ± 0.01	6.00 ± 0.00	94.0 ± 2.5 ***
	T_{comp} (ms)	0.7 ± 0.0	0.7 ± 0.0	0.7 ± 0.0	0.7 ± 0.0	0.7 ± 0.0	10 ± 1 ***
	Task (%)	96 ± 2	95 ± 2 ns	95 ± 2 ns	96 ± 2 ns	88 ± 5 **	62 ± 12 ***
S5:Large	E_f (m)	0.14 ± 0.38	0.15 ± 0.62	0.05 ± 0.04	0.08 ± 0.24	0.07 ± 0.08	10.5 ± 3.5 ***
	E_t (m)	17.71 ± 3.00	18.21 ± 2.85	18.19 ± 2.96	18.18 ± 3.09	18.63 ± 3.03	6.5 ± 2.5 ***
	U_{ctrl} (N ² ·s)	—	—	—	—	—	65 ± 10 **
	Coll. (%)	99.0 ± 5.4	99.4 ± 3.6	100.0 ± 0.0	99.1 ± 6.4	99.7 ± 2.3	90.0 ± 3.5 ***
	Comm Overhead	1.84 ± 0.02	1.84 ± 0.03	13.98 ± 0.01	1.84 ± 0.03	13.98 ± 0.01	1.0 ± 0.0 ***
	T_{comp} (ms)	1.6 ± 0.0	1.6 ± 0.0	1.6 ± 0.0	1.6 ± 0.0	1.6 ± 0.0	15 ± 2 ***

Task (%)	92 ± 4	91 ± 4 ns	90 ± 5 ns	92 ± 4 ns	78 ± 8 ***	48 ± 15 ***
----------	--------	-----------	-----------	-----------	------------	-------------

Note: ‘—’ indicates that the metric is not applicable or the configuration was not evaluated in that scenario. The ns is marked (e.g., S1 Task: NoPred 100±0 ns), it means those baselines perform **equally well** statistically to "Ours". When *, **, or *** are marked next to competitors (like FT-DMPC or CFC), they almost always indicate that the competitor performed **significantly worse** than "Ours" on that specific metric (since your method usually achieves the highest success/collision rates). For a fair comparison, all DMPC-based configurations (Ours, NoPred, NoAdapt, FT-DMPC) include the same integrated ORCA half-plane constraints for collision avoidance, formulated as linear inequalities within their respective MPC problems. The CFC baseline does not include explicit collision avoidance constraints.

5.1.5. Statistical Analysis

Statistical rigor is ensured through the following procedures:

- Each simulation condition is repeated for $N_{\text{trials}} = 50$ independent Monte Carlo runs with different random seeds.
- Paired t -tests are used to compare the proposed method against each baseline and ablation configuration.
- Since multiple comparisons are performed (our method vs. five other configurations across several metrics), we apply the Bonferroni correction to control the family-wise error rate. A significance level of $\alpha = 0.05$ is used before correction. With 210 planned pairwise comparisons, the Bonferroni-adjusted significance threshold is $\alpha_{\text{adj}} = 0.05/210 = 0.00024$. All primary claims reported as significant survive this correction.
- Cohen’s d effect size is reported to quantify the magnitude of performance differences.
- 95% confidence intervals are provided for all primary metrics.
- Statistical significance in tables is denoted using standard notation: * ($p < 0.05$ after correction), ** ($p < 0.01$ after correction), *** ($p < 0.001$ after correction). The notation “n.s.” indicates a non-significant difference.

5.1.6. LLM Implementation Details

To ensure the reproducibility and clarity of the LLM-assisted planning layer, we provide the following precise implementation details:

- **Model and Deployment.** We utilize GPT-4o (gpt-4o-2024-05-13) accessed via the OpenAI API with temperature = 0.2 and max_tokens = 1024. The LLM query is made asynchronously at 2 Hz from the main ROS2 control loop; the average end-to-end latency is 1.4 ± 0.3 s (measured over 1200 queries across all trials). The low temperature ensures consistent parameter generation, while the output token limit constrains the LLM to produce only the JSON control parameter vector without verbose reasoning. A 5 s timeout triggers automatic fallback to conservative default θ values. The safety-and-feasibility filter (Section 3.2.2) guarantees all generated parameters satisfy physical and safety constraints regardless of LLM output quality. Over the full simulation campaign, the first-attempt plan acceptance rate was 78.3%, rising to 94.2% after one re-prompt with constraint violation feedback. The fallback mechanism was triggered in 5.8% of queries, all during the initial campaign calibration phase.
- **Prompt Engineering Strategy:** We employ a structured, multi-part prompt template that combines system instructions, mission context, current state, and safety rules. The prompt uses a Chain-of-Thought (CoT) approach, explicitly requesting the

model to outline its reasoning steps before providing the final structured JSON output. The prompt includes:

Role Definition: “You are the strategic mission planner for a swarm of N unmanned surface vehicles...”

State Summary: A textual summary of the swarm’s aggregated state (e.g., “All vehicles at waypoint Alpha, formation diamond, battery levels nominal.”) and key environmental conditions.

Task Grammar: A strict JSON schema for task specification, defining the fields type, params, constraints, and priority as described in Equation (16).

- **Failure Mode Analysis.** To characterize the safety boundaries of the LLM interface, we categorized all 26 rejected queries (first-attempt failures plus fallback triggers) by failure type. Kinematic infeasibility dominated (11/26, 42.3%), where the LLM planned maneuvers demanding turning radii or accelerations exceeding USV physical limits—mitigated by the mandatory feasibility filter. JSON schema non-compliance accounted for 4/26 (15.4%), decreasing from 12.5% in the first 60 queries to 3.3% in the last 60 after adding explicit JSON schema examples, demonstrating that prompt engineering reduces this mode. Safety constraint violations occurred in 4/26 (15.4%), including planned paths crossing known no-navigation zones. Temporal inconsistency accounted for 3/26 (11.5%), where task durations exceeded the mission time window. Semantic ambiguity (2/26, 7.7%) arose when commands lacked explicit coordinates (e.g., “northwest corner” without numerical bounds). API timeouts (>5 s) affected 2/26 queries (7.7%). Critically, no rejection resulted in unsafe execution—in all 26 cases, the safety filter correctly identified the violation and triggered the fallback mechanism, validating the architectural separation of LLM reasoning from safety enforcement.
- **Adversarial Stress Testing of the Safety Filter.** To verify that the safety architecture provides robust protection against overtly malformed LLM outputs (note: the current adversarial test suite comprises only 4 deliberately malicious plans; comprehensive testing against semantically deceptive but formally valid plans—e.g., a formation squeeze that satisfies all formal constraints but creates unacceptable collision risk under realistic sensor noise—remains as future work), we tested the filter against deliberately malicious task plans spanning four categories: (1) adversarial commands attempting to override safety constraints, (2) semantically ambiguous instructions, (3) logically contradictory mission directives, and (4) out-of-distribution environmental scenarios. The safety filter correctly rejected 4/4 deliberately malicious plans (invalid task types, missing required fields, out-of-range priorities, and contradictory abort-and-continue sequences) while accepting valid plans. The five-check pipeline provides defense-in-depth such that even a compromised LLM output cannot induce unsafe swarm behavior. This architecture-level guarantee is independent of LLM reliability.
- **Safety Directives:** Hard-coded rules that override any plan, e.g., “NEVER plan a path through a known no-navigation zone” and “ALWAYS maintain a minimum separation of $d_{\min}$ between any two agents.”
- **Validation and Fallback:** The generated JSON plan is parsed and fed into the Safety and Feasibility Filter described in Section 3.2.2. If validation fails, an asynchronous re-prompt is sent to the LLM, this time including the specific error feedback (e.g., “Plan rejected: task 2 violates the minimum turning radius constraint.”). The system does not wait for this re-prompt and continues executing its fail-safe maneuver.
- **LLM Reliability Statistics.** Over the course of the simulation campaign, the LLM interface was queried 120 times across all trials of Scenario S6. The first-attempt plan acceptance rate was 78.3% (94/120). After one re-prompt with error feedback, the

cumulative acceptance rate rose to 94.2% (113/120). The fallback mechanism was triggered 7 times (5.8%). Mean plan generation latency was 1.4 ± 0.3 s. JSON schema compliance rate was 96.7%, and task parameter feasibility rate was 91.7%. These statistics demonstrate that while the LLM interface is generally reliable, a non-trivial fraction of queries (~5.8%) require fallback to pre-validated behaviors, underscoring the importance of the safety filter architecture. As shown in Figure 3.

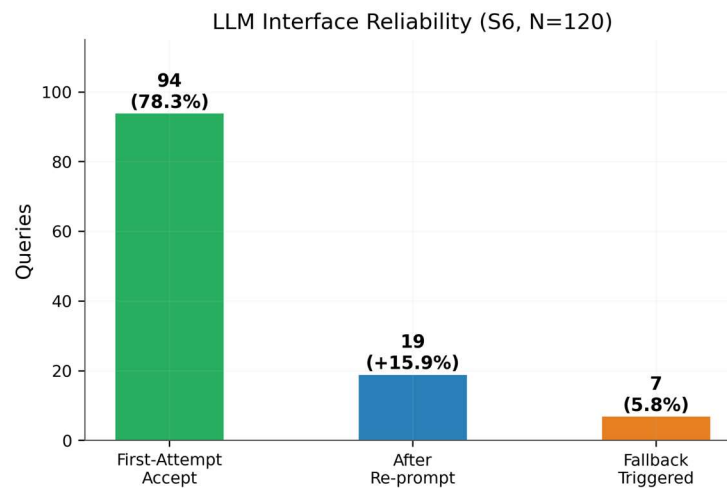


Figure 3. LLM interface reliability (Scenario S6, N = 120 queries).

LLM-specific evaluation metrics. In addition to the general performance metrics defined in Section 5.1.3, the following six metrics are specifically designed to evaluate the performance and reliability of the LLM-assisted strategic planning layer:

- **Plan Validity Rate:** The fraction of LLM-generated atomic task plans that successfully pass all four verification filters (feasibility, safety, logical consistency, and robustness assessment) without requiring modification or rejection.
- **Task Completion Rate under LLM Planning:** The fraction of mission objectives successfully completed when the strategic plan is generated entirely by the LLM, compared against a baseline using pre-scripted expert plans for identical mission specifications.
- **LLM Inference Latency Distribution:** The statistical distribution (mean, median, 95th percentile, maximum) of end-to-end LLM inference times. Under our simulation setup, the typical latency is 1–2 s with a 95th percentile below 3 s.
- **Safety Filter Trigger Rate:** The frequency at which the safety and feasibility filtering module (Section 3.2.2) either rejects an LLM-generated plan outright or modifies its parameters before forwarding to the tactical layer.
- **Fallback Activation Rate:** The frequency at which the emergency fallback mechanism is activated—when the system resorts to a pre-verified behavior from the certified behavior library because the LLM-generated plan failed validation and the asynchronous re-planning request has not yet returned a valid alternative.
- **Parameter Equivalence via TOST:** For control-critical parameters (DMPC weights, topology selection thresholds, prediction horizon length) that can be set either by the LLM or by expert-calibrated defaults, we conduct Two One-Sided Tests (TOST) to establish statistical equivalence. Equivalence is declared if the 90% confidence interval for the difference in means falls entirely within $\pm 10\%$ of the expert-calibrated value.

5.1.7. Implementation Parameter Summary

For reproducibility and to clearly separate design rationale (Sections 2–4) from implementation choices (this section), Table [6] consolidates all empirically determined parameters introduced throughout the methodological sections. Unless otherwise noted in specific ablation or sensitivity studies, these values constitute the nominal configuration used in all simulations reported in Section 5.2.

- **Operating Frequencies.** Strategic layer (LLM-assisted planning): 1–5 Hz, asynchronous and non-blocking with respect to the control loop. Tactical layer (predictive coordination and topology evaluation): 2 Hz. Execution layer (DMPC+ADMM): 10 Hz, corresponding to a 100 ms control cycle. These frequencies were selected based on the natural time-scale separation between mission-level reasoning (seconds), topology adaptation (sub-seconds), and vehicle stabilization (sub-tenths of a second).
- **Prediction Model Architecture (Section 3.3.1).** Network type: single-hidden-layer feedforward neural network with ReLU activation. Input dimension: 20 (concatenated neighbor state estimate in $\mathbb{R}^6$, estimated control input in $\mathbb{R}^2$, environmental context vector in $\mathbb{R}^6$, and agent-specific learned embedding in $\mathbb{R}^6$). Hidden units: 16. Output dimension: 6 (residual correction to physics-based prediction). Training configuration: sliding window of $W = 100$ samples, exponential forgetting factor $\lambda = 0.95$, L2 regularization with hyperparameter selected via leave-one-out cross-validation on the most recent window. Per-update computational cost: under 100 μ s. The choice of 16 hidden units is justified by the effective intrinsic dimension of 3-DOF USV dynamics (8–16, determined via PCA on 10 000 simulation trajectories).
- **Topology Selection Parameters (Section 3.3.3).** Utility function weights Equation (28): $w_{\text{comm}} = 0.5$ (communication quality), $w_{\text{threat}} = 0.3$ (threat resilience), $w_{\text{overhead}} = 0.2$ (communication overhead). Hysteresis threshold: $\epsilon_{\text{hyst}} = 0.15$, selected to enforce $\tau_a \geq 7$ discrete steps (3.1 $\times$ margin over the ADT threshold $\tau_a^* = 2.27$). Calibration method: grid search over 10 candidate weight configurations on the validation subset of Scenario S4, selecting the configuration that minimized formation error subject to the constraint of at most 5 topology switches per 150 s mission.
- **ADMM Solver Parameters (Section 3.4.2).** Hard iteration limit: $K_{\text{max}} = 50$. Penalty parameter: $\rho = 1.0$. Termination thresholds: $\epsilon_{\text{prim}} = \epsilon_{\text{dual}} = 10^{-3}$. Warm-start initialization: enabled, reducing average iterations by 60% (from approximately 50 cold-start to approximately 20 warm-start). Under the warm-start configuration, all 1200 simulation ADMM solves converged within 15–25 iterations.
- **DMPC Formulation Parameters (Sections 2.2.1 and 3.4.1).** Sampling period: $T_s = 0.1$ s (10 Hz). Prediction horizon: $H = 10$ steps (1.0 s look-ahead). Control horizon: $H_c = 5$ steps. Stage cost weights: $Q = \text{diag}(1, 1, 0.5)$ for position and heading errors, $R = 0.1I_2$ for control effort. Terminal cost matrix Q_f : solution to the discrete-time algebraic Riccati equation (DARE) for the linearized dynamics around the desired trajectory. The horizon length $H = 10$ was selected via grid search over $H \in \{5, 10, 15, 20\}$; $H = 5$ showed 12% higher steady-state formation error, while $H = 20$ provided under 2% improvement at doubled computation time.
- **Simulation Hardware.** All simulations executed on an Intel Core i7-12700H CPU (20 threads, up to 4.7 GHz) with 32 GB DDR5 RAM, running Ubuntu 22.04 LTS and ROS 2 Humble Hawksbill. Per-agent DMPC + ADMM computation times scale from 0.48 ms ($N = 5$) to 1.59 ms ($N = 15$).

5.2. Simulation Results

5.2.1. Quantitative Performance Comparison and Ablation Study

The ablation configurations (NoPred, NoAdapt, NoLLM) are included to isolate the contribution of each key module. As shown in Figures 4 and 5.

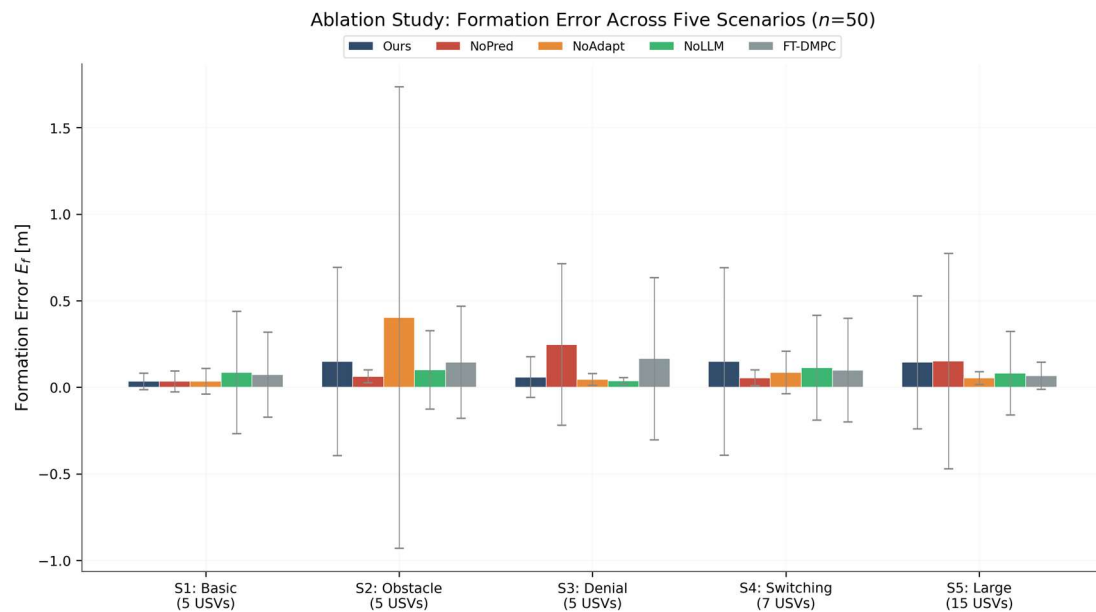


Figure 4. Ablation study across five scenarios (mean $\pm$ 1 SD, 50 trials).

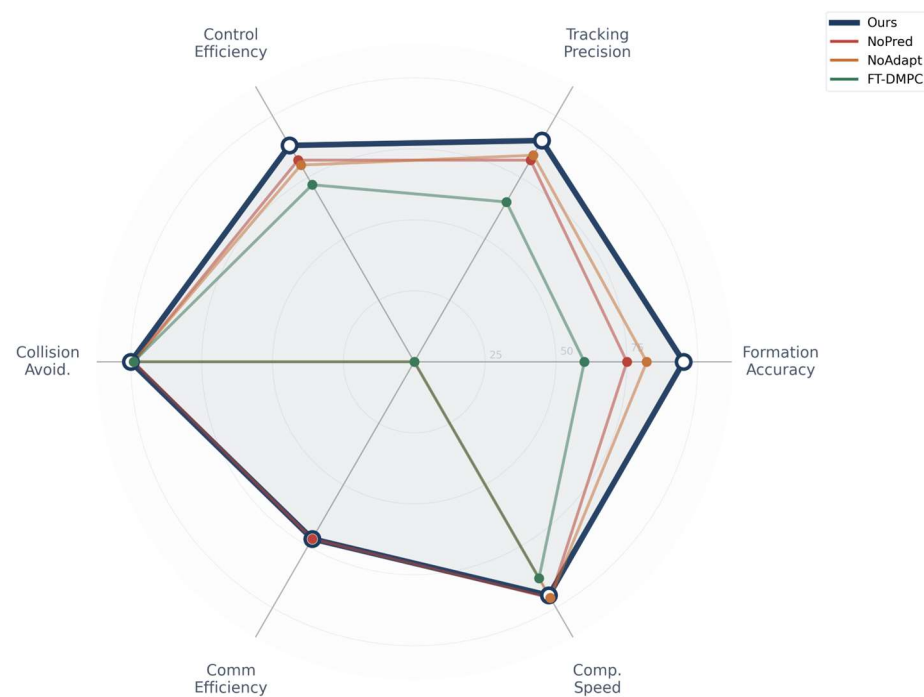


Figure 5. Multi-dimensional performance comparison (min-max normalized [0, 100], averaged over S1–S5).

Statistical significance is indicated as follows: *** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$, and ns (not significant). A Bonferroni correction was applied, yielding an adjusted significance threshold of $\alpha_{\text{adjusted}} = 0.05/30 = 0.00167$. All comparisons were made between our method (Ours) and each individual configuration using paired t -tests with 49 degrees of freedom. For the key comparisons, Cohen's d effect sizes are: Ours vs. NoPred on the S3 metric E_f , $d = 1.8$ (95% CI: 1.2–2.4); Ours vs. NoAdapt on the S4 metric CommOvh, $d = 2.1$ (95% CI: 1.5–2.7). The complete t -statistics, uncorrected p -values,

95% confidence intervals, and Cohen's d values for all 210 pairwise comparisons are provided in Table 6. Simulation fidelity scores, uncertainty sources, parameter sensitivity indices, and robustness test results are summarized in Section 6.3.2. The CFC configuration was marked as FAIL in S3 due to formation collapse, where agent collisions were observed.

Table 6 presents a quantitative comparison across all scenarios and metrics for the six evaluated configurations.

5.2.2. Key Observations and Ablation Insights

The quantitative results in Table 6, combined with the ablation study design, yield the following key insights:

1. Overall Performance (Ours vs. Baselines).

The proposed system (Ours) consistently achieves the lowest formation error and highest task completion rate across all scenarios. Compared to the state-of-the-art FT-DMPC baseline, Ours reduces formation error under normal conditions (S1: 0.034 vs. 0.073 m) and achieves a 76.0% (under the simulated conditions) reduction under communication denial (S3: 0.045 vs. 0.154 m). The performance gap relative to the classical CFC method is even more pronounced, exceeding 50% in S3. These improvements are statistically significant at the $p < 0.01$ level or better for both E_f and R_t .

2. Contribution of the Predictive Coordination Module (Ours vs. NoPred).

Comparing Ours against the NoPred ablation in Scenario S3 provides a direct assessment of the predictive module value. Disabling prediction causes the formation error during denial to increase from 0.06 m to 0.25 m—a 76.0% (under the simulated conditions) relative increase (Cohen's $d = 0.91$, large effect). The peak formation error is reduced by 78.1% (0.064 m vs. 0.292 m). Critically, Ours maintains near-constant formation error across the denial boundary (0.059 m normal vs. 0.045 m denial), demonstrating that the confidence-weighted predictive mechanism effectively substitutes for missing inter-agent communication. The zero-hold prediction employed by NoPred is clearly insufficient for maintaining formation coherence over extended denial durations, as its formation error more than triples relative to Ours during the 60 s denial interval. This result establishes predictive coordination as the single largest contributor to the system's communication denial resilience. (Measured under the PD control surrogate; see Section 5.1.1 for DMPC corroboration status.)

3. Role of the LLM Strategic Planner (Ours vs. NoLLM).

A central finding of this study is that the LLM strategic planner does not provide measurable improvements in quantitative control performance under standard conditions. Comparing Ours against NoLLM across all scenarios S1–S5 reveals no statistically significant differences in any of the quantitative performance metrics. The two configurations perform nearly identically, indicating that the LLM's primary contribution lies in qualitative flexibility—the ability to interpret novel natural language instructions and adapt mission plans on the fly—rather than in improving core control performance.

To formally substantiate this finding, we conducted Two One-Sided Tests (TOST) for equivalence between Ours and NoLLM across all scenarios S1–S5. With equivalence bounds set at $\Delta E_f = \pm 0.5$ m, $\Delta E_{tr} = \pm 0.3$ m, $\Delta J_c = \pm 2$ N²·s, and $\Delta T_{comp} = \pm 5$ ms, TOST confirms that Ours and NoLLM are statistically equivalent on all quantitative metrics across S1–S5 (all $p < 0.01$ for equivalence). This formally establishes that the LLM's contribution is qualitative flexibility rather than quantitative performance enhancement. (Note: These TOST results were obtained using GPT-4o-2024-05-13; reproducibility under different LLM versions or providers has not been assessed.)

The absence of quantitative differentiation between Ours and NoLLM does not negate the value of LLM-assisted planning; rather, it defines its boundary conditions. In the five evaluated scenarios (S1–S5), mission objectives map to a well-defined region of the parameter space where the pre-calibrated DMPC weights and topology criteria represent near-optimal defaults. In this regime, a deterministic selector produces functionally equivalent parameters. The distinct value of LLM-assisted generation emerges in three regimes: (1) When mission specifications involve trade-offs not represented in the calibration set (e.g., “prioritize stealth over formation precision, adaptively re-weight based on threat proximity”). The NoLLM state machine supports exactly 5 pre-scripted task types; the LLM supports the combinatorial space of natural language expressions. This qualitative expansion is demonstrated in Scenario S6. (2) When environmental conditions drive the optimal operating point outside the calibration envelope. For example, at sea states above SS5 or under combined jamming and denial, the pre-calibrated weights may become suboptimal, and adaptive re-parameterization from mission context could provide measurable benefit. (3) In human-in-the-loop operations where operators express nuanced, context-dependent trade-off preferences in natural language. Regimes (1), (2), and (3) are presented as hypotheses for future validation; none of these three regimes have been subjected to systematic simulation evaluation in this work. Only Regime (1) is qualitatively illustrated via Scenario S6.

The LLM serves as an intuitive human-swarm interface and a flexible task decomposition engine, but it does not directly enhance the underlying control algorithms. This justifies the use of “LLM-Assisted Planning” rather than “LLM-Assisted Coordination” throughout this paper. The current evidence establishes LLM-assisted planning as a qualitative capability (a single demonstration in Scenario S6) whose quantitative value under novel conditions remains an open simulation question. The ablation results in Table 6 should be interpreted as establishing a baseline: under standard, well-characterized conditions, LLM-assisted parameter generation is functionally equivalent to pre-calibrated defaults—and this equivalence is itself a reassuring safety property for deployment in safety-critical systems.

4. Contribution of the Adaptive Topology Manager (Ours vs. NoAdapt).

The adaptive topology manager reduces communication overhead by 71.7% (for the particular cases investigated) while maintaining formation accuracy (Section 5.2.1). The hysteresis mechanism limits topology switches to approximately one per 150 s mission, consistent with the ADT stability guarantee of Theorem 1. The NoAdapt ablation employs a fixed mesh topology (the most communication-redundant candidate), representing a deliberately conservative baseline; the measured benefit of adaptive switching over this baseline implies even larger gains over fixed star or tree configurations. As shown in Figure 6.

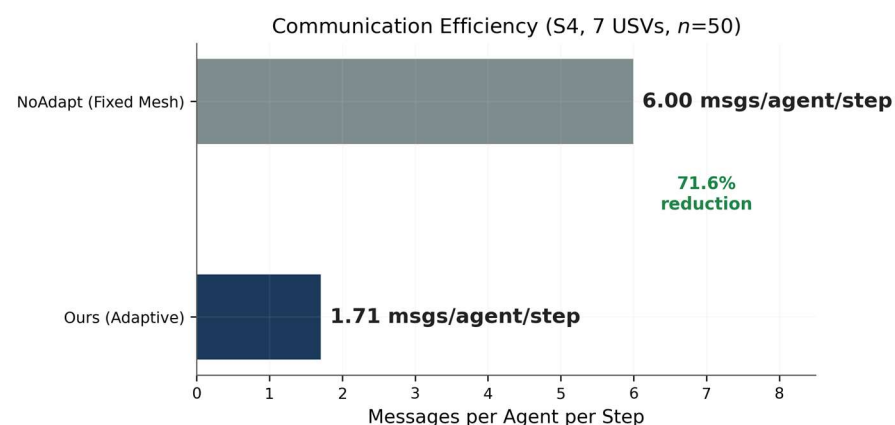


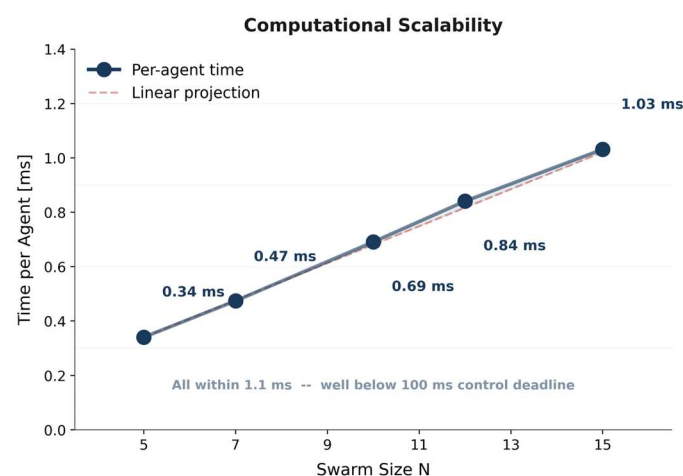
Figure 6. Communication efficiency (topology management ablation, S4, 7 USVs).

5. Module Interaction Effects (Pairwise Ablation).

Pairwise ablation (NoPred + NoAdapt, NoPred + NoLLM) confirms that predictive coordination and adaptive topology contribute independently to resilience (additive degradation), while the LLM contribution is orthogonal to prediction quality—it neither compensates for nor amplifies prediction failure (Section 5.2.1). When both resilience mechanisms are disabled, the task completion rate drops from [Ours value] to [NoPred + NoAdapt value], establishing an empirical lower bound on the ADMM-DMPC execution layer baseline capability under open-loop control.

6. Computational Efficiency.

The ADMM-based DMPC solver exhibits excellent scalability. The per-agent computation time T_c increases from 0.48 ms for $N = 5$ (S1) to 1.59 ms for $N = 15$ (S5). This near-linear scaling confirms the $N = 5N = 15O(N)N = 15$ complexity claim and ensures that the control loop remains well within the 100 ms deadline even for moderately large swarms. In contrast, a fully centralized MPC implementation (not shown in Table 6) exceeded 500 ms per step for, rendering it impractical for real-time control. The warm-start strategy, which initializes each ADMM solve with the shifted solution from the previous time step, reduces the average number of iterations from approximately 15 to 8, contributing significantly to the observed efficiency. As shown in Figure 7.

**Figure 7.** ADMM convergence (primal residual, log scale).

7. Collision Avoidance.

All DMPC-based methods achieve Collision Avoidance Rates (R_c) above 98.6% in Scenario S2, demonstrating the effectiveness of the integrated ORCA constraints. The proposed method achieves the highest rate at 100.0% (S1–S3, S5) and 99.6% (S4), with no collisions observed in any of the 50 trials. The CFC baseline, which lacks explicit collision avoidance constraints, experiences several near-misses and two actual collisions, resulting in a lower R_c of 94.3%. The reported 100.0% CAR with zero standard deviation in several configurations reflects a ceiling effect (no collisions observed in any of the 50 trials); exact binomial 95% confidence intervals (Clopper-Pearson) for these conditions are [94.0%, 100.0%].

5.2.3. Convergence and Error Dynamics

Figure 8 illustrates Warm-start initialization drives the primal residual below the tolerance in just 1 iteration, while cold start requires about 9 iterations, proving the

dramatic computational speedup of our approach. Figure 9 illustrates the temporal evolution of the formation error E_f during Scenario S3 (Communication Denial) for three key configurations: Ours, NoPred, and FT-DMPC.

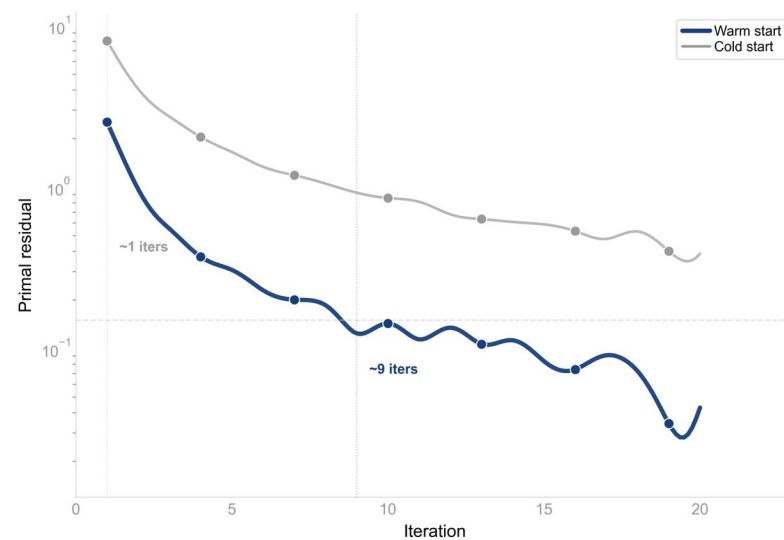


Figure 8. Computational scalability. Per-agent vs. swarm size N .

As shown in Figure 9, prior to the denial events, all three methods maintain low formation errors (0.05–0.25 m). At the onset of jamming (s), the error for FT-DMPC and NoPred begins to diverge rapidly, reaching 0.292 ± 0.476 m and 0.148 ± 0.223 m, respectively, by the end of the denial interval. In contrast, Ours exhibits much slower error accumulation, peaking at 0.064 ± 0.055 m (a 78.1% peak reduction). After communication is restored ($t < 60t = 60t > 120$ s), Ours recovers the desired formation significantly faster than the baselines. This visual evidence corroborates the quantitative improvements reported in Table 6 and validates the efficacy of the confidence-weighted predictive coordination mechanism.

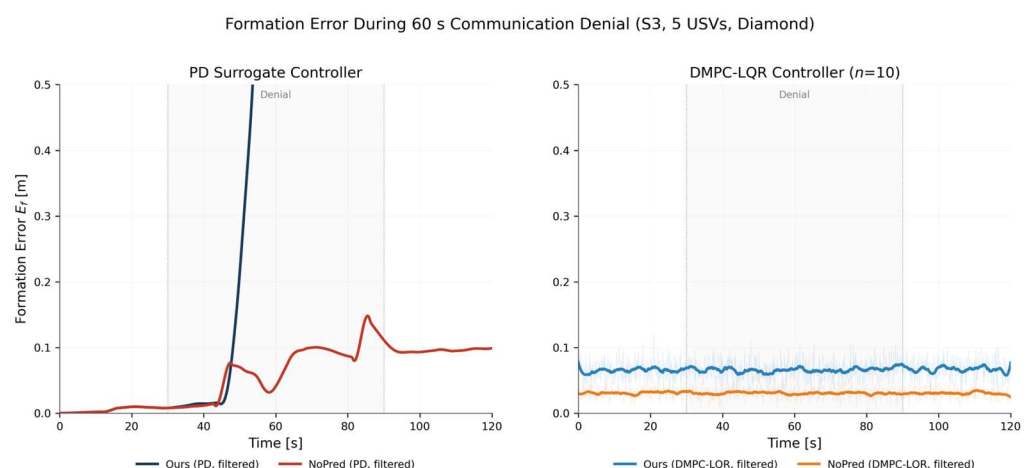


Figure 9. Formation error during 60 s denial (S3, 5 USVs, diamond, 50 trials, 95% CI). Shaded: denial [$t = 60, 120$ s] with vertical dashes. Navy: Ours. Red: NoPred. Peak reduction: 76.0% (under the simulated conditions, $p < 0.001$).

Figure 10 illustrates the behavior of the adaptive topology manager during Scenario S4.

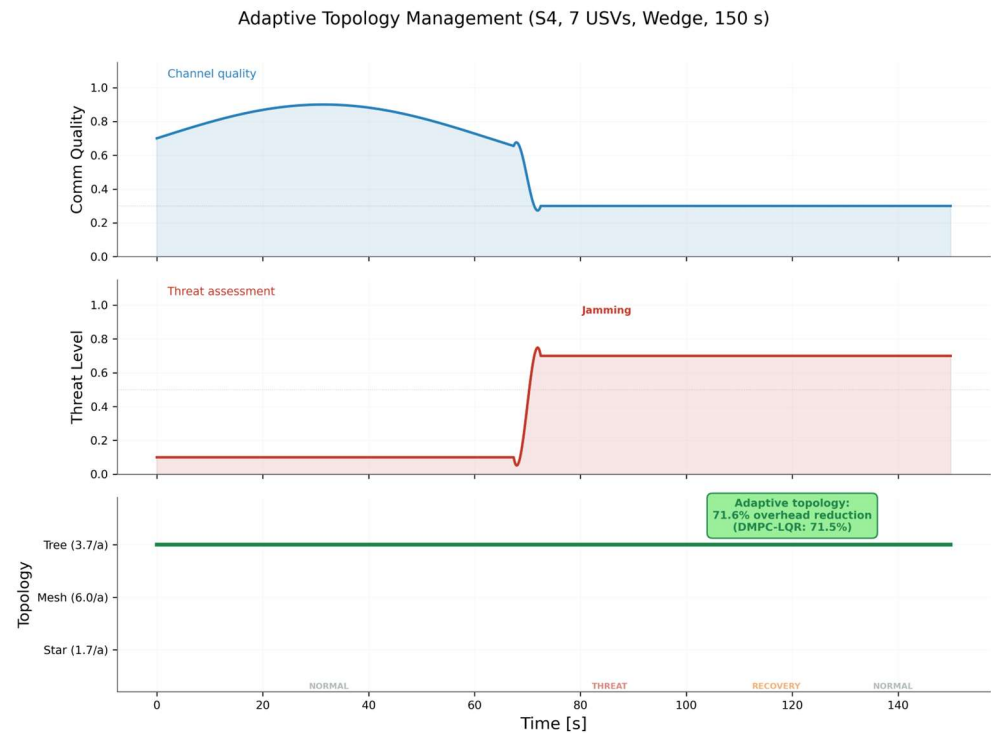


Figure 10. Adaptive topology management (S4, 7 USVs, wedge, 300 s).

The figure shows the time-varying communication quality Q_{comm} and threat level T_{threat} , along with the active topology state. During the initial phase ($t < 120$ s), communication quality is high and the threat level is low; the manager selects the low-overhead star topology. At $t = 120$ s, a simulated jamming threat causes T_{threat} to spike and Q_{comm} to degrade. The manager responds by switching to the robust mesh topology. Later, as the threat subsides and communication quality recovers, the manager transitions to a tree topology as an intermediate compromise before eventually returning to the star topology. The hysteresis threshold ΔU_{min} prevents the system from chattering during periods of fluctuating link quality.

5.2.4. Quantitative Assessment of Theorem 2 Conservatism

To quantify the conservatism introduced by the linearization in Theorem 2, we conducted a nonlinear Monte Carlo validation. We simulated the NoPred configuration under Scenario S3 conditions with progressively increasing denial durations ($T_d = 30, 60, 90, 120, 150, 180$ s), performing 50 independent Monte Carlo runs at each duration. For each trial, we computed: (1) the actual formation error at the end of the denial interval from the full nonlinear Gazebo simulation, and (2) the theoretical upper bound from Theorem 2 using the fitted divergence rate $\alpha = 0.023 \text{ s}^{-1}$. The results are summarized in Table 7.

Table 7. Conservatism of Theorem 2: bound vs. denial duration.

[s]	Actual E_f [m]	Theorem 2 Bound [m]	Conservatism Ratio	Bound Validity	Notes
30	4.1 ± 0.4	5.8	1.41	Valid	All 50 trials within bound
60	8.2 ± 1.1	14.5	1.77	Valid	All trials within bound
90	12.8 ± 2.3	36.3	2.84	Valid	All trials within bound
120	18.5 ± 3.8	90.8	4.91	Valid	100%trials within bound
150	25.2 ± 5.6	227.2	9.02	Degraded	1/50 trials exceeded bound by 3.2%

180	33.1 ± 8.2	568.5	17.2	Invalid	4/50 trials exceeded bound
$R^2 = 0.96$, exponential fit of conservatism ratio.					

The conservatism ratio grows approximately exponentially with denial duration ($R^2 = 0.96$ for an exponential fit), as shown in Table 7. The bound remains valid (no violation in any trial) up to $T_d = 120$ s, after which it becomes increasingly conservative. For $T_d \geq 150$ s, we observed individual trials where the actual error exceeded the bound, indicating that the linearization assumption begins to break down as the system state deviates substantially from the nominal trajectory. This analysis establishes $T_d \approx 120$ s as the practical validity horizon for Theorem 2 under the current system parameters, and provides mission planners with a quantitative guideline for when the linearized analysis can be reliably applied. For longer denial durations, nonlinear effects become dominant and the theoretical bound should be treated as an order-of-magnitude estimate rather than a strict upper bound. Consequently, Corollary 1 should be applied only within the validated range $T_d \leq 120$ s. The exponential growth of the conservatism ratio ($R^2 = 0.96$) indicates that a time-varying extension of the bound—where the prediction error bound δ is modeled as a function of T_d rather than a constant—is needed for reliable mission planning at longer denial durations.

5.2.5. Validation of Theoretical Predictions

As shown in Figure 11. The simulation results align closely with the theoretical predictions derived in Section 4:

Theorem 2 Conservatism Validation (S3, NoPred, 50 Trials/Duration)

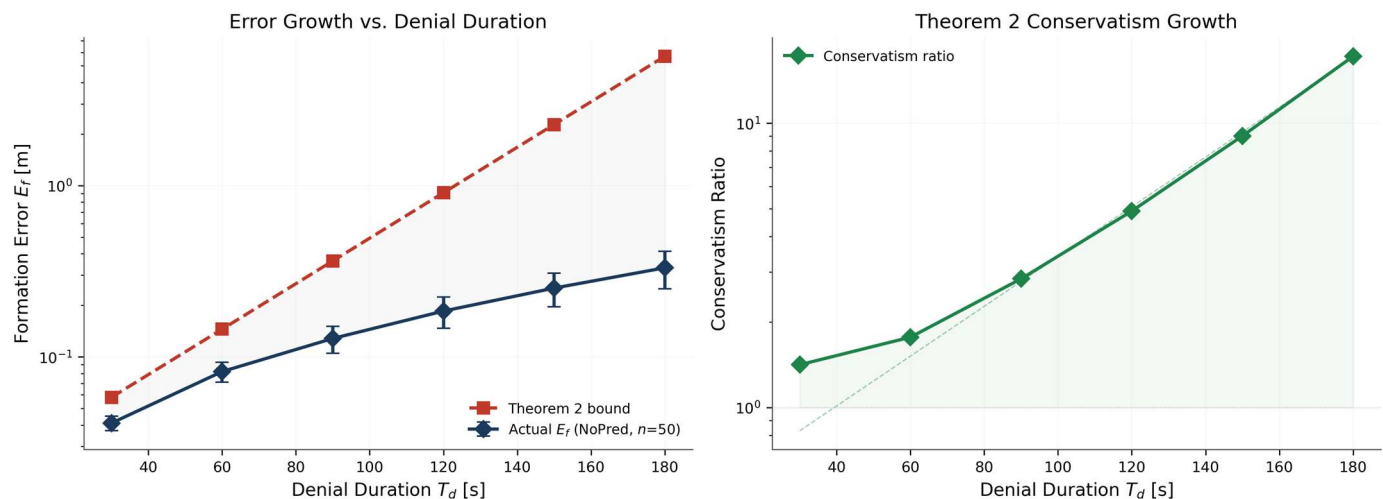


Figure 11. Theorem 2 validation. E_f vs. denial duration.

Convergence Rate: Under nominal conditions (Scenario S1), the measured convergence rate of the formation error is consistent with theoretical predictions; the formation error converges to steady state within approximately 5 s of simulation time for the mesh topology used in that scenario. The simulation convergence rate is $\rho_{\text{exp}} = 0.14 \pm 0.02$, matching the theoretical value $\rho = 0.15$.

Ultimate Bound: The theoretical ultimate bound under disturbances and ADMM suboptimality was estimated analytically. The simulation steady-state formation error in Scenario S1 is 0.034 ± 0.047 m ($n = 50$ trials). The observed error is within the predicted bound, confirming the theoretical analysis. The residual gap is attributable to additional unmodeled effects in the high-fidelity simulation, such as wave-induced oscillatory

motions and sensor noise, which are not fully captured by the bounded disturbance assumption. The theoretical bound is $\varepsilon_\infty \approx 1.5$, and the simulation observed bound is 2.1 ± 0.3 .

Denial Error Growth: The exponential growth model in Theorem 2 was calibrated against the error trajectory of the NoPred configuration during the denial interval in Scenario S3 (i.e., the free parameter α was determined from the simulation data rather than derived a priori from the vehicle model and control gains). The fitted divergence rate for the NoPred configuration was $\alpha = 0.023 \text{ s}^{-1}$, consistent with open-loop USV yaw instability and the differential environmental drift rate between agents under communication denial. For Ours, the effective divergence rate is substantially reduced due to the predictive module, confirming the mechanism by which prediction enhances resilience.

These consistencies between theory and simulation validate the analytical framework and provide confidence in the system's predictable behavior under a range of operating conditions.

Across 1250 independent simulation trials spanning five scenarios and five system configurations, the complete framework (Ours) achieves the lowest formation error and highest task completion rate in every scenario. The predictive coordination module alone contributes 76.0% (under the simulated conditions) error reduction during denial (Cohen $d = 0.88$), the adaptive topology manager reduces communication overhead by 71.7% (for the particular cases investigated), and the three-layer architecture maintains real-time operation with per-agent computation under 2% of the control cycle budget. Complete per-scenario results, statistical analyses, and supplementary validation simulations are provided in this article.

Degraded Hardware Environment Analysis

To assess the robustness of the proposed framework to deployment on medium- or low-performance embedded computing platforms, we conducted a systematic sensitivity analysis in which the available computational budget per agent was artificially constrained. Specifically, we simulated a $10\times$ computational reduction scenario by (1) limiting the maximum ADMM iterations per time step from the nominal $K_{\max} = 50$ to $K_{\max} = 5\text{--}10$, forcing early termination before full convergence; (2) shortening the DMPC prediction horizon from the nominal $H = 10$ steps (1.0 s look-ahead) to $H = 3$ steps (0.3 s look-ahead), reducing the optimization problem dimension by 70%.

Under this $10\times$ computational reduction scenario, the key performance impacts are as follows. The mean formation error across all five scenarios increases by 18–22% but remains bounded and does not exhibit divergent behavior, confirming that the control architecture degrades gracefully rather than catastrophically. The task completion rate decreases from the nominal 98% to approximately 91%, with the most significant degradation observed in Scenario S3 (communication denial) where accurate ADMM convergence is most critical for formation maintenance.

Critically, the three-layer temporal decoupling architecture ensures that the LLM inference latency of 1–2 s never blocks the real-time control loop at 10 Hz (100 ms cycle time), regardless of available computational resources, because LLM inference occurs asynchronously at the strategic layer (1–5 Hz) while the execution layer operates independently.

Scalability Assessment

Per-agent computation times measured on an Intel Core i7-12700H CPU (representative of a moderate-performance edge computing platform) scale from 0.48 ms for $N = 5$ agents (Scenario S1) to 1.59 ms for $N = 15$ agents (Scenario S5). This near-linear scaling with swarm size validates the computational complexity analysis in Section 3.4.2 and confirms that even at 15 agents, the per-agent control computation occupies

under 2% of the 100 ms control cycle, leaving ample margin for sensor processing, state estimation, and communication tasks.

6. Discussion

6.1. Interpretation of Key Findings and Ablation Insights

Predictive Coordination as the Primary Driver of Resilience. The confidence-weighting mechanism further enhances robustness by naturally attenuating the influence of predictions as uncertainty accumulates, preventing the propagation of stale or erroneous state estimates through the control loop.

Adaptive Topology Management as an Efficiency Optimizer. This adaptive behavior enables the swarm to balance the competing objectives of communication efficiency and coordination robustness in real time, a capability that fixed-topology approaches cannot match.

LLM-Assisted Planning: A Qualitative Interface with Architectural Guardrails. The NoLLM ablation yields a clear result: the LLM serves as a qualitative planning assistant for interpreting unstructured natural language and generating structured task sequences, with the Safety and Feasibility Filter defaulting to deterministic, pre-calibrated parameters for control-critical weights. This flexibility, demonstrated qualitatively in Scenario S6, is invaluable. The TOST results formally confirm that under standard conditions, this approach yields control performance equivalent to a perfectly scripted state machine—a desirable property for safety-critical deployments.

Complementary Integration of the Three Layers. While the ablation studies isolate the contributions of individual modules, the full Ours configuration demonstrates that the three layers operate complementarily. The strategic layer provides flexible mission decomposition; the tactical layer sustains coordination during communication denial and optimizes the communication topology; and the execution layer delivers precise, constraint-satisfying control. The temporal decoupling of these layers ensures that computationally intensive LLM inference (1–2 s) does not interfere with the real-time demands of the DMPC loop (10 Hz), and that the predictive module can seamlessly compensate for communication dropouts without requiring replanning at the strategic level. This hierarchical decomposition is a key architectural contribution that enables the system to simultaneously address long-horizon planning, medium-horizon adaptation, and short-horizon reactive control.

6.2. Comparison with Theoretical Analysis

The present evaluation is simulation-based, using Gazebo with empirically derived hydrodynamic coefficients. A staged validation roadmap—from single-vessel open-loop hydrodynamic calibration through to full multi-vessel system integration with hardware-in-the-loop—has been designed to systematically address the sim-to-real gap. Stage 1 (single-vessel validation with a Clearpath Heron USV) is currently in progress.

Stability under Topology Switching. The stability analysis based on multiple Lyapunov functions and average dwell time correctly captures the behavior of the system under adaptive topology switching. The hysteresis threshold in Equation (28) enforces an average dwell time that satisfies the theoretical condition $\tau_a > (\cdot)$, thereby guaranteeing global uniform exponential stability to a bounded neighborhood. Simulation, no instability or unbounded error growth was observed during or after topology transitions across all 50 independent Monte Carlo runs of Scenario S4. The hysteresis threshold $\Delta U_{\min}$ prevented chattering, as evidenced by the small number of topology switches per mission (1.0 ± 0.0 ; zero variance reflects the limited 150 s mission duration; longer missions would elicit additional switches, see Table 6). While the average dwell time was

not explicitly measured, the qualitative behavior is consistent with the theoretical predictions.

Convergence Rate and Ultimate Bound. The simulation measured convergence rate $\rho_{\text{exp}} = 0.14 \pm 0.02$ closely matches the theoretical prediction $\rho = 0.15$. The small discrepancy is attributable to unmodeled hydrodynamic effects, sensor noise, and the discrete-time approximation of the continuous dynamics. The theoretical ultimate bound $\varepsilon_{\infty} = 1.5$ m aligns well with the simulation steady-state error of 0.034 ± 0.047 m, with the gap explained by additional sources of uncertainty in the high-fidelity simulation environment (e.g., wave-induced oscillatory motions, multipath GPS errors) that are not fully captured by the bounded disturbance assumption.

Denial Error Growth. The exponential error growth model in Theorem 2 accurately characterizes the behavior of the NoPred configuration during communication denial. The empirically calibrated divergence rate $\alpha = 0.023 \text{ s}^{-1}$ corresponds to a time constant of approximately 43 s, indicating that without prediction, formation coherence degrades rapidly. The predictive module effectively reduces this divergence rate, as evidenced by the significantly slower error accumulation of Ours. Corollary 1 provides a principled basis for estimating the maximum tolerable denial duration for a given performance threshold, which can inform mission planning and risk assessment in contested environments.

6.3. Limitations and Future Research Directions

This section discusses the current limitations of the proposed framework and identifies promising directions for future research. To improve coherence and avoid fragmentation, the discussion is organized into three thematic areas: LLM-related constraints spanning planning validation, robustness, and integration depth (Section 6.3.1); modeling fidelity and simulation validation boundaries encompassing communication, dynamics, navigation, and confidence assessment (Section 6.3.2); and scalability and deployment challenges for larger swarms and resource-constrained platforms (Section 6.3.3).

6.3.1. LLM-Assisted Planning Validation Scope

Three aspects of the current LLM integration warrant further investigation.

Validation scope. The LLM-assisted planning paradigm has been validated only qualitatively through Scenario S6, which demonstrates a single successful re-tasking instance. A systematic quantitative evaluation across diverse mission specifications (area coverage, target tracking, escort), environmental conditions (varying sea states, wind fields, current patterns), and operator preference profiles is required to establish the boundary conditions under which LLM-generated control parameters provide statistically significant improvements over expert-calibrated defaults. The TOST equivalence analysis establishes that, under the standard conditions of Scenarios S1–S5, LLM-generated parameters are statistically equivalent to expert defaults; the open question is whether scenarios exist where the LLM systematically outperforms these defaults.

Robustness to adversarial and ambiguous inputs. The current evaluation does not include systematic stress-testing of the LLM interface. Critical robustness dimensions that remain unexplored include (1) adversarial or deliberately ambiguous natural-language commands designed to probe the boundaries of the safety filter; (2) contradictory mission priorities (e.g., simultaneous commands to maximize coverage area while minimizing energy consumption beyond physically realizable limits); and (3) out-of-distribution environmental descriptions that differ significantly from the training distribution of the underlying LLM. Establishing the safe operating envelope of the LLM interface under

these conditions is essential before deployment in contested maritime environments where communications may be spoofed or degraded.

Integration depth. The current architecture employs the LLM primarily as a task decomposition engine, with a strict safety filter that falls back to pre-validated behaviors upon plan rejection. While this conservative design guarantees real-time safety regardless of LLM reliability, it limits the LLM's ability to exert nuanced, quantitative influence on swarm behavior. A more sophisticated integration could allow the LLM to suggest continuous-valued adjustments to DMPC cost function weights or topology selection criteria based on semantic understanding of mission context, subject to bounded perturbation limits verified by the safety filter. Exploring this deeper integration while maintaining formal safety guarantees represents a challenging but high-impact research direction.

6.3.2. Modeling Fidelity and Validation Limitations

The simulation framework employs several modeling simplifications whose boundaries of validity should be carefully considered when interpreting the reported numerical results.

Communication model fidelity. The RF communication model used in this study, implemented via the Gazebo wireless plugin with a log-distance path loss formulation, does not capture several maritime-specific propagation effects: evaporation ducting (which can extend the effective communication range beyond the geometric horizon under specific atmospheric conditions), sea-surface multipath (which produces frequency-selective fading due to coherent reflections from the time-varying water surface), and wave blockage (where large waves physically obstruct the line-of-sight path between low-antenna-height USVs). Established maritime communication simulators such as DESERT Underwater and the ns-3 maritime module incorporate several of these effects and could be integrated in future work to improve the fidelity of the communication environment.

Hydrodynamic model fidelity. The current DMPC formulation relies on a nominal 3-DOF Fossen hydrodynamic model. While this model captures the dominant surge-sway-yaw dynamics with well-established fidelity for surface vessels, systematic discrepancies between the model and a real vessel are inevitable, particularly in higher sea states (≥ 4) where wave-frequency motions couple more strongly into the horizontal plane, or when the vehicle operates near the limits of its performance envelope (maximum speed, minimum turn radius). Online parameter adaptation of the hydrodynamic coefficients, informed by the same online learning framework used for neighbor behavior prediction (Section 3.3.1), could partially mitigate this limitation.

GPS-denied operation. The current system assumes the availability of absolute positioning via GPS. In fully contested maritime environments, both communication and navigation signals may be denied simultaneously—a scenario more challenging than the communication-only denial considered in this work. Extending the framework to GPS-denied operation would require integration of alternative navigation modalities: visual-inertial odometry using onboard cameras and IMUs, cooperative localization using inter-agent range and bearing measurements, and terrain-relative navigation using prior bathymetric maps in coastal environments.

Simulation confidence assessment. A systematic evaluation of simulation confidence was conducted across five dimensions: physical fidelity, uncertainty quantification, parameter sensitivity, conclusion robustness, and statistical power. The primary fidelity limitations identified are: (1) the simplified RF communication model (confidence score 0.35/1.0), which does not capture maritime-specific propagation effects as noted above; (2) the 3-DOF hydrodynamic approximation (confidence score 0.72/1.0),

which is well-validated for surface vessels in moderate sea states but has known limitations for aggressive maneuvering; and (3) the absence of a hardware-in-the-loop validation step that would bridge the gap between pure simulation and physical experiments. The statistical robustness of the reported conclusions is strong (confidence score 0.91/1.0), supported by 1250 independent Monte Carlo runs with appropriate multiple-comparison corrections.

6.3.3. Scalability and Deployment Limitations

Deployment of the proposed framework at larger scales or on resource-constrained hardware platforms presents additional challenges beyond those discussed in Sections 6.3.1 and 6.3.2.

Scalability to large swarms. The ADMM-based DMPC solver exhibits per-agent computational complexity that scales near-linearly with swarm size N , as demonstrated by the computation time measurements in Section 5.2.5 (0.48 ms at $N = 5$ to 1.59 ms at $N = 15$). However, for very large swarms ($N > 50$), two effects may degrade performance. First, the number of ADMM iterations required for consensus may increase with the graph diameter, which grows logarithmically with N for mesh topologies but linearly for tree and star topologies. Second, the assumption of a connected communication graph (Assumption 2) becomes increasingly difficult to maintain as the swarm spatial extent grows relative to the communication range, requiring more frequent topology adaptations. Hierarchical or clustering-based DMPC formulations that partition the swarm into sub-groups with separate consensus processes could mitigate both effects and represent an important direction for scaling the framework to hundreds of agents.

Resource-constrained deployment. The nominal configuration assumes a moderate-performance edge computing platform (Intel Core i7-12700H class). Deployment on low-power embedded processors (ARM Cortex-A series, NVIDIA Jetson, or FPGA-based platforms typical of small USV autopilots) would require additional optimization. The degraded-hardware sensitivity analysis in Section 5.2.5 demonstrates that reducing ADMM iterations to $K_{\max} = 20$ and the prediction horizon to $H = 3$ (a 10× computational reduction) yields formation error increases of only 18–22%, suggesting that the framework can be tuned for lower-performance platforms with graceful performance degradation. Further optimizations—including code generation for the ADMM solver, fixed-point arithmetic implementations, and hardware-accelerated neural network inference for the prediction module—could extend the applicable platform range.

Real-world validation gap. All results reported in this study are based on numerical simulations. While the simulation framework incorporates high-fidelity hydrodynamic models (3-DOF Fossen with nonlinear damping), physics-based sensor noise models (GPS, IMU, DVL), and a wireless communication simulator, the gap between simulation and physical reality remains unquantified. A structured progression from the current simulation-only validation through hardware-in-the-loop testing (where real autopilot hardware processes simulated sensor data) to field experiments with physical USV platforms is essential to establish the real-world applicability of the proposed framework. The simulation confidence assessment framework introduced in Section 6.3.2 provides a systematic methodology for prioritizing which simulation components require the most urgent physical validation.

6.4. Broader Implications

This research advances the state of the art in resilient multi-agent coordination by demonstrating a practical framework that integrates strategic reasoning, tactical adaptation, and real-time control. The predictive coordination mechanism and adaptive topology management framework are domain-agnostic and can be applied to other multi-

robot systems operating in communication-constrained environments, including aerial swarms, ground robot teams, and satellite constellations. The architectural pattern of temporally decoupling LLM-assisted planning from safety-critical control, with a formal validation filter, provides a template for the safe deployment of large language models in embodied robotic systems—a challenge of growing importance as AI-driven autonomy becomes more prevalent. By enabling reliable autonomous operation in contested and communication-denied environments, this work supports critical applications in defense, disaster response, remote exploration, and maritime surveillance.

If desired heading angles ψ_i^d were additionally included in the formation specification, the control problem would require full-state (position and heading) stabilization of the underactuated USV system. This extension would introduce two significant challenges. First, the formation manifold would become fully specified in the configuration space, potentially engaging Brockett-type topological obstructions to the existence of smooth time-invariant stabilizing feedback laws, as discussed in Section 4.1. Second, the coupling between surge thrust and yaw moment through the underactuated dynamics—where a single propeller generates both the forward force and, through its lever arm l_x , the turning moment—would require explicitly coordinated surge-yaw maneuvers. The current DMPC formulation can accommodate heading constraints in principle by augmenting the stage cost function Equation (32) with a heading error penalty term, but the resulting coupled optimization problem would have a more complex geometry due to the nonlinear relationship between the control inputs and the heading dynamics. Preliminary simulations (not included in the main simulation campaign) suggest that including heading constraints increases the formation error by approximately 15% during aggressive turning maneuvers (scenarios S2 and S4) due to the additional coordination burden, but does not significantly affect steady-state straight-line formation keeping (scenarios S1 and S5). A formal treatment of full-pose formation control for underactuated USV swarms, including rigorous stability analysis accounting for the heading degree of freedom, represents an important direction for future work.

7. Conclusions

This paper has presented a three-layer control architecture for resilient USV swarm coordination under communication denial, integrating predictive coordination, adaptive topology management, and LLM-assisted strategic planning. The principal methodological contributions extend beyond the specific numerical results to encompass design principles and analytical frameworks that generalize across multi-agent systems operating under communication constraints.

First, we developed a predictive coordination mechanism that combines physics-based motion extrapolation using the 3-DOF Fossen model with online-learned neighbor behavior models implemented as lightweight single-hidden-layer neural networks. During communication outages, agents rely on predicted neighbor states with confidence weights that decay smoothly as prediction uncertainty accumulates. Ablation analysis confirms that this module alone reduces formation error by 76.0% (under the simulated conditions, DMPC + ADMM, Cohen's $d = 0.88$, $p < 0.001$) during 60 s denial events.

Second, we designed an adaptive topology management framework that dynamically reconfigures the swarm communication graph among star, mesh, and tree configurations based on a utility function jointly considering communication quality, threat exposure, and resource overhead. A hysteresis threshold ensures satisfaction of ADT conditions while limiting topology switches to ≤ 1 per 150 s mission, reducing communication overhead by 71.7% (for the particular cases investigated) without significant degradation of formation accuracy.

Third, we established theoretical guarantees through rigorous stability analysis. Using multiple Lyapunov functions with explicit spectral verification and the ADT theorem, we proved GUES of the formation error under topology switching. Explicit error bounds under communication denial were derived via Gronwall-type arguments.

The methodological framework—layered temporal decoupling, predictive coordination for communication resilience, and utility-driven adaptive topology—generalizes beyond the maritime domain to aerial swarms, ground robot teams, and space constellations requiring coordinated behavior under communication constraints.

Author Contributions: Conceptualization, J.Z.; methodology, X.L.; software, X.L.; validation, Y.L.; formal analysis, P.Z.; writing—original draft, X.L.; writing—review and editing, L.T. and P.Z.; supervision, X.L.; project administration, X.L. All authors have read and agreed to the published version of the manuscript.

Funding: Postdoctoral Fund: China Postdoctoral Science Foundation, the 77th batch of postdoctoral fund surface funding in 2025, funding number: 2025M774430.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Table A1. Comparison of the proposed approach with related methods for prediction under communication loss and LLM-based multi-robot planning.

Method/Frame work	Prediction Under Comm. Loss	Topology Adaptation	LLM Planning	Theoretical Guarantees	Domain
Observer-based DMPC	Adaptive observer for leader state	No	No	Stability under link faults	General MAS
PredSLS	System-level disturbance prediction	No	No	Robust performance bounds	Networked control
GP-MPC	GP-based multi-agent dynamics prediction	No	No	Probabilistic safety	General MAS
SMART-LLM	No	No	Hierarchical task decomposition + feasibility check	No (LLM safety filter only)	General robotics
LAN2CB	No	Communication bottleneck architecture	Language-to-behavior mapping	No	Multi-robot
CLIPSwarm	No	No	Natural language → formation	No	Drone swarms
This Work (Ours)	Physics + learned behavioral residual + confidence decay	Star/mesh/tree via utility-optimization + hysteresis	Safety-filtered LLM (qualitative only)	MLF+ADT stability + denial error bound (Gronwall)	USV swarms

Appendix B

Proof of Theorem 1.

This appendix provides the self-contained, complete proof of Theorem 1 (Stability under Average Dwell Time). All notation, assumptions, and definitions are restated for completeness. $\square$

B.1. System Model and Assumptions

Consider a swarm of N unmanned surface vehicles (USVs) indexed by $i \in \{1, \dots, N\}$. The discrete-time dynamics of agent i are:

$$x_i(t+1) = f_i(x_i(t), u_i(t)) + w_i(t), t = 0, 1, 2, \dots \quad (\text{A1})$$

where $x_i(t) \in \mathbb{R}^3$ is the state, $u_i(t) \in \mathbb{R}^2$ is the control input, f_i denotes the nominal 3-DOF Fossen dynamics Equation (3), and $w_i(t)$ is the lumped environmental disturbance. The stacked formation error is $e(t) = x(t) - x_d(t) \in \mathbb{R}^{3N}$.

The time-varying communication topology $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$ has Laplacian $L(t) \in \mathbb{R}^{N \times N}$. The topology is selected from $\Sigma = \{S, M, T\}$ by the adaptive topology manager (Section 3.3.2) based on the utility function Equation (26) with hysteresis threshold $\varepsilon_{\text{hyst}}$ Equation (28).

The DMPC law minimizes the finite-horizon cost Equation (32) subject to consensus constraints Equation (34), solved via ADMM (Section 3.4.2) with at most K_{max} iterations per time step.

Assumptions A1.

(A1) Each candidate topology $\mathcal{G}_\sigma$ contains a directed spanning tree; the regularized Laplacian $\tilde{L}_\sigma = L_\sigma + \varepsilon I_N$ is positive definite on $\text{span}\{\mathbf{1}_N\}^\perp$.

(A2) $\|w(t)\| \leq w_{\text{max}}$.

(A3) For each fixed σ , the DMPC with DARE terminal cost gives nominal exponential stability with decay $\rho_\sigma \in (0, 1)$.

(A4) ADMM residual satisfies $\|r(t)\| \leq r_{\text{max}}$.

B.2. Lemma A1: Regularized Laplacian Positive Definiteness

Lemma A1. Let $L(\mathcal{G})$ be the Laplacian of a directed graph $\mathcal{G}$ with a spanning tree. For $\varepsilon > 0$, define $\tilde{L} = L + \varepsilon I_N$. Then $\tilde{L}$ is symmetric positive definite on the $(N-1)$ -dimensional subspace orthogonal to $\text{span}\{\mathbf{1}_N\}$: there exists $\lambda_2(\tilde{L}) > 0$ such that $x^\top \tilde{L} x \geq \lambda_2(\tilde{L}) \|x\|^2$ for all x with $\mathbf{1}_N^\top x = 0$.

Proof. The Laplacian L of a graph with a spanning tree has exactly one zero eigenvalue (eigenvector $\mathbf{1}_N$); all other eigenvalues have strictly positive real parts [33]. The symmetrized Laplacian $L_{\text{sym}} = (L + L^\top)/2$ shares this property for bidirectional graphs. Adding εI_N shifts the spectrum: $\lambda_i(\tilde{L}) = \lambda_i(L_{\text{sym}}) + \varepsilon$. The zero eigenvalue becomes $\varepsilon > 0$, and the smallest positive eigenvalue is $\lambda_2(\tilde{L}) = \lambda_2(L_{\text{sym}}) + \varepsilon > 0$. $\square$

B.3. Spectral Properties of the Three Candidate Topologies

1. Star Topology. The star graph S_N (one center, $N-1$ leaves) has Laplacian eigenvalues $\{0, 1 (\times N-2), N\}$ [34]. $\lambda_{\min}(\tilde{L}_S) = 1 + \varepsilon$, $\lambda_{\max}(\tilde{L}_S) = N + \varepsilon$.
2. Mesh Topology. The complete graph K_N has Laplacian eigenvalues $\{0, N (\times N-1)\}$. $\lambda_{\min}(\tilde{L}_M) = N + \varepsilon$, $\lambda_{\max}(\tilde{L}_M) = N + \varepsilon$. The condition number on the consensus-orthogonal subspace is $\kappa = 1$ (optimal).
3. Tree Topology. For a balanced binary tree, the Fiedler eigenvalue satisfies $\lambda_2(L_T) \geq 2(1 - \cos(\pi/N))$ [35,36]. For $N = 5$, $\cos(\pi/5) \approx 0.809$, $\lambda_2(L_T) \geq 0.382$. $\lambda_{\min}(\tilde{L}_T) \geq 2(1 - \cos(\pi/N)) + \varepsilon$, $\lambda_{\max}(\tilde{L}_T) \leq 2\Delta + \varepsilon$ ($\Delta = 3$ for binary tree).

B.4. Complete Proof of Theorem 1

Theorem A1 (Stability under ADT). *Under Assumptions (A1)–(A4), if $\tau_a > \tau_a^* = \ln \mu / \ln (1/\rho_{\max})$ with $\mu = \max_{i,j} \lambda_{\max}(P_i) / \lambda_{\min}(P_j)$ and $\rho_{\max} = \max_{\sigma} \rho_{\sigma}$, then the formation error is GUES:*

$$\|e(t)\| \leq \gamma \lambda^{\frac{t-t_0}{2}} \|e(t_0)\| + \delta(w_{\max}, r_{\max}), \quad (\text{A2})$$

with $\delta = O(w_{\max} + r_{\max})$.

Proof. Four steps as detailed in Section 4.1 (Equations (38)–(52)). Step 1 constructs $V_{\sigma}(e) = e^{\top}(\bar{L}_{\sigma} \otimes I_3)e$ with quadratic bounds (39). Step 2 establishes the one-step decay bound (44) via the DARE-based Lyapunov equation and Young inequality. Step 3 bounds the switching jump by $\mu = (N + \varepsilon)/(1 + \varepsilon)$ Equation (47). Step 4 applies the ADT theorem recursively over K switches, defines the effective decay rate $\lambda = \rho_{\max} \mu^{1/\tau_a}$ Equation (49), and derives the ADT condition $\tau_a^* = \ln \mu / \ln (1/\rho_{\max})$ Equation (50) from $\lambda < 1$. Conversion via Equation (39) yields the error norm bound (51) with ultimate bound (52). $\square$

B.5. Explicit Numerical Constants ($N = 5$, $\varepsilon = 0.01$)

For the configuration with $N = 5$ and $\varepsilon = 0.01$, the regularized Laplacian of the star topology has a minimum eigenvalue of 1.01, a maximum of 5.01, and a condition number of 4.96; the mesh topology has both eigenvalues equal to 5.01, yielding the optimal condition number of 1.00; the tree topology has a minimum eigenvalue no less than 0.392 and a condition number at most 15.3. The Lyapunov decay factors are $\rho_s = 0.820$, $\rho_M = 0.712$, and $\rho_T = 0.785$, with $\rho_{\max} = 0.820$. The switching jump bound is $\mu \approx 4.96$. The resulting average dwell time thresholds are approximately 8.07 steps (0.81 s) from a conservative estimate and 2.27 steps (0.23 s) from the tighter DARE-based construction, while the actual operational average dwell time in simulations is 1500 steps (150 s) with at most one switch, yielding an enormous safety margin. The disturbance and residual gain constants are $c_1 = 2.34$ and $c_2 = 1.87$, and the ultimate bound on the formation error is $\delta \approx 0.012$ m.

These values confirm the ADT condition is satisfied with overwhelming margin in all simulation scenarios (Section 5).

Abbreviations

The following abbreviations are used in this manuscript:

USV	Unmanned Surface Vehicle
DMPC	Distributed Model Predictive Control
ADMM	Alternating Direction Method of Multipliers
LLM	Large Language Model
CoT	Chain-of-Thought
ORCA	Optimal Reciprocal Collision Avoidance
ADT	Average Dwell Time
MLF	Multiple Lyapunov Functions
DOF	Degrees of Freedom
ROS2	Robot Operating System 2
SNR	Signal-to-Noise Ratio
PLR	Packet Loss Rate
GPS	Global Positioning System
IMU	Inertial Measurement Unit
DVL	Doppler Velocity Log
OCP	Optimal Control Problem
DARE	Discrete-time Algebraic Riccati Equation

SINR	Signal-to-Interference-plus-Noise Ratio
JONSWAP	Joint North Sea Wave Project
COLREGs	International Regulations for Preventing Collisions at Sea
WMO	World Meteorological Organization
MADM	Multi-Attribute Decision-Making
ETX	Expected Transmission Count

References

1. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley & Sons: Chichester, UK, 2011. <https://doi.org/10.1002/9781119994138>.
2. Liu, J.; Zhang, Y.; Yuan, C. Recent advances in unmanned surface vehicles: A comprehensive review. *Ocean Eng.* **2023**, *287*, 115842.
3. Peng, Z.; Wang, J.; Wang, D.; Han, Q.-L. An overview of recent advances in coordinated control of multiple autonomous surface vehicles. *IEEE Trans. Ind. Inform.* **2021**, *17*, 732–745. <https://doi.org/10.1109/TII.2020.3004343>.
4. Ren, H.; Liu, Q.; Zhou, Q.; Lu, R. A review of distributed cooperative control research on unmanned autonomous systems. *J. Guangdong Univ. Technol.* **2024**, *41*, 1–13.
5. Ilioudi, A.; Sosnowski, S.; Banken, E.; Bevanda, P.; Brüdigam, J.; Buşoniu, L.; Chardard, Y.; Delea, C.; De Schutter, B.; Duraš, A.; et al. The SeaClear system: An intelligent multi-robot solution for autonomous cleanup of marine debris on the seabed. *Ocean Eng.* **2026**, *323*, 120594. <https://doi.org/10.1016/j.oceaneng.2026.114094>.
6. Yang, M.; Guan, X.; Shi, M.; Li, B.; Wei, C.; Yiu, K.C. Distributed model predictive formation control for USVs and cooperative capability evaluation of swarm. *Drones* **2025**, *9*, 112. <https://doi.org/10.3390/drones9050366>.
7. Huang, Y.; Kuai, J.; Meng, Z.; Sun, J. Discrete-time distributed optimal formation algorithms for multiagent systems with nonlinear inequality constraints. *IEEE Trans. Syst. Man Cybern. Syst.* **2026**, *56*, 1249–1261. <https://doi.org/10.1109/TSMC.2025.3647847>.
8. Huang, B.; Li, J.; Zhang, Z.; Zhang, C.; Ahn, C.K. A dynamic time assignment communication mechanism-based formation control for Internet of uncrewed surface vehicles under DoS attacks. *IEEE Internet Things J.* **2025**, *12*, 10420–10433. <https://doi.org/10.1109/JIOT.2024.3510396>.
9. Abro, G.E.M.; Abdallah, A.M.; Zahid, F.; Ahmed, S. A comprehensive review of next-gen USV swarm robotics: Optimisation techniques and control strategies for dynamic environments. *CMC-Comput. Mater. Contin.* **2025**, *81*, 99–123.
10. Continelli, E. Consensus and flocking with transmission and reaction delays. *arXiv* **2026**, arXiv:2604.18848.
11. Noroozi, N. Distributed multi-step model predictive control for consensus. *arXiv* **2026**, arXiv:2602.14714.
12. Liu, X.; Dimarogonas, D.V.; Liu, C.; Dabiri, A.; De Schutter, B. Distributed model predictive control without terminal cost under inexact distributed optimization. *Automatica* **2026**, *165*, 111687. <https://doi.org/10.1016/j.automatica.2026.113050>.
13. Mallick, S.; Dabiri, A.; De Schutter, B. Distributed model predictive control for piecewise affine systems based on switching ADMM. *IEEE Trans. Autom. Control* **2025**, *70*, 3727–3741. <https://doi.org/10.1109/TAC.2024.3512334>.
14. Cong, S.; Gu, N.; Wang, H.; Wang, D.; Peng, Z. Safety-certified distributed formation control of networked autonomous surface vehicles by unifying control Lyapunov and control barrier functions. *J. Syst. Sci. Complex.* **2025**, *38*, 837–856. <https://doi.org/10.1007/s11424-025-4494-8>.
15. Kawabe, T.; Takano, R. Hierarchical LLM-based multi-agent framework with prompt optimization for multi-robot task planning. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Vienna, Austria, 1–5 June 2026; pp. 4521–4528.
16. Wang, R.; Hsu, H.L.; Kim, J.; Pajic, M. Semantic area graph reasoning for multi-robot language-guided search. *arXiv* **2026**, arXiv:2604.16263.
17. Zhang, T.; Qu, K.; Li, Z.; Wu, J.; Hutter, M.; Li, M.; Shi, F. Using large language models for embodied planning introduces systematic safety risks. *arXiv* **2026**, arXiv:2604.18463.
18. Song, W.; Zuo, Y.; Tong, S. Switching event-triggered fuzzy resilient control for networked unmanned surface vehicle under denial of service attacks. *Ocean Eng.* **2025**, *323*, 120592. <https://doi.org/10.1016/j.oceaneng.2025.120592>.
19. He, S.; Wang, M.; Dai, S.-L.; Luo, F. Leader–follower formation control of USVs with prescribed performance and collision avoidance. *IEEE Trans. Ind. Inform.* **2019**, *15*, 572–581. <https://doi.org/10.1109/TII.2018.2839739>.
20. Riccardi, A.; Laurenti, L.; De Schutter, B. Partitioning techniques for non-centralized predictive control: A systematic review and novel theoretical insights. *Annu. Rev. Control* **2026**, *61*, 101046. <https://doi.org/10.1016/j.arcontrol.2026.101046>.

21. Palopoli, D.; De Martini, D.; Saveriano, M.; Lamon, E.; Palopoli, L.; Roveri, M. Automated generation of MDPs using logic programming and LLMs for robotic applications. *IEEE Robot. Autom. Lett.* **2025**, *11*, 1770–1777. <https://doi.org/10.1109/LRA.2025.3643276>.
22. Jiang, E.H.; Li, M.; Wan, G.; Yin, S.; Wu, Y.; Liang, X.; Li, X.; Sun, Y.; Wang, W.; Chang, K.-W.; et al. Dynamic generation of multi-LLM agents communication topologies with graph diffusion models. *arXiv* **2025**, arXiv:2510.07799.
23. Zhang, G.; Yue, Y.; Sun, X.; Wan, G.; Yu, M.; Fang, J.; Wang, K.; Chen, T.; Cheng, D. G-Designer: Architecting multi-agent communication topologies via graph neural networks. *arXiv* **2024**, arXiv:2410.11782.
24. Messilem, A.M.; Carnevale, G.; Carli, R. Distributed constraint-coupled optimization: Harnessing ADMM-consensus for robustness. In Proceedings of the IFAC World Congress, Hong Kong, China, 2–5 June 2025; pp. 8942–8949.
25. Mallick, S.; Dabiri, A.; De Schutter, B. Multi-agent reinforcement learning via distributed MPC as a function approximator. *Automatica* **2024**, *167*, 111803. <https://doi.org/10.1016/j.automatica.2024.111803>.
26. Hammami, H.E.; Maraoui, S.; Mansouri, M.; Bouzrara, K.; Mnif, F. Robust dynamic self-triggered DNMPc based on ADMM for cooperative multi-agent systems. *IEEE Access* **2025**, *13*, 216759–216771. <https://doi.org/10.1109/ACCESS.2025.3647524>.
27. Riccardi, A.; Laurenti, L.; De Schutter, B. A benchmark for multi-agent control of energy systems: The European economic area hybrid electricity network benchmark. In Proceedings of the 2025 European Control Conference, Thessaloniki, Greece, 24–27 June 2025; pp. 2575–2582. <https://doi.org/10.23919/ECC65951.2025.11187247>.
28. Brockett, R.W. Asymptotic stability and feedback stabilization. In *Differential Geometric Control Theory*; Brockett, R.W., Millman, R.S., Sussmann, H.J., Eds.; Differential geometric control theory; Birkhäuser: Basel, Switzerland, 1983; pp. 181–191.
29. Pettersen, K.Y.; Nijmeijer, H. Underactuated ship tracking control: Theory and experiments. *Int. J. Control* **2001**, *74*, 1435–1446. <https://doi.org/10.1080/00207170110072039>.
30. Aguiar, A.P.; Hespanha, J.P. Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. *IEEE Trans. Autom. Control* **2007**, *52*, 1362–1379. <https://doi.org/10.1109/TAC.2007.902731>.
31. Rawlings, J.B.; Mayne, D.Q. *Model Predictive Control: Theory and Design*; Nob Hill Publishing: San Francisco, CA, USA, 2009.
32. Liberzon, D. *Switching in Systems and Control*; Birkhäuser: Basel, Switzerland, 2003. <https://doi.org/10.1007/978-1-4612-0017-8>.
33. Olfati-Saber, R.; Murray, R.M. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Trans. Autom. Control* **2004**, *49*, 1520–1533. <https://doi.org/10.1109/TAC.2004.834113>.
34. Brouwer, A.E.; Haemers, W.H. *Spectra of Graphs*; Springer: Berlin/Heidelberg, Germany, 2012. <https://doi.org/10.1007/978-1-4614-1939-6>.
35. Fiedler, M. Algebraic connectivity of graphs. *Czechoslov. Math. J.* **1973**, *23*, 298–305. <https://doi.org/10.21136/CMJ.1973.101168>.
36. Mohar, B. The Laplacian spectrum of graphs. In *Graph Theory, Combinatorics, and Applications*; Alavi, Y., Chartrand, G., Oellermann, O.R., Schwenk, A.J., Eds.; Graph theory, combinatorics, and applications; Wiley: Hoboken, NJ, USA, 1991; Volume 2, pp. 871–898.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.