

Article

Cooperative Coverage Control for Heterogeneous AUVs Based on Control Barrier Functions and Consensus Theory

Fengxiang Mao *, Dongsong Zhang , Liang Xu and Rui Wang

School of Big Data and Artificial Intelligence, Xinyang College, Xinyang 464000, China

* Correspondence: mfxwj2008@163.com

Abstract

This paper addresses the problem of cooperative coverage control for heterogeneous Autonomous Underwater Vehicle (AUV) swarms operating in complex underwater environments. The objective is to achieve optimal coverage of a target region while simultaneously ensuring collision avoidance—both among AUVs and with static obstacles—and satisfying the inherent dynamic constraints of the AUVs. To this end, we propose a hierarchical control framework that fuses Control Barrier Functions (CBFs) with consensus theory. First, addressing the heterogeneity and limited sensing ranges of the AUVs, a cooperative coverage model based on a modified Voronoi partition is constructed. A nominal controller based on consensus theory is designed to balance the ratio of task workload to individual capability for each AUV. By minimizing a Lyapunov-like function via gradient descent, the swarm achieves self-organized optimal coverage. Second, to guarantee system safety, multiple safety constraints are designed for the AUV double-integrator dynamics, utilizing Zeroing Control Barrier Functions (ZCBFs) and High-Order Control Barrier Functions (HOCBFs). This approach unifies the handling of collision avoidance and velocity limitations. Finally, the nominal coverage controller and safety constraints are integrated into a Quadratic Programming (QP) formulation. This constitutes a safety-critical layer that modifies the control commands in a minimally invasive manner. Theoretical analysis demonstrates the stability of the framework, the forward invariance of the safe set, and the convergence of the coverage task. Simulation experiments verify the effectiveness and robustness of the proposed method in navigating obstacles and efficiently completing heterogeneous cooperative coverage tasks in complex environments.

Keywords: heterogeneous AUV swarms; cooperative coverage control; control barrier functions; consensus theory; quadratic programming



Academic Editor: Fabio Leccese

Received: 28 December 2025

Revised: 21 January 2026

Accepted: 22 January 2026

Published: 26 January 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Covering over 70% of the Earth's surface, the ocean harbors abundant natural resources and holds significant strategic importance for the global climate, economy, and security. As intelligent platforms capable of autonomous navigation and the execution of exploration and operational missions, Autonomous Underwater Vehicles (AUVs) have emerged as a pivotal technology for marine exploitation and utilization [1–3].

In recent years, driven by advancements in Multi-Agent System (MAS) theory, the deployment of multiple AUVs for cooperative missions has emerged as an inevitable trend in underwater operations [4,5]. The theoretical foundations of networked control in MASs have been extensively studied, providing rigorous frameworks for analyzing system stability and consensus [6,7]. Furthermore, concepts such as bearing rigidity have extended

the capabilities of formation control and estimation beyond simple distance constraints [8]. In practical applications, ensuring collision avoidance while maintaining formation or tracking trajectories remains a critical challenge. Recent works have explored distributed formation tracking for second-order systems, emphasizing the importance of robust control protocols in dynamic environments [9]. Compared to a single AUV, an AUV swarm offers superior efficiency, enhanced robustness, and a more extensive operational range. In this context, cooperative coverage control represents a core application [10,11]. Its primary objective is to coordinate the AUVs to achieve an optimal distribution across the target region, thereby facilitating tasks such as seafloor mapping, hydrographic data collection, and search and rescue operations.

However, achieving efficient and safe cooperative coverage in practical underwater environments presents numerous challenges. First, AUV swarms are typically heterogeneous; that is, individual agents exhibit disparities in endurance, sensing capabilities, and data processing power [12]. Employing homogeneous control strategies in such scenarios can lead to the underutilization of capable AUVs and the overloading of less capable ones, thereby degrading overall system efficiency and operational endurance [13]. Consequently, it is imperative to design a heterogeneous cooperative coverage strategy capable of balancing task loads effectively. Second, the underwater environment is complex and unstructured, characterized by numerous known or unknown static obstacles (e.g., submerged reefs, shipwrecks, and subsea infrastructure). Furthermore, during mission execution, there is an inherent risk of inter-agent collisions [14]. Safety is the paramount prerequisite for AUV swarm operations. Traditional obstacle avoidance methods, such as Artificial Potential Fields (APFs), often suffer from local minima and struggle to provide rigorous safety guarantees [15]. Finally, AUVs are subject to complex nonlinear dynamics and physical constraints (e.g., velocity and acceleration saturation) [16], and their onboard sensors (e.g., sonar) typically possess limited sensing ranges [17]. Collectively, these factors significantly exacerbate the complexity of the control system design.

In the domain of AUV cooperative coverage, the Voronoi-based Lloyd's algorithm stands as one of the most classical methods [18,19]. This approach operates by iteratively computing the centroids of Voronoi regions and steering agents toward these centroids, ultimately achieving optimal coverage of the region [20]. However, the conventional Lloyd's algorithm is primarily tailored for homogeneous systems and fails to account for heterogeneity [21]. To address the challenge of heterogeneity, researchers have introduced workload-based consensus theory [22]. The core concept shifts from merely converging to region centroids to leveraging information exchange among neighbors; the objective is to drive the ratio of task workload to individual capability for each AUV toward a consensus value, thereby achieving load balancing in coverage control [23].

In terms of system safety assurance, Control Barrier Functions (CBFs) have garnered significant attention as an emerging technique [24]. By integrating CBFs with a nominal controller (e.g., a coverage controller) within a Quadratic Programming (QP) framework, it is possible to compute a control input online that prioritizes the satisfaction of safety constraints while executing the nominal task to the greatest extent possible [25]. This approach offers a powerful mathematical tool for addressing control problems in safety-critical systems.

To date, research applying CBFs to obstacle avoidance in multi-agent systems has yielded promising results; however, most existing studies predominantly concentrate on homogeneous systems or simple collision avoidance tasks. Research regarding the integration of the CBF safety framework with load-balanced coverage control for heterogeneous AUV swarms remains insufficient [26]. To address the aforementioned challenges, the aim of this study was to design a cooperative coverage control algorithm tailored for heterogeneous

AUV swarms that provides rigorous safety guarantees. The main contributions of this paper are summarized as follows:

- (a) A cooperative coverage model is proposed that explicitly accounts for both the heterogeneous capabilities of AUVs and their limited sensor sensing ranges. Based on a modified Voronoi partition, this model formalizes the coverage objective as the minimization of a global cost function that characterizes the variance of the task workload-to-capability ratio.
- (b) A nominal controller based on gradient descent is designed specifically for the double-integrator dynamics of AUVs. The derivation of the gradient integrates consensus theory to drive the convergence of the workload-to-capability ratios, thereby achieving dynamic task load balancing at the system level.
- (c) A unified safety-critical controller is constructed within a QP framework. By leveraging CBFs, this controller transforms inter-AUV collision avoidance, static obstacle avoidance, and velocity constraints into linear inequality constraints on the control input, ensuring rigorous system safety in complex environments.
- (d) The effectiveness of the proposed control framework is rigorously proven. Utilizing Lyapunov stability theory and the invariance principle, it is demonstrated that the closed-loop system converges to the optimal equilibrium point of the coverage task subject to the satisfaction of all safety constraints.

The remainder of this paper is organized as follows. Section 2 presents the problem formulation and system modeling, covering preliminaries such as the CBF, the AUV dynamic model, and the modeling of the heterogeneous cooperative coverage problem. Section 3 details the design of the CBF-integrated coverage controller, including the derivation of the nominal controller gradient and the formalization of multiple safety constraints. Section 4 provides a system stability analysis of the proposed framework, encompassing proofs for both safety and coverage convergence. Section 5 validates the effectiveness of the proposed algorithm through numerical simulations. Finally, Section 6 concludes the paper and outlines directions for future research.

2. Problem Formulation and Modeling

This section serves to establish the theoretical and modeling foundations for the subsequent controller design. First, we introduce the preliminaries, including Control Barrier Functions—the cornerstone of the safety control strategy in this paper—and Voronoi theory for region partitioning. Next, the dynamic model of the AUV is established. Finally, we mathematically formulate the AUV cooperative coverage control problem by incorporating heterogeneity, limited sensing ranges, and safety constraints.

2.1. Preliminaries

2.1.1. Control Barrier Functions

CBFs serve as a powerful tool for guaranteeing system safety. This is achieved by defining a safe set and ensuring that the system state consistently remains within the interior of this set.

Consider a nonlinear control-affine system described by

$$\dot{x} = f(x) + g(x)u, \quad (1)$$

where $x \in \mathbb{R}^n$ denotes the system state, and $u \in U \subset \mathbb{R}^m$ represents the control input. The functions f and g are assumed to be locally Lipschitz-continuous [27].

Definition 1 (Safe Set and Forward Invariance). Consider a set C defined by a continuously differentiable function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ as follows:

$$\begin{cases} C = \{x \in \mathbb{R}^n : h(x) \geq 0\} \\ \partial C = \{x \in \mathbb{R}^n : h(x) = 0\} \\ \text{Int}(C) = \{x \in \mathbb{R}^n : h(x) > 0\} \end{cases} \quad (2)$$

The set C is referred to as the safe set. And C is said to be forward-invariant with respect to system (1) if, for every initial state $x(t_0) \in C$, any solution $x(t)$ to system (1) satisfies $x(t) \in C$ for all $t \geq t_0$.

To ensure the forward invariance of C , the following definition is introduced:

Definition 2 (Extended Class $\mathcal{K}$ Function [28]). A continuous function $\alpha : (-b, a) \rightarrow (-\infty, \infty)$ (for some $a, b > 0$) is called an extended class $\mathcal{K}$ function if it is strictly increasing and satisfies $\alpha(0) = 0$.

Definition 3 (Zeroing Control Barrier Function, ZCBF). Given system (1) and the set C defined by (2), the function $h(x)$ is called a Zeroing Control Barrier Function (ZCBF) if there exists an extended class $\mathcal{K}$ function α such that, for all $x \in C$, there exists a control input $u \in U$ satisfying

$$\sup_{u \in U} [L_f h(x) + L_g h(x)u + \alpha(h(x))] \geq 0, \quad (3)$$

where $\sup$ denotes the supremum, and $L_f h(x) = \frac{\partial h}{\partial x} f(x)$ and $L_g h(x) = \frac{\partial h}{\partial x} g(x)$ are the Lie derivatives [28] of $h(x)$ along $f(x)$ and $g(x)$, respectively.

Theorem 1 (ZCBF and Safety [29]). Given a ZCBF $h(x)$, define the set of feasible controls $K_{cbf}(x)$ as

$$K_{cbf}(x) = \{u \in U \mid L_f h(x) + L_g h(x)u + \alpha(h(x)) \geq 0\}. \quad (4)$$

For any initial state $x(t_0) \in C$, if there exists a locally Lipschitz-continuous controller $u(x)$ such that $u(x) \in K_{cbf}(x)$, then the system state $x(t)$ will remain within C for all time, implying that C is forward-invariant.

Definition 4 (Relative Degree). The relative degree r of a function $h : \mathbb{R}^n \rightarrow \mathbb{R}$ with respect to system (1) is defined as the minimum number of times h must be differentiated with respect to time for the control input u to explicitly appear in the derivative $\frac{d^r h}{dt^r}$.

When the relative degree of $h(x)$ satisfies $r > 1$ (i.e., $L_g h(x) = 0$), the control input u does not appear in inequality (4), rendering the ZCBF ineffective. In such cases, High-Order Control Barrier Functions (HOCBFs) are required.

Definition 5 (High-Order Control Barrier Function, HOCBF). For a function $h(x)$ with a relative degree of r , a series of functions can be recursively defined as

$$\begin{cases} \psi_0(x) = h(x) \\ \psi_1(x) = \dot{\psi}_0(x) + k_1 \psi_0(x) \\ \vdots \\ \psi_{r-1}(x) = \dot{\psi}_{r-2}(x) + k_{r-1} \psi_{r-2}(x) \end{cases} \quad (5)$$

where $k_1, \dots, k_{r-1} > 0$ are constants. The function $h(x)$ is referred to as a HOCBF if there exists an extended class $\mathcal{K}$ function α and a constant $k_r > 0$ such that the set $C_H = \{x \in \mathbb{R}^n \mid \psi_0(x) \geq$

$0, \dots, \psi_{r-1}(x) \geq 0\}$, defined by $\psi_0, \dots, \psi_{r-1}$, is forward-invariant. This can be achieved by satisfying the following inequality:

$$L_f \psi_{r-1}(x) + L_g \psi_{r-1}(x)u + k_r \psi_{r-1}(x) \geq 0. \quad (6)$$

A controller u satisfying this inequality guarantees that the system state consistently remains within the safe set $C \subseteq C_H$.

2.1.2. Control Lyapunov Functions

Control Lyapunov Functions (CLFs) serve as a fundamental tool for analyzing system stability and synthesizing stabilizing controllers. By extending Lyapunov stability theory to control systems, CLFs provide a means to identify a control input u that guarantees system stability.

Definition 6 (Control Lyapunov Function [30]). *For the control-affine system (1), a continuously differentiable function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ is termed an (exponentially stabilizing) Control Lyapunov Function (CLF) if there exist positive constants c_1, c_2, c_3 such that for all $x \in \mathbb{R}^n$:*

$$c_1 \|x\|^2 \leq V(x) \leq c_2 \|x\|^2, \quad (7)$$

$$\inf_{u \in U} [L_f V(x) + L_g V(x)u + c_3 V(x)] \leq 0. \quad (8)$$

Inequality (7) ensures the positive definiteness of $V(x)$. Inequality (8) constitutes the core condition, indicating the existence of at least one control input u capable of rendering $\dot{V}(x, u) = L_f V(x) + L_g V(x)u$ negative definite or essentially causing $V(x)$ to decay.

Theorem 2 (CLF and Controller). *Given a CLF $V(x)$, any locally Lipschitz-continuous controller $u(t)$ satisfying (8) stabilizes system (1) to the zero equilibrium state. In this work, the principles of CLFs are leveraged to design a nominal controller $\hat{u}_i$ that ensures the coverage cost function decays over time, thereby achieving convergence to the optimal coverage configuration.*

2.1.3. Voronoi Region Partitioning

Voronoi partitioning stands as a classical and highly effective method for region decomposition in coverage control problems. Consider a swarm of N AUVs operating within a target region Q , with their positions denoted by $P = \{p_1, p_2, \dots, p_N\}$.

Definition 7 (Voronoi Partition [31]). *The Voronoi region V_i associated with the i -th AUV p_i is defined as the set of points within region Q whose distance to p_i*

$$V_i(P) = \{q \in Q \mid f(p_i, q) \leq f(p_j, q), \forall j \in \mathcal{N} \setminus \{i\}\}, \quad (9)$$

where $f(p_i, q)$ typically represents the Euclidean distance between p_i and q , i.e., $f(p_i, q) = \|p_i - q\|$.

Lemma 1. *For a given cost function $H(P, W) = \sum_{i=1}^N \int_{W_i} f(p_i, q) \phi(q) dq$, where $\phi(q)$ represents the scalar density function characterizing the information importance distribution (as defined in Section 2.3.1), under the condition that the AUV positions P are fixed, the optimal region partition $W = \{W_1, \dots, W_N\}$ that minimizes H is precisely the Voronoi partition $V = \{V_1, \dots, V_N\}$.*

2.2. AUV Dynamic Model

The complete six-degree-of-freedom (6-DOF) nonlinear dynamic model of an AUV is inherently complex [32]. In cooperative coverage control scenarios, AUV swarms typically execute planar missions at a constant depth or with fixed depth differentials. To re-

duce model complexity while retaining the essential inertial and velocity characteristics required for collision avoidance, the horizontal motion of the AUV is simplified to a double-integrator model [33]. Consider a swarm consisting of N AUVs, indexed by the set $\mathcal{N} = \{1, 2, \dots, N\}$. The state of the i -th AUV is denoted as $x_i = [p_i^T, v_i^T]^T \in \mathbb{R}^4$, where $p_i = [x_i, y_i]^T \in \mathbb{R}^2$ represents its position in the horizontal plane, and $v_i = [\dot{x}_i, \dot{y}_i]^T \in \mathbb{R}^2$ denotes its velocity. The dynamic model is expressed as follows:

$$\begin{cases} \dot{p}_i = v_i \\ \dot{v}_i = u_i \end{cases} \quad (10)$$

where $u_i \in \mathbb{R}^2$ serves as the control input, representing the horizontal acceleration of the AUV. This model aligns with the control-affine form described in (1), with the state defined as $x_i = [p_i^T, v_i^T]^T$ and the system dynamics given by

$$f(x_i) = \begin{bmatrix} v_i \\ 0_{2 \times 1} \end{bmatrix}, \quad g(x_i) = \begin{bmatrix} 0_{2 \times 2} \\ I_{2 \times 2} \end{bmatrix}$$

Furthermore, the AUVs are subject to physical constraints, specifically limitations on the magnitudes of their velocity and acceleration:

$$\begin{cases} \|v_i\| \leq v_{max} \\ \|u_i\| \leq u_{max} \end{cases} \quad (11)$$

where v_{max} and u_{max} denote the maximum velocity and maximum acceleration, respectively.

Although the double-integrator model simplifies complex hydrodynamics (e.g., drag and added mass), it serves as a high-level kinematic planner in this framework. The generated control input u_i is treated as a reference acceleration, which is typically tracked by a low-level dynamic controller (e.g., PID or Sliding Mode) that handles the specific actuator dynamics and environmental disturbances in practical implementation.

2.3. Modeling of the Cooperative Coverage Control Problem

This section presents the mathematical formulation of the cooperative coverage problem for heterogeneous AUV swarms, explicitly accounting for limited sensor capabilities and safety constraints.

2.3.1. Mission Space and Sensing Model

The mission space $Q \subset \mathbb{R}^2$ for the AUV swarm is defined as a bounded convex region. The distribution of information importance within this region is characterized by a known density function $\phi : Q \rightarrow \mathbb{R}_{\geq 0}$, where a larger value of $\phi(q)$ corresponds to a higher detection value or task priority at point q .

Each AUV is equipped with sonar sensors possessing limited sensing ranges. Consequently, the sensing range C_i is modeled as a circular region centered at the AUV's position p_i with a radius r_i :

$$C_i(p_i) = \{q \in Q \mid \|q - p_i\| \leq r_i, \forall i \in \mathcal{N}\}. \quad (12)$$

The AUV swarm is heterogeneous, with each AUV possessing a distinct comprehensive mission capability denoted by $E_i \in \mathbb{R}^+$. Here, E_i is a scalar that comprehensively encapsulates factors such as sensor performance, data processing speed, and endurance. A larger value of E_i indicates a stronger capability to execute missions.

Task Partitioning: Integrating Voronoi partitioning with sensor limitations, the region actually covered by an AUV is the intersection of its Voronoi cell and its sensing range, denoted as the sensing-limited Voronoi cell $V_i^{r_i}$:

$$V_i^{r_i}(p) = V_i(p) \cap C_i(p_i). \quad (13)$$

This partitioning method may result in certain regions within Q (e.g., $V_j \setminus C_j$) remaining uncovered by any AUV, as these regions lie within the Voronoi cell of agent j but fall outside its sensing range.

To address this issue, a modified Voronoi partition $\Omega_i(p)$ is adopted. This approach reallocates the uncovered regions $S_i = (\cup_{j \in \mathcal{N} \setminus \{i\}} (V_j \setminus V_j^{r_j})) \cap C_i$ to a neighbor i capable of sensing them. The final responsibility region Ω_i for AUV i is defined as

$$\Omega_i = V_i^{r_i} \cup S_i \setminus \left(\bigcup_{k \in \mathcal{N} - \{i\}} S_{ik}^* \right), \quad (14)$$

where S_{ik}^* represents the portion of the overlap between S_i and S_k that is closer to k . This modified partition Ω_i ensures that all sensible regions are rationally allocated.

To compute the local Voronoi cell V_i , each AUV i only requires the relative positions of neighboring AUVs $j \in \mathcal{N}_i$. Although the global domain boundary is assumed to be known a priori (as a stored map), the partition is computed in a fully distributed manner based on local interactions.

Workload Modeling: Based on this modified partition $\Omega_i(p)$, the task workload $T_i(p)$ of AUV i is defined as the integral of the information density within its responsibility region:

$$T_i(p) = \int_{\Omega_i(p)} \phi(q) dq. \quad (15)$$

To balance the workload among the heterogeneous AUVs, the workload-to-capability ratio $R_i(p)$ is defined as

$$R_i(p) = \frac{T_i(p)}{E_i}. \quad (16)$$

where $T_i(p)$ is the workload computed in Equation (15), and E_i represents the heterogeneous mission capability coefficient of the i -th AUV.

To explicitly quantify the heterogeneous capability index E_i from multi-dimensional physical attributes, we adopt a weighted linear aggregation method. The capability E_i is computed as

$$E_i = \kappa \left(\omega_v \frac{v_{max,i}}{\bar{v}_{base}} + \omega_s \frac{R_{s,i}}{\bar{R}_{base}} + \omega_e \frac{T_{end,i}}{\bar{T}_{base}} \right),$$

where $v_{max,i}$, $R_{s,i}$, and $T_{end,i}$ represent the maximum velocity, sensing radius, and battery endurance of the i -th AUV, respectively. The denominators $\bar{v}_{base}$, $\bar{R}_{base}$, $\bar{T}_{base}$ are the maximum values of these attributes across the entire swarm, used for normalization to ensure all terms are dimensionless and scale-consistent. The coefficients $\omega_v, \omega_s, \omega_e \geq 0$ are user-defined weights satisfying $\omega_v + \omega_s + \omega_e = 1$, and κ is a scaling factor.

The selection of these weighting coefficients dictates the sensitivity of the coverage strategy to specific attributes, allowing the framework to adapt to diverse mission requirements. For instance, in a time-critical rapid search mission, a higher value for ω_v can be assigned to prioritize agents with superior mobility. Conversely, for a detailed seabed mapping task requiring high-resolution data, ω_s would be dominant to leverage agents equipped with high-precision sonars. In the simulations presented in this paper, we assume a balanced configuration with equal weights to reflect a general-purpose scenario.

2.3.2. Consensus and Cost Function Lemma

To achieve workload balancing among heterogeneous AUVs (i.e., driving the ratios $R_i = T_i/E_i$ toward a consensus), a global cost function is required such that its minimum corresponds to the state where all R_i are consistent. The following lemma provides the mathematical basis for selecting the cost function $V_c = \sum (T_i^2/E_i)$.

Lemma 2 ([34]). Consider n variables $\{\delta_1, \delta_2, \dots, \delta_n\}$ satisfying the constraint that their sum is a fixed constant, $\sum_{i=1}^n \delta_i = \Delta$. For a set of positive real numbers $\{\beta_1, \beta_2, \dots, \beta_n\}$, consider the objective function $V = \sum_{i=1}^n \frac{\delta_i^2}{\beta_i}$. The minimum value of V is given by

$$\min(V) = \min\left(\sum_{i=1}^n \frac{\delta_i^2}{\beta_i}\right) = \frac{\Delta^2}{\sum_{i=1}^n \beta_i}. \quad (17)$$

This minimum is attained if and only if the ratios of all variables $\frac{\delta_i}{\beta_i}$ are equal, i.e.,

$$\frac{\delta_1}{\beta_1} = \frac{\delta_2}{\beta_2} = \dots = \frac{\delta_n}{\beta_n} = \frac{\Delta}{\sum_{i=1}^n \beta_i}. \quad (18)$$

Application to Coverage: In the context of the coverage problem, let the variable δ_i correspond to the task workload T_i of AUV i , and let the positive real number β_i correspond to its mission capability E_i . Assuming that the total task workload $\sum T_i = \int_Q \phi(q) dq = \Delta$ is a constant, the constructed cost function $V_c(p) = \sum_{i=1}^N \frac{T_i(p)^2}{E_i}$ achieves its minimum if and only if $\frac{T_i}{E_i} = \frac{T_j}{E_j}$ (i.e., $R_i = R_j$). Consequently, designing a controller to minimize $V_c(p)$ effectively drives the system to realize heterogeneous load balancing.

2.3.3. Control Objectives

The control objectives of this paper are classified into two aspects: the coverage objective and the safety objective.

- (a) **Coverage Objective:** Through the autonomous movement of the AUV swarm, the positions p are dynamically adjusted such that the task workload-to-capability ratios R_i tend toward consensus:

$$\lim_{t \rightarrow \infty} (R_i(p(t)) - R_j(p(t))) = 0, \quad \forall i, j \in \mathcal{N}. \quad (19)$$

This realization of heterogeneous load balancing—where capable agents assume greater workloads—constitutes the core of the optimal cooperative coverage control.

- (b) **Safety Objective:** System safety must be rigorously guaranteed throughout the execution of the coverage task. Leveraging CBF theory, the safety objective is formalized as maintaining the system state $x(t)$ consistently within a safe set C_{safe} . The set C_{safe} is defined as the intersection of multiple sub-safe sets:

$$C_{safe} = C_{agent} \cap C_{obs} \cap C_{vel}, \quad (20)$$

C_{agent} (Inter-AUV Safety Set): To prevent collisions between AUV i and j , a safety function considering predictive braking is adopted. Let d_i be the safety radius of AUV i , and the safe distance be $d_{safe,ij} = d_i + d_j$. The safety function $h_{ij}(x) \geq 0$ is defined as

$$h_{ij}(p, v) = 2\sqrt{u_{max}(\|\Delta p_{ij}\| - d_{safe,ij})} + \frac{\Delta p_{ij}^T \Delta v_{ij}}{\|\Delta p_{ij}\|} \geq 0, \quad \forall i \neq j, \quad (21)$$

where $\Delta p_{ij} = p_i - p_j$ and $\Delta v_{ij} = v_i - v_j$. This function acts as a constraint primarily when $\Delta p_{ij}^T \Delta v_{ij} \leq 0$, i.e., when the AUVs are approaching each other.

C_{obs} (Obstacle Safety Set): To avoid collision between AUV i and a static obstacle o located at p_o with radius d_o , the safe distance is defined as $d_{obs} = d_i + d_o$. The safety function $h_{io}(x) \geq 0$ is defined as

$$h_{io}(p) = \|p_i - p_o\| - d_{obs} \geq 0, \quad \forall o \in \mathcal{O}. \quad (22)$$

This constitutes a constraint with a relative degree of 2, necessitating the use of HOCBFs.

C_{vel} (Velocity Limit Set): To ensure the velocity of AUV i does not exceed v_{max} , the safety function $h_{v,i}(x) \geq 0$ is defined as

$$h_{v,i}(v) = v_{max}^2 - \|v_i\|^2 \geq 0, \quad \forall i \in \mathcal{N}. \quad (23)$$

This constitutes a constraint with a relative degree of 1.

In practical AUV systems, non-negligible stopping distances arise due to inertia and drag. A conservative enlargement of the safety distance can be used to implicitly account for such effects. Qualitatively, the safety buffer d_{safe} should be selected such that $d_{safe} > \frac{v_{max}^2}{2a_{brake}} + v_{max}\tau_{delay}$, where a_{brake} is the maximum braking deceleration capability of the physical AUV subject to hydrodynamic drag, and τ_{delay} is the system response latency.

In summary, the core problem addressed in this paper is the design of a distributed control law u_i such that, subject to the AUV dynamics (10) and physical constraints (11), the system state $x(t)$ consistently satisfies the safety constraints (20) while simultaneously achieving the coverage objective (19).

3. Design of Coverage Control Method

This section details the design of a cooperative coverage controller for heterogeneous AUVs that explicitly incorporates safety constraints. A hierarchical control framework called CBF-QP is adopted. As illustrated in Figure 1, this framework comprises two primary components: a nominal controller and a safety layer.

Nominal Controller ($\hat{u}_i$): Designed based on CLF concepts, this controller aims to achieve the coverage objective defined in Section 2.3.3—specifically, the consensus of the task workload-to-capability ratio R_i .

Safety Layer: Designed based on the CBF, this layer transforms all safety objectives defined in Section 2.3.3 (inter-AUV collision avoidance, obstacle avoidance, and velocity limits) into linear inequality constraints on the control input u_i .

The final control input u_i^* is obtained by solving a QP problem. This QP formulation aims to minimize the deviation from the nominal controller $\hat{u}_i$ while strictly satisfying all CBF safety constraints:

$$u_i^* = \arg \min_{u_i \in \mathbb{R}^2} \|u_i - \hat{u}_i\|^2 \quad \text{s.t.} \quad \text{CBF safety constraints}, \quad (24)$$

where u_i^* denotes the final applied safety control input, and $\hat{u}_i$ represents the nominal control input designed to achieve coverage performance. The term $\|\cdot\|^2$ is the squared Euclidean norm; minimizing this term implies that u_i^* is the control action closest to $\hat{u}_i$ subject to the satisfaction of safety constraints.

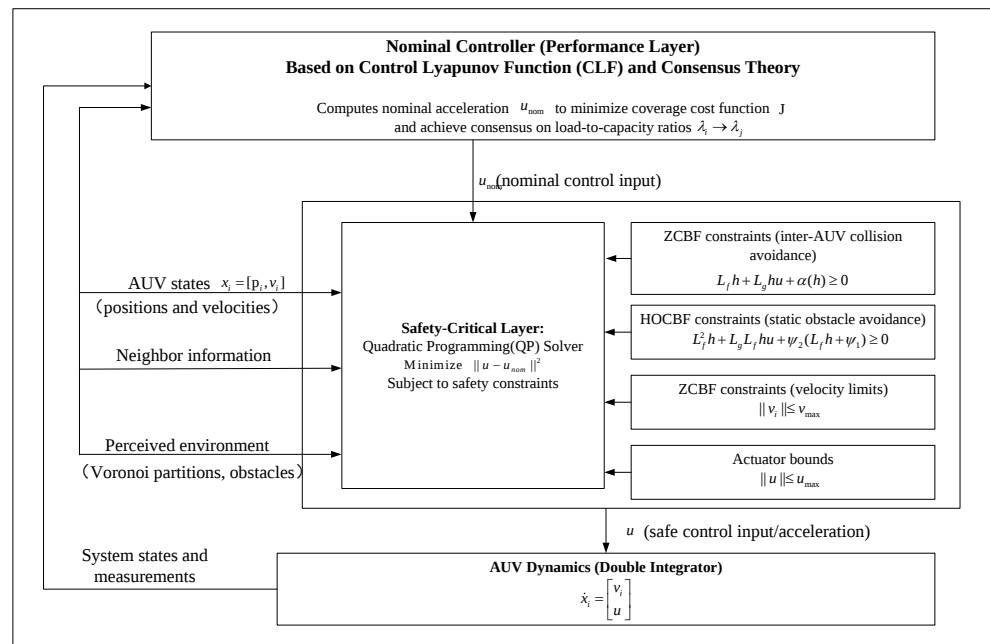


Figure 1. Hierarchical CBF-QP control framework for heterogeneous AUV cooperative coverage control.

3.1. Design of CLF-Based Nominal Controller $\hat{u}_i$

The primary objective of the nominal controller is to drive the system to converge to the coverage objective (19), specifically ensuring $R_i \rightarrow R_j$. This is realized by minimizing the global cost function $V_c(p)$.

3.1.1. Control Lyapunov Function for Coverage Cost

A Control Lyapunov Function, denoted as $V_{clf}(x)$, is constructed comprising two components: the potential energy $V_c(p)$ characterizing the workload balance and the total kinetic energy of the system $V_k(v)$.

$$V_{clf}(x) = V_c(p) + V_k(v) = \sum_{i=1}^N \frac{T_i(p)^2}{E_i} + \frac{1}{2} \sum_{i=1}^N v_i^T v_i, \quad (25)$$

where $x = [p_1^T, v_1^T, \dots, p_N^T, v_N^T]^T$ represents the state vector of the entire system; $T_i(p)$ denotes the task workload of AUV i ; E_i represents the mission capability of AUV i ; and v_i is the velocity of AUV i . Based on Lemma 2, $V_c(p)$ attains its minimum if and only if the ratios $R_i = T_i/E_i$ are consistent across all agents.

The control objective is to design u_i such that $\dot{V}_{clf} \leq 0$, thereby driving the system to converge to the minimum of V_{clf} . At this equilibrium, $v_i = 0$, and the workload-to-capability ratios R_i achieve consensus.

3.1.2. Nominal Controller

First, compute the time derivative of the Control Lyapunov Function $V_{clf}(x)$:

$$\dot{V}_{clf} = \frac{d}{dt} V_c(p) + \frac{d}{dt} V_k(v) = \sum_{i=1}^N \left(\frac{\partial V_c}{\partial p_i} \dot{p}_i + \frac{\partial V_k}{\partial v_i} \dot{v}_i \right). \quad (26)$$

By substituting the AUV dynamic model (10) and noting that $\frac{\partial V_k}{\partial v_i} = v_i$, Equation (26) can be rewritten as

$$\dot{V}_{clf} = \sum_{i=1}^N \left((\nabla_{p_i} V_c)^T v_i + v_i^T u_i \right) = \sum_{i=1}^N v_i^T (\nabla_{p_i} V_c + u_i), \quad (27)$$

where $\nabla_{p_i} V_c$ denotes the gradient of the coverage cost with respect to the position of AUV i .

To render $\dot{V}_{clf}$ negative semi-definite, the nominal controller $\hat{u}_i$ is designed in the form of a proportional–derivative (PD) controller:

$$\hat{u}_i = -k_p(\nabla_{p_i} V_c) - k_d v_i, \quad (28)$$

where $k_p > 0$ and $k_d > 0$ are control gains, corresponding to the proportional and derivative terms, respectively. For the sake of algebraic simplicity and without loss of generality, we set $k_p = 1$ in the following derivation (noting that for any arbitrary $k_p > 0$, the stability analysis follows identically by scaling the potential energy term in the Lyapunov candidate). Substituting Equation (28) into Equation (27) yields

$$\dot{V}_{clf} = \sum_{i=1}^N v_i^T (\nabla_{p_i} V_c + (-\nabla_{p_i} V_c - k_d v_i)) = -k_d \sum_{i=1}^N \|v_i\|^2 \leq 0. \quad (29)$$

This result demonstrates that the nominal controller $\hat{u}_i$ ensures the decay of V_{clf} , driving the system velocities $v_i \rightarrow 0$. Furthermore, according to LaSalle's Invariance Principle [35], the system converges to the largest invariant set where $\dot{V}_{clf} = 0$. This corresponds to the state where $v_i = 0$ and $\nabla_{p_i} V_c = 0$, which is precisely the coverage objective.

3.1.3. Coverage Cost Gradient $\nabla_{p_i} V_c$

The core of the control law (28) lies in the computation of the gradient of $V_c(p)$ with respect to p_i , denoted as $\nabla_{p_i} V_c$.

$$\nabla_{p_i} V_c = \frac{\partial V_c}{\partial p_i} = \frac{\partial}{\partial p_i} \sum_{k=1}^N \frac{T_k(p)^2}{E_k} = \sum_{k=1}^N \frac{2T_k(p)}{E_k} \frac{\partial T_k(p)}{\partial p_i}. \quad (30)$$

Based on Equation (16), where $R_k(p) = T_k(p)/E_k$, the above expression can be rewritten as

$$\nabla_{p_i} V_c = 2 \sum_{k=1}^N R_k(p) \frac{\partial T_k(p)}{\partial p_i}. \quad (31)$$

Since the task workload $T_k(p) = \int_{\Omega_k(p)} \phi(q) dq$ for AUV k is defined over the responsibility region $\Omega_k(p)$, the boundary of $\Omega_k(p)$ depends solely on the position p_k and the positions of its neighbors p_j (where $j \in \mathcal{N}_i$, and $\mathcal{N}_i$ denotes the set of neighbors of i). Consequently, for any $k \notin \{i\} \cup \mathcal{N}_i$, the derivative vanishes, i.e., $\frac{\partial T_k}{\partial p_i} = 0$. Equation (31) can thus be simplified to

$$\nabla_{p_i} V_c = 2R_i \frac{\partial T_i}{\partial p_i} + 2 \sum_{j \in \mathcal{N}_i} R_j \frac{\partial T_j}{\partial p_i}. \quad (32)$$

To compute $\frac{\partial T_k}{\partial p_i}$, the Leibniz integral rule [36] is applied. Since the density function $\phi(q)$ is independent of the AUV position p_i (i.e., $\frac{\partial \phi(q)}{\partial p_i} = 0$), only the boundary terms need to be considered:

$$\frac{\partial T_k(p)}{\partial p_i} = \frac{\partial}{\partial p_i} \int_{\Omega_k(p)} \phi(q) dq = \int_{\partial \Omega_k} \phi(q) n_k^T(q) \frac{\partial q}{\partial p_i} ds, \quad (33)$$

where $\partial \Omega_k$ denotes the boundary of Ω_k , and $n_k(q)$ represents the outward unit normal vector at point q on the boundary.

- (a) Case $k = i$: The boundary $\partial\Omega_i$ consists of two parts: the boundary $\partial\Omega_{ij}$ shared with neighbor j , and the sensing boundary $\partial\Omega_i^s$ (the arc where $\|q - p_i\| = r_i$). Thus,

$$\frac{\partial T_i}{\partial p_i} = \sum_{j \in \mathcal{N}_i} \int_{\partial\Omega_{ij}} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds + \int_{\partial\Omega_i^s} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds. \quad (34)$$

- (b) Case $k = j$ ($j \in \mathcal{N}_i$): For a neighbor j , only the portion of the boundary $\partial\Omega_j$ that is shared with i (i.e., $\partial\Omega_{ji} = \partial\Omega_{ij}$) depends on p_i . Thus,

$$\frac{\partial T_j}{\partial p_i} = \int_{\partial\Omega_{ji}} \phi(q) n_j^T(q) \frac{\partial q}{\partial p_i} ds. \quad (35)$$

Note that on the shared boundary, the normal vectors are opposite, i.e., $n_j(q) = -n_i(q)$. Substituting (34) and (35) into the gradient formula (32) yields

$$\begin{aligned} \nabla_{p_i} V_c = & 2R_i \left(\sum_{j \in \mathcal{N}_i} \int_{\partial\Omega_{ij}} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds + \int_{\partial\Omega_i^s} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds \right) \\ & + 2 \sum_{j \in \mathcal{N}_i} R_j \left(\int_{\partial\Omega_{ij}} \phi(q) (-n_i^T(q)) \frac{\partial q}{\partial p_i} ds \right), \end{aligned} \quad (36)$$

where $\nabla_{p_i} V_c$ is the gradient of the global cost function V_c with respect to the position p_i of the AUV i . R_i and R_j are the task load-capability ratios of AUV i and its neighbor j , respectively, and $R_i = T_i/E_i$. $\mathcal{N}_i$ is the neighbor set of AUV i . $\partial\Omega_{ij}$ is the shared Voronoi boundary between AUV i and j . $\partial\Omega_i^s$ is the sensing boundary of AUV i that is not shared with any neighbors (i.e., the part of $\|q - p_i\| = r_i$). $\phi(q)$ is the task density at point q . $n_i^T(q)$ is the transpose of the outward normal vector of the boundary $n_i(q)$ at point q . $\frac{\partial q}{\partial p_i}$ is the sensitivity of the points q on the boundary to the movement of p_i . ds denotes the line integral along the boundary.

By rearranging the terms associated with the shared boundaries, the following is obtained:

$$\nabla_{p_i} V_c = 2 \sum_{j \in \mathcal{N}_i} (R_i - R_j) \left[\int_{\partial\Omega_{ij}} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds \right] + 2R_i \left[\int_{\partial\Omega_i^s} \phi(q) n_i^T(q) \frac{\partial q}{\partial p_i} ds \right]. \quad (37)$$

The gradient $\nabla_{p_i} V_c$ possesses distinct physical interpretations:

- (a) Consensus Term (First Term): This term represents the weighted sum of the differences between R_i and R_j . The term $(R_i - R_j)$ indicates that if the workload ratio of AUV i exceeds that of its neighbor j , the gradient generates a force driving p_i to move—typically away from j —to reduce T_i and increase T_j , thereby driving R_i and R_j toward consensus.
- (b) (Second Term): This term reflects the task density along the sensing boundary $\partial\Omega_i^s$ of AUV i . If the density $\phi(q)$ at the boundary is high, this term generates a force pushing AUV i toward the high-density region to expand coverage.

Consequently, the nominal controller $\hat{u}_i$ in Equation (28) is explicitly expressed as

$$\hat{u}_i = -k_p \left(2 \sum_{j \in \mathcal{N}_i} (R_i - R_j) \mathbf{F}_{ij}(p) + 2R_i \mathbf{F}_{is}(p_i) \right) - k_d v_i, \quad (38)$$

where $\mathbf{F}_{ij}(p)$ and $\mathbf{F}_{is}(p_i)$ represent the two boundary integral vectors defined in (37), respectively.

To enable the distributed computation of Equation (38) and the Voronoi partition, each AUV i is required to broadcast a local state packet to its neighbors $j \in \mathcal{N}_i$ at each control cycle. The exchanged information set includes $\{p_i, v_i, E_i, T_i\}$, where p_i, v_i are for collision avoidance and Voronoi partitioning, and E_i, T_i (or the ratio $R_i = T_i/E_i$) are for the consensus controller.

3.2. Design of Safety Constraints

(1) Constraint 1: Inter-AUV Collision Avoidance

Utilizing the Zeroing Control Barrier Function $h_{ij}(x)$ defined in (21),

$$h_{ij}(p, v) = 2\sqrt{u_{\max}(\|\Delta p_{ij}\| - d_{\text{safe},ij})} + \frac{\Delta p_{ij}^T \Delta v_{ij}}{\|\Delta p_{ij}\|} \geq 0, \quad (39)$$

where $h_{ij}(p, v)$ is the safety function between AUV i and j . $\Delta p_{ij} = p_i - p_j$ denotes the relative position, and $\Delta v_{ij} = v_i - v_j$ denotes the relative velocity. $u_{\max}$ is the maximum acceleration, and $d_{\text{safe},ij}$ is the minimum safety distance.

This function possesses a relative degree of $r = 1$. According to Theorem 1, the safety constraint is formulated as $\dot{h}_{ij} + \lambda_1 h_{ij} \geq 0$, where $\lambda_1 > 0$ is a tunable parameter. The complete form of $\dot{h}_{ij}$ is given by

$$\dot{h}_{ij} = L_f h_{ij} + L_g h_{ij} \Delta u_{ij} = L_f h_{ij} + L_g h_{ij} (u_i - u_j), \quad (40)$$

where $\Delta u_{ij} = u_i - u_j$. The Lie derivatives are defined as $L_f h_{ij} = (\nabla_p h_{ij})^T \dot{p} + (\nabla_v h_{ij})^T \dot{v}$ and $L_g h_{ij} = (\nabla_v h_{ij})^T g(x) = \frac{\Delta p_{ij}^T}{\|\Delta p_{ij}\|}$.

Consequently, the safety constraint transforms into

$$L_f h_{ij} + \frac{\Delta p_{ij}^T}{\|\Delta p_{ij}\|} (u_i - u_j) + \lambda_1 h_{ij} \geq 0. \quad (41)$$

Remark 1. (Parameter Tuning for λ_1). The parameter $\lambda_1 > 0$ in the ZCBF constraint serves as a stiffness gain that dictates how aggressively the AUV is allowed to approach the safety boundary. A larger λ_1 permits the system to operate closer to the neighbor before intervening, resulting in less conservative behavior but demanding higher control authority (acceleration) to brake or turn sharply. Conversely, a smaller λ_1 enforces a more conservative safety margin, triggering avoidance maneuvers earlier but with smoother control inputs. In practical implementation, λ_1 should be tuned empirically to maximize maneuverability while strictly ensuring that the required safety control input does not exceed the actuator's physical saturation limit.

To decouple the control inputs u_i and u_j within the QP formulation, a symmetrical Responsibility Allocation strategy is adopted. This distributes the safety constraint burden equally between agents i and j :

$$\frac{\Delta p_{ij}^T}{\|\Delta p_{ij}\|} u_i \geq -\frac{1}{2} (L_f h_{ij} + \lambda_1 h_{ij}). \quad (42)$$

It is noted that the current formulation primarily addresses static obstacles. For dynamic non-cooperative obstacles, the constraint (42) can be adapted by estimating the obstacle's velocity vector and treating it as a time-varying boundary constraint.

The inter-AUV collision avoidance constraint is thus formally defined as a linear inequality:

$$\mathbf{A}_{ij} u_i \geq \mathbf{b}_{ij}(x), \quad (43)$$

where the coefficient matrix $\mathbf{A}_{ij}$ and the boundary vector $\mathbf{b}_{ij}(x)$ are defined as

$$\mathbf{A}_{ij} = \frac{\Delta p_{ij}^T}{\|\Delta p_{ij}\|}, \quad \mathbf{b}_{ij}(x) = -\frac{1}{2}(L_f h_{ij}(p, v) + \lambda_1 h_{ij}(p, v))$$

and the Lie derivative $L_f h_{ij}$ is explicitly calculated as

$$L_f h_{ij} = \frac{u_{max} \Delta p_{ij}^T \Delta v_{ij}}{\sqrt{u_{max}(\|\Delta p_{ij}\| - d_{safe,ij})\|\Delta p_{ij}\|}} + \frac{\|\Delta v_{ij}\|^2}{\|\Delta p_{ij}\|} - \frac{(\Delta p_{ij}^T \Delta v_{ij})^2}{\|\Delta p_{ij}\|^3}$$

(2) Constraint 2: Static Obstacle Avoidance

Using the function $h_{io}(p) = \|p_i - p_o\| - d_{obs} \geq 0$ defined in Equation (22), this function has a relative degree of $r = 2$ with respect to u_i (since $\dot{h}_{io}$ depends only on v_i , and $\ddot{h}_{io}$ depends on u_i). Based on the HOCBF definition in Definition 5, define $\psi_0(x) = h_{io}(x)$ and $\psi_1(x)$ as

$$\psi_1(x) = \dot{\psi}_0(x) + k_1 \psi_0(x) = L_f h_{io}(x) + k_1 h_{io}(x)$$

The HOCBF safety constraint is then given by $\psi_1(x) + k_2 \psi_1(x) \geq 0$, where $k_1, k_2 > 0$. Then,

$$\dot{\psi}_1 = \frac{\partial \psi_1}{\partial x} \dot{x} = L_f \psi_1 + L_g \psi_1 u_i. \quad (44)$$

Expanding $L_f \psi_1$ and $L_g \psi_1$ yields

$$\begin{cases} L_f \psi_1 = L_f(L_f h_{io} + k_1 h_{io}) = L_f^2 h_{io} + k_1 L_f h_{io} \\ L_g \psi_1 = L_g(L_f h_{io} + k_1 h_{io}) = L_g L_f h_{io} + k_1 L_g h_{io} \end{cases} \quad (45)$$

Since h_{io} is solely a function of p , $L_g h_{io} = \frac{\partial h_{io}}{\partial v_i} = 0$. Substituting (45) into (44), the constraint becomes

$$(L_f^2 h_{io} + k_1 L_f h_{io}) + (L_g L_f h_{io}) u_i + k_2 (L_f h_{io} + k_1 h_{io}) \geq 0, \quad (46)$$

where the terms are explicitly calculated as

$$\begin{aligned} L_f h_{io} &= (\nabla_{p_i} h_{io})^T v_i = \frac{(p_i - p_o)^T v_i}{\|p_i - p_o\|}, \\ \mathbf{A}_{io} &= L_g L_f h_{io} = (\nabla_{v_i} (L_f h_{io}))^T = \frac{(p_i - p_o)^T}{\|p_i - p_o\|}, \\ L_f^2 h_{io} &= (\nabla_{p_i} (L_f h_{io}))^T v_i = \frac{\|v_i\|^2}{\|p_i - p_o\|} - \frac{((p_i - p_o)^T v_i)^2}{\|p_i - p_o\|^3}, \\ \mathbf{b}_{io}(x) &= -(L_f^2 h_{io} + (k_1 + k_2) L_f h_{io} + k_1 k_2 h_{io}). \end{aligned}$$

(3) Constraint 3: Velocity Limit

Using the function $h_{v,i}(v) = v_{max}^2 - \|v_i\|^2 \geq 0$ defined in Equation (22), this function has a relative degree of $r = 1$. The ZCBF constraint is given by $\dot{h}_{v,i} + \lambda_3 h_{v,i} \geq 0$. Thus,

$$\dot{h}_{v,i} = L_f h_{v,i} + L_g h_{v,i} u_i. \quad (47)$$

Since $h_{v,i}$ depends solely on v , $L_f h_{v,i} = \frac{\partial h_{v,i}}{\partial p_i} v_i = 0$, and $L_g h_{v,i} = \frac{\partial h_{v,i}}{\partial v_i} = -2v_i^T$. The constraint transforms into

$$-2v_i^T u_i + \lambda_3(v_{max}^2 - \|v_i\|^2) \geq 0. \quad (48)$$

The velocity limit constraint is formally defined as a linear inequality:

$$\mathbf{A}_{v,i} u_i \leq \mathbf{b}_{v,i}(x), \quad (49)$$

where $\mathbf{A}_{v,i} = 2v_i^T$ and $\mathbf{b}_{v,i}(x) = \lambda_3(v_{max}^2 - \|v_i\|^2)$.

The nominal controller $\hat{u}_i$ from (38) and all safety constraints (43), (47), and (50) are combined into a single Quadratic Programming problem. Simultaneously, the acceleration limits from Equation (11) are incorporated as boundary constraints for the QP.

To guarantee the feasibility of the optimization problem under all conditions and to explicitly define the constraint activation logic, we reformulate the QP with a relaxation (slack) variable δ . The revised optimization problem for each AUV i is formulated as

$$\begin{aligned} u_i^*, \delta^* = \arg \min_{u_i, \delta} \quad & \|u_i - \hat{u}_i\|^2 + p \cdot \delta^2 \\ \text{s.t.} \quad & A_{ij} u_i \geq b_{ij}(x) - \delta, \quad \forall j \in \mathcal{N}_i^{\text{active}} \\ & A_{io} u_i \geq b_{io}(x) - \delta, \quad \forall o \in \mathcal{O}_i^{\text{active}} \\ & A_{v,i} u_i \leq b_{v,i}(x) \\ & \|u_i\| \leq u_{max} \end{aligned} \quad (50)$$

where $p > 0$ is a sufficiently large penalty coefficient prioritizing safety, and u_{max} represents the physical actuator saturation limit. Here, u_i^* denotes the optimal control input, and δ^* represents the optimal value of the slack variable, which quantifies the minimum necessary violation of safety constraints. The scalar slack variable $\delta \geq 0$ transforms the original hard safety constraints into soft constraints. This formulation provides a sufficient condition for feasibility: For any system state x , there implies the existence of a sufficiently large δ such that the feasible set of (50) is non-empty. Consequently, in extreme scenarios where strictly satisfying all safety constraints is physically impossible (e.g., deadlock), the solver yields a non-zero δ^* as an automatic relaxation strategy, finding a solution that minimizes the safety violation to maintain control continuity rather than failing.

The active sets are determined by the AUV's local sensing capability. The active neighbor set is defined as $\mathcal{N}_i^{\text{active}} = \{j \in \mathcal{N} \mid \|p_i - p_j\| \leq R_s\}$ and the active obstacle set as $\mathcal{O}_i^{\text{active}} = \{o \in \mathcal{O} \mid \|p_i - p_o\| \leq R_s\}$, where R_s is the AUV's maximum sensing radius defined in Equation (12). To prevent high-frequency chattering when obstacles are near the sensing boundary, a hysteresis mechanism is applied: a constraint is deactivated only when the distance exceeds $R_s + \epsilon_{hys}$, where $\epsilon_{hys} > 0$ is a small user-defined distance buffer constant (e.g., $\epsilon_{hys} = 0.5 \text{ m}$).

This QP formulation constitutes a convex optimization problem, which allows for efficient online solution. The resulting u_i^* represents the final control command applied to AUV i , which guarantees safety while maximally approximating the nominal control objective for coverage.

4. System Stability Analysis

This section aims to provide a rigorous mathematical proof of the two fundamental properties characterizing the CBF-QP control framework (51) proposed in Section 3: system safety and coverage convergence.

(1) Safety Analysis: Under the effect of the proposed QP controller u_i^* , the system state $x(t)$ remains invariant within the safety set C_{safe} defined in Equation (20). Specifically, the AUVs are guaranteed to avoid collisions with other agents and obstacles and to strictly adhere to velocity limits at all times.

(2) Convergence Analysis: Subject to the safety guarantees, the system state $x(t)$ asymptotically converges to a stable equilibrium. At this equilibrium, the AUV velocities vanish (i.e., $v_i = 0$), and the heterogeneous coverage cost function $V_c(p)$ attains a local minimum.

4.1. Safety Analysis

System safety is realized by ensuring the forward invariance of the safety set C_{safe} .

Theorem 3. (System Safety) For the AUV swarm system, consider the safety set C_{safe} defined in Equation (20). If the initial system state $x(0) \in C_{safe}$, then the closed-loop system formed by the QP controller u_i^* guarantees that the safe set C_{safe} is forward-invariant, provided that the optimal slack variable satisfies $\delta^*(t) = 0$ for all $t \geq 0$. In extreme scenarios where strictly satisfying all constraints is physically impossible ($\delta^* > 0$), the controller minimizes the violation of the safety boundary to maintain operational continuity.

Proof of Theorem 3. According to Equation (20), the safety set C_{safe} is the intersection of three subsets: $C_{safe} = C_{agent} \cap C_{obs} \cap C_{vel}$. By set theory, to prove that C_{safe} is forward-invariant, it suffices to prove that each of the three sets— C_{agent} , C_{obs} , and C_{vel} —is individually forward-invariant.

(a) Forward Invariance of C_{agent} (Inter-AUV Collision Avoidance):

The set C_{agent} is characterized by the safety function $h_{ij}(x) \geq 0$, where

$$h_{ij}(p, v) = 2\sqrt{u_{max}(\|\Delta p_{ij}\| - d_{safe,ij})} + \frac{\Delta p_{ij}^T \Delta v_{ij}}{\|\Delta p_{ij}\|} \geq 0$$

This constitutes a safety function with a relative degree of $r = 1$; therefore, the ZCBF theory is employed for the proof.

According to Theorem 1, to guarantee the forward invariance of the set defined by $h_{ij} \geq 0$, the controller u_i must satisfy the condition $\dot{h}_{ij} + \lambda_1 h_{ij} \geq 0$ (choosing the class $\mathcal{K}$ function as $\alpha(h) = \lambda_1 h$). In Equation (41), it was derived that this condition is equivalent to $L_f h_{ij} + L_g h_{ij}(u_i - u_j) + \lambda_1 h_{ij} \geq 0$. Through the symmetric allocation strategy in Equation (42), the constraint on u_i was formulated as $\mathbf{A}_{ij} u_i \geq \mathbf{b}_{ij}(x)$. The feasible set $K(x)$ of the QP controller u_i^* in Equation (51) is defined such that it must satisfy Equation (43). Consequently, the optimal solution u_i^* necessarily satisfies the ZCBF condition. By Theorem 1, u_i^* guarantees that C_{agent} is forward-invariant.

(b) Forward Invariance of C_{vel} (Velocity Limit):

The set C_{vel} is characterized by the safety function $h_{v,i}(x) \geq 0$, defined as

$$h_{v,i}(v) = v_{max}^2 - \|v_i\|^2 \geq 0$$

This is similarly a safety function with a relative degree of $r = 1$, necessitating the use of ZCBF theory.

To guarantee the forward invariance of $h_{v,i} \geq 0$, the controller u_i must satisfy the condition $\dot{h}_{v,i} + \lambda_3 h_{v,i} \geq 0$. In Equation (49), this condition was derived to be equivalent to $L_f h_{v,i} + L_g h_{v,i} u_i + \lambda_3 h_{v,i} \geq 0$, which corresponds to the linear inequality $\mathbf{A}_{v,i} u_i \leq \mathbf{b}_{v,i}(x)$. The feasible set $K(x)$ of the QP controller u_i^* is similarly defined such

that it must strictly satisfy Equation (50). Consequently, u_i^* guarantees that C_{vel} is forward-invariant.

(c) Forward Invariance of C_{obs} (Obstacle Avoidance):

The set C_{obs} is characterized by the safety function $h_{io}(x) \geq 0$, where

$$h_{io}(p) = \|p_i - p_o\| - d_{obs} \geq 0$$

This function has a relative degree of $r = 2$, necessitating the use of HOCBF theory (Definition 5).

To guarantee the forward invariance of $h_{io} \geq 0$, we define $\psi_0 = h_{io}$ and $\psi_1 = \dot{\psi}_0 + k_1\psi_0$ and require the controller u_i to satisfy

$$L_f\psi_1 + L_g\psi_1 u_i + k_2\psi_1 \geq 0$$

In Equation (46), it was proven that this HOCBF condition is equivalent to the QP constraint $\mathbf{A}_{io}u_i \geq \mathbf{b}_{io}(x)$. The feasible set $K(x)$ of the QP controller u_i^* is defined such that it must satisfy Equation (47). Consequently, u_i^* guarantees that C_{obs} is forward-invariant.

In summary, the QP controller u_i^* is formulated to optimize the control input subject to the safety constraints characterizing C_{agent} , C_{obs} , and C_{vel} . Consequently, under the condition that the hard constraints are feasible (i.e., the optimal slack variable satisfies $\delta^* = 0$), the control input strictly satisfies all barrier function inequalities, thereby guaranteeing that the intersection set C_{safe} is forward-invariant. In extreme scenarios where $\delta^* > 0$, the strictly forward-invariant property is locally relaxed, and the controller operates to minimize the violation of the safety boundaries to maintain system solvability and operational continuity.

The proof is complete. $\square$

4.2. Convergence Analysis

In Section 4.1, it was proven that the system remains safe under the effect of the QP controller u_i^* , meaning that C_{safe} is forward-invariant. Building on this, this section further analyzes the system's convergence, proving that while maintaining safety, the system converges to the desired coverage objective (i.e., $R_i \rightarrow R_j$).

LaSalle's Invariance Principle [35] is employed to analyze system convergence. The Control Lyapunov Function $V_{clf}(x)$ defined in Section 3.1 is selected as the Lyapunov candidate function for this analysis:

$$V_{clf}(x) = V_c(p) + V_k(v) = \sum_{i=1}^N \frac{T_i(p)^2}{E_i} + \frac{1}{2} \sum_{i=1}^N v_i^T v_i. \quad (51)$$

According to Theorem 3, the system trajectory $x(t)$ is confined within a compact set $\Omega = \{x \in C_{safe} \mid V_{clf}(x) \leq V_{clf}(x(0))\}$, and $V_{clf}(x) \geq 0$ holds true.

Theorem 4. (System Convergence) Under the conditions of Theorem 3, and employing the random perturbation mechanism (Remark 2) to escape non-optimal local minima (deadlocks), the closed-loop system formed by the QP controller u_i^* in Equation (51) will converge to the largest invariant set M . This set M is contained within the set S defined as

$$S = \{x \in \Omega \mid v_i = 0, \forall i \in \mathcal{N}\}. \quad (52)$$

This implies that the system will ultimately converge to a static state where all AUV velocities are zero.

Proof of Theorem 4. To apply LaSalle's Principle, first compute the time derivative $\dot{V}_{clf}$ of the Lyapunov function (52) under the effect of the actual QP controller u_i^* . Based on the derivation in Equation (27), we have

$$\dot{V}_{clf} = \sum_{i=1}^N v_i^T (\nabla_{p_i} V_c + u_i^*). \quad (53)$$

To simplify the derivation, following the settings in Equations (28) and (29), let the proportional gain of the nominal controller be $k_p = 1$.

Express the actual controller u_i^* as $u_i^* = \hat{u}_i + (u_i^* - \hat{u}_i)$, where $\delta_u = u_i^* - \hat{u}_i$ represents the correction term generated by the QP to satisfy safety constraints. Substituting this into (54),

$$\dot{V}_{clf} = \sum_{i=1}^N v_i^T (\nabla_{p_i} V_c + \hat{u}_i + (u_i^* - \hat{u}_i)). \quad (54)$$

Now, substitute the nominal controller $\hat{u}_i = -\nabla_{p_i} V_c - k_d v_i$ into Equation (55):

$$\dot{V}_{clf} = \sum_{i=1}^N v_i^T (\nabla_{p_i} V_c + (-\nabla_{p_i} V_c - k_d v_i) + (u_i^* - \hat{u}_i)). \quad (55)$$

Simplifying yields

$$\dot{V}_{clf} = \sum_{i=1}^N v_i^T (-k_d v_i + (u_i^* - \hat{u}_i)) = \underbrace{-k_d \sum_{i=1}^N \|v_i\|^2}_{\text{Nominal Damping}} + \underbrace{\sum_{i=1}^N v_i^T (u_i^* - \hat{u}_i)}_{\text{CBF Perturbation}}. \quad (56)$$

Equation (57) indicates that $\dot{V}_{clf}$ is composed of two parts. As long as $v_i \neq 0$, the first term is negative semi-definite, representing the desired damping term. The second term is the CBF perturbation term, whose sign is indefinite. When safety constraints are active (i.e., $u_i^* \neq \hat{u}_i$), these constraints may compromise the descent of the CLF, potentially causing $\dot{V}_{clf}$ to be temporarily non-negative.

Although $\dot{V}_{clf}$ is not a strictly negative semi-definite function, since the system trajectory $x(t)$ is proven to be confined within the compact set Ω , LaSalle's Invariance Principle dictates that the system trajectory $x(t)$ must converge to the largest invariant set M contained within Ω . This set M is a subset of the set S where $\dot{V}_{clf} = 0$.

$$S = \{x \in \Omega \mid \dot{V}_{clf} = 0\} = \{x \in \Omega \mid -k_d \sum \|v_i\|^2 + \sum v_i^T (u_i^* - \hat{u}_i) = 0\}. \quad (57)$$

If $v_i = 0$ (for all i), then $\dot{V}_{clf} = 0 + 0 = 0$. Consequently, all static states where the system velocities are zero are included in the set S .

$$\{x \in \Omega \mid v_i = 0, \forall i\} \subseteq S. \quad (58)$$

The largest invariant set M to which the system converges must satisfy $M \subseteq S$. According to the definition of an invariant set, any trajectory starting from within M must remain in M forever. For a trajectory to remain in M permanently, it must consistently satisfy $\dot{V}_{clf} = 0$. Based on the analysis of Equation (59), this requires that the trajectory must satisfy $v_i(t) \equiv 0$ identically.

If $v_i(t) \equiv 0$ holds identically, then its derivative $\dot{v}_i(t) = 0$ must also hold. According to the system dynamics in Equation (10), where $\dot{v}_i = u_i^*$, this implies that all states within the largest invariant set M must satisfy

$$M \subseteq \{x \in \Omega \mid v_i = 0, \dot{v}_i = u_i^* = 0, \forall i\}. \quad (59)$$

The proof is complete. $\square$

The static equilibrium point to which the system converges must satisfy $v_i = 0$ and $u_i^* = 0$. We now analyze the condition under which $u_i^* = 0$. When $v_i = 0$, the nominal controller simplifies to $\hat{u}_i = -k_p \nabla_{p_i} V_c$. Consequently, the QP problem transforms into

$$u_i^* = \arg \min_{u_i \in K(x|_{v=0})} \|u_i - (-k_p \nabla_{p_i} V_c)\|^2, \quad (60)$$

where $K(x|_{v=0})$ denotes the safe feasible set at the instant when $v_i = 0$. The condition $u_i^* = 0$ being the solution to this QP occurs in two scenarios:

- (a) **Ideal Equilibrium:** In this case, the safety constraints are inactive or the nominal control naturally satisfies them. The system stops because the coverage objective is achieved, i.e., $\nabla_{p_i} V_c = 0$, leading to $\hat{u}_i = 0$. This corresponds to the global or local minimum of the coverage cost function where workload consensus is reached.
- (b) **Constrained Equilibrium (Deadlock):** In this case, the safety constraints are active. The system stops not because the task is complete ($\nabla_{p_i} V_c \neq 0$) but because the nominal control input $\hat{u}_i$ is exactly strictly opposed by the safety correction term required to maintain the forward invariance of C_{safe} . Mathematically, this occurs when the negative gradient direction aligns perfectly with the normal vector of the safety boundary, creating a local minimum where $u_i^* = 0$ despite $\hat{u}_i \neq 0$.

With the activation of the perturbation mechanism described in Remark 2, the system is capable of escaping these unstable equilibria. Consequently, the system will ultimately converge to the ideal equilibrium (Case a), where the following conditions are satisfied:

$$v_i = 0 \quad \text{and} \quad \nabla_{p_i} V_c = 0, \quad \forall i \in \mathcal{N}. \quad (61)$$

According to Lemma 2, $\nabla_{p_i} V_c = 0$ is the necessary condition for the coverage cost function $V_c(p) = \sum \frac{T_i(p)^2}{E_i}$ to attain a (local) minimum. This condition is satisfied if and only if the workload-to-capability ratios $R_i(p) = T_i/E_i$ achieve consensus across all AUVs. Consequently, the system is proven to converge to a static state with zero velocity, simultaneously realizing the optimal coverage state characterized by heterogeneous task workload balance.

Remark 2. While (Case b) represents a theoretical local minimum (deadlock) inherent to gradient-based methods in cluttered environments, it is often an unstable equilibrium. To ensure the system converges to the Ideal Equilibrium (Case a), a practical random perturbation mechanism is implemented. If the system detects a static state ($\|v_i\| < \epsilon_v$) while the coverage gradient is non-zero ($\|\nabla_{p_i} V_c\| > \epsilon_g$), a small random perturbation vector δ_{rand} is temporarily added to the nominal controller $\hat{u}_i$. This breaks the symmetry of the force balance, allowing the agent to slide along the safety boundary and escape the deadlock.

5. Simulation Experiments and Analysis

Numerical simulations were conducted to verify the effectiveness of the heterogeneous AUV cooperative coverage control framework proposed in this paper, which integrates CBFs with consensus theory. The simulation experiments were performed on a device equipped with an Intel Core i7-8750H processor (2.6 GHz), running the Windows 11 64-bit operating system. The algorithms were implemented using MATLAB R2023a. Two sets of comparative experiments were designed to separately verify the two core performance indicators of the control framework:

- (a) Coverage Performance Verification: In an obstacle-free environment, this experiment validates whether the nominal controller can drive the heterogeneous AUV swarm to achieve consensus convergence of the task workload-to-capability ratio ($R_i \rightarrow R_j$).
- (b) Safety Verification: In an environment containing static obstacles, this experiment validates whether the CBF-QP-based controller can strictly satisfy collision avoidance and state constraints while performing the coverage mission, thereby ensuring system safety.

The simulation mission area $\mathcal{D}$ is defined as a 5000 m $\times$ 5000 m two-dimensional rectangular sea region. The information density function (task weight) within the region is modeled as a Mixture of Gaussians [37] to simulate the distribution of regions of interest in a vast underwater environment:

$$\phi(q) = \exp(-10^{-5}\|q - [2500, 4000]^T\|^2) + \exp(-10^{-5}\|q - [1200, 1200]^T\|^2) + \exp(-10^{-5}\|q - [3800, 1200]^T\|^2) + 0.3$$

This density function exhibits three high-value targets located at (2500, 4000), (1200, 1200), and (3800, 1200).

Consider a swarm composed of $N = 5$ heterogeneous AUVs, whose initial positions are randomly distributed within an error range of $(-300, 300)$ around the region's center coordinates (2500, 2500). To reflect heterogeneity, the comprehensive capability parameters E_i are set to be different for each AUV. The sensing radius R_s is set to 600 m, and the AUV safety distance is set to 50 m. Specific parameters and controller gains are listed in Table 1 below.

The core optimization QP problem (50) was solved using the MATLAB quadprog solver with the interior-point-convex algorithm. To ensure numerical stability and strict reproducibility, the discrete control update interval was set to $\Delta t = 0.1$ s, and the spatial domain was discretized with a grid resolution of 1×1 m for Voronoi mass computation. Sensing radius $R_s = 600$ m, safety threshold $d_{safe} = 50$ m, and constraint activation hysteresis buffer $\epsilon_{hys} = 0.5$ m. Furthermore, the random number generator seed was fixed (`rng(1)`) to guarantee consistent initialization of agent positions and perturbation logic across trials.

Table 1. Simulation parameter settings.

Parameter Symbol	Physical Meaning	Value/Unit
N	Number of AUVs	5
E	Heterogeneous capability vector	[1.0, 1.5, 2.0, 2.5, 3.0]
R_s	Sensor sensing radius	600 m
v_{max}	Maximum speed limit	5 m/s
u_{max}	Maximum acceleration limit in X/Y direction	1.5 m/s ²
d_{safe}^{AUV}	AUV safety distance	50 m
k_p, k_d	Nominal controller gain	0.5, 1.0
γ	CBF class function coefficient	0.5

5.1. Experiment 1: Heterogeneous Cooperative Coverage in Obstacle-Free Environment

The experimental results of heterogeneous cooperative coverage in an obstacle-free environment are shown in Figures 2 and 3.

Figure 2 illustrates the optimal coverage trajectories and the final Voronoi partition of the heterogeneous AUV swarm in an obstacle-free environment. Upon initiation of the simulation, the five AUVs rapidly disperse from their initial clustered state towards the three high-intensity Gaussian hotspots distributed within the mission area. Driven

by the nominal controller, the AUVs automatically allocate target regions according to their individual capabilities: The stronger agents, AUV 5 and AUV 4, do not congregate at the same hotspot. Instead, they maneuver towards the high-density core regions located in the bottom-left and top areas, respectively, thereby occupying the primary task peaks. Conversely, the weaker AUV 1 is assigned to the peripheral region of the top hotspot. The black Voronoi partition lines in the figure clearly demonstrate the result of the capability-based weighted partition. The partition boundaries are not simple geometric perpendicular bisectors; instead, they are curved towards the weaker side. This curvature allows high-capability AUVs to govern broader high-value regions, providing intuitive validation of the algorithm's adaptability in spatial allocation based on heterogeneity.

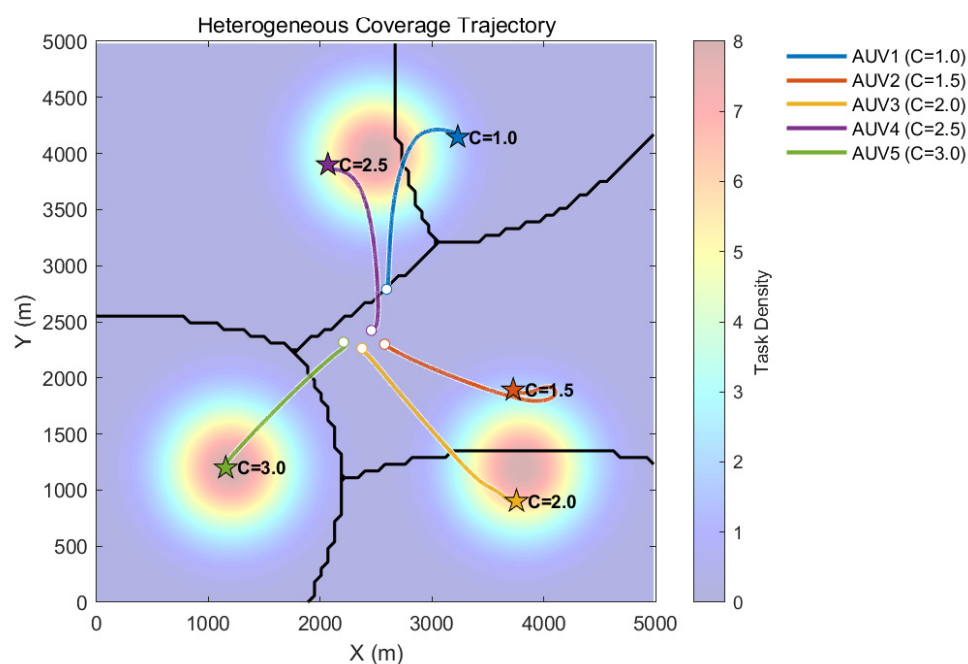


Figure 2. Cooperative coverage trajectories and spatial distribution of a heterogeneous AUV swarm in an obstacle-free environment.

Figure 3a illustrates the dynamic evolution of the core performance metric—the task workload-to-capability ratio. The system begins in a state of extreme imbalance due to the random and clustered initial positions of the AUVs, evidenced by drastic disparities in load ratios ranging from near 0 to over 2500. However, as the control algorithm intervenes, the adaptive weighting mechanism drives the curves to rapidly converge, reaching a unified steady-state value of approximately 908 around $t = 400$ s. While the zoomed-in view reveals minor periodic fluctuations attributed to the dynamic adjustment of Voronoi boundaries on the discrete simulation grid, these deviations are negligible and strictly confined near the convergence value. This result provides robust evidence that the algorithm successfully achieves the consensus equilibrium objective, ensuring that the actual workload undertaken by each AUV is strictly proportional to its physical capability.

Figure 3b illustrates the temporal evolution of the physical area occupied by each AUV, corroborating the impact of system heterogeneity. In contrast to the uniform convergence observed in workload ratios, the physical area trajectories exhibit a distinct stratified distribution at steady state, with AUV 4 occupying the largest area (approximately 0.7×10^7 m²) and the least capable agent, AUV 1, occupying the smallest (approximately 0.35×10^7 m²). Notably, the physical area is not strictly proportional to capability magnitude—exemplified by the most capable agent, AUV 5, occupying a slightly smaller area than AUV 4—because

spatial allocation is heavily influenced by environmental information density; AUV 5 secures a high-density core region where a smaller physical footprint suffices to accumulate a substantial workload, whereas AUV 4 covers a larger portion of the low-density periphery. This phenomenon underscores the algorithm's ability to synergistically balance capability heterogeneity and environmental distribution to achieve an optimal spatial allocation.

Figure 3c,d depict the temporal evolution of the control inputs for the AUV swarm. During the initial phase (0–300 s), the control inputs are predominantly saturated, adhering tightly to the physical constraint boundaries of $\pm 1.5 \text{ m/s}^2$ (indicated by red dashed lines) as the AUVs traverse significant distances; this demonstrates the algorithm's capacity to fully exploit the AUVs' maximum maneuvering capabilities to minimize convergence time. Upon approaching the target regions, the control magnitudes rapidly decrease and stabilize, though some high-frequency switching is observed in the steady state—an artifact attributed to minor discrete jumps of Voronoi centroids at grid edges and the continuous fine-tuning required for strict workload balancing. Crucially, all control commands remain strictly within the predefined hard constraints throughout the entire process, thereby verifying the controller's safety and feasibility regarding dynamic compliance.

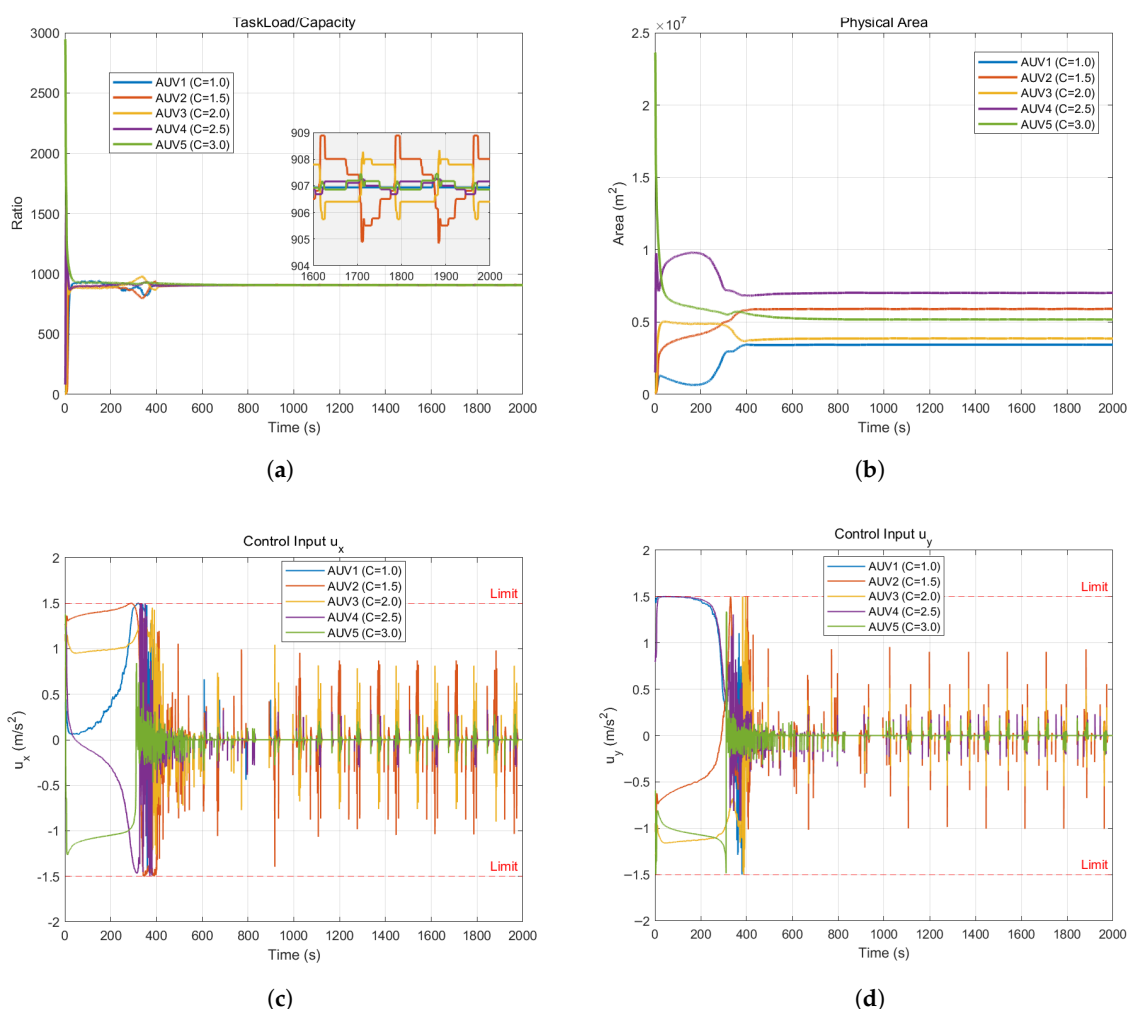


Figure 3. Coverage control state in an obstacle-free environment. (a) The taskload-to-capability ratio. (b) Spatial area allocation. (c) Control input u_x . (d) Control input u_y .

As observed in Figure 3c,d, high-frequency oscillations occur when the AUVs operate near safety boundaries. This switching behavior is characteristic of QP-based controllers strictly enforcing hard constraints in a discrete-time setting. While mathematically ensuring forward invariance, such chattering is undesirable for physical marine actuators due to

potential mechanical wear and energy inefficiency. For practical implementation, this issue can be mitigated through several strategies:

- (a) **Signal Smoothing:** Applying a low-pass filter to the QP-generated acceleration command u_i^* before it reaches the thruster drivers.
- (b) **Relaxed Barrier Functions:** Implementing a boundary layer or using relaxed class- $\mathcal{K}$ functions in the CBF formulation to allow for smoother transitions between safe and unsafe sets.
- (c) **Input Regularization:** Increasing the regularization weight in the QP cost function to penalize rapid changes in control effort ($\|\Delta u\|^2$).

The results from Experiment 1 demonstrate that, in an obstacle-free environment, the proposed control framework effectively accommodates the heterogeneous characteristics of the AUV swarm. By leveraging the adaptive adjustment of Voronoi weights, the algorithm successfully steers AUVs toward task regions commensurate with their respective capabilities, thereby realizing a global consensus on the workload-to-capability ratio and validating the effectiveness of the nominal controller.

5.2. Experiment 2: Safe Coverage Control in Obstacle Environments

To evaluate the obstacle avoidance capability, three circular obstacles were established within the simulation area at coordinates (2500, 3300), (1800, 1900), and (3200, 1900), with radii of 300 m, 250 m, and 250 m, respectively. The experimental results regarding safe coverage control in this obstacle-laden environment are presented in Figures 4–6.

Figure 4 visually demonstrates the path planning and spatial allocation capabilities of the heterogeneous AUV swarm in the presence of static obstacles. Observing the trajectories, the five AUVs depart from their initial clustered region and accurately identify the gray circular obstacles located along their paths toward the three Gaussian target hotspots. Unlike the linear trajectories observed in the obstacle-free environment, the trajectories in this experiment exhibit smooth curvilinear circumvention characteristics when approaching obstacles. Ultimately, the AUVs not only safely bypass the obstacles but also form a rational Voronoi coverage configuration based on their respective capability weights E_i . The Voronoi partition lines exhibit adaptive curvature in response to the environmental geometric constraints, verifying the effectiveness of the heterogeneous planning algorithm in complex topological environments.

Figure 5a reflects the cooperative performance and robustness of the swarm under perturbed conditions. Compared to the obstacle-free environment, the workload ratio curves in this experiment exhibit more significant fluctuations during the convergence process, particularly between $t = 200$ s and 800 s. This is attributed to the presence of obstacles, which temporarily obstruct the sensing line-of-sight of certain AUVs or alter the Voronoi neighborhood topology, leading to sudden jumps in local workload calculations. Nevertheless, as the AUVs circumnavigate the obstacles and enter the target regions, the adaptive weight update law rapidly eliminates these disturbance-induced biases. Ultimately, all curves stably converge to a unified constant value after $t = 1000$ s, demonstrating that the proposed consensus algorithm possesses effective disturbance rejection capabilities.

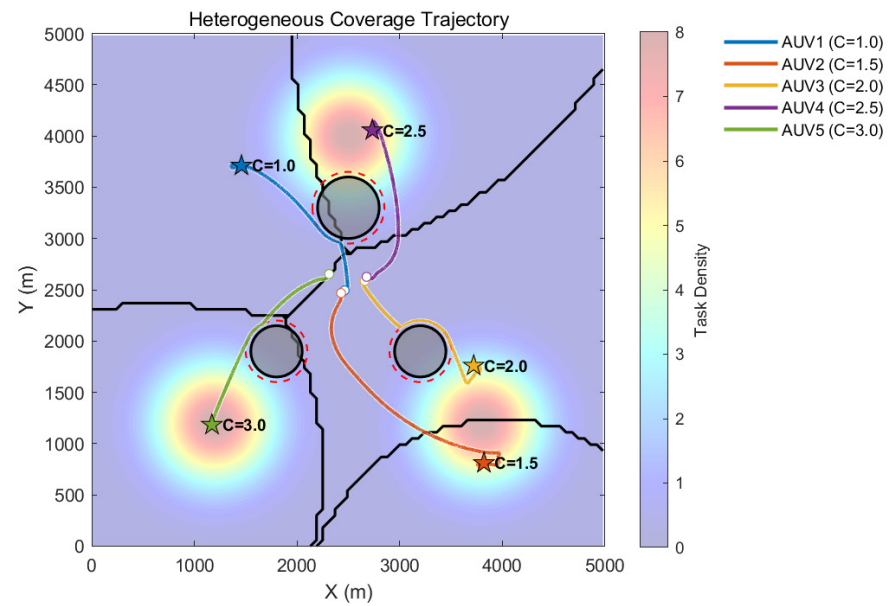


Figure 4. Cooperative coverage trajectories and spatial distribution of a heterogeneous AUV swarm in an obstacle environment.

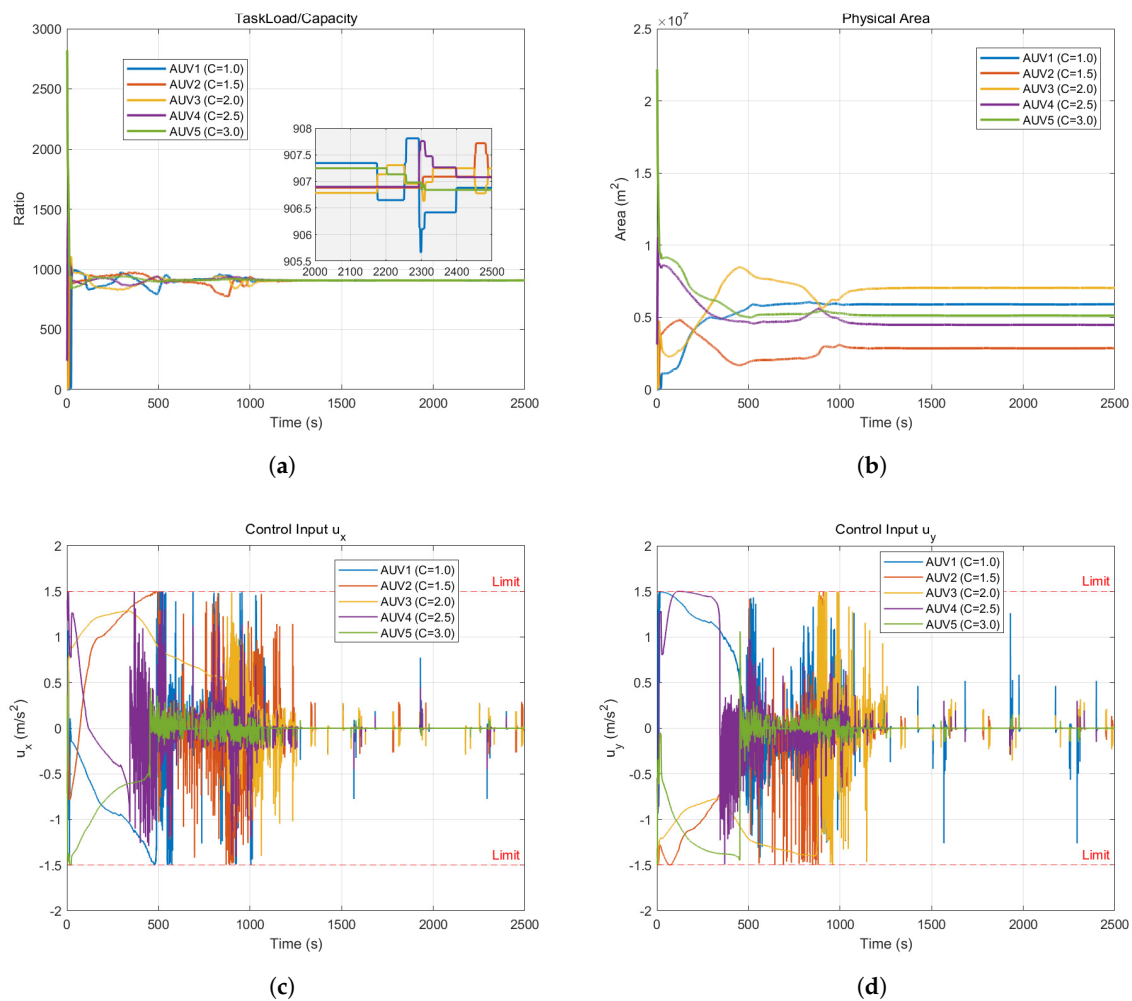


Figure 5. Coverage control state in an obstacle environment. (a) The taskload-to-capability ratio. (b) Spatial area allocation. (c) Control input u_x . (d) Control input u_y .

The stratification phenomenon in the curves of Figure 5b clearly embodies the heterogeneous characteristics: At steady state, AUVs with higher capabilities tend to occupy larger physical areas than those with lower capabilities. Notably, during the obstacle avoidance phase around $t = 400$ s, some curves display sharp spikes or dips. This occurs because the AUVs are forced to temporarily compress or stretch their Voronoi cells to avoid obstacles. This dynamic area adjustment mechanism proves that the algorithm can flexibly manage the conflict between geometric spatial constraints and mission workload requirements, effectively trading temporary spatial optimality for system safety and eventual workload balance.

Figure 5c,d depict the time-domain evolution of acceleration control inputs. Following an initial phase of saturation at the $\pm 1.5 \text{ m/s}^2$ boundaries to maximize speed, the intermediate obstacle avoidance phase features dense high-frequency switching, representing the real-time interplay between the nominal controller (target seeking) and safety constraints (obstacle avoidance).

Figure 6 displays the real-time Euclidean distance between each AUV and the nearest obstacle surface, with the red dashed line indicating the mandatory 50 m collision boundary. Simulation results show that during the avoidance period ($t = 100$ s to 500 s), while distance curves for AUV 4 and AUV 3 drop sharply as they approach obstacles, the minima strictly remain above the red dashed line. This confirms that the HOCBF-based safety controller effectively intervenes upon detecting collision risks to generate sufficient repulsive acceleration, ensuring the system state remains within the safety set for a rigorous zero-collision guarantee.

The results of Experiment 2 demonstrate the robustness of the proposed control framework. Facing complex conditions with multiple static obstacles, the algorithm not only strictly enforces safe collision avoidance distances but also successfully guides the heterogeneous swarm to overcome disturbances caused by environmental topology changes, ultimately achieving efficient and balanced cooperative coverage.

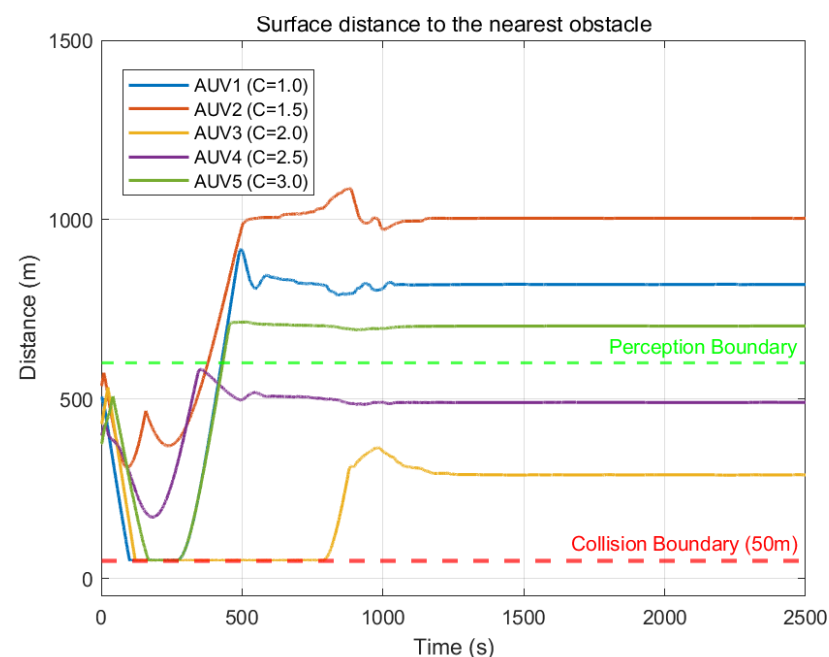


Figure 6. Distance between AUV and nearest obstacle.

5.3. Experiment 3: Comparative Experiments and Quantitative Analysis

To comprehensively evaluate the performance of the proposed hierarchical control framework, comparative experiments were conducted against three baseline methods

under the same environmental conditions as Experiment 2 (containing multiple static obstacles). The simulation duration was set to 1500 s to ensure sufficient time for convergence. The baseline strategies are defined as follows:

- (a) Standard Lloyd's Algorithm: A classic coverage control method based on Voronoi partitioning that drives agents toward centroids without explicit obstacle avoidance constraints.
- (b) APF-Based Method: A method combining the nominal coverage controller with Artificial Potential Fields to handle obstacle avoidance reactively.
- (c) CBF-Only Method: A simplified strategy that applies the proposed safety constraints but utilizes a standard position-based nominal controller without the heterogeneous workload-to-capability consensus mechanism.

The quantitative results are summarized in Table 2. The performance metrics include

- (a) Coverage Cost (CCost): Defined as the standard deviation of the workload-to-capability ratio (R_i) across the swarm. A lower V_c indicates a more balanced task allocation aligned with agent heterogeneity.
- (b) Coverage Time (CTime): Time required to reach the steady-state coverage configuration.
- (c) Min Obs Dist (MOD): The minimum Euclidean distance recorded between AUV and the nearest obstacle during the entire simulation (safety threshold $d_{safe} = 50$ m).
- (d) Min AUV Dist (MAD): The minimum Euclidean distance recorded between any AUV during the entire simulation.
- (e) Violations (Vio.): Total accumulated count of time steps where safety constraints were violated (i.e., distance < 50 m).
- (f) Runtime: Average computation time per control step.

Table 2. Quantitative performance comparison of different control strategies.

Method	CCost	CTime (s)	MOD (m)	MAD (m)	Vio.	Runtime (ms)
Proposed	0.15	702	50.02	170.45	0	0.71
Standard Lloyd	396.41	N/A	0.00	170.45	4252	0.20
APF-Based	5.85	730	0.00	72.67	6978	0.21
CBF-Only	396.41	N/A	50.01	170.45	0	0.73

The evolution of coverage performance, quantified by the coverage cost in Table 2 and visualized in Figure 7, demonstrates the superior efficiency of the proposed framework. As indicated by the rapidly descending yellow curve in Figure 7, the proposed method achieves a steady-state cost of 0.15 within 702 s, signifying a heterogeneous load balance where the swarm reaches the global optimum efficiently. In contrast, the Standard Lloyd and CBF-only methods fail to converge, maintaining a high cost of 396.41 due to unresolved conflicts, while the APF-based method settles at a suboptimal local minimum of 5.85 after a convergence process.

Safety performance is validated through the obstacle distance metrics in Table 2 and the time-domain evolution in Figures 8 and 9. Specifically, the Standard Lloyd and APF-based strategies exhibit failures, with their distance curves in Figure 8 frequently dropping to 0.00 m (crossing the 50 m safety threshold), which corresponds to the massive violation counts of 4252 and 6978, respectively. Conversely, the proposed method strictly maintains a minimum distance of 50.02 m with 0 violations, and as shown in Figure 9, it maintains a stable inter-agent separation, proving that the hierarchical ZCBF/HOCBF constraints effectively enforce hard safety boundaries.

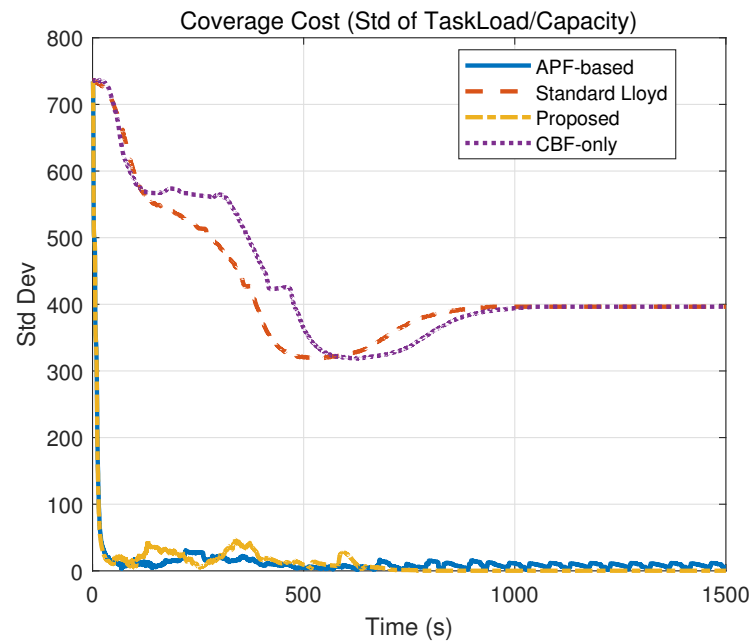


Figure 7. Comparison of coverage cost evolution.

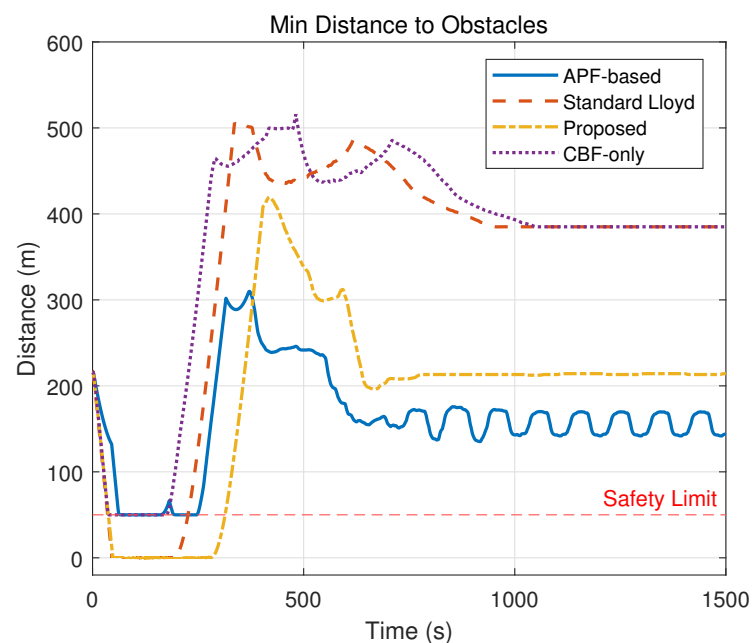


Figure 8. Minimum distance to static obstacles.

The spatial trajectories presented in Figure 10, combined with the safety violation statistics in Table 2, highlight the critical role of the consensus mechanism in preventing deadlocks. The proposed method generates paths (Figure 10a) that successfully navigate obstacles. Meanwhile, the collision-prone paths of the Lloyd method (Figure 10b) and the oscillating paths of the APF method (Figure 10c) further confirm their unsuitability for complex environments compared to the feasible solution found by this paper.

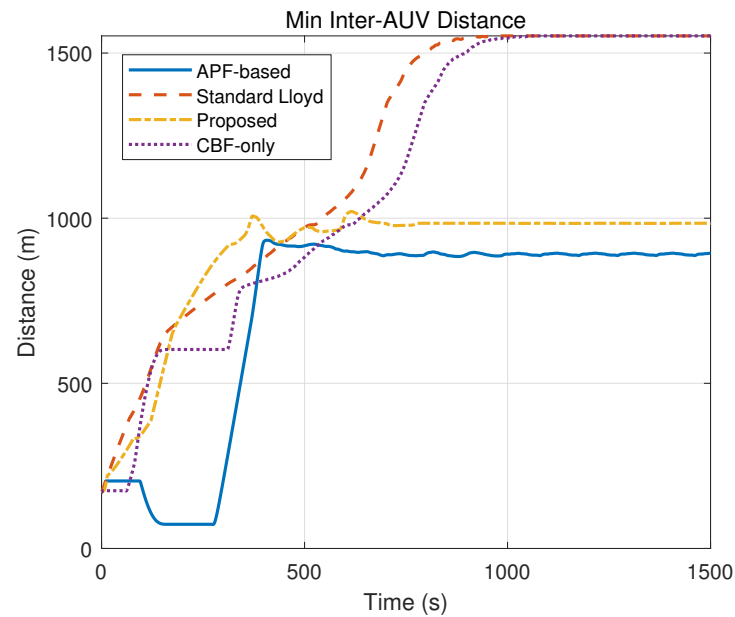


Figure 9. Minimum inter-agent distance.

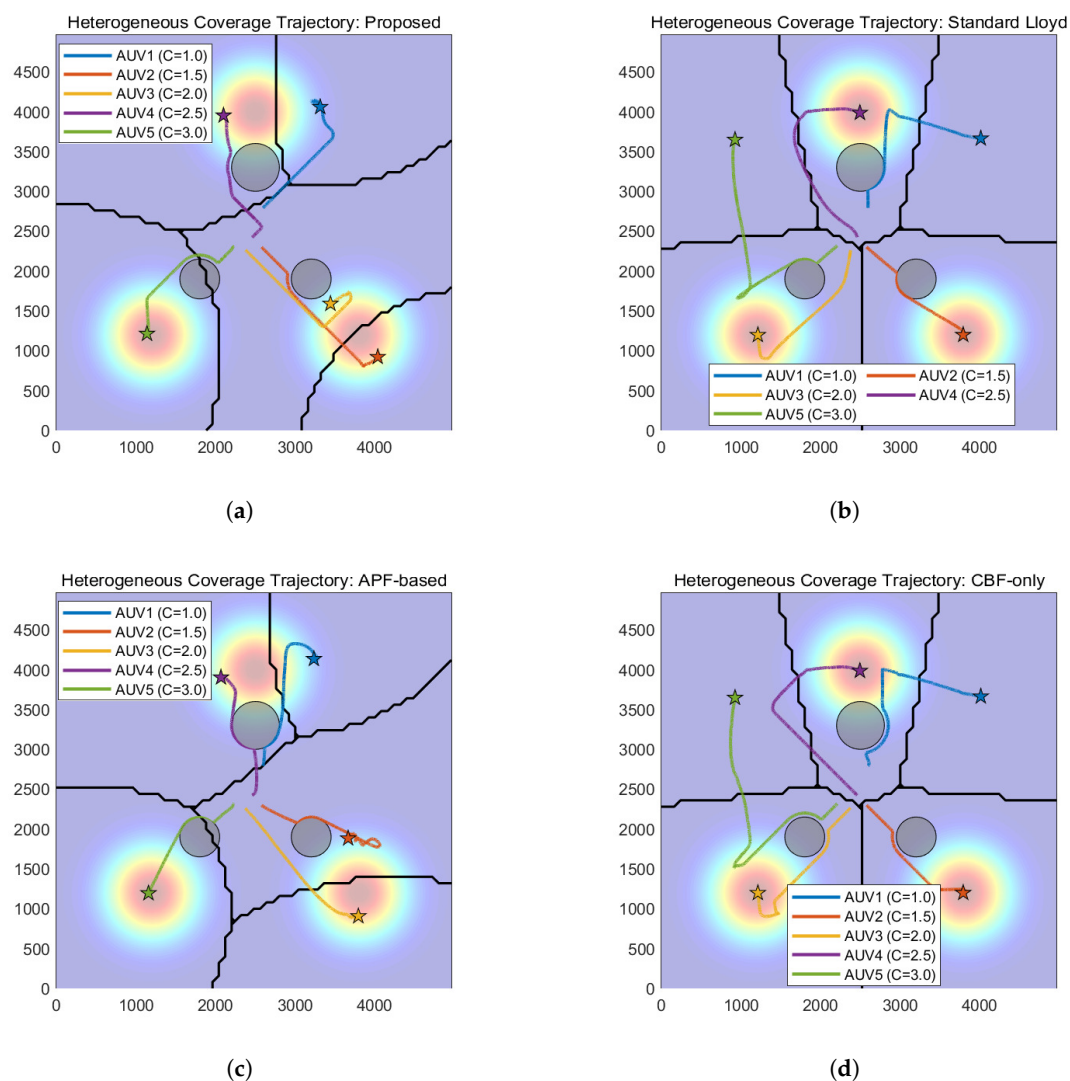


Figure 10. Spatial trajectory comparison. (a) The proposed method. (b) Lloyd's method. (c) APF-based method. (d) CBF-only method.

Figure 11 provides visual confirmation of the heterogeneous load balancing capability, which is mathematically represented by the low cost standard deviation of $V_c = 0.15$. In the proposed method (Figure 11a), the workload-to-capability ratio curves for all five heterogeneous AUVs converge to a single consensus value, demonstrating that the task load is allocated proportional to each AUV's capability E_i . This contrasts with the divergent or chaotic curves observed in the baseline methods (Figure 11b–d)—such as the Standard Lloyd case where capable AUVs remain underutilized while weaker ones are overloaded—thereby validating that the proposed consensus-based nominal controller effectively exploits system heterogeneity to optimize global coverage.

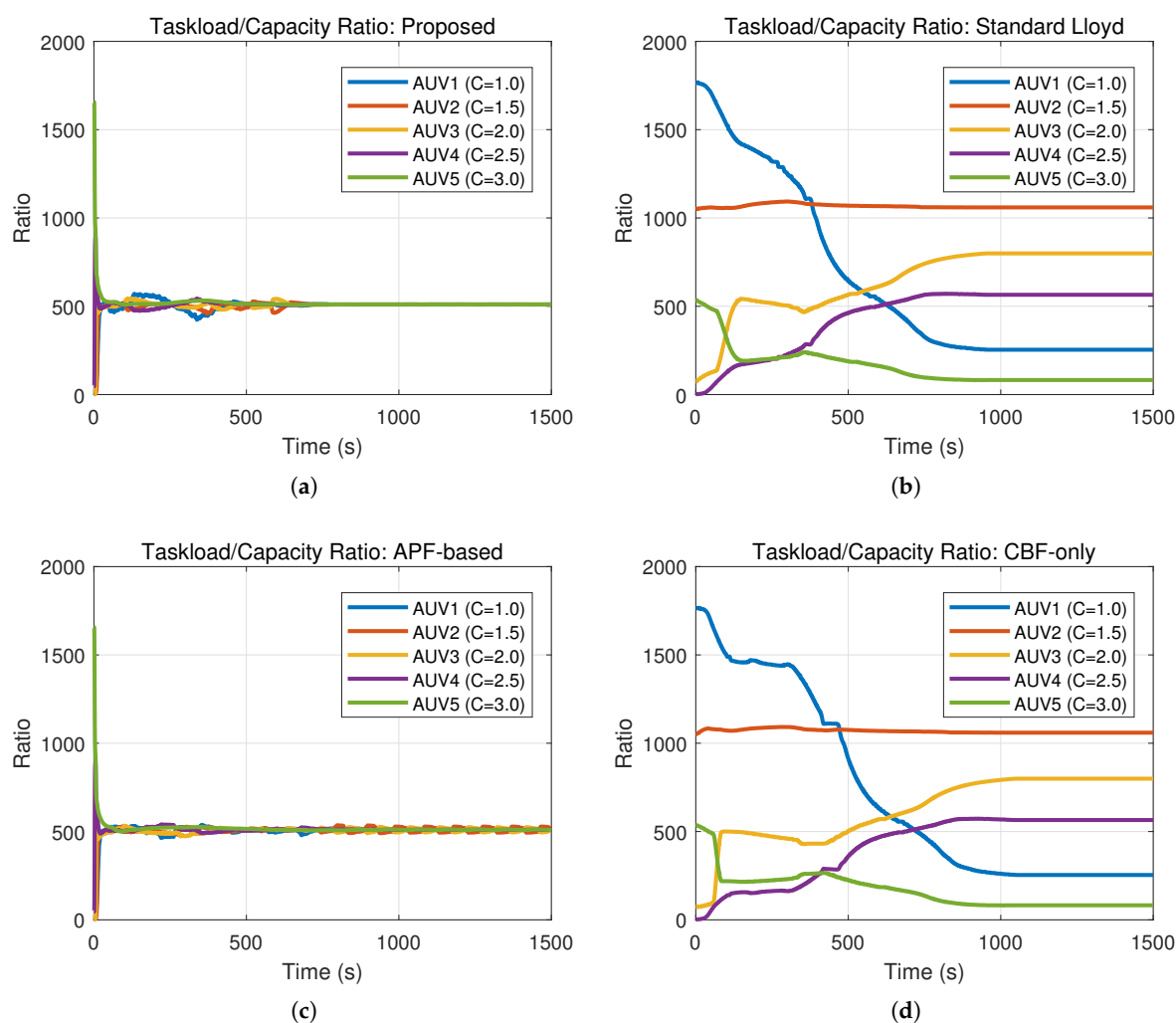


Figure 11. Taskload-to-capability ratio (R_i) evolution. (a) The proposed method. (b) Lloyd's method. (c) APF-based method. (d) CBF-only method.

6. Conclusions

This paper addresses the cooperative coverage control problem for heterogeneous AUV swarms in complex underwater environments by proposing a hierarchical control framework based on QP that integrates CBFs with consensus theory. At the performance layer, a nominal controller $\hat{u}_i$ is designed leveraging CLF concepts and consensus theory to minimize the cost function $V_c = \sum (T_i^2 / E_i)$, thereby driving the workload-to-capability ratio R_i toward global consensus. At the safety layer, ZCBFs and HOCBFs are employed to transform inter-AUV collision avoidance, obstacle avoidance, and velocity limits into linear constraints for the controller. Theoretical analysis confirms the forward invariance of the safe set and the convergence of the coverage mission, while simulation experiments

validate that the swarm achieves efficient heterogeneous load balancing while prioritizing safety constraints via a soft-constraint formulation.

Regarding the gap between the theoretical model and practical deployment, future work will focus on integrating the proposed high-level planner with high-fidelity AUV dynamic models. Specifically, we will investigate the combination of this framework with robust low-level controllers to actively compensate for hydrodynamic damping and actuator saturation. Furthermore, incorporating braking-distance-aware safety margins into the QP formulation will be explored to enhance the system's applicability in real-world ocean currents. We also plan to extend the collision avoidance capabilities to address dynamic non-cooperative obstacles by integrating state estimation and time-varying barrier functions into the CBF-QP optimization.

Author Contributions: Conceptualization, F.M.; methodology, F.M.; software, D.Z.; validation, F.M., L.X., and R.W.; formal analysis, D.Z.; investigation, F.M.; resources, D.Z.; data curation, D.Z.; writing—original draft preparation, D.Z.; writing—review and editing, F.M.; visualization, L.X.; supervision, F.M.; project administration, F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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.

References

1. Zhang, J.; Zhu, Z.; Xue, Y.; Deng, Z.; Qin, H. Local path planning of under-actuated AUV based on VADWA considering dynamic model. *Ocean Eng.* **2024**, *310*, 118705. [\[CrossRef\]](#)
2. Zhang, Z.; Xu, J.; Xie, G.; Wang, J.; Han, Z.; Ren, Y. Environment-and energy-aware auv-assisted data collection for the internet of underwater things. *IEEE Internet Things J.* **2024**, *11*, 26406–26418. [\[CrossRef\]](#)
3. Huang, H.; Bian, X.; Jiang, T.; Tang, Q.; Qin, H.; Wang, Z. Hydrodynamic analysis and manipulation control on a streamlined I-AUV. *ISA Trans.* **2024**, *153*, 453–466. [\[CrossRef\]](#)
4. Pang, W.; Zhu, D.; Chen, M.; Xu, W. RG-SAPF: A Scheme for Cooperative Escorting of Underwater Moving Target by Multi-AUV Formation Systems Based on Rigidity Graph and Safe Artificial Potential Field. *Sensors* **2025**, *25*, 6823. [\[CrossRef\]](#)
5. Petritoli, E.; Cagnetti, M.; Leccese, F. Simulation of autonomous underwater vehicles (auvs) swarm diffusion. *Sensors* **2020**, *20*, 4950. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Lunze, J. Networked Control of Multi-Agent Systems. Available online: <https://www.editionmora.de/ncs/downloads/NCSA2Flyer.pdf> (accessed on 13 January 2022).
7. Chen, F.; Ren, W. On the control of multi-agent systems: A survey. *Found. Trends Syst. Control* **2019**, *6*, 339–499. [\[CrossRef\]](#)
8. Zhao, S.; Zelazo, D. Bearing rigidity theory and its applications for control and estimation of network systems: Life beyond distance rigidity. *IEEE Control Syst. Mag.*, **2019**, *39*, 66–83. [\[CrossRef\]](#)
9. Fabris, M.; Fattore, G.; Cenedese, A. Optimal time-invariant distributed formation tracking for second-order multi-agent systems. *Eur. J. Control* **2024**, *77*, 100985. [\[CrossRef\]](#)
10. Li, L.; Li, Y.; Wang, Y.; Xu, G.; Wang, H.; Gao, P.; Feng, X. Multi-AUV coverage path planning algorithm using side-scan sonar for maritime search. *Ocean Eng.* **2024**, *300*, 117396. [\[CrossRef\]](#)
11. Zhang, Y.; Wang, Q.; Shen, Y.; Wang, T.; Dai, N.; He, B. Multi-AUV cooperative search method based on dynamic optimal coverage. *Ocean Eng.* **2023**, *288*, 116168. [\[CrossRef\]](#)
12. Wang, Z.; Zhang, Z.; Wang, J.; Jiang, C.; Han, Z.; Ren, Y. Design of heterogeneous multi-AUV IoUT: Architecture, key technologies and applications. *IEEE Wirel. Commun.* **2024**, *in press*.
13. Chen, B.; Hu, J.; Ghosh, B.K. Finite-time tracking control of heterogeneous multi-AUV systems with partial measurements and intermittent communication. *Sci. China Inf. Sci.* **2024**, *67*, 152202. [\[CrossRef\]](#)

14. Liu, T.; Huang, J.; Zhao, J. Research on obstacle avoidance of underactuated autonomous underwater vehicle based on offline reinforcement learning. *Robotica* **2025**, *43*, 194–218. [\[CrossRef\]](#)
15. Pang, W.; Zhu, D.; Sun, C. Multi-AUV formation reconfiguration obstacle avoidance algorithm based on affine transformation and improved artificial potential field under ocean currents disturbance. *IEEE Trans. Autom. Sci. Eng.* **2023**, *21*, 1469–1487. [\[CrossRef\]](#)
16. Li, D.; Du, L. AUV trajectory tracking models and control strategies: A review. *J. Mar. Sci. Eng.* **2021**, *9*, 1020. [\[CrossRef\]](#)
17. Mu, X.; Gao, W. Coverage path planning for multi-AUV considering ocean currents and sonar performance. *Front. Mar. Sci.* **2025**, *11*, 1483122. [\[CrossRef\]](#)
18. Zhou, M.; Li, J.; Wang, C.; Wang, J.; Wang, L. Applications of Voronoi diagrams in multi-robot coverage: A review. *J. Mar. Sci. Eng.* **2024**, *12*, 1022. [\[CrossRef\]](#)
19. Feng, F.; Xiong, S.; Liu, Z.; Xian, Z.; Zhou, Y.; Kobayashi, H.; Kawamoto, A.; Nomura, T.; Zhu, B. Cellular topology optimization on differentiable Voronoi diagrams. *Int. J. Numer. Methods Eng.* **2023**, *124*, 282–304. [\[CrossRef\]](#)
20. Wang, J.; Wang, R. Multi-UAV area coverage track planning based on the voronoi graph and attention mechanism. *Appl. Sci.* **2024**, *14*, 7844. [\[CrossRef\]](#)
21. Ebrahimi, B.; Bataleblu, A.A.; Roshanian, J. Bi-level Voronoi strategy for cooperative search and coverage. *Swarm Evol. Comput.* **2025**, *97*, 102064. [\[CrossRef\]](#)
22. Chen, J.; Liu, X.; Liu, J.; Hao, M.; Liu, Z. Full-coverage obstacle avoidance path planning based on RF-SMO for AUV in irregular regions. *Clust. Comput.* **2025**, *28*, 836. [\[CrossRef\]](#)
23. Guo, X.; Chen, Y.; Zhao, D.; Luo, G. A static area coverage algorithm for heterogeneous AUV group based on biological competition mechanism. *Front. Bioeng. Biotechnol.* **2022**, *10*, 845161. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Xiao, W.; Belta, C. High-order control barrier functions. *IEEE Trans. Autom. Control* **2021**, *67*, 3655–3662. [\[CrossRef\]](#)
25. Valmorbida, G.; Hovd, M. Quadratic programming with ramp functions and fast online QP-MPC solutions. *Automatica* **2023**, *153*, 111011. [\[CrossRef\]](#)
26. Liu, J.; Yang, J.; Mao, J.; Zhu, T.; Xie, Q.; Li, Y.; Wang, X.; Li, S. Flexible active safety motion control for robotic obstacle avoidance: A CBF-guided MPC approach. *IEEE Robot. Autom. Lett.* **2025**, in press. [\[CrossRef\]](#)
27. Zühlke, M.M.; Kudenko, D. Adversarial robustness of neural networks from the perspective of lipschitz calculus: A survey. *ACM Comput. Surv.* **2025**, *57*, 142. [\[CrossRef\]](#)
28. Khalil, H.K.; Grizzle, J.W. *Nonlinear Systems*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
29. Ames, A.D.; Xu, X.; Grizzle, J.W.; Tabuada, P. Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. Autom. Control* **2016**, *62*, 3861–3876. [\[CrossRef\]](#)
30. Xiao, W.; Li, A.; Cassandras, C.G.; Belta, C. Toward model-free safety-critical control with humans in the loop. *Annu. Rev. Control* **2024**, *57*, 100944. [\[CrossRef\]](#)
31. Liu, Y.; Chen, J.; Pan, S.; Cohen-Or, D.; Zhang, H.; Huang, H. Split-and-fit: Learning b-reps via structure-aware voronoi partitioning. *ACM Trans. Graph. (TOG)* **2024**, *43*, 108. [\[CrossRef\]](#)
32. Wang, H.; Wang, H.; Guo, X. Distributed Adaptive Fault-Tolerant Formation Control for Heterogeneous USV-AUV Swarms Based on Dynamic Event Triggering. *J. Mar. Sci. Eng.* **2025**, *13*, 2116. [\[CrossRef\]](#)
33. Li, B.; Gao, X.; Huang, H.; Yang, H. Improved adaptive twisting sliding mode control for trajectory tracking of an AUV subject to uncertainties. *Ocean Eng.* **2024**, *297*, 116204. [\[CrossRef\]](#)
34. Sayyaadi, H.; Moarref, M. A distributed algorithm for proportional task allocation in networks of mobile agents. *IEEE Trans. Autom. Control* **2010**, *56*, 405–410. [\[CrossRef\]](#)
35. Lee, T.C.; Zhang, Y.; Mareels, I. Multi-agreements in social networks based on LaSalle invariance principle and positive definiteness. *Automatica* **2024**, *165*, 111666. [\[CrossRef\]](#)
36. Tremblay, R.; Gaboury, S.; Fugere, B.J. A new Leibniz rule and its integral analogue for fractional derivatives. *Integral Transform. Spec. Funct.* **2013**, *24*, 111–128. [\[CrossRef\]](#)
37. Adeniran, A.T.; Faweya, O.; Ogunlade, T.O.; Balogun, K.O. Derivation of Gaussian Probability Distribution: A New Approach. *Appl. Math.* **2020**, *11*, 436–446. [\[CrossRef\]](#)

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.