A Decoupled Non-uniform Guiding Vector Field for Multi-robot Coordinated Path-Following on 2D Manifolds

JINGZONG LIU
ZEXU ZHANG
Harbin Institute of Technology, Harbin, China

LINAG ZHANG
Anhui University, Anhui, China

KAI ZHANG
CHAO YAN
Harbin Institute of Technology, Harbin, China

***Abstract*— This paper proposes a decoupled non-uniform guiding vector field (DNGVF) for coordinated path-following of a robot team on 2D manifolds embedded in 3D space. The DNGVF comprises three components: a propagation term that steers the robots along the desired path, a convergence term that drives path-following errors to zero, and a coordination term, decoupled from the convergence term, that enforces the predefined formation on the surface. Additionally, a speed coordination strategy is developed to construct the non-uniform guiding vector field, which assigns desired velocities to each robot in the vector field, thereby improving coordination accuracy. Finally, a surface generation method based on moving frame is introduced to enhance surface design flexibility. The key contribution of this work is the decoupling of the convergence and coordination terms by formulating the latter within the tangent space of the surface, thereby improving both path-following accuracy and coordination performance. Numerical and software-in-the-loop simulations validate that the proposed DNGVF reduces path-following errors while simultaneously improving coordination accuracy.**

***Index Terms*— Motion coordination, Multi-robot systems, Non-uniform guiding vector field, Path-following.**

Manuscript received XXXXX 00, 0000; revised XXXXX 00, 0000; accepted XXXXX 00, 0000. This work was supported in part by the National Natural Science Foundation of China under grant No. 62303002 *(Corresponding author: ZEXU ZHANG)*

Authors' address: Jingzong Liu, Zexu Zhang, Kai Zhang and Chao Yan are with the School of Astronautics, Harbin Institute of Technology, Harbin 150001, China, E-mail: (liujingzong@stu.hit.edu.cn; zexuzhang@hit.edu.cn; kaizhang2023@hit.edu.cn; yanchao_hit@163.com); Liang Zhang are with the School of Electrical Engineering and Automation, Anhui University, Anhui 230039, China, E-mail: (liangzhang@ahu.edu.cn).

0018-9251 © 2022 IEEE

I. INTRODUCTION

DURING the past few years, multi-robot coordinated path-following has been widely studied [1]–[4]. It involves guiding a group of robots along a given path while forming the desired formation through local interactions [5]. Typical coordinated path-following is implemented by integrating a multi-robot coordination framework with a path-following algorithm [6].

Typical path-following algorithms include control-based methods, geometric-based methods, and guiding vector fields (GVF). Control-based path-following methods [7] tended to result in complex control laws, especially under nonholonomic constraints, making them challenging for practical engineering applications. Geometric-based methods primarily consist of line-of-sight techniques and their variations [8], [9]. The performance of these methods is significantly influenced by the selection strategy for target points. Numerical simulations presented in [10] demonstrate that, in comparison to alternative approaches, the GVF technique more accurately follows the path and requires the least control effort. These methods guide robots to the desired path by constructing a vector field [11]. Goncalves et al. [12] represented the desired path by the zero level sets of a family of smooth implicit functions, thereby obviating the calculation of the closest point on the desired path and effectively decreasing the computational burden. On this basis, Yao et al. [13] achieved global convergence through path parameterization and the construction of a higher-dimensional GVF.

Coordination frameworks can be categorized as centralized [14]–[17] and distributed [6], [13], [18], [19] approaches. The leader-follower method [14]–[16] exemplifies a centralized framework. In [17], a coordination algorithm for followers leveraged the leader's single path parameter (e.g., arc length) to reduce communication overhead. Wang et al. [16] employed a leader-follower structure, designing non-uniform guiding vector fields for both the leader and followers to enable two-aircraft formation flight under wind conditions. Conversely, distributed coordination frameworks offer enhanced scalability and reduced communication requirements. Chen et al. [6] adopted a hierarchical strategy, employing optimization techniques to design path-following algorithms for individual robots. Distributed coordination is subsequently activated upon convergence of tracking errors within a predefined coordination set, realizing the desired arc length distribution among robots. De et al. [19] utilized a consensus algorithm to coordinate the phase of UAVs along distinct circular trajectories, achieving a specified phase offset between vehicles. In [20], the desired path is extended into a parameterized curve, and both propagation and convergence terms are designed to guide the robots along this trajectory, and a consensus algorithm was employed to adjust the robots' positions in the parameter space, thereby achieving the predefined parameter differences and ensuring an orderly distribution along the curve. Furthermore, Yao et al. [21] used parameterized surface as the desired surface, and the two coordinates in the parameter space were coordinated simultaneously, enabling multi-robot formation on the surface. Additionally, another approach achieved a more flexible ordering in formation motion on a two-dimensional manifold [5]. However, the coordination and convergence terms in these methods are inherently coupled, which may lead to mutual interference between coordination and convergence behaviors, ultimately compromising the performance of coordinated path-following.

In this paper, we consider the impact of the coupling between following and coordination behaviors on the performance of coordinated path-following, as well as the challenges posed by coordinated motion along trajectories with varying curvature. To address these issues, we propose a decoupled non-uniform guiding vector field that provides both the motion direction and speed for the robots. Specifically, our contributions are summarized as follows:

1) We propose a novel multi-robot coordination method that eliminates the impact of virtual coordinate coordinated motion on path-following. Unlike existing results, the coordination term in this paper ensures that coordinated motion occurs within the level set where the robots are located, thus avoiding mutual influence between the coordination term and the convergence term.
2) We propose a non-uniform guiding vector field for robot speed coordination, which theoretically guarantees the first-order consistency of virtual coordinates while maintaining the speed of the robot formation, thereby improving the precision of formation maintenance.
3) We introduce a parallel transport frame (PT-Frame) based method for desired surface generation. This approach divides the desired surface into two components: the reference curve and the formation curve, both expressed using parametric curves, thereby enabling a more flexible design of the desired trajectory surface.
4) We validate the proposed approach through numerical simulations and software-in-the-loop (SITL) simulations. The results confirm that our method effectively reduces path-following error and improves formation accuracy.

Notations: Let $\dot{f}(\tau)$ denote the derivation of any differentiable function $f(\tau)$ with respect to the parameter, i.e. $\frac{\mathrm{d}f}{\mathrm{d}\tau}$. Similarly, $\ddot{f}(\tau)$ and $f^{(n)}(\tau)$ represent the second derivative $\frac{\mathrm{d}^2 f}{\mathrm{d}\tau^2}$ and the n-th derivative $\frac{\mathrm{d}^n f}{\mathrm{d}\tau^n}$. The symbol ∇ denotes the Jacobian matrix or the gradient. For a vector-value function $\boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), \cdots, f_n(\boldsymbol{x})]^\top$, where $\boldsymbol{x} = [x_1, \cdots, x_m]^\top$, the Jacobian matrix is defined as $\nabla \boldsymbol{f} = [\frac{\partial f_i}{\partial x_j}] \in \mathbb{R}^{n \times m}$. When $f(\boldsymbol{x})$ is a scalar function, the gradient vector is given by $\nabla f(\boldsymbol{x}) = [\frac{\partial f}{\partial x_1}, \cdots, \frac{\partial f}{\partial x_m}]$. Let $(\cdot)^{[i]}$ denote the i-th element of a vector, and $(\cdot)^{[i,j]}$ denote the element in the i-th row and j-th column of a matrix. All vectors are considered column vectors in this paper unless noted differently.

II. PRELIMINARIES AND PROBLEM STATEMENT

A. Research Objectives and Framework

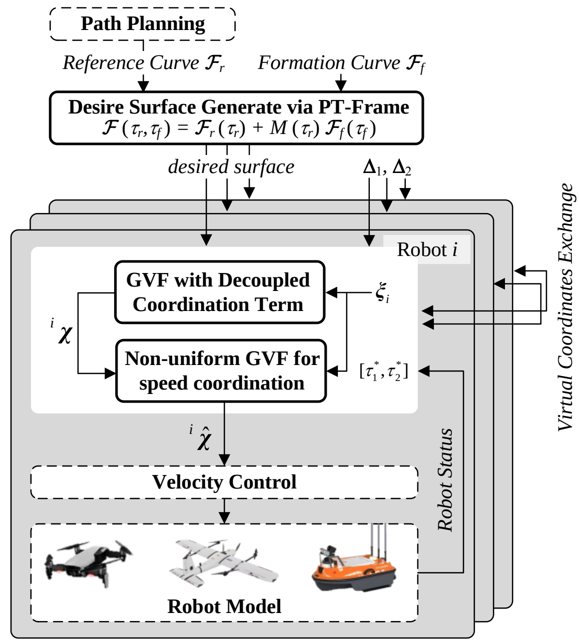


Fig. 1: Coordinated path-following framework using decoupled non-uniform guiding vector field.

The objective of this work is to guide a group of robots to move in a prescribed formation along a desired path. Rather than directly combining path-following algorithms with formation controllers, the proposed framework constructs a surface around the desired path and achieves coordinated motion of the robots on this surface. Consequently, the coordinated path-following task is reformulated as a surface construction problem together with a motion coordination problem defined on the surface.

As illustrated in Fig. 1, the overall procedure consists of two main stages. First, a desired surface is constructed using the PT-frame-based surface generation method by propagating the formation curve along the reference curve (i.e., the desired path). The reference curve specifies the longitudinal direction of the surface, whereas the formation curve determines its cross-sectional geometry. Second, the decoupled guiding vector field computes the desired motion direction ${}^i\chi$ from the states of each robot and its neighbors, and the speed coordination strategy determines the velocity magnitude to obtain the desired velocity ${}^i\hat{\chi}$. Each robot then tracks this desired velocity through its low-level controller, thereby achieving coordinated motion of all robots on the surface.

The following subsection introduces the motion coordination problem on the surface. The technical details of the decoupled coordination GVF, speed coordination strategy, and the PT-frame-based surface generation method are provided in Section III.

B. Motion Coordination on 2D Manifolds

A parametric surface $\mathcal{S}^{ph} \in \mathbb{R}^3$ can be defined as a set of points:

$$\mathcal{S}^{ph} = \{[x_1, x_2, x_3]^\top | x_i = f_i(\tau_1, \tau_2), i \in \mathbb{Z}_1^3\} \tag{1}$$

where the superscript $(\cdot)^{ph}$ denotes the objects in *physical space* which is spanned by coordinate axes x_1, x_2 and x_3, $\tau_1, \tau_2 \in \mathbb{R}$ are the parameters of surface $\mathcal{S}^{ph}$, and function $f_i(\tau_1, \tau_2)$ belongs to C^2. Define the space spanned by axes τ_1 and τ_2 as the *virtual space*, and $(\cdot)^{vi}$ represents the objects in virtual space.

The combination of the physical and virtual spaces enables the surface to be formulated as a two-dimensional manifold embedded in a higher-dimensional space, thereby eliminating potential singularities [21]. Specifically, this manifold is defined as the intersection of the zero-level sets of three implicit functions:

$$\mathcal{S} := \{\boldsymbol{\xi} \in \mathbb{R}^5 | \phi_i(\boldsymbol{\xi}) = 0, i = 1, 2, 3\} \tag{2}$$

where $\phi_i(\cdot) := x_i - f_i(\tau_1, \tau_2)$, $\boldsymbol{\xi} = [x_1, x_2, x_3, \tau_1, \tau_2]^\top$ is the generalized coordinate. Let $\boldsymbol{\xi}^{ph} = [x_1, x_2, x_3]^\top$ be *physical coordinate*, $\boldsymbol{\xi}^{vi} = [\tau_1, \tau_2]^\top$ be *virtual coordinate*.

Motion coordination requires a group of robots to maintain a desired formation while moving along a desired surface [5]. Since the mapping between the virtual coordinate $\boldsymbol{\xi}^{vi}$ and the physical point $\boldsymbol{\xi}^{ph}$ is bijective for all $\boldsymbol{\xi} \in \mathcal{S}$, the desired formation of n robots on the physical surface $\mathcal{S}^{ph}$ can be generated by defining their relative offsets in the virtual coordinate space. Fig. 2 illustrates examples of circular and square formations moving on the cylindrical surface

The *desired virtual coordinate differences matrices* are defined as follows:

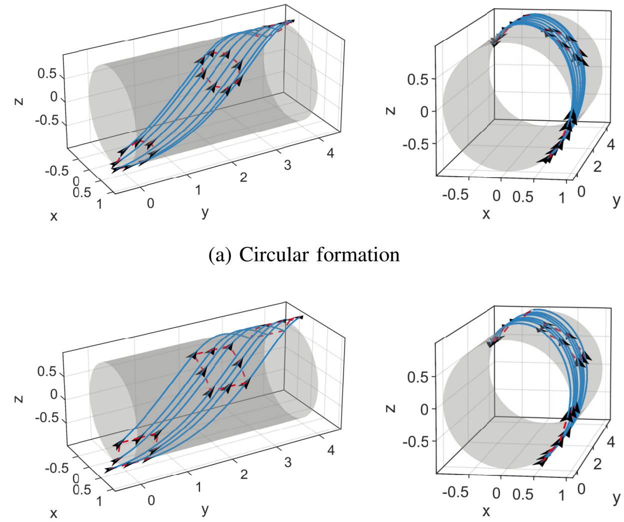
$$\begin{aligned}
\boldsymbol{\Delta}_1 &= \begin{bmatrix} \Delta_1^{[1,1]} & \cdots & \Delta_1^{[1,n]} \\ \vdots & \ddots & \vdots \\ \Delta_1^{[n,1]} & \cdots & \Delta_1^{[n,n]} \end{bmatrix} \\
\boldsymbol{\Delta}_2 &= \begin{bmatrix} \Delta_2^{[1,1]} & \cdots & \Delta_2^{[1,n]} \\ \vdots & \ddots & \vdots \\ \Delta_2^{[n,1]} & \cdots & \Delta_2^{[n,n]} \end{bmatrix}
\end{aligned} \tag{3}$$

where $\Delta_k^{[i,j]}$ is the desired difference in τ_k between robots i and j.

The desired virtual coordinate differences matrices satisfy the following properties:

$$\Delta_k^{[i,j]} = -\Delta_k^{[j,i]}, \quad \forall i, j \in \mathbb{Z}_1^N and\ i \neq j \tag{4a}$$
$$\Delta_k^{[i,i]} = 0, \quad \forall i \in \mathbb{Z}_1^N \tag{4b}$$
$$\Delta_k^{[i,j]} = \Delta_k^{[i,p]} - \Delta_k^{[j,p]}, \quad \forall i, j, p \in \mathbb{Z}_1^N \tag{4c}$$

The coordinated motion on the two-dimensional manifold is then defined as follows:



(a) Circular formation

(b) Square formation

Fig. 2: The motion of circular and square formations on cylinder.

DEFINITION 1 (Motion coordination on a surface). *Suppose a desired surface $\mathcal{S}$ is defined by (2). A group of N robots is said to achieve motion coordination on the surface if the following conditions are satisfied:*

1) *Each robot $i \in \mathbb{Z}_1^N$ asymptotically converges to the desired surface $\mathcal{S}$.*
2) *For any two robots $i, j \in \mathbb{Z}_1^N$, the differences between their virtual coordinates satisfy*

$$\tau_1^{[i]} - \tau_1^{[j]} = \Delta_1^{[i,j]}, \quad \tau_2^{[i]} - \tau_2^{[j]} = \Delta_2^{[i,j]},$$

where $\Delta_1^{[i,j]}$ and $\Delta_2^{[i,j]}$ are defined in (3).

REMARK 1. *In this work, the desired path is defined as a one-dimensional manifold (i.e., a curve), represented by the reference curve $F_r(\tau_r)$. A desired surface $\mathcal{S}$, defined as a two-dimensional manifold, is generated in a neighborhood of $F_r(\tau_r)$ using the method described in Subsection III.C. Coordinated path following is considered to be achieved when the robots converge to $\mathcal{S}$ and accomplish motion coordination on the surface.*

C. Basics of Graph Theory

The communication topology of a multi-robot system can be modeled as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} := \{1, \cdots, n\}$ is a finite non-empty node set and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. An edge $(i, j) \in \mathcal{E}$ indicates that robot j can receive information from robot i, but not necessarily vice versa. A directed graph $\mathcal{G}$ contains a directed spanning tree if and only if $\mathcal{G}$ has at least one node with directed paths to all other nodes. The adjacency matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$ of graph $\mathcal{G}$ is defined such that $a_{ij} = 1$ if $(j, i) \in \mathcal{E}$, and $a_{ij} = 0$ otherwise. The Laplacian

This article has been accepted for publication in IEEE Transactions on Aerospace and Electronic Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAES.2026.3659453

matrix $\mathcal{L} \in \mathbb{R}^{n\times n}$ is defined as follows:

$$l_{ii} = \sum_{j=1, j\neq i}^{n} a_{ij}, \qquad l_{ij} = -a_{ij}, i \neq j \tag{5}$$

$\mathcal{L}$ satisfies:

$$l_{ij} \leq 0,\ \ i \neq j, \quad \sum_{j=1}^{n} l_{ij} = 0, i = 1, \cdots, n \tag{6}$$

LEMMA 1 ([22]). *Suppose that $\boldsymbol{z} = [z_1, \cdots, z_n]^\top \in \mathbb{R}^n$, and let $\mathcal{L} \in \mathbb{R}^{n\times n}$ be the Laplacian matrix associated with the directed graph $\mathcal{G}$. Then, the following conditions are equivalent:*

1) *The directed graph $\mathcal{G}$ has a directed spanning tree.*
2) *$\mathcal{L}\boldsymbol{z} = 0$ if and only if $z_1 = \cdots = z_n$*
3) *Consensus is reached for the closed-loop system $\dot{\boldsymbol{z}} = -\mathcal{L}\boldsymbol{z}$. That is for all $\boldsymbol{z}_i(0) \in \mathbb{R}^m$, $||z_i(t) - z_j(t)|| \to 0$ as $t \to \infty$, where $i, j = 1, \cdots, n$.*

D. Guiding Vector Field for 2D Manifolds

The surface tracking error of the robots is defined as $\boldsymbol{e}_s = [g(\phi_1), g(\phi_2), g(\phi_3)]^\top$, where ϕ_i is defined in (2), and $g(\cdot)$ is a strictly increasing function satisfying $g(0) = 0$. When the surface is constructed in a neighborhood of the desired path, $\boldsymbol{e}_s$ also quantifies the path-following accuracy and is therefore referred to as the *path-following error* throughout the paper. For simplicity, letting $g(\phi_i) = \phi_i$ yields:

$$\boldsymbol{e}_s = \boldsymbol{\Phi}(\boldsymbol{\xi}) = \begin{bmatrix}\phi_1(\boldsymbol{\xi}) & \phi_2(\boldsymbol{\xi}) & \phi_3(\boldsymbol{\xi})\end{bmatrix}^\top \tag{7}$$

Based on the definition of $\phi_i(\boldsymbol{\xi})$, we have $\boldsymbol{e}_s = \boldsymbol{0}$ if and only if $\boldsymbol{\xi} \in \mathcal{S}$.

The corresponding surface guiding vector field $\chi \in \mathbb{R}^5$ is given by:

$$\boldsymbol{\chi}(\boldsymbol{\xi}) = \boldsymbol{\chi}_t(\boldsymbol{\xi}) + \boldsymbol{\chi}_n(\boldsymbol{\xi}) \tag{8}$$

where $\boldsymbol{\chi}_t(\boldsymbol{\xi})$ is the *propagation term* and $\boldsymbol{\chi}_n(\boldsymbol{\xi})$ is the *convergence term*, each defined as follows [21]:

$$\begin{aligned}\boldsymbol{\chi}_t(\boldsymbol{\xi}) &= \wedge\left(\nabla\phi_1(\boldsymbol{\xi}), \nabla\phi_2(\boldsymbol{\xi}), \nabla\phi_3(\boldsymbol{\xi}), \boldsymbol{v}\right)\\ \boldsymbol{\chi}_n(\boldsymbol{\xi}) &= -\sum_{i=1}^{3} k_i\phi_i(\boldsymbol{\xi})\nabla\phi_i(\boldsymbol{\xi})\end{aligned} \tag{9}$$

where $\wedge(\cdot)$ is the wedge product [13], $\boldsymbol{v}$ is the extra vector can be chosen arbitrarily, $k_i > 0$ is the convergence gain. And the gradient vector $\nabla\phi_i(\boldsymbol{\xi})$ is:

$$\nabla\phi_i(\boldsymbol{\xi}) = \begin{bmatrix}\nabla\phi_i^{[1]} & \nabla\phi_i^{[2]} & \nabla\phi_i^{[3]} & -\frac{\partial f_i}{\partial\tau_1} & -\frac{\partial f_i}{\partial\tau_2}\end{bmatrix}^\top \tag{10}$$

where $\nabla\phi_i^{[i]} = 1$ and $\nabla\phi_i^{[j]} = 0$ for all $j \neq i$.

Evidently, $\boldsymbol{\chi}_t(\boldsymbol{\xi})$ is orthogonal to $\boldsymbol{\chi}_n(\boldsymbol{\xi})$ and lies within the tangent plane of the level set containing ξ. The specific direction of $\boldsymbol{\chi}_t(\boldsymbol{\xi})$ is determined by $\boldsymbol{v}$. For the sake of simplicity, let $\boldsymbol{v} = [0, 0, 0, v_4, v_5]$. Then the propagation term can be expressed as follows:

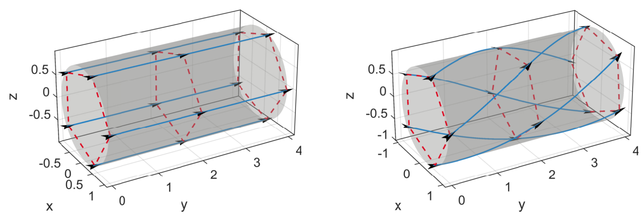
$$\boldsymbol{\chi}_t(\boldsymbol{\xi}) = -v_5\begin{bmatrix}\frac{\partial \boldsymbol{f}}{\partial\tau_1} & 1 & 0\end{bmatrix}^\top + v_4\begin{bmatrix}\frac{\partial \boldsymbol{f}}{\partial\tau_2} & 0 & 1\end{bmatrix}^\top \tag{11}$$

where $\frac{\partial \boldsymbol{f}}{\partial\tau_1} = [\frac{\partial f_1}{\partial\tau_1}, \frac{\partial f_2}{\partial\tau_1}, \frac{\partial f_3}{\partial\tau_1}]$ and $\frac{\partial \boldsymbol{f}}{\partial\tau_2} = [\frac{\partial f_1}{\partial\tau_2}, \frac{\partial f_2}{\partial\tau_2}, \frac{\partial f_3}{\partial\tau_2}]$.

Eq. (11) shows that the robot's motion on $\mathcal{S}^{ph}$ is a linear combination of $\frac{\partial \boldsymbol{f}}{\partial\tau_1}$ and $\frac{\partial \boldsymbol{f}}{\partial\tau_2}$, with the v_1/v_2 ratio determining the trajectory. Taking a cylindrical surface as an illustrative example, and its equation is given by:

$$\begin{cases} x = \cos\tau_1\\ y = \tau_2\\ z = \sin\tau_1\end{cases} \tag{12}$$

As shown in Fig. 3, when $v_4 = 1$ and $v_5 = 0$, the desired motion on $\mathcal{S}^{ph}$ is a straight line parallel to the y-axis. Increasing v_5 to 0.5 (with $v_4 = 1$) results in a helical trajectory on $\mathcal{S}^{ph}$.



(a) $v_4 = 1$ and $v_5 = 0$ (b) $v_4 = 1$ and $v_5 = 0.5$

Fig. 3: Different trajectories on cylinder for different choices of $\boldsymbol{v}$.

To achieve the desired virtual coordination differences defined in (3), a guiding vector field (GVF) incorporating a *coordination term* is constructed [21]:

$${}^i\boldsymbol{\chi}(\boldsymbol{\xi}) = {}^i\boldsymbol{\chi}_t(\boldsymbol{\xi}) + {}^i\boldsymbol{\chi}_n(\boldsymbol{\xi}) + {}^i\boldsymbol{\chi}_c(\boldsymbol{\xi}). \tag{13}$$

where superscript ${}^i(\cdot)$ denotes the i-th robot, and according to (9), ${}^i\boldsymbol{\chi}_t$ and ${}^i\boldsymbol{\chi}_n$ are defined as follows:

$$\begin{aligned}{}^i\boldsymbol{\chi}_t({}^i\boldsymbol{\xi}) &= -v_5\begin{bmatrix}\frac{\partial \boldsymbol{f}}{\partial {}^i\tau_1} & 1 & 0\end{bmatrix}^\top + v_4\begin{bmatrix}\frac{\partial \boldsymbol{f}}{\partial {}^i\tau_2} & 0 & 1\end{bmatrix}^\top && \text{(14a)}\\ {}^i\boldsymbol{\chi}_n({}^i\boldsymbol{\xi}) &= -\sum_{j=1}^{3} k_j\phi_j({}^i\boldsymbol{\xi})\nabla\phi_j({}^i\boldsymbol{\xi}) && \text{(14b)}\end{aligned}$$

A straightforward coordination strategy, following [21], introduces the coordination term ${}^i\boldsymbol{\chi}_c$ as:

$${}^i\boldsymbol{\chi}_c({}^i\boldsymbol{\xi}) = \begin{bmatrix} 0\\ 0\\ 0\\ k_{c1}\sum_{j=1}^{n} a_{ij}({}^i\tau_1 - {}^j\tau_1 - \Delta_1^{i,j})\\ k_{c2}\sum_{j=1}^{n} a_{ij}({}^i\tau_2 - {}^j\tau_2 - \Delta_2^{i,j})\end{bmatrix}^\top, \tag{15}$$

where k_{c1} and k_{c2} are the coordination gains.

Equation (15) provides a simple yet limited formulation of the coordination term, which guarantees convergence to the desired formation when the communication graph contains a directed spanning tree. However, since the terms defined in (14b) and (15) are not orthogonal, their interaction may cause conflicting effects, thereby slowing error convergence and degrading coordinated path-following performance.

III. MAIN RESULT

A. Decoupled Coordination Term

The *coordination error* of robot i is given by

$$\begin{aligned} {}^{i}e_{c1} &= \sum_{j=1}^{n} a_{ij}({}^{i}\tau_1 - {}^{j}\tau_1 - \Delta_1^{[i,j]}) \\ {}^{i}e_{c2} &= \sum_{j=1}^{n} a_{ij}({}^{i}\tau_2 - {}^{j}\tau_2 - \Delta_2^{[i,j]}) \end{aligned} \tag{16}$$

where a_{ij} is the element in the ith row and jth column of $\mathcal{A}$. The coordination errors of all robots are expressed as vectors $\boldsymbol{e}_{c1} = [{}^{1}e_{c1}, \cdots, {}^{n}e_{c1}]$ and $\boldsymbol{e}_{c2} = [{}^{1}e_{c2}, \cdots, {}^{n}e_{c2}]$. The corresponding consensus control algorithm is designed as follows:

$$\begin{aligned} {}^{i}c_1 &= -k_{c1}{}^{i}e_{c1} \\ {}^{i}c_2 &= -k_{c2}{}^{i}e_{c2} \end{aligned} \tag{17}$$

where k_{c1} and k_{c2} are the coordination gains.

Equation (15) is obtained by setting ${}^{i}\boldsymbol{\chi}_c(\boldsymbol{\xi}) = [0, 0, 0, {}^{i}c_1, {}^{i}c_2]^\top$. When combined with (14), this ensures both convergence and coordination (as shown in [21]). However, a critical prerequisite for this guarantee is the near-simultaneous convergence of $\|{}^{i}\boldsymbol{e}_s\|$, ${}^{i}e_{c1}$ and ${}^{i}e_{c2}$ to zero. For instance, consider the scenario where ${}^{i}\boldsymbol{e}_s = \boldsymbol{0}$ at point ${}^{i}\boldsymbol{\xi}$ but ${}^{i}e_{c1}$ and ${}^{i}e_{c2}$ are not both zero. In this case, the projection of ${}^{i}\boldsymbol{\chi}_c$ onto the gradient of the level set function ϕ_i becomes ${}^{i}\boldsymbol{\chi}_c \cdot \nabla\phi_i = -{}^{i}c_1(\partial f_i/\partial\tau_1) - {}^{i}c_2(\partial f_i/\partial\tau_2) \neq 0$. Consequently, ${}^{i}\boldsymbol{\chi}({}^{i}\boldsymbol{\xi})$ would drive ${}^{i}\boldsymbol{\xi}$ away from the desired manifold $\mathcal{S}$, leading to an increase in $\|{}^{i}\boldsymbol{e}_s\|$. Conversely, if ${}^{i}e_{c1} = {}^{i}e_{c2} = 0$ but $\|{}^{i}\boldsymbol{e}_s\| \neq 0$, guaranteeing the coordination error remains at zero necessitates that ${}^{1}\boldsymbol{\chi}_n^{[4]} = \cdots = {}^{n}\boldsymbol{\chi}_n^{[4]}$ and ${}^{1}\boldsymbol{\chi}_n^{[5]} = \cdots = {}^{n}\boldsymbol{\chi}_n^{[5]}$, which is rarely achieved in practice.

To eliminate this mutual influence and decouple the coordination term from the convergence term, two auxiliary vectors $\boldsymbol{s}_1, \boldsymbol{s}_2 \in \mathbb{R}^5$ are proposed for constructing the coordination term. The following properties are required for $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$:

$$\boldsymbol{s}_1 \cdot \nabla\phi_i(\boldsymbol{\xi}) = \boldsymbol{s}_2 \cdot \nabla\phi_i(\boldsymbol{\xi}) = 0, i = 1, 2, 3 \tag{18a}$$

$$\boldsymbol{s}_1 \cdot \boldsymbol{s}_2 \neq |\boldsymbol{s}_1||\boldsymbol{s}_2| \tag{18b}$$

REMARK 2. *For any $\boldsymbol{\xi}^* \in \mathbb{R}^5$, define the manifold it lies on as $\bar{\mathcal{S}} = \{\boldsymbol{\xi} | \phi_i(\boldsymbol{\xi}) = \phi_i(\boldsymbol{\xi}^*), i = 1, 2, 3\}$. Property (18a) indicates that $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ lie in the tangent space of $\bar{\mathcal{S}}$, while property (18b) shows that $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ are linearly independent. Hence, $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ can serve as a basis for representing vectors in the tangent space of $\bar{\mathcal{S}}$.*

To facilitate the construction of the coordination term, let $\boldsymbol{s}_1^{vi} = [1, 0]$ and $\boldsymbol{s}_2^{vi} = [0, 1]$. Then the corresponding expression for the auxiliary vectors are given in (19).

$$\begin{aligned} \boldsymbol{s}_1 &= \begin{bmatrix} \frac{\partial f_1}{\partial\tau_1} & \frac{\partial f_2}{\partial\tau_1} & \frac{\partial f_3}{\partial\tau_1} & 1 & 0 \end{bmatrix}^\top \\ \boldsymbol{s}_2 &= \begin{bmatrix} \frac{\partial f_1}{\partial\tau_2} & \frac{\partial f_2}{\partial\tau_2} & \frac{\partial f_3}{\partial\tau_2} & 0 & 1 \end{bmatrix}^\top \end{aligned} \tag{19}$$

Using $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ to construct the coordination term:

$${}^{i}\boldsymbol{\chi}_c(\boldsymbol{\xi}) = ({}^{i}c_1 - {}^{i}\boldsymbol{\chi}_n^{[4]}(\boldsymbol{\xi}))\boldsymbol{s}_1 + ({}^{i}c_2 - {}^{i}\boldsymbol{\chi}_n^{[5]}(\boldsymbol{\xi}))\boldsymbol{s}_2 \tag{20}$$

REMARK 3. *The interaction between the convergence term and the consensus term is mutual. In (20), the coordination term is constructed within the tangent space using $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$, thereby minimizing their interference with the convergence behavior. Meanwhile, the terms $-{}^{i}\boldsymbol{\chi}_n^{[4]}(\boldsymbol{\xi})$ and $-{}^{i}\boldsymbol{\chi}_n^{[5]}(\boldsymbol{\xi})$ act as feedforward components, reducing the influence of the convergence term on the coordination behavior and thus achieving decoupling between the two.*

To clearly demonstrate the necessity of decoupling the convergence and coordination terms, we consider a simplified case of two robots (indexed 0 and 1) moving cooperatively along a circular parametric path defined as:

$$\mathcal{S} = \{\boldsymbol{\xi} | \phi_1(\boldsymbol{\xi}) = 0, \phi_2(\boldsymbol{\xi}) = 0\}$$

where $\boldsymbol{\xi} = [x, y, \tau]^\top$, $\phi_1(\boldsymbol{\xi}) = x - \cos(2\tau)$ and $\phi_2(\boldsymbol{\xi}) = y - \sin(2\tau)$. The corresponding propagation term and convergence term are defined as:

$$\begin{aligned} {}^{i}\boldsymbol{\chi}_t({}^{i}\boldsymbol{\xi}) &= \begin{bmatrix} -2\sin(2{}^{i}\tau) & 2\cos(2{}^{i}\tau) & 1 \end{bmatrix}^\top \\ {}^{i}\boldsymbol{\chi}_n({}^{i}\boldsymbol{\xi}) &= -10\phi_1({}^{i}\boldsymbol{\xi}) \begin{bmatrix} 1 & 0 & 2\sin(2{}^{i}\tau) \end{bmatrix}^\top - \\ &\quad 10\phi_2({}^{i}\boldsymbol{\xi}) \begin{bmatrix} 0 & 1 & -2\cos(2{}^{i}\tau) \end{bmatrix}^\top \end{aligned}$$

Let $\Delta = 0$. The baseline coordination term ${}^{i}\boldsymbol{\chi}_c^o$ and the decoupled one ${}^{i}\boldsymbol{\chi}_c^d$ as follow:

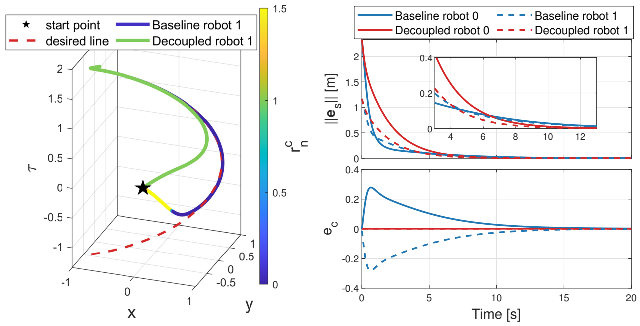
$$\begin{aligned} {}^{i}\boldsymbol{\chi}_c^o &= -\begin{bmatrix} 0 & 0 & c({}^{i}\tau) \end{bmatrix}^\top \\ {}^{i}\boldsymbol{\chi}_c^d &= -\left(c({}^{i}\tau) - \boldsymbol{\chi}_n^{[3]}({}^{i}\boldsymbol{\xi}) \right) \boldsymbol{s} \end{aligned} \tag{21}$$

where $c({}^{i}\tau) = 10({}^{i}\tau - {}^{1-i}\tau)$, and $\boldsymbol{s} = [-2\sin(2{}^{i}\tau), 2\cos(2{}^{i}\tau), 1]^\top$. The path following error is ${}^{i}\boldsymbol{e} = [\phi_1({}^{i}\boldsymbol{\xi}), \phi_2({}^{i}\boldsymbol{\xi})]^\top$, coordination error is ${}^{i}e_c = {}^{i}\tau - {}^{1-i}\tau$. We consider two distinct initial cases:

1) **Case 1:** Random initial positions with aligned coordination parameters (i.e. ${}^{0}\tau_0 = {}^{1}\tau_0$), implying no initial coordination errors.
2) **Case 2:** Random initial virtual coordinate ${}^{i}\tau$ with with initial position defined by ${}^{i}\boldsymbol{\xi}_0 = [\cos(2{}^{i}\tau_0), \sin(2{}^{i}\tau_0), {}^{i}\tau_0]$, implying no initial path following errors.

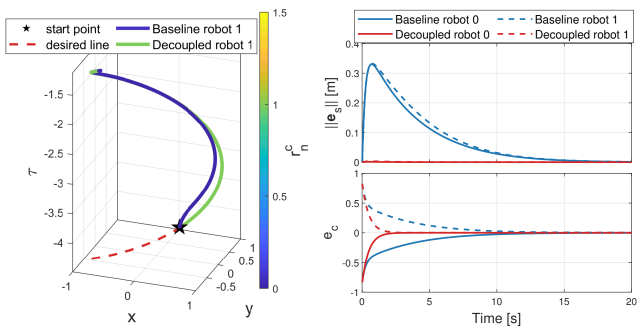
Additionally, define $r_n^c = 1 + (\boldsymbol{\chi}_n\boldsymbol{\chi}_c)/(\boldsymbol{\chi}_n\boldsymbol{\chi}_n)$, which characterizes the interaction between $\boldsymbol{\chi}_n$ and $\boldsymbol{\chi}_c$. Specifically, when $r_n^c = 1$, the two components are orthogonal and do not influence each other. If $r_n^c < 1$, $\boldsymbol{\chi}_c$ projects onto the negative direction of $\boldsymbol{\chi}_n$, indicating an antagonistic relationship. Conversely, when $r_n^c > 1$, the two components mutually reinforce each other.

The simulation result of Case 1 is presented in Fig. 4. As shown in Fig. 4a, the decoupled coordination term ${}^{i}\boldsymbol{\chi}_c^d$ guarantees orthogonality with ${}^{i}\boldsymbol{\chi}_n$, thereby ensures $r_n^c = 1$. In contrast, the baseline GVF exhibits a rapid shift from a reinforcing to an antagonistic relationship between ${}^{i}\boldsymbol{\chi}_c^o$ and ${}^{i}\boldsymbol{\chi}_n$. Consequently, Fig 4b reveals that although the baseline GVF achieves a faster initial decrease in $\|\boldsymbol{e}_s\|$, its overall convergence is slower than that of the decoupled approach. Moreover, the baseline GVF's convergence behavior leads to an increase in e_c due to the inability to guarantee ${}^{0}\boldsymbol{\chi}_n^{[3]} = {}^{1}\boldsymbol{\chi}_n^{[3]}$, whereas



(a) Trajectories of robot 1. (b) Variation of $\|\boldsymbol{e}_s\|$ and e_c.

Fig. 4: Simulation result of Case 1: no initial coordination error.



(a) Trajectories of robot 1. (b) Variation of $\|\boldsymbol{e}_s\|$ and e_c.

Fig. 5: Simulation result of Case 2: no initial path following error.

the feedforward term in ${}^i\boldsymbol{\chi}_c^d$ maintains e_c at zero by eliminating the influence of ${}^i\boldsymbol{\chi}_n$.

Similar conclusions are drawn from the results of Case 2 (Fig. 5). The non-decoupled coordination term ${}^1\boldsymbol{\chi}_c^o$ exhibits an antagonistic relationship with ${}^1\boldsymbol{\chi}_n$ throughout the robot's motion (Fig. 5a). This significantly impedes its coordination error convergence compared to the decoupled approach, as shown in Fig. 5b. Furthermore, the non-decoupled coordination term also leads to an increase in $\|\boldsymbol{e}_s\|$, a phenomenon not observed with the decoupled vector field.

THEOREM 1 (Path-following). *Consider the autonomous system* ${}^i\dot{\boldsymbol{\xi}}(t) = {}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi})(t)$, *where* ${}^i\boldsymbol{\chi}(\boldsymbol{\xi}_i)$ *is defined by* (13), (14) *and* (20). *Then* $\forall\, {}^i\boldsymbol{\xi}_0 \in \mathbb{R}^5$, ${}^i\boldsymbol{\xi}(t)$ *globally converges to the set* $\mathcal{S}$ *as* $t \to \infty$.

Proof:
Select the Lyapunov function ${}^iV_s = \frac{1}{2}{}^i\boldsymbol{e}_s{}^\top \boldsymbol{K} \cdot {}^i\boldsymbol{e}_s$. Its derivative can be calculated:

$$
\begin{aligned}
{}^i\dot{V}_s &= {}^i\boldsymbol{\Phi}^\top \boldsymbol{K} \nabla^i\boldsymbol{\Phi}\ {}^i\dot{\boldsymbol{\xi}} \\
&\overset{(13),(14)}{=} -{}^i\boldsymbol{\chi}_n^\top \left({}^i\boldsymbol{\chi}_t(\boldsymbol{\xi}) + {}^i\boldsymbol{\chi}_n(\boldsymbol{\xi}) + {}^i\boldsymbol{\chi}_c(\boldsymbol{\xi})\right)
\end{aligned}
$$

where $\boldsymbol{K} = \operatorname{diag}\{k_1, k_2, k_3\}$ is a diagonal matrix composed of the gains of the convergence term, $\nabla^i\boldsymbol{\Phi} = [\nabla^i\phi_1, \nabla^i\phi_2, \nabla^i\phi_3]^\top$.

Due to the orthogonality of $\boldsymbol{\chi}_n$ with $\boldsymbol{\chi}_t$ and $\boldsymbol{\chi}_c$, the above equation can be simplified to:

$$
\begin{aligned}
{}^i\dot{V}_s &= -{}^i\boldsymbol{\chi}_n^\top {}^i\boldsymbol{\chi}_n \\
&= -\left(\sum_{j=1}^{3} k_j^{2\,i}\phi_j^2 + {}^i\boldsymbol{\chi}_n^{[4]^2} + {}^i\boldsymbol{\chi}_n^{[5]^2}\right) \le 0
\end{aligned}
$$

Substituting (7), (10) and (14b), ${}^i\dot{V}_s$ is expressed as:

$$
{}^i\dot{V}_s = -\sum_{j=1}^{3}\left(\left({}^i\boldsymbol{e}_s^{[j]}k_j\right)^2 + \left({}^i\boldsymbol{e}_s^{[j]}\frac{\partial f_j}{\partial \tau_1}\right)^2 + \left({}^i\boldsymbol{e}_s^{[j]}\frac{\partial f_j}{\partial \tau_2}\right)^2\right) \le 0
$$

The equality holds if and only if ${}^i\boldsymbol{e}_s = \mathbf{0}$. According to Lyapunov stability criterion, $\lim_{t\to\infty}{}^i\boldsymbol{e}_s = 0$, which means ${}^i\boldsymbol{\xi}$ converges to the set $\mathcal{S}$ as $t \to \infty$. ■

THEOREM 2 (Motion coordination). *Consider the autonomous system* ${}^i\dot{\boldsymbol{\xi}}(t) = {}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi}(t))$, *where* ${}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi})$ *is defined by* (13), (14) *and* (20). *Suppose the communication graph* $\mathcal{G}$ *among the robots has a directed spanning tree, and* $\boldsymbol{\Delta}_1$ *and* $\boldsymbol{\Delta}_2$ *are given. Then* $\forall\, {}^i\boldsymbol{\xi}_0 \in \mathbb{R}^5$, *the error vectors* $\mathbf{e}_{c1}, \mathbf{e}_{c2}$ *converge to* $\mathbf{0}$ *as* $t \to \infty$.

Proof:
Derivation of (16) yields ${}^i\dot{e}_{ck} = \sum_{j=1}^n a_{ij}({}^i\dot{\tau}_k - {}^j\dot{\tau}_k)$ for $k = 1, 2$. It is known that ${}^i\dot{\tau}_1 = {}^i\dot{\boldsymbol{\xi}}^{[4]}$ and ${}^i\dot{\tau}_2 = {}^i\dot{\boldsymbol{\xi}}^{[5]}$. By substituting (13), (14a) and (20), we obtain ${}^i\dot{\tau}_1 = -v_5 + {}^ic_1$ and ${}^i\dot{\tau}_2 = v_4 + {}^ic_2$. Further substituting (17) into the expressions for ${}^i\dot{e}_{ck}$, we have:

$$
\begin{aligned}
{}^i\dot{e}_{c1} &= -k_{c1}\sum_{j=1}^{n} a_{ij}({}^ie_{c1} - {}^je_{c1}) \\
{}^i\dot{e}_{c2} &= -k_{c2}\sum_{j=1}^{n} a_{ij}({}^ie_{c2} - {}^je_{c2})
\end{aligned}
$$

These equations can be expressed compactly in matrix form as:

$$
\begin{aligned}
\dot{\boldsymbol{e}}_{c1} &= -k_{c1}\boldsymbol{\mathcal{L}}\boldsymbol{e}_{c1} \\
\dot{\boldsymbol{e}}_{c2} &= -k_{c2}\boldsymbol{\mathcal{L}}\boldsymbol{e}_{c2}
\end{aligned}
$$

According to Lemma 1, as $t \to \infty$, $\boldsymbol{e}_{c1}$ and $\boldsymbol{e}_{c2}$ converge to $\mathbf{0}$. ■

THEOREM 3 (Decoupling). *Consider the autonomous system* ${}^i\dot{\boldsymbol{\xi}}(t) = {}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi})(t)$, *where* ${}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi})$ *is defined by* (13), (14) *and* (20). *Then the following statements holds:*

1) *If the convergence error satisfies* ${}^i\boldsymbol{e}_s({}^i\boldsymbol{\xi}(t_p)) = \mathbf{0}$ *at time* t_p, *then for any* $t > t_p$ *it holds that* ${}^i\boldsymbol{e}_s({}^i\boldsymbol{\xi}(t)) = \mathbf{0}$.
2) *If the coordination error satisfies* ${}^ie_{c1}({}^i\boldsymbol{\xi}(t_c)) = 0$ *(or* ${}^ie_{c2}({}^i\boldsymbol{\xi}(t_c)) = 0$*) at time* t_c, *then for any* $t > t_c$ *it holds that* ${}^ie_{c1}({}^i\boldsymbol{\xi}(t)) = 0$ *(or* ${}^ie_{c2}({}^i\boldsymbol{\xi}(t)) = 0$*).*

Proof:

Differentiating both sides of (7), we obtain ${}^i\dot{\boldsymbol{e}}_s = \nabla{}^i\boldsymbol{\Phi}{}^i\dot{\boldsymbol{\xi}}$. Substituting ${}^i\dot{\boldsymbol{\xi}}$ with ${}^i\boldsymbol{\chi}({}^i\boldsymbol{\xi})$, and utilizing the orthogonality of $\nabla\phi_k$ with ${}^i\boldsymbol{\chi}_n$ and ${}^i\boldsymbol{\chi}_c$, we have:

$$\begin{aligned}{}^i\dot{\boldsymbol{e}}_s &= \left[\nabla{}^i\phi_1({}^i\boldsymbol{\xi}) \quad \nabla{}^i\phi_2({}^i\boldsymbol{\xi}) \quad \nabla{}^i\phi_3({}^i\boldsymbol{\xi})\right]^\top {}^i\boldsymbol{\chi}_n \\ &= -k\nabla{}^i\boldsymbol{\Phi}^\top \mathrm{diag}({}^i\boldsymbol{e}_s)\nabla{}^i\boldsymbol{\Phi}\end{aligned}$$

When $t = t_p$, ${}^i\boldsymbol{e}_s = 0$, which implies ${}^i\dot{\boldsymbol{e}}_s = 0$. Thus, for any $t > t_p$, $\int_{t_p}^t {}^i\boldsymbol{e}_s dt = 0$, and Statements 1 holds.

Similarly, according to Theorem 2, we obtain:

$$\begin{aligned}\dot{\boldsymbol{e}}_{c1} &= -k_{c1}\mathcal{L}\boldsymbol{e}_{c1} \\ \dot{\boldsymbol{e}}_{c2} &= -k_{c2}\mathcal{L}\boldsymbol{e}_{c2}\end{aligned}$$

Thus, once $\boldsymbol{e}_{ck}$ converges to $\mathbf{0}$ at $t = t_c$, it follows that $\dot{\boldsymbol{e}}_{ck} = 0$, thereby statement 2 holds. ■

B. Non-uniform GVF for Speed Coordination

Traditional approaches typically use GVF to guide the desired movement direction, while robots maintain speeds based on external requirements or a constant value. Which mean the velocity $\boldsymbol{v}_i$ of robot i is:

$${}^i\boldsymbol{v} = {}^iv(t)\frac{{}^i\boldsymbol{\chi}^{ph}({}^i\boldsymbol{\xi})}{||{}^i\boldsymbol{\chi}^{ph}({}^i\boldsymbol{\xi})||}$$

where ${}^iv(t)$ is the speed of robot i at t, ${}^i\boldsymbol{\chi}^{ph}({}^i\boldsymbol{\xi}) = [{}^i\boldsymbol{\chi}^{[1]}({}^i\boldsymbol{\xi}), {}^i\boldsymbol{\chi}^{[2]}({}^i\boldsymbol{\xi}), {}^i\boldsymbol{\chi}^{[3]}({}^i\boldsymbol{\xi})]$ represents the components of the vector field in the physical space. And according to (14a), the corresponding virtual coordinate speed is give by:

$${}^i\boldsymbol{v}^{vi} = \left[-\frac{v_5}{||{}^i\boldsymbol{\chi}^{ph}({}^i\boldsymbol{\xi})||} \quad \frac{v_4}{||{}^i\boldsymbol{\chi}^{ph}({}^i\boldsymbol{\xi})||}\right]^\top$$

However, this approach is incompatible with coordination via virtual coordinates, particularly when robots follow trajectories with different curvatures. To simplify the analysis, assume a group of n robots has already converged to the desired surface, with ${}^i\boldsymbol{e}_s = \mathbf{0}$ and ${}^ie_{c1} = {}^ie_{c2} = 0$. The coordination errors for robot i are derived as:

$$\begin{aligned}{}^i\dot{e}_{c1} &= -v_5\sum_{j=1}^n a_{ij}\left(\frac{1}{\|{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi})\|} - \frac{1}{\|{}^j\boldsymbol{\chi}_t^{ph}({}^j\boldsymbol{\xi})\|}\right) \\ {}^i\dot{e}_{c2} &= v_4\sum_{j=1}^n a_{ij}\left(\frac{1}{\|{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi})\|} - \frac{1}{\|{}^j\boldsymbol{\chi}_t^{ph}({}^j\boldsymbol{\xi})\|}\right)\end{aligned}$$

Combining the errors of all robots into a vector, the above equation simplifies to the following compact form:

$$\begin{aligned}\dot{\boldsymbol{e}}_{c1} &= \mathcal{L}\left[\frac{1}{\|{}^1\boldsymbol{\chi}_t^{ph}({}^1\boldsymbol{\xi})\|} \quad \cdots \quad \frac{1}{\|{}^n\boldsymbol{\chi}_t^{ph}({}^n\boldsymbol{\xi})\|}\right]^\top \\ \dot{\boldsymbol{e}}_{c2} &= \mathcal{L}\left[\frac{1}{\|{}^1\boldsymbol{\chi}_t^{ph}({}^1\boldsymbol{\xi})\|} \quad \cdots \quad \frac{1}{\|{}^n\boldsymbol{\chi}_t^{ph}({}^n\boldsymbol{\xi})\|}\right]^\top\end{aligned}$$

where $\|{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi})\| = \sqrt{v_5^2\left(\partial\boldsymbol{f}/\partial{}^i\tau_1\right)^2 + v_4^2\left(\partial\boldsymbol{f}/\partial{}^i\tau_2\right)^2}$ according to (14b).

Based on the Property 2 of Lemma 1, the condition for $\dot{\boldsymbol{e}}_{ck} = \mathbf{0}$ is that $\|{}^1\boldsymbol{\chi}_t^{ph}({}^1\boldsymbol{\xi})\| = \cdots = \|{}^n\boldsymbol{\chi}_t^{ph}({}^n\boldsymbol{\xi})\|$. If this equality holds, it ensures that $\boldsymbol{e}_{c1} = \boldsymbol{e}_{c2} = \mathbf{0}$, thus maintaining formation stability. However, due to the arbitrary nature of the desired surface, this condition cannot be strictly guaranteed. More detrimentally, without the decoupled coordination term presented in (20), the accumulation of ${}^ie_{c1}$ and ${}^ie_{c2}$ could adversely impact the tracking performance, resulting in an increase in ${}^i\boldsymbol{e}_s$.

To address the problem of unstable formation maintenance caused by virtual coordinate speed mismatch, which arises from vector field normalization, we propose a non-uniform guiding vector field approach. This method establishes a reference point ${}^i\boldsymbol{\xi}^*$ for each robot. By applying the predefined desired speed to the physical components of ${}^i\boldsymbol{\xi}^*$ and inversely deriving the desired speed of robot i, we ensure its virtual coordinate velocity matches the reference point. Subsequently, the demonstrated convergence of each robot's reference point to a common location ensures stable formation movement along the preset speed.

Define *coordination speed* as follows:

$${}^iv = v_c\frac{||{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi})||}{||{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi}^*)||} \tag{22}$$

where $v_c \in \mathbb{R}$ is the desired speed of the robot team, ${}^i\boldsymbol{\xi}^* = [{}^i\boldsymbol{x}^*, {}^i\tau_1^*, {}^i\tau_2^*]^\top$ is the reference point, and ${}^i\boldsymbol{x}^* = [f_1({}^i\tau_1^*, {}^i\tau_2^*), f_2({}^i\tau_1^*, {}^i\tau_2^*), f_3({}^i\tau_1^*, {}^i\tau_2^*)]^\top$, where ${}^i\tau_1^*$ and ${}^i\tau_2^*$ are defined as follows:

$$\begin{aligned}{}^i\tau_1^* &= {}^i\tau_1 - {}^i\Delta_1^* \\ {}^i\tau_2^* &= {}^i\tau_2 - {}^i\Delta_2^*\end{aligned} \tag{23}$$

where ${}^i\Delta_j^* = \sum_{k=1}^n \Delta_j^{[i,k]}/n,\ j = 1,2$ is the desired difference between the i-th robot's virtual coordinate and its' reference virtual coordinate.

Then the speed offset ratio is defined as:

$${}^ir = \frac{{}^iv}{v_c} = \frac{||{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi})||}{||{}^i\boldsymbol{\chi}_t^{ph}({}^i\boldsymbol{\xi}^*)||} \tag{24}$$

and the corresponding non-uniform guiding vector field is:

$${}^i\hat{\boldsymbol{\chi}} = {}^i\hat{\boldsymbol{\chi}}_n + {}^i\hat{\boldsymbol{\chi}}_t + {}^i\hat{\boldsymbol{\chi}}_c \tag{25}$$

with:

$${}^i\hat{\boldsymbol{\chi}}_t = v_c\frac{{}^ir{}^i\boldsymbol{\chi}_t}{||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}||} \tag{26a}$$

$${}^i\hat{\boldsymbol{\chi}}_n = v_c\frac{{}^i\boldsymbol{\chi}_n}{||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}||} \tag{26b}$$

$${}^i\hat{\boldsymbol{\chi}}_c = v_c\frac{({}^ir{}^ic_1 - \boldsymbol{\chi}_n^{[4]})\boldsymbol{s}_1 + ({}^ir{}^ic_2 - \boldsymbol{\chi}_n^{[5]})\boldsymbol{s}_2}{||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}||} \tag{26c}$$

where the denominator, notably, does not include ${}^i\boldsymbol{\chi}_c^{ph}$ to prevent the normalization process from negating the effect of the coordination term when ${}^i\boldsymbol{\chi}_c$ is aligned with ${}^i\boldsymbol{\chi}_t$.

ASSUMPTION 1. *For all* $\tau_1, \tau_2 \in \mathbb{R}$, *it holds that* $\|v_4\partial\boldsymbol{f}/\partial\tau_2 - v_5\partial\boldsymbol{f}/\partial\tau_1\| \neq 0$

Assumption 1 is crucial as it precludes the physical space component of the propagation term from vanishing. For normalized vector fields, this assumption remains necessary, further requiring a stronger condition: $\|{}^i\boldsymbol{\chi}^{ph}\| \neq 0$.

This article has been accepted for publication in IEEE Transactions on Aerospace and Electronic Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAES.2026.3659453

In practice, to avoid a zero denominator in (26), the gain of the convergence term can be adaptively increased or decreased when $||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}||$ approaches zero.

To facilitate the analysis of the non-uniform vector field, the properties of the reference virtual coordinates are first introduced.

LEMMA 2. *Assuming the reference virtual coordinate ${}^i\tau_1^*$ and ${}^i\tau_2^*$ are defined by (23), $\boldsymbol{\Delta}_1$ and $\boldsymbol{\Delta}_2$ are defined by (3). Then, for $k = 1, 2$, the following statements are equivalent:*

1) ${}^1\tau_k^* = \cdots = {}^n\tau_k^*$.
2) ${}^i\tau_k - {}^j\tau_k = \Delta_k^{[i,j]}$ *for all* $i, j \in \{1, \cdots, n\}$.

Proof:
According to the definition of ${}^i\Delta_k^*$, we have:

$$\begin{aligned} {}^i\tau_k^* - {}^j\tau_k^* &= {}^i\tau_k - {}^j\tau_k - \left(\sum_{m=1}^{n} \frac{\Delta_k^{[i,m]} - \Delta_k^{[j,m]}}{n}\right) \\ &\overset{(4c)}{=} {}^i\tau_k - {}^j\tau_k - \Delta_k^{[i,j]} \end{aligned}$$

From this derived equality, it is evident that ${}^i\tau_k^* = {}^j\tau_k^*$ for all i, j if and only if ${}^i\tau_k - {}^j\tau_k = \Delta_k^{[i,j]}$ for all i, j. Thus, the two statements are equivalent. ∎

The following analysis shows that the proposed non-uniform guiding vector field (25) achieves motion coordination while ensuring convergence to uniform virtual coordinate speeds.

THEOREM 4. *Consider the system ${}^i\dot{\boldsymbol{\xi}} = {}^i\hat{\boldsymbol{\chi}}({}^i\boldsymbol{\xi})$, where $\hat{\boldsymbol{\chi}}$ is defined by (25) and (26). For all initial conditions ${}^i\boldsymbol{\xi}_0 \in \mathbb{R}^5$, the following three propositions hold as $t \to \infty$:*

1) *The path-following error converge to zero, i.e., ${}^i\boldsymbol{e}^s \to \boldsymbol{0}$, where $i \in \mathbb{Z}_1^n$.*
2) *The coordination errors converge to zero, i.e., ${}^ie_{c1}, {}^ie_{c2} \to 0$, where $i \in \mathbb{Z}_1^n$.*
3) *Robots moving on $\mathcal{S}$ with identical virtual coordinate velocities, i.e. ${}^i\dot{\tau}_1 = {}^j\dot{\tau}_1$ and ${}^i\dot{\tau}_2 = {}^j\dot{\tau}_2$, for all $i, j \in \mathbb{Z}_1^n$.*

Proof:
For the first proposition, a similar analysis in the proof of Theorem 1 can be applied. Consider the Lyapunov function ${}^iV_s = \frac{1}{2}{}^i\boldsymbol{e}_s^\top \cdot \boldsymbol{K} \cdot {}^i\boldsymbol{e}_s \geq 0$, the corresponding differential equation is:

$$\begin{aligned} {}^i\dot{V}_s =& {}^i\boldsymbol{\Phi}^\top \boldsymbol{K} \nabla\, {}^i\boldsymbol{\Phi}\, {}^i\dot{\boldsymbol{\xi}} \\ \overset{(26)}{=}& - \frac{v_c}{||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}||}\left(\sum_{j=1}^{3} k_j^2 {}^i\phi_j{}^2 + {}^i\boldsymbol{\chi}_n^{[4]}{}^2 + {}^i\boldsymbol{\chi}_n^{[5]}{}^2\right) \end{aligned}$$

Since $v_c/||{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}|| > 0$, it follows that ${}^i\dot{V}_s \leq 0$ and equality holds if and only if ${}^i\boldsymbol{e}_s = \boldsymbol{0}$. According to Lyapunov stability criterion, the first proposition holds.

As for the second proposition, we still rely on Lemma 1 for the proof. The derivative of virtual coordinate is:

$$\begin{aligned} {}^i\dot{\tau}_1 &= \frac{v_c{}^ir(-v_5 + c_1)}{\|{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}\|} \\ {}^i\dot{\tau}_2 &= \frac{v_c{}^ir(v_4 + c_2)}{\|{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}\|} \end{aligned} \tag{27}$$

Let $\hat{\mathcal{L}} = \hat{\boldsymbol{K}}\mathcal{L}$ where $\hat{\boldsymbol{K}} = \mathrm{diag}(v_c{}^1r, \cdots, v_c{}^nr)/\|{}^i\boldsymbol{\chi}_t^{ph} + {}^i\boldsymbol{\chi}_n^{ph}\|$, $\hat{\mathcal{L}}$ still satisfies (6). Combining all virtual coordinate derivatives, Eq. (27) can be written as follows:

$$\begin{aligned} \dot{\boldsymbol{\tau}}_1 &= \hat{\mathcal{L}}\boldsymbol{\tau}_1^* \\ \dot{\boldsymbol{\tau}}_2 &= \hat{\mathcal{L}}\boldsymbol{\tau}_2^* \end{aligned}$$

Based on Lemma 1, the components of $\boldsymbol{\tau}_1^*$ (and $\boldsymbol{\tau}_2^*$) converge to a common value. This, combined with Lemma 2 ensuring that ${}^i\tau_1$ and ${}^i\tau_2$ converge to their desired differences, leads to ${}^ie_{c1}$ and ${}^ie_{c2}$ converges to 0. Thus the second proposition holds.

Based on the first two propositions, it is derived that $\lim_{t\to\infty} {}^i\dot{\boldsymbol{\xi}}^{ph} = {}^i\hat{\boldsymbol{\chi}}_t^{ph} = [-v_5, v_4]$, the third proposition holds. ∎

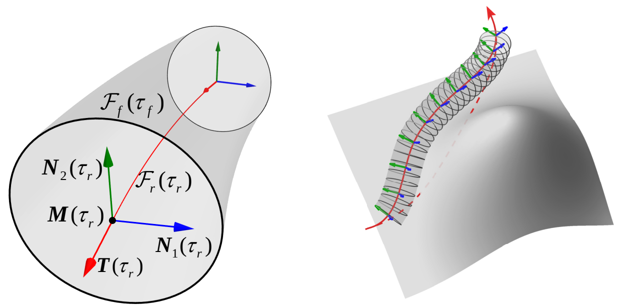
REMARK 4. *The proposed vector field defined by (26) addresses the formation instability associated with normalized GVFs without requiring additional communication. While directly using unnormalized vector fields (constructed from (13), (14), and (20)) can also achieve a stable formation, the inherent variation of $|{}^i\boldsymbol{\chi}_t^{ph}|$ during robot motion makes it difficult to regulate the overall formation speed. In contrast, the proposed non-uniform GVF explicitly introduces a speed modulation mechanism that preserves formation stability while allowing the collective motion speed to be specified or adjusted in a controlled manner.*

C. Desired Surface Generation via PT-Frame

Parametric-curve-based path planning problems have been extensively studied. To address the challenges posed by obstacles, we propose a moving-frame-based desired surface generation method. As illustrated in Fig. 6a, this method establishes a continuous moving frame $\boldsymbol{M}$ along a reference curve $\mathcal{F}_r(\tau_r) = [f_{r,1}(\tau_r), f_{r,2}(\tau_r), f_{r,3}(\tau_r)]^\top$. Subsequently, a formation curve $\mathcal{F}_f(\tau_f) = [f_{f,1}(\tau_f), f_{f,2}(\tau_f), f_{f,3}(\tau_f)]^\top$ is prescribed to move along this reference curve, thereby generating the desired surface $\mathcal{F}(\tau_r, \tau_f)$. By applying path planning methods to $\mathcal{F}_r(\tau_r)$, an obstacle-free surface $\mathcal{F}(\tau_r, \tau_f)$ is established, as shown in Fig. 6b. The expression of $\mathcal{F}(\tau_r, \tau_f)$ is given by:

$$\mathcal{F}(\tau_r, \tau_f) = \mathcal{F}_r(\tau_r) + \boldsymbol{M}(\tau_r)\mathcal{F}_f(\tau_f) \tag{28}$$

where $\tau_r, \tau_f \in \mathbb{R}$ are the parameters of $\mathcal{F}_r$ and $\mathcal{F}_f$, respectively. $\boldsymbol{M}(\tau_r) \in \mathbb{R}^{3\times 3}$ is the rotation matrix whose columns are the basis vectors of the moving frame expressed in the inertial frame. The component functions $f_{r,i}$ and $f_{f,i}$ for $i = 1, 2, 3$ belong to C^2.



(a) (b)

Fig. 6: Diagram of PT-frame-based surface construction method. The red solid lines denote the reference curve $\mathcal{F}_r(\tau_r)$ and the dashed one in (b) is the original reference curve, the black curve denotes the formation curve $\mathcal{F}_f(\tau_f)$, the frames consisting of red, blue and green vectors are the moving frames. (a) Components of the moving-frame-based desired surface. (b) Illustration of obstacle avoidance through reference curve replanning.

The moving frame defines the rotational motion of $\mathcal{F}_f$ along $\mathcal{F}_r$. The Frenet frame is a moving frame commonly used in autonomous driving and trajectory planning [23]–[25]. However, it becomes undefined when the curvature vanishes. The PT-frame offers an robust approach, ensuring a well-defined moving frame even when the curve's second derivative vanishes [26].

PT-frame consists of mutually orthogonal unit vectors $\boldsymbol{T}$, $\boldsymbol{N}_1$ and $\boldsymbol{N}_2$, where $\boldsymbol{T}$ is the tangent unit vector of curve $\mathcal{F}_r(\tau_r)$. The rotation matrix is then given by:

$$\boldsymbol{M}(\tau_r) = \begin{bmatrix} \boldsymbol{T}(\tau_r) \\ \boldsymbol{N}_1(\tau_r) \\ \boldsymbol{N}_2(\tau_r) \end{bmatrix}^\top \tag{29}$$

The corresponding derivative satisfies the following differential equation:

$$\begin{bmatrix} \dot{\boldsymbol{T}}(\tau_r) \\ \dot{\boldsymbol{N}}_1(\tau_r) \\ \dot{\boldsymbol{N}}_2(\tau_r) \end{bmatrix} = \frac{ds}{d\tau_r} \begin{bmatrix} 0 & k_1(\tau_r) & k_2(\tau_r) \\ -k_1(\tau_r) & 0 & 0 \\ -k_2(\tau_r) & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{T}(\tau_r) \\ \boldsymbol{N}_1(\tau_r) \\ \boldsymbol{N}_2(\tau_r) \end{bmatrix} \tag{30}$$

where $\frac{ds}{d\tau_r} = \sqrt{\dot{f}_{r,1}^2 + \dot{f}_{r,2}^2 + \dot{f}_{r,3}^2}$ is the arc length, $k_1(\tau_r)$ and $k_2(\tau_r)$ satisfy the following equations:

$$\kappa(\tau_r) = \sqrt{{k_1}^2 + {k_2}^2} \tag{31}$$

$$\theta(\tau_r) = \arctan\frac{k_1}{k_2} \tag{32}$$

$$t(\tau_r) = -\frac{d\theta(\tau_r)}{d\tau_r} \tag{33}$$

where $\kappa(\tau_r)$ is the scalar curvature, θ is the total rotation accumulated along the curve, and $t(\tau_r)$ is the torsion. By choosing different integrating constant θ_0 for $\theta = -\int t(\tau_r)d\tau_r + \theta_0$, different initial PT-frame can be obtained. Specifically, when $\theta_0 = 0$, the initial PT-frame coincides with the Frenet frame.

κ and t are, in principle, calculated in terms of the parameterized or numerical local values of $\mathcal{F}_r(\tau_r)$ and its first three derivatives as follows:

$$\begin{aligned} \kappa(\tau_r) &= \frac{||\dot{\mathcal{F}}_r(\tau_r) \times \ddot{\mathcal{F}}_r(\tau_r)||}{||\dot{\mathcal{F}}_r(\tau_r)||^3} \\ t(\tau_r) &= \frac{(\dot{\mathcal{F}}_r(\tau_r) \times \ddot{\mathcal{F}}_r(\tau_r)) \cdot \mathcal{F}_r^{(3)}(\tau_r)}{||\dot{\mathcal{F}}_r(\tau_r) \times \ddot{\mathcal{F}}_r(\tau_r)||^2} \end{aligned} \tag{34}$$

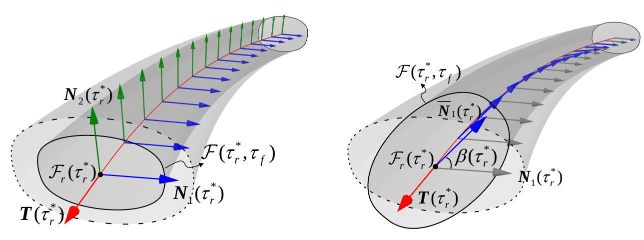
The values $k_1(\tau_r)$ and $k_2(\tau_r)$ are obtained by solving (31) to (34). Then the partial derivative of $\mathcal{F}(\tau_r, \tau_f)$ can be analytically derived.

To further enhance the success rate of desired surface generation in obstacle-rich environments, an affine transformation is additionally applied to $\mathcal{F}_f$. Equation (28) is then expressed as:

$$\mathcal{F}(\tau_r, \tau_f) = \mathcal{F}_r(\tau_r) + k(\tau_r)\boldsymbol{M}(\tau_r)\boldsymbol{R}_x(\beta(\tau_r))\mathcal{F}_f(\tau_f) \tag{35}$$

where $k(\tau_r) > 0$ is the scaling coefficient that scales $\mathcal{F}_f$ at τ_r (as shown in Fig. 7a), rotation matrix $\boldsymbol{R}_x(\beta(\tau_r))$ defines the rotation of $\mathcal{F}_f$ around $\boldsymbol{T}(\tau_r)$, with $\beta(\tau_r)$ being the rotation coefficient that specifies the rotation angle (as shown in Fig. 7a).

For the continuity of $\mathcal{F}(\tau_r, \tau_f)$, $k(\tau_r)$ and $\beta(\tau_r)$ are required to be C^2 functions. Additionally, a lower bound $k(\tau_r) \geq k_{min} > 0$ is imposed to prevent $\mathcal{F}_f(\tau_f)$ from shrinking to an unacceptably small size.



(a) $k(\tau_r^*) = 0.7$ (b) $\beta(\tau_r^*) = \pi/4$

Fig. 7: Illustration of the scaling and rotation coefficients.

REMARK 5 (Application in Obstacle Environments). *The proposed PT-frame-based desired surface generation method (35) significantly enhances surface design flexibility and efficiency, thereby extending the applicability of GVF on 2D manifolds, especially in obstacle-rich environments. For a given reference curve $\mathcal{F}_r(\tau_r)$ and formation curve $\mathcal{F}_f(\tau_f)$, an obstacle avoidance planning workflow can be outlined as follows:*

1) ***Reference Curve Replanning:*** *Utilize replanning methods, such as those described in [27], to obtain an obstacle-free reference curve $\mathcal{F}_r$. To enhance the success rate of surface planning, obstacles can be inflated to ensure $\mathcal{F}_r$ maintains a safe distance from them.*
2) ***Surface Initialization:*** *Initialize the scaling coefficient $k(\tau_r) = 1$ and the rotation coefficient $\beta(\tau_r) = 0$. Substitute these into (35) to obtain*

an initial desired surface $\mathcal{F}$. Initialize empty lists $\boldsymbol{\tau}_l$, $\boldsymbol{k}_l$, and $\boldsymbol{\beta}_l$ to store the planned reference curve parameters, scaling coefficients and rotation coefficients, respectively.

3) ***Collision Point Selection:*** *Calculate the distance of points on $\mathcal{F}$ that either collide with obstacles or are closer than a safe distance d_{safe} (where distance is considered negative if inside an obstacle). Select the point corresponding to the minimum distance and denote its corresponding reference curve parameter as τ_r^*.*
4) ***Affine Transformation Coefficients Calculation:*** *An optimization problem is then solved to minimally perturb the current $k(\tau_r^*)$ and $\beta(\tau_r^*)$ values while ensuring the cross-section $\mathcal{F}(\tau_r^*, \tau_f)$ is within the safe region. A simplified objective could be:*

$$\begin{aligned} &\min J(k,\beta) = (\beta - \beta(\tau_r^*))^2 + (k - k(\tau_r^*))^2 \\ s.t. \quad & distance(\mathcal{F}(\tau_r^*), O) > d_{safe} \end{aligned}$$

where O denotes the obstacles. The optimal affine transformation coefficients k^* and β^* are then obtained. τ_r^* is inserted into $\boldsymbol{\tau}_l$ in order, and k^* and β^* are inserted into the corresponding indices of $\boldsymbol{k}_l$ and $\boldsymbol{\beta}_l$, respectively. Subsequently, polynomial or other smooth interpolation methods are applied to the sequence pairs $[\boldsymbol{\tau}_l, \boldsymbol{k}_l]$ and $[\boldsymbol{\tau}_l, \boldsymbol{\beta}_l]$ to obtain revised $k(\tau_r)$ and $\beta(\tau_r)$. These updated coefficients are then used in (35) to generate a revised surface $\mathcal{F}$.
5) ***Iterative Optimization:*** *Repeat Steps 3 and 4 until the entire surface satisfies the obstacle-free condition, i.e., $distance(p, O) > d_{safe}$ for all $p \in \mathcal{F}$.*

IV. SIMULATION AND EXPERIMENTS

A. Ablation Study

To clearly demonstrate the effectiveness of the decoupled coordination and speed coordination strategies, we conduct four comparative simulations: the baseline guiding vector field (GVF), guiding vector field with decoupled coordination term (DGVF), guiding vector field with speed coordination strategy (NGVF) and guiding vector field with both (DNGVF). For clarity, the baseline coordination term ${}^i\boldsymbol{\chi}_c^o$ and the decoupled one ${}^i\boldsymbol{\chi}_c^d$ are defined in (21), the non-uniform vector field with speed coordination is defined by (26). The specific definitions for each GVF as follows:

$$\begin{aligned} \text{GVF}: \quad & {}^i\hat{\boldsymbol{\chi}}_B = \frac{{}^i\boldsymbol{\chi}_t + {}^i\boldsymbol{\chi}_n + {}^i\boldsymbol{\chi}_c^o}{\|{}^i\boldsymbol{\chi}_t{}^{ph} + {}^i\boldsymbol{\chi}_n{}^{ph}\|} \\ \text{DGVF}: \quad & {}^i\hat{\boldsymbol{\chi}}_D = \frac{{}^i\boldsymbol{\chi}_t + {}^i\boldsymbol{\chi}_n + {}^i\boldsymbol{\chi}_c^d}{\|{}^i\boldsymbol{\chi}_t{}^{ph} + {}^i\boldsymbol{\chi}_n{}^{ph}\|} \\ \text{NGVF}: \quad & {}^i\hat{\boldsymbol{\chi}}_N = {}^i\hat{\boldsymbol{\chi}}_t + {}^i\hat{\boldsymbol{\chi}}_n + {}^i\hat{\boldsymbol{\chi}}_c^o \\ \text{DNGVF}: \quad & {}^i\hat{\boldsymbol{\chi}}_{DN} = {}^i\hat{\boldsymbol{\chi}}_t + {}^i\hat{\boldsymbol{\chi}}_n + {}^i\hat{\boldsymbol{\chi}}_c^d \end{aligned} \tag{36}$$

To validate the efficacy of the proposed method and demonstrate its enhanced performance relative to the baseline strategy in ideal settings, numerical simulations are conducted with ${}^i\dot{\boldsymbol{\xi}} = {}^i\hat{\boldsymbol{\chi}}(\boldsymbol{\xi})$. The convergence gains are uniformly set as $k_1 = k_2 = k_3 = 0.1$, and the coordination gains k_{c1} and k_{c2} are set to 0.5 and 0.2, respectively. The desired cruising speed of the robot team is $v_c = 10$ m/s and the guiding signal frequency is 10 Hz. The initial virtual coordinates are initialized as ${}^1\tau_1 = \cdots = {}^n\tau_1 = 0$ and ${}^1\tau_2 = \cdots = {}^n\tau_2 = 0$. The robots communicate using a ring topology, as shown in Fig. 8.

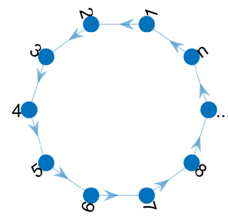


Fig. 8: Communication topology graph of 10 nodes.

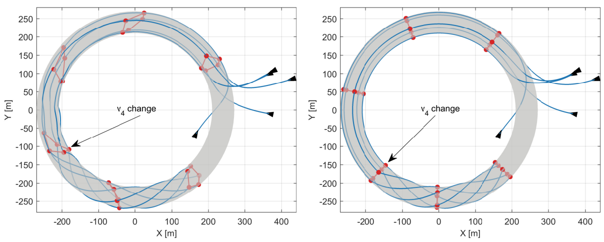
We conduct a simulation of five robots moving on a toroidal surface. The virtual coordinates of all robots are initialized to zero, and their physical coordinates are randomly distributed. The 300-second simulation consists of three sequential phases:

1) From 0 s to 100 s, the desired formation is uniformly distributed along the formation curve with $\boldsymbol{v} = [0, 0, 0, 0, 1]$.
2) From 100 s to 200 s, v_4 is modified to 0.5.
3) From 200 s to 300 s, the desired formation changes to a circular formation on the surface.

As shown in Figs. 10 and 11, during 0 s to 100 s, the coordination error along τ_1 is considerably larger than that along τ_2. This results from formation instability caused by the identical speed setting under the given surface and $\boldsymbol{v}$ configurations (see Fig. 9a), which mainly affects coordination alone τ_1. The increased coordination error propagates through the coupled dynamics, resulting in larger path-following errors. When v_4 is increased to 0.5 at 100s, the introduced motion along the formation curve enables the robots to oscillate around the desired surface, thereby alleviating curvature-induced formation distortions and reducing path-following errors.

The speed coordination strategy stabilizes the formation by synchronizing robot speeds and mitigating the instability caused by speed mismatches. As shown in Fig. 9b, the formation under NGVF avoids the non-convergent behavior observed in Fig. 9a. Furthermore, Fig. 9 indicates that NGVF yields the smallest steady-state formation error, enabling convergence of the path-following error even in the absence of the decoupled coordination term.

The decoupled coordination term constrains the coordination motion within the normal plane of the convergence term, effectively preventing interference between the path-following and coordination errors. Consequently, when $\boldsymbol{\Delta}_1$ and $\boldsymbol{\Delta}_2$ change at 200 s, the coordination error increases sharply. Due to the coupling effect, the path-following errors of GVF and NGVF also rise, whereas the DGVF remains stable.



(a) GVF (b) NGVF

Fig. 9: Trajectories of 5 robots from 0s to 150s.

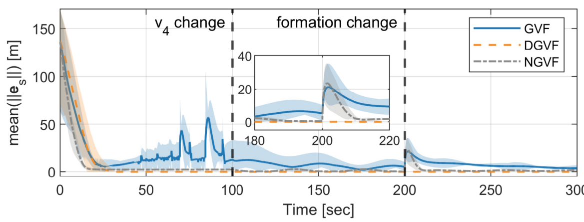
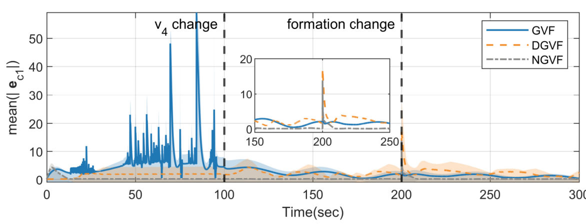
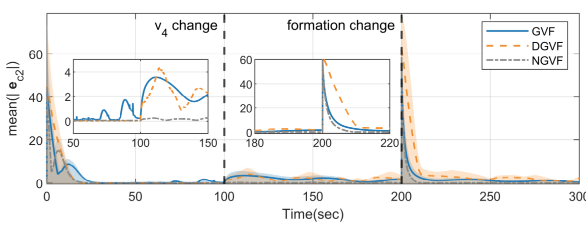


Fig. 10: Average path-following errors of 5 robots.



(a) τ_1 mean errors

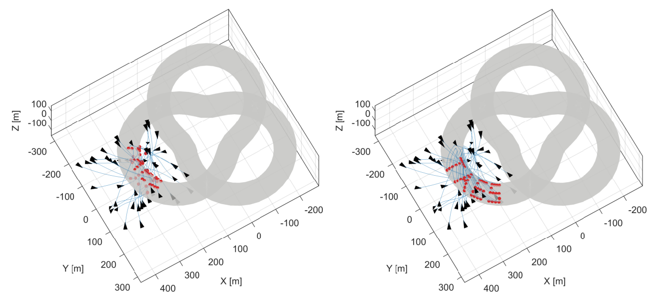


(b) τ_2 mean errors

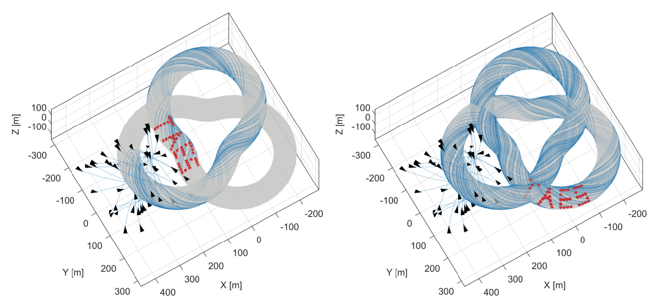
Fig. 11: Average coordination errors of 5 robots.

To further validate the path-following and coordination effectiveness of the proposed method with a larger swarm size, we consider a team of $n = 51$ robots. The desired differences matrices $\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2 \in \mathbb{R}^{51\times 51}$ are manually set to form a “TAES” letter combination on the desired surface. The desired surface is constructed using the PT-frame-based method described in Subsection III.C, where the reference curve is a closed curve containing three self-intersection points. The specific parameters for $\mathcal{F}_r(\tau_r)$ and $\mathcal{F}_f(\tau_f)$ are provided in Appendix A. The extra vector in (9) is set as $\boldsymbol{v} = [0, 0, 0, 0.5, 1]$.

Fig. 12 depicts the moving trajectories of robots guiding by DNGVF from 0 to 500 seconds. It is observed that under the DNGVF, the robots achieve the preset formation shape by 25 seconds and subsequently maintain this formation as they continuously move along the surface, with their motion consistent with the specified $\boldsymbol{v}$. Fig. 13 and Fig. 14 illustrate the evolution of the reference points ${}^i\boldsymbol{\xi}^*$ for each robot under different vector fields. According to the analysis in Section III.B, when ${}^ie_{c1}$ and ${}^ie_{c2}$ converge to zeros, ${}^i\boldsymbol{\xi}^*$ should converge to a common point. It is evident that in DNGVF, the reference points of individual robots are more concentrated, resulting in higher formation accuracy.



(a) The formation pattern at 15s. (b) The formation pattern at 25s.



(c) The formation pattern at 145s. (d) The formation pattern at 415s.

Fig. 12: Simulated motion trajectories and formation configurations of 51 robots at different time.

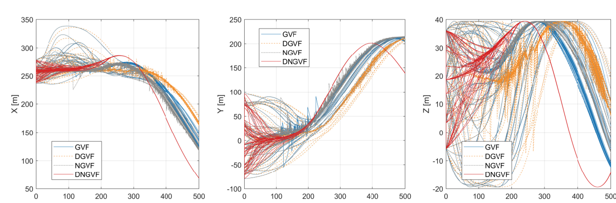


Fig. 13: Changes in the physical coordinates of the reference points, ${}^i\boldsymbol{\xi}^{*ph}$, for GVF, DGVF, NGVF and DNGVF with respect of time.

To quantify the tracking performance, the mean path-following error is defined as $\text{mean}||\boldsymbol{e}_s|| =$

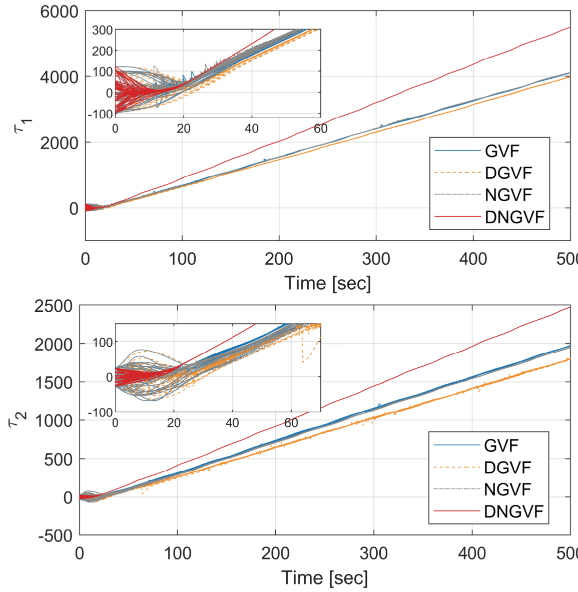


Fig. 14: Changes in the virtual coordinates of the reference points, ${}^{i}\boldsymbol{\xi}^{*vi}$, for GVF, DGVF, NGVF and DNGVF with respect of time.

$\sum_{i=1}^{51} ||{}^{i}\boldsymbol{e}_s({}^{i}\boldsymbol{\xi})||/51$. Additionally, the time-averaged path-following error over the interval $[a, b]$ is given by $\text{avg}_a^b(||\boldsymbol{e}_s||) = \frac{1}{b-a}\int_a^b \text{mean}||\boldsymbol{e}_s||dt$. Using the same approach, $\text{mean}|e_{c1}|$, $\text{mean}|e_{c2}|$, $\text{avg}_\text{a}^\text{b}|e_{c1}|$ and $\text{avg}_a^b|e_{c1}|$ are defined. Table. I summarizes the time-averaged errors of $\text{avg}_a^b(||\boldsymbol{e}_s||)$, $\text{avg}_\text{a}^\text{b}|e_{c1}|$ and $\text{avg}_a^b|e_{c1}|$ for each method. Additionally, Fig. 15 illustrates the variation of $\text{mean}||\boldsymbol{e}_s||$ across different methods, while Fig. 16 shows $\text{mean}|e_{c1}|$ and $\text{mean}|e_{c2}|$.

Table I shows that DGVF and DNGVF achieve significantly higher tracking accuracy compared to vector fields without the decoupled coordination term. Moreover, the coordination error of DNGVF and NGVF which incorporate speed coordination, is markedly lower than other vector fields, and DNGVF demonstrates the highest coordination accuracy.

TABLE I: Average path-following errors and average coordination errors of the four methods from 150 to 500 seconds.

	$\text{avg}_{150}^{500}(\|\|\boldsymbol{e}_s\|\|)$ [m]	$\text{avg}_{150}^{500}\|e_{c1}\|$	$\text{avg}_{150}^{500}\|e_{c2}\|$
GVF	3.90	2.00	2.59
DGVF	**0.33**	2.34	2.69
NGVF	2.23	1.21	1.14
DNGVF	0.73	**0.38**	**0.19**

The error plots confirm analogous findings. Fig. 15 and Fig. 16 illustrate that DGVF, despite its higher coordination error, achieves superior tracking accuracy. This outcome underscores the decoupled coordination term's ability to effectively segregate the influences between coordination and tracking dynamics. Furthermore, GVF and NGVF display correlated trends in tracking and coordination errors between 350 and 500 seconds. As elucidated in Section III.A, this arises because the coordination behavior's component along $\nabla\phi_j({}^{i}\boldsymbol{\xi})$ can undesirably draw ${}^{i}\boldsymbol{\xi}$ away from $\mathcal{S}$ as $||{}^{i}\boldsymbol{e}_s||$ approaches zero, leading to an antagonistic interplay. Likewise, if ${}^{i}\boldsymbol{\chi}_n^{ph}$ are not uniformly equal for all $i \in n$, the convergence term can also displace ${}^{i}\boldsymbol{\xi}$ from the ideal zero formation error state.

Moreover, as analyzed in Section III.B, vector field normalization can lead to formation instability. Fig. 16 corroborates this, showing that the normalized GVF and DGVF result in significant coordination errors and considerable fluctuations. Despite NGVF's speed coordination mechanism effectively improving formation accuracy, its coordination accuracy is slightly lower than DNGVF due to the mutual interference between convergence and coordination behaviors.

In conclusion, DNGVF leverages its decoupled coordination term to prevent interference between coordination and tracking behaviors. Simultaneously, its speed coordination strategy mitigates the formation instability arising from normalization without increasing robot interaction information. This combined approach consequently enhances both path tracking and formation accuracies.

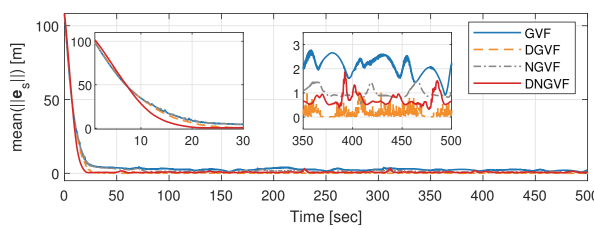
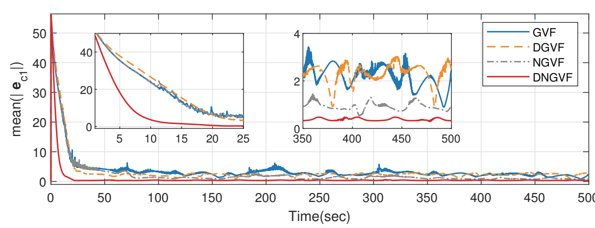
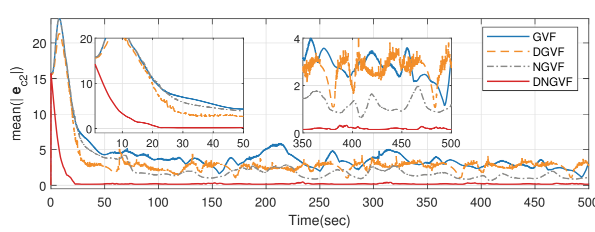


Fig. 15: Path-following errors of different methods.



(a) τ_1 mean errors



(b) τ_2 mean errors

Fig. 16: Coordination errors of different methods.

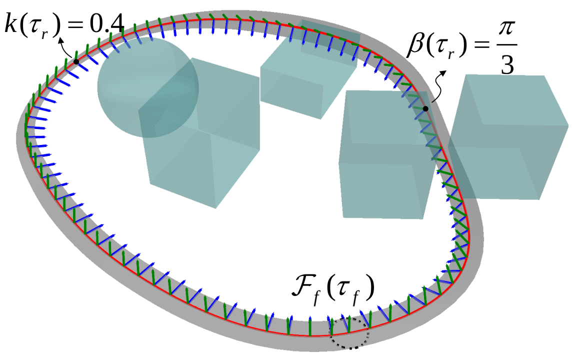


Fig. 17: Obstacle-free desired surface generated via the PT-Frame-based method.

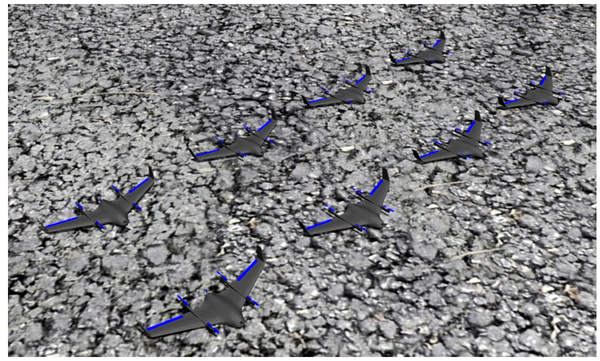


Fig. 18: 8 VTOLs on the ground in SITL experiments.

B. Software-In-The-Loop Experiments

Velocity-tracking errors in practical flight control often cause deviations from the desired vector-field velocity. To examine the algorithm's performance under such non-ideal conditions, software-in-the-loop (SITL) simulations are carried out using the PX4 autopilot suite. Eight VTOL models are employed, with the GVF/DNGVF providing navigation signals, while coordinated turn controller (37) and TECS controller (38) handle attitude regulation.

Under the no sideslip assumption, the desired yaw angle of UAV i is

$$^{i}\psi_d = \arctan 2(\dot{y}_d, \dot{x}_d)$$

where $\dot{x}_d = {}^{i}\hat{\boldsymbol{\chi}}^{[1]}$ and $\dot{y}_d = {}^{i}\hat{\boldsymbol{\chi}}^{[2]}$ denote the desired velocity components of UAV i in the XY-plane.

According to the coordinated turn, the desired roll angle is derived as:

$$^{i}\phi_d = \arctan\left(\frac{||^{i}v_d||k_\psi(\psi_d - \psi)}{g}\right) \tag{37}$$

where g is the gravitational acceleration.

Based on TECS [28], the desired thrust and pitch angle are obtained:

$$^{i}\theta_d, {}^{i}T_d = \mathrm{TECS}(\dot{h}_d, {}^{i}v_d, UavStatus) \tag{38}$$

where $\dot{h}_d = {}^{i}\hat{\boldsymbol{\chi}}^{[3]}$ denotes the desired longitudinal speed, $^{i}v_d = ||^{i}\hat{\boldsymbol{\chi}}^{ph}||$ is the desired speed.

UAVs often operate in ground-parallel formations for missions such as ground tracking, area search, and road inspection. Accordingly, in the SITL flight simulation, we accordingly design the formation curve with $\mathcal{F}_f^{[3]} = 0$, resulting in a strip-like desired surface as depicted in Fig. 17. Moreover, through strategic design of the scaling and rotation coefficients in (35), the surface is configured to pass through narrow gaps and circumvent nearby obstacles. The specific parameters of the desired surface are detailed in Appendix B.

The key parameters for the simulation are set as follows: $\boldsymbol{v} = [0, 0, 0, 0, -1]$, $k_{c1} = 0.02$, $k_{c2} = 0.01$, $k_1 = k_2 = k_3 = 0.002$, $v_c = 15$ m/s, and the guiding signal frequency is 5 Hz. The desired differences matrix $\boldsymbol{\Delta}_1$ is set to be $\mathbf{0}$, while $\boldsymbol{\Delta}_2$ is configured to ensure the UAVs are uniformly distributed along $\mathcal{F}_f$.

8 UAVs are initialized with virtual coordinates $^{1}\tau_r = \cdots = {}^{8}\tau_r = 0$ and $^{1}\tau_f = \cdots = {}^{8}\tau_f = 0$, and their physical starting positions set near the origin, as shown in Fig. 18. The simulation begins with UAVs taking off from the ground in copter mode, ascending to 80 m altitude over approximately 30 seconds. Following this ascent, they transition to fixed-wing mode, flying under the guidance of GVF/DNGVF. Their attitude control is determined by (37) and (38).

Fig. 19a displays the simulated flight trajectories of UAVs guided by DNGVF. To better illustrate the formation performance, Figs. 19b and 19c depict the trajectories of the group of UAV flight on the affinely transformed desired surface under GVF and DNGVF guidance, respectively. It can be observed that, owing to the normalization process, UAVs guided by GVF struggle to maintain the prescribed formation shape, while DNGVF effectively ensures the swarm flies in the predetermined formation.

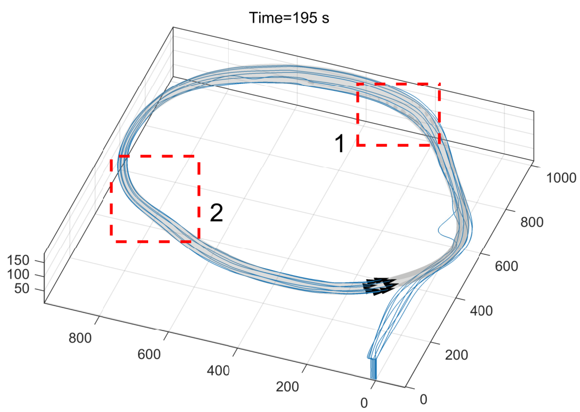
Figs. 20 and 21 present the average path-following and coordination errors, respectively. Compared to GVF, UAVs guided by DNGVF exhibit enhanced tracking and coordination accuracies. Moreover, it can be observed that once the group largely stabilizes (at approximately 50 s), the values of $\mathrm{mean}(\|\boldsymbol{e}_s\|)$, $\mathrm{mean}(|e_{c1}|)$ and $\mathrm{mean}(|e_{c2}|)$ under GVF guidance display similar trends, aligning with the findings in Section IV.A.

Furthermore, the errors e_{c1} are significantly greater than e_{c2}. This discrepancy arises from the speed control inaccuracies inherent in the chosen UAV attitude controller. Specifically, the configurations $v_4 = 0$ and $v_5 = -1$, leading to $^{i}\boldsymbol{\chi}_t = [\partial\mathcal{F}_r/\partial\tau_r, 1, 0]^\top$. When $\boldsymbol{e}_s$, e_{c1} and e_{c2} are small, the flight direction of UAVs largely aligns with $^{i}\boldsymbol{\chi}_t$. Therefore, the speed control error prevents the UAVs from accurately moving at the desired speed, consequently leading to errors in $^{i}\boldsymbol{\xi}^{[4]}$.

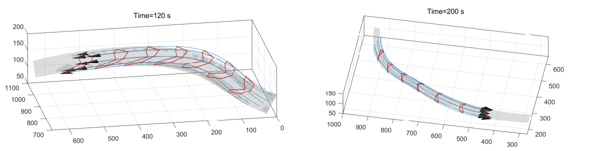
V. CONCLUSION

In this paper, we propose a guiding vector field that decouples path-following behavior and coordination behavior. The proposed method effectively reduces path-

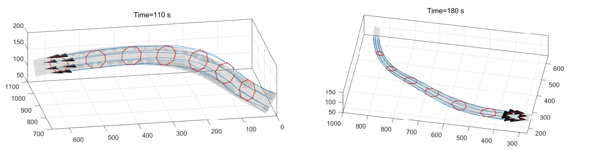
This article has been accepted for publication in IEEE Transactions on Aerospace and Electronic Systems. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TAES.2026.3659453



(a) Trajectories of UAVs guided by DNGVF.



(b) Flight trajectories and formation of UAVs guided by GVF on segments 1 and 2 of the desired surface.



(c) Flight trajectories and formation of UAVs guided by DNGVF on segments 1 and 2 of the desired surface.

Fig. 19: UAV trajectories under GVF and DNGVF navigation in SITL experiments.

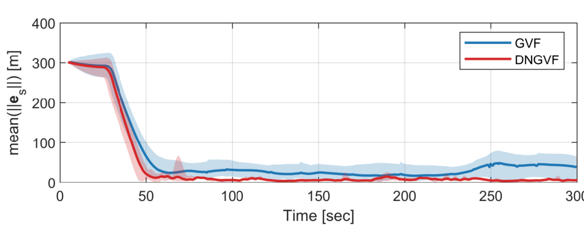
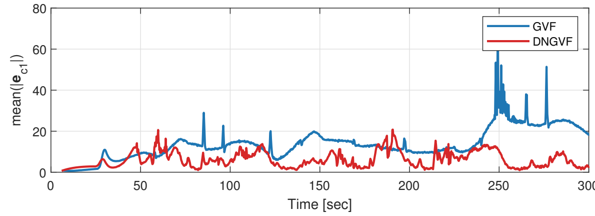
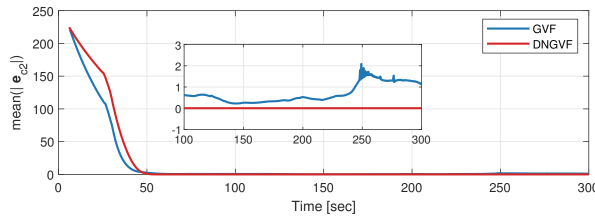


Fig. 20: Path-following errors of UAVs under GVF and DNGVF navigation in the SITL experiments.

following error and coordination error of robots. Specifically, we first introduce auxiliary vectors orthogonal to the convergence term, projecting the virtual coordinate coordination quantity onto the auxiliary vector to achieve decoupling from the convergence term. Then, utilizing the consistency of reference virtual coordinates, we construct a non-uniform GVF to ensure that the virtual coordinate speeds of the robots converge to the same value, improving formation maintenance accuracy. We analyze the stability of the proposed method. Additionally, we propose a PT-Frame-based desired surface construction method to enhance surface design flexibility.



(a) τ_1 mean errors



(b) τ_2 mean errors

Fig. 21: Coordination errors of UAVs under GVF and DNGVF navigation in the SITL experiments.

Finally, we conduct numerical simulation ablation experiments and Software-in-the-Loop (SITL) comparison experiments. The results demonstrate that the proposed method effectively improves path-following accuracy and formation coordination.

APPENDIX

A. The Desired Surface Parameters in Ablation Study

A three-dimensional, third-order polynomial curve is determined by a point list $\boldsymbol{p} = [\boldsymbol{p}^{[1]}, \cdots, \boldsymbol{p}^{[n]}]^\top \in \mathbb{R}^{n\times 3}$, a velocity list $\boldsymbol{v} = [\boldsymbol{v}^{[1]}, \cdots, \boldsymbol{v}^{[n]}]^\top \in \mathbb{R}^{n\times 3}$ and a parameter list $\boldsymbol{\tau} = [\tau_1, \cdots, \tau_n] \in \mathbb{R}^n$, where $n \geq 2$ and $\tau_i < \tau_j$ for any $i < j$, $i, j \in \mathbb{Z}_1^n$. If this curve is closed, then $n \geq 3$ and $p_1 = p_n$ and $v_1 = v_n$.

A three-dimensional, third-order polynomial curve is expressed as:

$$\begin{aligned} x(\tau) &= \boldsymbol{A}_{1,i} T_i(\tau) \\ y(\tau) &= \boldsymbol{A}_{2,i} T_i(\tau) \\ z(\tau) &= \boldsymbol{A}_{3,i} T_i(\tau) \end{aligned} \tag{39}$$

where the segment index i is determined by $i = \arg(0 \leq \bar{\tau} < \tau_{i+1} - \tau_1)$ and $\bar{\tau} = \tau \mod (\tau_n - \tau_1)$, $\boldsymbol{A}_{k,i}$ is the k-th coefficient matrix of the i-th segment of the curve, $T_i(\tau)$ is the parameter vector of the i-th segment of the curve. $\boldsymbol{A}_{k,i}$ and $T_i(\tau)$ are determined by (40) and (41).

$$\boldsymbol{A}_{k,i} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3/\Gamma_i^2 & -2/\Gamma_i & 3/\Gamma_i^2 & -1/\Gamma_i \\ 2/\Gamma_i^3 & 1/\Gamma_i^2 & -2/\Gamma_i^3 & 1/\Gamma_i^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{p}(i,k) \\ \boldsymbol{v}(i,k) \\ \boldsymbol{p}(i+1,k) \\ \boldsymbol{v}(i+1,k) \end{bmatrix} \tag{40}$$

$$T_i(\tau) = \begin{bmatrix} 1 & \tau - \boldsymbol{\tau}_i & (\tau - \boldsymbol{\tau}_i)^2 & (\tau - \boldsymbol{\tau}_i)^3 \end{bmatrix}^\top \tag{41}$$

where $\Gamma_i = \boldsymbol{\tau}_{i+1} - \boldsymbol{\tau}_i$ is the parameter gap.

For the ablation study, the conference curve is obtained by fitting the curve defined by the equation below using (39):

$$\begin{aligned} f_1(\omega) &= 100\cos(2\omega)\left(\cos(3\omega)+2\right) \\ f_2(\omega) &= 100\sin(2\omega)\left(\cos(3\omega)+2\right) \\ f_3(\omega) &= 10 \end{aligned}$$

The point list, velocity list and parameter gap of reference curve $(F)_r$ are determined by:

$$\begin{aligned} \boldsymbol{p}_r^{[i]} &= \begin{bmatrix} f_1(2\pi/n*(i-1)) \\ f_2(2\pi/n*(i-1)) \\ f_3(2\pi/n*(i-1)) \end{bmatrix} \\ \boldsymbol{v}_r^{[i]} &= 0.1\mathrm{unit}\left(\frac{d_{i,i-1}(\boldsymbol{p}_r^{[i+1]}-\boldsymbol{p}_r^{[i]})}{d_{i+1,i}+d_{i,i-1}} + \frac{d_{i+1,i}(\boldsymbol{p}_r^{[i]}-\boldsymbol{p}_r^{[i-1]})}{d_{i+1,i}+d_{i,i-1}}\right) \\ \Gamma_i &= 10\|\boldsymbol{p}_r^{[i+1]}-\boldsymbol{p}_r^{[i]}\| \end{aligned} \tag{42}$$

where $n = 100$ is the sample point amount, and $d_{i+1,i} = \|\boldsymbol{p}_r^{[i+1]} - \boldsymbol{p}_r^{[i]}\|$.

The point list $\boldsymbol{p}_f^{[i]}$ of $\mathcal{F}_f$ is:

$$\boldsymbol{p}_f = \begin{bmatrix} 0 & 40 & 10 \\ 0 & -40 & 10 \\ 0 & -40 & -10 \\ 0 & 40 & -10 \end{bmatrix}$$

And the velocity list $\boldsymbol{v}_f$ and parameter gap Γ_i are derived using the same method as (42)

B. The Desired Surface Parameters in SITL Experiments

The point list of $\mathcal{F}_r$ used in SITL simulation is:

$$\boldsymbol{p}_r = \begin{bmatrix} 280 & 440 & 800 & 910 & 1000 & 880 & 440 & 320 & 240 \\ 80 & -40 & 80 & 180 & 560 & 880 & 880 & 720 & 480 \\ 100 & 100 & 100 & 140 & 140 & 100 & 100 & 100 & 80 \end{bmatrix}$$

And the corresponding $\boldsymbol{v}_r$ and Γ_i are derived using the same method as (42).

The parameters of formation curve are obtained by fitting a circle:

$$\begin{aligned} f_1(\omega) &= 30\cos(\omega) \\ f_2(\omega) &= 30\sin(\omega) \\ f_2(\omega) &= 0 \end{aligned}$$

REFERENCES

[1] J. Ghommam and F. Mnif, "Coordinated path-following control for a group of underactuated surface vessels," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 10, pp. 3951–3963, 2009.

[2] L. Sabattini, C. Secchi, M. Cocetti, A. Levratti, and C. Fantuzzi, "Implementation of coordinated complex dynamic behaviors in multirobot systems," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 1018–1032, 2015.

[3] A. Matese, P. Toscano, S. F. Di Gennaro, L. Genesio, F. P. Vaccari, J. Primicerio, C. Belli, A. Zaldei, R. Bianconi, and B. Gioli, "Intercomparison of uav, aircraft and satellite remote sensing platforms for precision viticulture," *Remote sensing*, vol. 7, no. 3, pp. 2971–2990, 2015.

[4] M. Qian, Z. Wu, and B. Jiang, "Cerebellar model articulation neural network-based distributed fault tolerant tracking control with obstacle avoidance for fixed-wing uavs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 5, pp. 6841–6852, 2023.

[5] B.-B. Hu, H.-T. Zhang, W. Yao, Z. Sun, and M. Cao, "Coordinated guiding vector field design for ordering-flexible multi-robot surface navigation," *IEEE Transactions on Automatic Control*, 2024.

[6] H. Chen, Y. Cong, X. Wang, X. Xu, and L. Shen, "Coordinated path-following control of fixed-wing unmanned aerial vehicles," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 4, pp. 2540–2554, 2021.

[7] A. Akhtar, S. L. Waslander, and C. Nielsen, "Path following for a quadrotor using dynamic extension and transverse feedback linearization," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 3551–3556.

[8] T. I. Fossen, K. Y. Pettersen, and R. Galeazzi, "Line-of-sight path following for dubins paths with adaptive sideslip compensation of drift forces," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 820–827, 2014.

[9] W. Caharija, M. Candeloro, K. Y. Pettersen, and A. J. Sørensen, "Relative velocity control and integral los for path following of underactuated surface vessels," *IFAC Proceedings Volumes*, vol. 45, no. 27, pp. 380–385, 2012.

[10] P. Sujit, S. Saripalli, and J. B. Sousa, "Unmanned aerial vehicle path following: A survey and analysis of algorithms for fixed-wing unmanned aerial vehicless," *IEEE Control Systems Magazine*, vol. 34, no. 1, pp. 42–59, 2014.

[11] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, "Vector field path following for miniature air vehicles," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 519–529, 2007.

[12] V. M. Goncalves, L. C. Pimenta, C. A. Maia, B. C. Dutra, and G. A. Pereira, "Vector fields for robot navigation along time-varying curves in n-dimensions," *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 647–659, 2010.

[13] W. Yao, H. G. de Marina, B. Lin, and M. Cao, "Singularity-free guiding vector field for robot navigation," *IEEE Transactions on Robotics*, vol. 37, no. 4, pp. 1206–1221, 2021.

[14] G. V. Pratama, E. Ekawati, and F. Mukhlish, "Cyclic leader-follower quadrotor formation control with squared fiducial marker guides," in *2024 International Conference on Electrical and Information Technology (IEIT)*. IEEE, 2024, pp. 66–71.

[15] X. Dong, Y. Li, C. Lu, G. Hu, Q. Li, and Z. Ren, "Time-varying formation tracking for uav swarm systems with switching directed topologies," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 12, pp. 3674–3685, 2018.

[16] X. Wang, S. Baldi, X. Feng, C. Wu, H. Xie, and B. De Schutter, "A fixed-wing uav formation algorithm based on vector field guidance," *IEEE Transactions on Automation Science and Engineering*, vol. 20, no. 1, pp. 179–192, 2022.

[17] X. Xiang, D. Chen, C. Yu, and L. Ma, "Coordinated 3d path following for autonomous underwater vehicles via classic pid controller," *IFAC Proceedings Volumes*, vol. 46, no. 20, pp. 327–332, 2013.

[18] A.-M. Zou and K. D. Kumar, "Distributed attitude coordination control for spacecraft formation flying," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1329–1346, 2012.

[19] H. G. De Marina, Z. Sun, M. Bronz, and G. Hattenberger, "Circular formation control of fixed-wing uavs with constant speeds," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 5298–5303.

[20] W. Yao, H. G. de Marina, Z. Sun, and M. Cao, "Distributed coordinated path following using guiding vector fields," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 10 030–10 037.

[21] ——, "Guiding vector fields for the distributed motion coordination of mobile robots," *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1119–1135, 2023.

[22] W. Ren and Y. Cao, *Distributed coordination of multi-agent networks: emergent problems, models, and issues*. Springer Science & Business Media, 2010.

[23] Z. Xu, J. Zhou, D. Tian, X. Duan, K. Qu, D. Zhao, Z. Sheng, and J. Zhai, “Motion planning and control of quadrotor based on mpc in frenet frame,” in *2023 IEEE International Conference on Unmanned Systems (ICUS)*. IEEE, 2023, pp. 1421–1426.

[24] K. Tanaka, M. Tanaka, Y. Takahashi, A. Iwase, and H. O. Wang, “3-d flight path tracking control for unmanned aerial vehicles under wind environments,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 12, pp. 11 621–11 634, 2019.

[25] S. Zhao, X. Wang, D. Zhang, and L. Shen, “Curved path following control for fixed-wing unmanned aerial vehicles with control constraint,” *Journal of Intelligent & Robotic Systems*, vol. 89, pp. 107–119, 2018.

[26] A. J. Hanson and H. Ma, “Parallel transport approach to curve framing,” *Indiana University, Techreports-TR425*, vol. 11, pp. 3–7, 1995.

[27] F. Gao, L. Wang, B. Zhou, X. Zhou, J. Pan, and S. Shen, “Teach-repeat-replan: A complete and robust system for aggressive flight in complex environments,” *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1526–1545, 2020.

[28] C. Voth and U.-L. Ly, “Design of a total energy control autopilot using constrained parameter optimization,” *Journal of Guidance, Control, and Dynamics*, vol. 14, no. 5, pp. 927–935, 1991.

Jingzong Liu received the B.S. degree in mechanical electronics engineering from Chongqing University, Chongqing, China, in 2021, the M.S. degree from the Harbin Institute of Technology, Harbin, China, in 2023.

He is currently working toward the Ph.D. degree in aeronautical and astronautical science and technology with the School of Astronautics, Harbin Institute of Technology, Harbin, China.

His research interests include multi-robot, path planning and target tracking.

Zexu Zhang received the B.S. degree in electronics and communication engineering from Air Force Engineering University, Xi’an, China, in 1993, the M.S. degree in astronautics engineering and the Ph.D. degree in aerospace science and technology from the School of Astronautics, Harbin Institute of Technology, Harbin, China, in 2000 and 2004, respectively.

He is a Full Professor with the Harbin Institute of Technology, where he is also the Director of the Institute of Aircraft Dynamics and Control. His research interests include aircraft autonomous navigation and control, intelligent cooperative perception and autonomous decision-making in drone swarm, data visualization.

Dr. Zhang was the Elected Director of the Committee of Space Intelligence of the Chinese Society of Space Research (2021–2025).

Liang Zhang received his B.Eng. degree in Automation from Shandong University, China in 2015 and Ph.D. degree in the School of Astronautics, Harbin Institute of Technology, China in 2021. He was also a visiting Ph.D. student in the Autonomous System Lab, ETH Zurich, Switzerland from 2018 to 2019 under the foundation from China Scholarship Council.

He is currently an associate professor at the School of Electrical Engineering and Automation, Anhui University, China.

His research interests are mainly around the multi-robot or multi-agent system (MAS/MRS), including distributed estimation, cooperative control, planning under uncertainty.

Kai Zhang received the B.S. and M.S. degrees from Hefei University of Technology, Anhui, China, in 2016 and 2019, respectively. He earned the D.Eng degree from the Department of Control Science and Engineering from Harbin Institute of Technology, Harbin, China, in 2023.

He is currently working as an Assistant Professor at Harbin Institute of Technology, Harbin, China.

His research interests include multiagent systems, time-delay systems and finite-time control.

Chao Yan received the B.S. degree in aircraft design and engineering from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2020, the M.S. degree from the Harbin Institute of Technology, Harbin, China, in 2022.

He is currently working toward the Ph.D. degree in aeronautical and astronautical science and technology with the School of Astronautics, Harbin Institute of Technology, Harbin, China.

His research interests include multi-agent and distributed cooperation control.