Cooperative exploration of level surfaces of three dimensional scalar fields

Article *in* Automatica · September 2011

DOI: 10.1016/j.automatica.2011.06.001 · Source: DBLP

2 authors:

Wencen Wu
San Jose State University
52 PUBLICATIONS **521** CITATIONS
SEE PROFILE

Fumin Zhang
Georgia Institute of Technology
245 PUBLICATIONS **3,923** CITATIONS
SEE PROFILE

Cooperative Exploration of Level Surfaces of Three Dimensional Scalar Fields ⋆

Wencen Wu [a] and Fumin Zhang [a]

[a] *Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332*

Abstract

We develop strategies for a group of mobile sensing agents to cooperatively explore level surfaces of an unknown 3D scalar field. A cooperative Kalman filter is constructed to combine sensor readings from all agents and give estimates of the field value and gradient at the center of the formation formed by the sensing agents. The formation formed by the agents is controlled to track curves on a level surface in the field under steering control laws. We prove that the formation center can move to a desired level surface and can follow a curve with known frame and curvatures. In particular, we present results on tracking lines of curvature on a desired level surface, revealing the 3D geometry of the scalar field. Taubin's algorithm is modified and applied to detect and estimate principal curvatures and principal directions for lines of curvature. We prove the sufficient and necessary conditions that ensure reliable estimates using Taubin's algorithm. We also theoretically justify the minimum number of agents that can be utilized to accomplish the exploration tasks. Simulation results demonstrate that a line of curvature on a desired level surface can be detected and traced successfully.

Key words: Cooperative exploration; Curvature estimation; Cooperative filtering.

1 Introduction

The problem of cooperative exploration investigates how to deploy a group of collaborative mobile agents to explore an unknown scalar field efficiently and adaptively [3] [6]. Cooperative exploration missions in the literature include, but are not limited to, climbing gradients [15], cooperative path following [7], and monitoring environmental boundaries [9] [10]. Most existing results are for exploring two dimensional (2D) scalar fields. In this paper, we focus on the problem of exploring three dimensional (3D) scalar fields. This work significantly extends the earlier results on 2D cooperative exploration introduced by Zhang and Leonard [21].

We control a group of agents to move in a formation so that the local structure of the field can be estimated from the measurements taken by all the agents. To combine the measurements from the agents, a cooperative Kalman filter is constructed to give estimates of the field value and gradient at the formation center. We estimate the Hessian matrix that is used in the implementation of the cooperative Kalman filter by relating each of its elements to the curvatures of the local level surfaces of the field in a neighborhood of the formation center.

The desired formation is maintained by the formation shape control law described in [19–21], which is based on Jacobi transform. The Jacobi transform decouples the dynamics of the formation center from the dynamics of the formation shape, which allows us to develop separate control laws. Following a differential geometric approach [16] [12] [11], we develop steering control laws that control the formation center to detect and move to a desired level surface and track a curve on the surface with known frame and curvatures. Once the formation shape control and the formation center motion control are combined, the formation can be viewed as a "super-agent" that is able to accomplish curve tracking tasks.

Among all possible curves that the formation may detect and follow, we study the problem of controlling the formation to detect and track one of the lines of curvature on a desired level surface. Lines of curvature are curves that are associated with principal directions and principal curvatures [5] [14], which measure how the surface bends. To implement the tracking control, we apply Taubin's algorithm [17], which is modified to generate estimates of prin-

⋆ The research work is supported by ONR grants N00014-08-1-1007, N00014-09-1-1074, and N00014-10-10712 (YIP), and NSF grants ECCS- 0841195 (CAREER), CNS-0931576, and ECCS-1056253.

Email addresses: wwencen3@gatech.edu (Wencen Wu), fumin@gatech.edu (Fumin Zhang).

cipal directions and principal curvatures with a small number of agents in a formation at each time instant. An important concern here is the quantity of agents required to obtain valid estimates. However, Taubin's algorithm and other related works [4] [8] were developed for computer vision applications and do not contain conditions about the minimum number of agents and their arrangements to generate valid estimates. We establish sufficient and necessary conditions for Taulin's algorithm to provide non-singular estimates. The conditions theoretically justify the minimum number of agents required and constraints on the formation shape. Our results have not been reported in literature on Taubin's algorithm.

We have found that some techniques developed for the 2D exploration in earlier works [21] can be applied to the 3D case with only slight extension. These include the cooperative Kalman filtering algorithm and the formation control law based on Jacobi transform. Such techniques are only briefly reviewed in this paper. On the other hand, we have discovered that the 3D exploration offers significantly more difficult challenges to curvature estimation, Hessian estimation, and tracking control, which are the main topics of this paper.

This paper is organized as follows. In Section 2, we review the information dynamics and the formation shape control for the cooperative exploration problems. In Section 3, we develop control laws to control the formation center to track a curve on a level surface. In Section 4, the principal curvature estimation algorithm and the constraints on agent quantity are discussed. Estimation of the Hessian matrix is performed in Section 5. Simulation results are shown in Section 6, and concluding remarks are presented in Section 7.

2 Extension of the Information Dynamics and the Formation Control to 3D

Assume that $z(\mathbf{r})$ is an unknown smooth scalar field where $\mathbf{r} \in \mathbb{R}^3$. The field consists of level surfaces $\Gamma(\mathbf{r}) = \{\mathbf{r} | z(\mathbf{r}) = C_i, i = 1, \cdots\}$, where C_i are constants that correspond to different field values. The scalar field $z(\mathbf{r})$ is perturbed by noises. We consider the problem of estimating the local geometric structure of the field by deploying a group of sensing agents in the field. In this section, we review the information dynamic model and the cooperative control for the cooperative exploration problem discussed in [21] and make extensions to 3D space.

2.1 Information Dynamics

Suppose N sensing agents are deployed to explore an unknown field. In most applications, the measurements are taken discretely over time. We assume that each agent can only take one measurement of the field at each time instant k where k is an integer. At time instant k, the position of the ith agent is denoted by $\mathbf{r}_{i,k}$ and the field value at the position $\mathbf{r}_{i,k}$ is denoted as $z_{i,k}$, where $i = 1, 2, \cdots, N$. The measurement taken by the ith agent can be written as

$$p_{i,k} = z_{i,k} + w_{i,k} + n_{i,k}, \tag{1}$$

where $n_{i,k}$ is i.i.d Gaussian noise and $w_{i,k}$ is spatially correlated Gaussian noise.

Since the group of agents can be considered as a "super-agent" when exploring the field, we are interested in the estimates of the field value and gradient at the formation center $\mathbf{r}_{c,k}$ at each time instant, where $\mathbf{r}_{c,k}$ is defined as the average of the positions of all the agents in the form of $\frac{1}{N}\sum_{i=1}^{N}\mathbf{r}_{i,k}$. Therefore, the state is chosen as $\mathbf{s}_k = (z_{c,k}, \nabla z_{c,k}^T)^T$ where $z_{c,k}$ is the field value and $\nabla z_{c,k}$ is the field gradient at $\mathbf{r}_{c,k}$. Using Taylor's expansion to approximate $z_{i,k}$, we can get

$$z_{i,k} \approx z_{c,k} + (\mathbf{r}_{i,k} - \mathbf{r}_{c,k})^T \nabla z_{c,k} + \frac{1}{2}(\mathbf{r}_{i,k} - \mathbf{r}_{c,k})^T \nabla^2 z_{c,k} (\mathbf{r}_{i,k} - \mathbf{r}_{c,k}). \tag{2}$$

where $\nabla^2 z_{c,k}$ is the Hessian of the field at $\mathbf{r}_{c,k}$. Let C_k be a $N \times 4$ matrix with the ith row defined by $[1, (\mathbf{r}_{i,k} - \mathbf{r}_{c,k})^T]$ for $i = 1, 2, ..., N$. Let D_k be a $N \times 9$ matrix with its ith row defined by $\frac{1}{2}((\mathbf{r}_{i,k} - \mathbf{r}_{c,k}) \otimes (\mathbf{r}_{i,k} - \mathbf{r}_{c,k}))^T$ where $\otimes$ is the Kronecker product. Define $\mathbf{H}_{c,k}$ as the estimate of the Hessian $\nabla^2 z_{c,k}$. Suppose $\vec{\mathbf{H}}_{c,k}$ is defined by rearranging the elements of $\mathbf{H}_{c,k}$ as $\vec{\mathbf{H}}_{c,k} = [H_{c,k(11)}, H_{c,k(21)}, H_{c,k(31)}, H_{c,k(12)}, H_{c,k(22)}, H_{c,k(32)}, H_{c,k(13)}, H_{c,k(23)}, H_{c,k(33)}]$. Now the measurement equation can be written as

$$\mathbf{p}_k = C_k \mathbf{s}_k + D_k \vec{\mathbf{H}}_{c,k} + \mathbf{w}_k + D_k \mathbf{e}_k + \mathbf{n}_k, \tag{3}$$

where $\mathbf{e}_k$ is the error vector associated with estimating the Hessian. The variables $\mathbf{p}_k$, $\mathbf{w}_k$ and $\mathbf{n}_k$ are $N \times 1$ vectors, i.e. $\mathbf{p}_k = [p_{i,k}]$, $\mathbf{w}_k = [w_{i,k}]$ and $\mathbf{n}_k = [n_{i,k}]$ where $i = 1, \cdots, N$. When the formation center moves, the state $\mathbf{s}_k$ evolves according to

$$\begin{aligned} z_{c,k} &= z_{c,k-1} + (\mathbf{r}_{c,k} - \mathbf{r}_{c,k-1})^T \nabla z_{c,k-1}, \\ \nabla z_{c,k} &= \nabla z_{c,k-1} + \mathbf{H}_{c,k-1}(\mathbf{r}_{c,k} - \mathbf{r}_{c,k-1}). \end{aligned} \tag{4}$$

Define $\mathbf{h}_{k-1} = (0, E[\mathbf{H}_{c,k}(\mathbf{r}_{c,k} - \mathbf{r}_{c,k-1})]^T)^T$, $\mathbf{I}_{3\times 3}$ as the 3×3 identity matrix, and $\mathbf{A}_{k-1}^s = \begin{pmatrix} 1 & (\mathbf{r}_{c,k} - \mathbf{r}_{c,k-1})^T \\ 0 & \mathbf{I}_{3\times 3} \end{pmatrix}$. Then the state equation can be expressed as

$$\mathbf{s}_k = \mathbf{A}_{k-1}^s \mathbf{s}_{k-1} + \mathbf{h}_{k-1} + \boldsymbol{\varepsilon}_{k-1}, \tag{5}$$

where $\boldsymbol{\varepsilon}_{k-1}$ is a $N \times 1$ noise vector that is independent of the measurement noise $\mathbf{n}_k$.

2.2 Cooperative Kalman Filter

Once the state equation (5) and measurement equation (3) are known, a cooperative Kalman filter is constructed to re-

duce the measurement noise. Denote $U_k = E[\mathbf{e}_k\mathbf{e}_k^T]$, $R_k = E[\mathbf{n}_k\mathbf{n}_k^T]$ and $M_k = E[\varepsilon_k\varepsilon_k^T]$ where $\mathbf{e}_k$, $\mathbf{n}_k$ and ε_k are introduced in section 2.1. The cooperative Kalman filter equations are as follows

$$\begin{aligned}
\mathbf{s}_{k(-)} &= \mathbf{A}_{k-1}^s \mathbf{s}_{k-1(+)} + \mathbf{h}_{k-1},\\
P_{k(-)} &= \mathbf{A}_{k-1}^s P_{k-1(+)} \mathbf{A}_{k-1}^{sT} + M_{k-1},\\
K_k &= P_{k(-)} C_k^T [C_k P_{k(-)} C_k^T + D_k U_k D_k^T + R_k]^{-1},\\
\mathbf{s}_{k(+)} &= \mathbf{s}_{k(-)} + K_k(\mathbf{p}_k - C_k \mathbf{s}_{k(-)} - D_k \vec{\mathbf{H}}_{c,k}),\\
P_{k(+)}^{-} &= P_{k(-)}^{-1} + C_k^T [D_k U_k D_k^T + R_k]^{-1} C_k. \qquad (6)
\end{aligned}$$

The subscript $(-)$ and $(+)$ indicate the predictions and the updated estimates, respectively. The convergence of the cooperative Kalman filter can be proved in a similar way as the proof in [21]. We will discuss the estimates of the Hessian term $\mathbf{H}_{c,k}$ in Section 5.

Note: For the rest of the paper, we drop the subscript k for simplicity whenever only the kth step is concerned.

2.3 Formation Shape Control

We use Jacobi vectors to describe the formation of the agents: $[\mathbf{r}_c, \mathbf{q}_1, \cdots, \mathbf{q}_{N-1}] = [\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N]\Psi$, where Ψ is the Jacobi transform [20]. For example, if $N = 3$, we can define the Jacobi vectors to be $\mathbf{q}_1 = \frac{\sqrt{2}}{2}(\mathbf{r}_2 - \mathbf{r}_3), \mathbf{q}_2 = \frac{\sqrt{6}}{6}(2\mathbf{r}_1 - \mathbf{r}_2 - \mathbf{r}_3)$. Assume that each agent has unit mass. The dynamics of the agents are described by Newton's equations: $\ddot{\mathbf{r}}_i = \mathbf{f}_i, i = 1, \cdots, N$, where $\mathbf{f}_i$ is the control force to the ith agent. Given the Jacobi vectors and the dynamics of the agents, the following relationships hold: $\ddot{\mathbf{q}}_j = \mathbf{u}_j, j = 1, \cdots, N-1$ and $N\ddot{\mathbf{r}}_c = \mathbf{f}_c$, where $\mathbf{u}_j$ are the formation control forces that need to be designed and $\mathbf{f}_c$ is the force applied to the formation center. The forces $\mathbf{u}_j$ that use $\mathbf{q}_j$ as feedback, the force $\mathbf{f}_c$ that uses $\mathbf{r}_c$ as feedback, and the forces $\mathbf{f}_i$ are related by the Jacobi transform:

$$[\mathbf{f}_c, \mathbf{u}_1, \cdots, \mathbf{u}_{N-1}] = [\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_N]\Psi. \qquad (7)$$

The control of the formation shape and the control of the formation center motion are decoupled via the Jacobi transform Ψ.

Let $\mathbf{q}_j^0, j = 1, \cdots, N-1$ be the desired Jacobi vectors that define a certain formation. For example, if $N = 3$ and the agents form an equilateral triangle with side length a, then $\mathbf{q}_1^0 = \frac{\sqrt{2}}{2}a\mathbf{e}_1$ and $\mathbf{q}_2^0 = \frac{\sqrt{2}}{2}a\mathbf{e}_2$ where $\mathbf{e}_1$ and $\mathbf{e}_2$ are two desired directions with unit length. In order to keep all the sensing agents in the desired formation so that the $\mathbf{q}_j$'s converge to $\mathbf{q}_j^0$, we use the control laws in the form of: $\mathbf{u}_j = -K_1(\mathbf{q}_j - \mathbf{q}_j^0) - K_2\dot{\mathbf{q}}_j$ for $j = 1, \cdots, N-1$, where K_1 and K_2 are positive gains. It can be proved that under this control law, the sensing agents converge to the desired formation with an exponential rate of convergence [21].

3 Curve Tracking on a Level Surface

In this section, we design 3D steering control laws that control the formation center to move to a desired level surface and track a curve with known curvatures and frame on the level surface.

3.1 Curve Tracking Dynamics

At each time instant, for a 3D scalar field, consider a level surface with the level value z_c passing through the formation center $\mathbf{r}_c$. The gradients of the 3D scalar field are perpendicular to the level surfaces. At the formation center $\mathbf{r}_c$, a unit normal vector $\mathbf{n}$, which is perpendicular to the surface can be defined as $\mathbf{n} = \frac{\nabla z_c}{\|\nabla z_c\|}$. When the formation is moving in the field at unit speed, its velocity vector is a unit vector $\mathbf{X}_1$. The field value z_c, which is estimated by the cooperative Kalman filter, is changing with respect to time:

$$\dot{z}_c = \nabla z_c \cdot \frac{d\mathbf{r}_c}{dt} = \nabla z_c \cdot \mathbf{X}_1 = \|\nabla z_c\| \mathbf{n} \cdot \mathbf{X}_1. \qquad (8)$$

Suppose $\gamma(s)$ is a curve passing through the formation center $\mathbf{r}_c$ that lies on the level surface, where s is the arc-length parameter. Then a right-handed orthonormal frame $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{n})$ for the curve is established where $\mathbf{x}_1$ is the unit tangent vector to the curve and $\mathbf{x}_2$ is defined by $\mathbf{n} \times \mathbf{x}_1$. To describe the trajectory traced by the formation center moving with unit speed, a natural frame [2] can be established. Let $\mathbf{X}_1$ be the unit tangent vector to the trajectory of the formation center, and let $\mathbf{N}_c$ and $\mathbf{X}_2$ be unit normal vectors to the trajectory that are parallel transported along the trajectory from an arbitrarily chosen initial configuration so that $\mathbf{X}_1$, $\mathbf{X}_2$, and $\mathbf{N}_c$ always form an orthornormal basis of $\mathbf{R}^3$. Fig. 1 illustrates the frame $[\mathbf{x}_1, \mathbf{x}_2, \mathbf{n}]$ of the curve $\gamma(s)$ on a level surface that passing through the formation center and the frame $[\mathbf{X}_1, \mathbf{X}_2, \mathbf{N}]$ of the formation center trajectory.

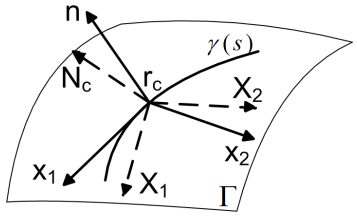


Fig. 1. The frame $[\mathbf{x}_1, \mathbf{x}_2, \mathbf{n}]$ of a curve $\gamma(s)$ on a level surface that is passing through the formation center and the natural frame $[\mathbf{X}_1, \mathbf{X}_2, \mathbf{N}]$ of the trajectory of the formation center.

There are two sets of dynamic equations that are similar to the well-known Frenet-Serret equations that describe the changes of the two frames, one set for the curve $\gamma(s)$ on the level surface, and the other for the trajectory of the formation

center. We list the two sets of equations side by side as follows.

$$\begin{aligned} \dot{\gamma} &= \alpha \mathbf{x}_1 & \dot{\mathbf{r}}_c &= \mathbf{X}_1 \\ \dot{\mathbf{x}}_1 &= \alpha\kappa_n \mathbf{n} + \alpha\kappa_g \mathbf{x}_2 & \dot{\mathbf{X}}_1 &= u\mathbf{N}_c + v\mathbf{X}_2 \\ \dot{\mathbf{x}}_2 &= -\alpha\kappa_g \mathbf{x}_1 + \alpha\tau_g \mathbf{n} & \dot{\mathbf{X}}_2 &= -v\mathbf{X}_1 \\ \dot{\mathbf{n}} &= -\alpha\kappa_n \mathbf{x}_1 - \alpha\tau_g \mathbf{x}_2 & \dot{\mathbf{N}}_c &= -u\mathbf{X}_1. \end{aligned} \tag{9}$$

The term $\alpha = ds/dt$ is the instantaneous rate of change for the curve length of $\gamma(s)$ when the formation center moves. The terms κ_n, κ_g and τ_g are the normal curvature, the geodesic curvature, and the geodesic torsion of the curve $\gamma(s)$ on the level surface. We will discuss their geometric meaning in more detail in Section 4.1. The terms u and v are the steering controls for the formation center moving at the unit speed.

3.2 Steering Control Law Design

We define the steering control problem for the formation center as follows:

Problem 3.1 *Consider the motion of the formation center* $\mathbf{r_c}$ *moving at unit speed and the following assumptions about the 3D scalar field:*

- *A1 Suppose there exists a unique level surface* $\Gamma(\mathbf{r}_c)$ *passing through* $\mathbf{r_c}$ *along the trajectory of* $\mathbf{r_c}$.
- *A2 Suppose a unit tangent vector to a curve* $\gamma(s) \in \Gamma(\mathbf{r}_c)$ *passing through* $\mathbf{r_c}$ *is well defined at* $\mathbf{r}_c$ *and known as* $\mathbf{x_1}$. *This implies that* $\mathbf{x}_1$ *is known or accurately measured at every point of the trajectory of* $\mathbf{r}_c$.
- *A3 Suppose the curvatures (*$\kappa_n(s)$, $\kappa_g(s)$, $\tau_g(s)$*) are bounded and known at* $\mathbf{r}_c$ *for the curve* $\gamma(s)$ *. This implies that the curvatures are known or accurately measured at every point of the trajectory of* $\mathbf{r}_c$.

Given a desired field value C, design the steering control laws u and v so that the formation center converges to the level surface with value C and moves along the curve $\gamma(s)$ *with the tangent direction* $\mathbf{x}_1$. *In other words, as* $t \to \infty$, *the goal is to achieve* $z_c \to C$ *and* $\mathbf{X}_1 \to \mathbf{x}_1$.

Remark 3.2 *Assumptions (A2) and (A3) usually do not specify a unique curve on a level surface to track. Instead, we aim to track one out of a class of curves with desired curvatures and tangent directions. We will use the formation to estimate the tangent* $\mathbf{x}_1$ *and the curvatures. In Section 4, we will show that the lines of curvature of a surface can be traced in this setting.*

The relative displacement between the two frames at the formation center can be described by a set of "shape variables" [12] [20] as $((\mathbf{x}_1\cdot\mathbf{X}_1),(\mathbf{x}_2\cdot\mathbf{X}_1),(\mathbf{n}\cdot\mathbf{X}_1),z_c)$. Define two 3×3 matrices $g_1 = (\mathbf{x}_1,\mathbf{x}_2,\mathbf{n})$ and $g_2 = (\mathbf{X}_1,\mathbf{X}_2,\mathbf{N}_c)$. From the fact that $g_1, g_2 \in SO(3)$, we have the orthonormality conditions that $g_1^T g_1 = \mathbf{I}_{3\times 3}$, $g_2^T g_2 = \mathbf{I}_{3\times 3}$ and $(g_1^T g_2)(g_1^T g_2)^T = \mathbf{I}_{3\times 3}$ [1]. Hence, the last equation and the orthonomality of the frames give

$$\begin{aligned} &(\mathbf{x}_2\cdot\mathbf{N}_c)(\mathbf{x}_1\cdot\mathbf{N}_c) + (\mathbf{x}_2\cdot\mathbf{X}_2)(\mathbf{x}_1\cdot\mathbf{X}_2) = -(\mathbf{x}_2\cdot\mathbf{X}_1)(\mathbf{x}_1\cdot\mathbf{X}_1), \\ &(\mathbf{x}_1\cdot\mathbf{N}_c)(\mathbf{n}\cdot\mathbf{N}_c) + (\mathbf{x}_1\cdot\mathbf{X}_2)(\mathbf{n}\cdot\mathbf{X}_2) = -(\mathbf{x}_1\cdot\mathbf{X}_1)(\mathbf{n}\cdot\mathbf{X}_1), \\ &(\mathbf{x}_1\cdot\mathbf{X}_2)^2 + (\mathbf{x}_1\cdot\mathbf{N}_c)^2 = 1 - (\mathbf{x}_1\cdot\mathbf{X}_1)^2. \end{aligned} \tag{10}$$

These identities will be used to simplify the dynamics of the shape variables.

From the equation $\dot{\mathbf{X}}_1 = u\mathbf{N}_c + v\mathbf{X}_2$, we can derive that $u = \dot{\mathbf{X}}_1\cdot\mathbf{N}_c$ and $v = \dot{\mathbf{X}}_1\cdot\mathbf{X}_2$. Since $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{n}$ form an orthogonal basis of $\mathbf{R}^3$, $\dot{\mathbf{X}}_1$ can be expressed by the linear combination of $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{n}$ as $\dot{\mathbf{X}}_1 = a_1\mathbf{x}_1 + a_2\mathbf{x}_2 + a_3\mathbf{n}$, where a_1, a_2 and a_3 are scalars that depend on the dynamics of the formation center and the curve. Hence, u and v can be represented as

$$\begin{aligned} u &= a_1(\mathbf{x}_1\cdot\mathbf{N}_c) + a_2(\mathbf{x}_2\cdot\mathbf{N}_c) + a_3(\mathbf{n}\cdot\mathbf{N}_c), \\ v &= a_1(\mathbf{x}_1\cdot\mathbf{X}_2) + a_2(\mathbf{x}_2\cdot\mathbf{X}_2) + a_3(\mathbf{n}\cdot\mathbf{X}_2). \end{aligned} \tag{11}$$

The design of u and v becomes finding the parameters (a_1, a_2, a_3). With u and v as in (11), we can obtain that

$$\begin{aligned} &\frac{d(\mathbf{x}_1\cdot\mathbf{X}_1)}{dt} = \dot{\mathbf{x}}_1\cdot\mathbf{X}_1 + \mathbf{x}_1\cdot\dot{\mathbf{X}}_1 \\ &= (\alpha\kappa_n\mathbf{n} + \alpha\kappa_g\mathbf{x}_2)\cdot\mathbf{X}_1 + \mathbf{x}_1\cdot(u\mathbf{N}_c + v\mathbf{X}_2) \\ &= \alpha\kappa_n(\mathbf{n}\cdot\mathbf{X}_1) + \alpha\kappa_g(\mathbf{x}_2\cdot\mathbf{X}_1) + a_1((\mathbf{x}_1\cdot\mathbf{N}_c)^2 + (\mathbf{x}_1\cdot\mathbf{X}_2)^2) \\ &+ a_2((\mathbf{x}_2\cdot\mathbf{N}_c)(\mathbf{x}_1\cdot\mathbf{N}_c) + (\mathbf{x}_2\cdot\mathbf{X}_2)(\mathbf{x}_1\cdot\mathbf{X}_2)) \\ &+ a_3((\mathbf{n}\cdot\mathbf{N}_c)(\mathbf{x}_1\cdot\mathbf{N}_c) + (\mathbf{n}\cdot\mathbf{X}_2)(\mathbf{x}_1\cdot\mathbf{X}_2)). \end{aligned} \tag{12}$$

Applying the identities in (10), $\frac{d(\mathbf{x}_1\cdot\mathbf{X}_1)}{dt}$ becomes

$$\begin{aligned} &\frac{d(\mathbf{x}_1\cdot\mathbf{X}_1)}{dt} = \alpha\kappa_n(\mathbf{n}\cdot\mathbf{X}_1) + \alpha\kappa_g(\mathbf{x}_2\cdot\mathbf{X}_1) + a_1(1-(\mathbf{x}_1\cdot\mathbf{X}_1)^2) \\ &- a_2(\mathbf{x}_2\cdot\mathbf{X}_1)(\mathbf{x}_1\cdot\mathbf{X}_1) - a_3(\mathbf{x}_1\cdot\mathbf{X}_1)(\mathbf{n}\cdot\mathbf{X}_1) \end{aligned} \tag{13}$$

which only depends on the shape variables. Applying similar calculations to $\frac{d(\mathbf{x}_2\cdot\mathbf{X}_1)}{dt}$ and $\frac{d(\mathbf{n}\cdot\mathbf{X}_1)}{dt}$ gives us

$$\begin{aligned} &\frac{d(\mathbf{x}_2\cdot\mathbf{X}_1)}{dt} = -\alpha\kappa_g(\mathbf{x}_1\cdot\mathbf{X}_1) + \alpha\tau_g(\mathbf{n}\cdot\mathbf{X}_1) - a_1(\mathbf{x}_1\cdot\mathbf{X}_1)(\mathbf{x}_2\cdot\mathbf{X}_1) \\ &+ a_2(1-(\mathbf{x}_2\cdot\mathbf{X}_1)^2) - a_3(\mathbf{x}_2\cdot\mathbf{X}_1)(\mathbf{n}\cdot\mathbf{X}_1), \end{aligned} \tag{14}$$

$$\begin{aligned} &\frac{d(\mathbf{n}\cdot\mathbf{X}_1)}{dt} = -\alpha\kappa_n(\mathbf{x}_1\cdot\mathbf{X}_1) - \alpha\tau_g(\mathbf{x}_2\cdot\mathbf{X}_1) - a_1(\mathbf{x}_1\cdot\mathbf{X}_1)(\mathbf{n}\cdot\mathbf{X}_1) \\ &- a_2(\mathbf{x}_2\cdot\mathbf{X}_1)(\mathbf{n}\cdot\mathbf{X}_1) + a_3(1-(\mathbf{n}\cdot\mathbf{X}_1)^2). \end{aligned} \tag{15}$$

If the control laws u and v (e.g. a_1, a_2, and a_3) are designed as feedback laws using only the shape variables, we can then focus on analyzing the closed-loop dynamics of the shape variables described by the equations (8), (13)-(15) as a time-varying nonlinear system. We want to stabilize the equilibrium of the closed-loop dynamics that corresponds to the desired tracking behavior.

Suppose the scalar field has extrema $z_{min} < z_{max}$. Consider a Lyapunov candidate function that is analogous to the one chosen in [12]:

$$V = -\ln(\mathbf{x}_1 \cdot \mathbf{X}_1) + h(z_c), \tag{16}$$

where $h(z_c)$ satisfies the following assumptions:

B1 $h(z_c)$ is continuously differentiable on (z_{min}, z_{max}) and $f(z_c) = \frac{dh}{dz_c}$ is a Lipschitz continuous function.

B2 $f(C) = 0$, and $f(z) \neq 0$ if $z \neq C$ where C is the desired level surface value.

B3 $\lim_{z \to z_{min}} h(z) = \infty, \lim_{z \to z_{max}} h(z) = \infty$ and $\exists \tilde{z}$ such that $h(\tilde{z}) = 0$.

The term $\ln(\mathbf{x}_1 \cdot \mathbf{X}_1)$ in the Lyapunov function aims to align the moving direction of the formation center with the tangent direction of the curve on the level surface. We will prove that as long as we set $\mathbf{x}_1 \cdot \mathbf{X}_1 > 0$ initially, $0 < \mathbf{x}_1 \cdot \mathbf{X}_1 \leq 1$ all the time, which makes the term $-\ln(\mathbf{x}_1 \cdot \mathbf{X}_1) \geq 0$. The other term $h(z_c)$ serves to control the agent to stay on a desired level surface. The derivative of the Lyapunov candidate function can be calculated as

$$\dot{V} = -\frac{1}{\mathbf{x}_1 \cdot \mathbf{X}_1} \frac{d(\mathbf{x}_1 \cdot \mathbf{X}_1)}{dt} + f(z_c)\dot{z}_c \tag{17}$$

If we choose $a_1 = \mu, a_2 = \frac{\alpha \kappa_g}{\mathbf{x}_1 \cdot \mathbf{X}_1}$, and $a_3 = \frac{\alpha \kappa_n}{\mathbf{x}_1 \cdot \mathbf{X}_1} - f(z_c)\|\nabla z_c\|$, where μ is a positive constant and plug (a_1, a_2, a_3) into u and v in equation (11), we get

$$\begin{aligned} u &= \mu(\mathbf{x}_1 \cdot \mathbf{N}_c) + \frac{\alpha \kappa_g}{\mathbf{x}_1 \cdot \mathbf{X}_1}(\mathbf{x}_2 \cdot \mathbf{N}_c) + \frac{\alpha \kappa_n}{\mathbf{x}_1 \cdot \mathbf{X}_1}(\mathbf{n} \cdot \mathbf{N}_c) \\ &\quad - f(z_c)\|\nabla z_c\|(\mathbf{n} \cdot \mathbf{N}_c), \\ v &= \mu(\mathbf{x}_1 \cdot \mathbf{X}_2) + \frac{\alpha \kappa_g}{\mathbf{x}_1 \cdot \mathbf{X}_1}(\mathbf{x}_2 \cdot \mathbf{X}_2) + \frac{\alpha \kappa_n}{\mathbf{x}_1 \cdot \mathbf{X}_1}(\mathbf{n} \cdot \mathbf{X}_2) \\ &\quad - f(z_c)\|\nabla z_c\|(\mathbf{n} \cdot \mathbf{X}_2). \end{aligned} \tag{18}$$

If we plug a_1, a_2 and a_3 into (13) and then use (8), we can calculate that

$$\dot{V} = -\frac{\mu}{\mathbf{x}_1 \cdot \mathbf{X}_1}(1 - (\mathbf{x}_1 \cdot \mathbf{X}_1)^2) \leq 0. \tag{19}$$

We have the following proposition.

Proposition 3.3 *Consider a smooth scalar field and the formation center satisfying assumptions (A1-A3) and the following additional assumptions:*

A4 All level surfaces are compact.

A5 The field has isolated extrema at a finite set of points $\mathbf{R}_{sup}$. Suppose the infimums are all equal to z_{min} and the supremums are all equal to z_{max}.

Let the desired level value $C \in (z_{min}, z_{max})$ be given. Then under the control laws u and v in equations (18) *with assumptions (B1-B3), as $t \to \infty$, we have $\mathbf{X}_1 \to \mathbf{x}_1$ and $z_c \to C$ from all initial states satisfying $\mathbf{x}_1 \cdot \mathbf{X}_1 > 0$ and $\mathbf{r}_c(t_0) \notin \mathbf{R}_{sup}$.*

Proof Consider the Lyapunov candidate function V in (16) and $\dot{V}$ in (19). Since $V \to \infty$ as $\mathbf{x}_1 \cdot \mathbf{X}_1 \to 0$, $z_c \to z_{max}$, or $z_c \to z_{min}$, if the trajectory of the formation center initially satisfies $\mathbf{x}_1 \cdot \mathbf{X}_1 > 0$ and $z_c \in (z_{min}, z_{max})$, then the trajectory will stay in a compact sub-level set of the Lyapunov function V. Let E be the following set within the sub-level set where $\dot{V} = 0$:

$$\begin{aligned} E = \{&((\mathbf{x}_1 \cdot \mathbf{X}_1), (\mathbf{x}_2 \cdot \mathbf{X}_1), (\mathbf{n} \cdot \mathbf{X}_1), z_c) | \\ &(\mathbf{x}_1 \cdot \mathbf{X}_1) = 1, (\mathbf{x}_2 \cdot \mathbf{X}_1) = 0, (\mathbf{n} \cdot \mathbf{X}_1) = 0\}. \end{aligned} \tag{20}$$

Because the closed-loop system is time-varying, we can not apply the classical LaSalle's Invariance Principle. Instead, a more advanced invariance theorem can be applied (Theorem 8.4 in [13]) to claim that the trajectory will converge to the set E when $t \to \infty$. At points in E, the closed loop system becomes

$$\begin{aligned} &\dot{z}_c = 0, \ \frac{d(\mathbf{x}_1 \cdot \mathbf{X}_1)}{dt} = 0, \ \frac{d(\mathbf{x}_2 \cdot \mathbf{X}_1)}{dt} = 0, \\ &\frac{d(\mathbf{n} \cdot \mathbf{X}_1)}{dt} = -f(z_c)\|\nabla z_c\|. \end{aligned} \tag{21}$$

In the current context, $\mathbf{n} \cdot \mathbf{X}_1 = 0$ on set E and we have shown that the dynamics will converge to set E, hence $\mathbf{n} \cdot \mathbf{X}_1 \to 0$. According to the Barbalat Lemma (Lemma 8.2 in [13]), if $f(z_c)\|\nabla z_c\|$ is uniformly continuous and $\mathbf{n} \cdot \mathbf{X}_1 \to 0$, then $\frac{d(\mathbf{n} \cdot \mathbf{X}_1)}{dt} \to 0$ must hold. Since all level surfaces are compact and the field is smooth, it is straightforward to show that $\|\nabla z_c\|$ is uniformly continuous along smooth curves with bounded curvatures on the level surfaces. Therefore, we conclude that $f(z_c)\|\nabla z_c\| = 0$, which implies that $f(z_c) = 0$ on E. This means the tangent vector $\mathbf{X}_1$ to the trajectory of the formation center will be aligned with the known tangent vector $\mathbf{x}_1$ along the curve and the field value at the formation center will converge to the desired constant value C. ■

4 Curvature Estimation Using Formations

For exploration problems, the field that is going to be explored is unknown. Assumptions (A2) and (A3) can only be satisfied based on estimates made by sensing agents. We consider a special case, which is to detect and track one of the lines of curvature on a desired level surface [18]. We design a formation formed by N agents so that by combining the measurements taken by all of the agents, the curvatures and the directions of a line of curvature can be estimated.

4.1 Principal Curvatures and Directions

We start with reviewing the definition of the lines of curvature briefly [14]. As shown in Fig. 2, $\gamma(s)$ is a curve that lies on a smooth surface Γ, which can be described by the equations (9) (left). $\gamma_1(s_1)$ is another curve which also lies on Γ and intersects with $\gamma(s)$ at the point $\mathbf{r}_c$. As introduced in Section 3.1, the frame $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{n})$ is used to describe the

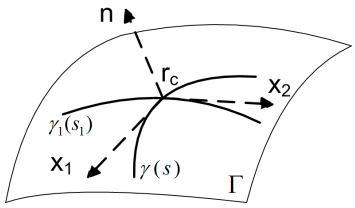


Fig. 2. Two curves on a level surface Γ. $\mathbf{x}_1$ and $\mathbf{x}_2$ are the tangent vectors of $\gamma(s)$ and $\gamma_1(s_1)$. $\mathbf{n}$ is the normal vector to Γ at $\mathbf{r}_c$.

curve $\gamma(s)$. If the curve $\gamma_1(s_1)$ has $\mathbf{x}_2$ as its unit tangent vector at $\mathbf{r}_c$, then at the same point $\mathbf{r}_c$, the frame for $\gamma_1(s_1)$ is $(\mathbf{x}_2, -\mathbf{x}_1, \mathbf{n})$.

Suppose κ_n and κ_{1n} are the normal curvatures of $\gamma(s)$ and $\gamma_1(s_1)$ at the point $\mathbf{r}_c$, which are also known as the directional curvatures of the surface Γ at $\mathbf{r}_c$ in the directions $\mathbf{x}_1$ and $\mathbf{x}_2$. Among all possible directional curvatures of the surface Γ at $\mathbf{r}_c$, if κ_n takes the maximum value along $\mathbf{x}_1$, then κ_n is one of the principal curvatures and $\mathbf{x}_1$ is the corresponding principal direction of Γ at $\mathbf{r}_c$. Since $\mathbf{x}_1$ and $\mathbf{x}_2$ are perpendicular to each other, then $\mathbf{x}_2$ is another principal direction and κ_{1n} is the corresponding principal curvature with the minimum value among all directional curvatures of Γ at $\mathbf{r}_c$. Note that the principal directions may not be unique for some smooth surfaces such as a sphere. If the tangent direction $\mathbf{x}_1$ of $\gamma(s)$ at each point is a principal direction at that point, then $\gamma(s)$ is a line of curvature of the surface Γ. Another important property of lines of curvature is that the geodesic torsion τ_g is zero. Examples of the lines of curvature are the meridians and circles of latitude of a surface of revolution, such as a cylinder.

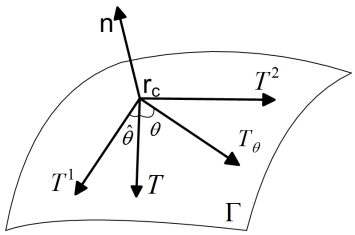


Fig. 3. $\mathbf{T}^1$ and $\mathbf{T}^2$ are the two principal directions of the surface at $\mathbf{r}_c$. $\mathbf{T}$ and $\mathbf{T}_\theta$ are two arbitrarily chosen tangent vectors that form certain angles with $\mathbf{T}^1$. $\mathbf{n}$ is the normal vector to the surface at $\mathbf{r}_c$.

4.2 Taubin's Algorithm

To estimate the principal directions and the principal curvatures of a line of curvature on a level surface, we introduce the curvature estimation algorithm described by Taubin in [17]. As shown in Fig. 3, let $\mathbf{T}^1$ and $\mathbf{T}^2$ denote the two principal directions of the surface Γ at the point $\mathbf{r}_c$ with corresponding principal curvatures κ^1 and κ^2 where $\kappa^1 > \kappa^2$. Choose an arbitrary unit tangent vector $\mathbf{T}$ to the surface at $\mathbf{r}_c$ that forms an angle $\hat{\theta}$ with $\mathbf{T}^1$ where $\hat{\theta}$ is unknown. For $-\pi < \theta < \pi$, define another unit tangent vector $\mathbf{T}_\theta$ to the surface at $\mathbf{r}_c$ that forms an angle θ with $\mathbf{T}$. Let $\kappa_p(\mathbf{T}_\theta)$ be the directional curvature associated with the direction $\mathbf{T}_\theta$. Then a symmetric matrix $\mathbf{M}_p$ can be constructed by an integral formula as

$$\mathbf{M}_p = \frac{1}{2\pi}\int_{-\pi}^{+\pi} \kappa_p(\mathbf{T}_\theta)\mathbf{T}_\theta\mathbf{T}_\theta^T d\theta. \tag{22}$$

It can be shown that the principal directions and the unit normal vector are the eigenvectors of $\mathbf{M}_p$, which can be computed by diagonalizing $\mathbf{M}_p$ as

$$\mathbf{M}_p = \begin{pmatrix} \mathbf{T}^1 & \mathbf{T}^2 & \mathbf{n} \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{T}^1 & \mathbf{T}^2 & \mathbf{n} \end{pmatrix}^T, \tag{23}$$

where λ_1 and λ_2 are the two non-zero eigenvalues of $\mathbf{M}_p$. It is further shown in [17] that the principal curvatures can be calculated as $\kappa^1 = 3\lambda_1 - \lambda_2$ and $\kappa^2 = 3\lambda_2 - \lambda_1$.

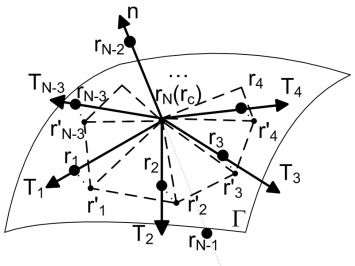


Fig. 4. $\mathbf{r}_c$ is the center of the formation. $\mathbf{r}'_i, i = 1, \cdots, N-3$ are points on the level surface obtained by searching along either the negative or positive direction of the normal vector $\mathbf{n}$ starting from $\mathbf{r}_i, i = 1, \cdots, N-3$. $\mathbf{T}_i, i = 1, \cdots, N-3$ are projections of $\mathbf{r}'_i - \mathbf{r}_c$ to the tangent plane of Γ at $\mathbf{r}_c$.

We introduce a *discretized Taubin's algorithm* for estimating curvatures using formations. We arrange a formation formed by N agents as illustrated in Fig 4. We allocate $N-3$ agents on a plane in a circular fashion, among which we arbitrarily select one as $\mathbf{r}_1$ and label the others $\mathbf{r}_2, \cdots, \mathbf{r}_{N-3}$ counterclockwisely. The remaining three agents are allocated along a line perpendicular to the plane with the Nth agent located at the center of the formation formed by the $N-3$ agents and the agents $N-1$ and $N-2$ located symmetrically on the opposite sides of the plane. The position of the Nth agent $\mathbf{r}_N$ overlaps with the formation center $\mathbf{r}_c$. Note that this configuration requires $N > 4$. If $N = 5$, the first two agents only form a line instead of a plane. The formation can be stabilized with the cooperative control laws based on the Jacobi vectors. With the formation control law described in Section 2.3, the $N-3$ agents can be controlled to lie on the tangent plane of $\Gamma(\mathbf{r}_c)$ and agents $\mathbf{r}_{N-2}, \mathbf{r}_{N-1}$, and $\mathbf{r}_N$ can be controlled to be aligned with the direction of $\mathbf{n}$ by correctly selecting $\mathbf{q}_i^0, i = 1, \cdots, N-1$. We assume that all such formation control goals have been achieved.

The discretized Taubin's algorithm is as follows.

Algorithm 4.1 *Denote $\hat{\mathbf{n}}$ as the estimate of $\mathbf{n}$. Starting from $\mathbf{r}_1,...,\mathbf{r}_{N-3}$ and searching along the positive or negative directions of $\hat{\mathbf{n}}$ obtained at the previous time instant, we can find $\mathbf{r}'_1,...,\mathbf{r}'_{N-3}$, which lie on the level surface $\Gamma(\mathbf{r}_c)$ and divide $\Gamma(\mathbf{r}_c)$ into $N-3$ triangular faces. We label the triangular faces as $f_i, i=1,\cdots,N-3$. The unit vectors $\mathbf{T}_i, i=1,...,N-3$ represent the projections of the vectors $\mathbf{r}'_i-\mathbf{r}_c$ to the tangent plane of the surface $\Gamma(\mathbf{r}_c)$. With this setting, the steps to estimate the principal curvatures and principal directions with N agents are as follows:*

(1) estimate the unit normal vector $\mathbf{n}$ at $\mathbf{r}_c$. Let $\mathbf{n}_{f_i}$ be the unit normal vector to the face f_i. For $i=1,\cdots,N-4$, $\mathbf{n}_{f_i}=\frac{\mathbf{r}_i-\mathbf{r}_c}{\|\mathbf{r}_i-\mathbf{r}_c\|}\times\frac{\mathbf{r}_{i+1}-\mathbf{r}_c}{\|\mathbf{r}_{i+1}-\mathbf{r}_c\|}$. For the face f_{N-3}, $\mathbf{n}_{f_{N-3}}=\frac{\mathbf{r}_{N-3}-\mathbf{r}_c}{\|\mathbf{r}_{N-3}-\mathbf{r}_c\|}\times\frac{\mathbf{r}_1-\mathbf{r}_c}{\|\mathbf{r}_1-\mathbf{r}_c\|}$. Then $\mathbf{n}$ can be estimated by $\hat{\mathbf{n}}=\frac{\sum_{i=1}^{N-3}|f_k|\mathbf{n}_{f_i}}{\|\sum_{i=1}^{N-3}|f_i|\mathbf{n}_{f_i}\|}$, where $|f_i|$ are the areas of the faces f_i.

(2) compute the projections $\mathbf{T}_i$. Since the tangent plane of $\Gamma(\mathbf{r}_c)$ at $\mathbf{r}_c$ is perpendicular to $\hat{\mathbf{n}}$, $\mathbf{T}_i$ can be estimated using $\mathbf{T}_i=\frac{(\mathbf{r}'_i-\mathbf{r}_c)-((\mathbf{r}'_i-\mathbf{r}_c)\cdot\hat{\mathbf{n}})\hat{\mathbf{n}}}{\|(\mathbf{r}'_i-\mathbf{r}_c)-((\mathbf{r}'_i-\mathbf{r}_c)\cdot\hat{\mathbf{n}})\hat{\mathbf{n}}\|}$.

(3) approximate the matrix $\mathbf{M}_p$ in (22) as

$$\mathbf{M}_v=\sum_{i=1}^{N-3}\omega_i\kappa_i\mathbf{T}_i\mathbf{T}_i^T, \tag{24}$$

where ω_i are the weights that depend on $|f_i|$ and satisfy $\sum\omega_i=1$. κ_i are the directional curvatures associated with $\mathbf{T}_i$ and are approximated by $\kappa_i=\frac{2\hat{\mathbf{n}}^T(\mathbf{r}'_i-\mathbf{r}_c)}{\|\mathbf{r}'_i-\mathbf{r}_c\|^2}$.

(4) diagonalize $\mathbf{M}_v$ to obtain the estimated principal directions $\hat{\mathbf{T}}^1$ and $\hat{\mathbf{T}}^2$, as well as the estimated principal curvatures $\hat{\kappa}^1$ and $\hat{\kappa}^2$. Therefore, the frame of a line of curvature that is associated with the larger principal curvature can be estimated by $\hat{\mathbf{x}}_1=\hat{\mathbf{T}}^1$, $\hat{\mathbf{x}}_2=\hat{\mathbf{T}}^2$ and $\hat{\kappa}_n=\hat{\kappa}^1$.

Remark 4.2 *In the step (2), the projections $\mathbf{T}_i$ can be approximated by $\mathbf{r}_i-\mathbf{r}_c$ when the formation converges and the agents $1,\cdots,N-3$ stay in the tangent plane of the surface at the position $\mathbf{r}_c$.*

4.3 Geodesic Curvature Estimation

The geodesic curvature measures how a curve is curving in the surface M. The geodesic curvature κ_g, the normal curvature κ_n, and the Frenet-Serret curvature κ of a curve are related by $\kappa^2=\kappa_n^2+\kappa_g^2$.

Algorithm 4.3 *Knowing the consecutive positions of the formation center $\mathbf{r}_{c,k-2},\mathbf{r}_{c,k-1},\mathbf{r}_{c,k}$ and $\mathbf{r}_{c,k+1}$,*

(1) compute the unit tangent vector to the trajectory of $\gamma(s)$ at time instant k, which should be aligned with $\mathbf{x}_1$ that can be approximated by $\hat{\mathbf{T}}_k=\frac{\mathbf{r}_{c,k+1}-\mathbf{r}_{c,k-1}}{\|\mathbf{r}_{c,k+1}-\mathbf{r}_{c,k-1}\|}$.

(2) compute the Frenet-Serret curvature. With the estimated tangent vectors $\hat{\mathbf{T}}_k$ and $\hat{\mathbf{T}}_{k-1}$, the Frenet-Serrat curvature κ can be estimated as $\hat{\kappa}=\frac{\arccos(\hat{\mathbf{T}}_k\cdot\hat{\mathbf{T}}_{k-1})}{\|\mathbf{r}_{c,k}-\mathbf{r}_{c,k-1}\|}$.

(3) estimate the geodesic curvature. Since we have obtained $\hat{\kappa}_n$, the geodesic curvature κ_g can be calculated by $\hat{\kappa}_g=\sqrt{\hat{\kappa}^2-\hat{\kappa}_n^2}$.

Until now, we have estimated all the information needed by assumptions (A2) and (A3) for tracking a line of curvature with $\hat{\tau}_g=0$.

4.4 Constraints on Agent Quantity and Formation Design

The discretized Taubin's algorithm approximates the integral formula for $\mathbf{M}_p$ with a finite sum that computes $\mathbf{M}_v$. The number of agents and the formation will affect the estimation accuracy. Under this concern, we discuss the constraints on the agent quantity. For $\Gamma(\mathbf{r}_c)$, assume that there exist two unique principal directions $\mathbf{T}^1\in T_c\Gamma$ and $\mathbf{T}^2\in T_c\Gamma$ where $T_c\Gamma$ is the tangent plane of $\Gamma(\mathbf{r}_c)$ at $\mathbf{r}_c$. With the configuration shown in Fig. 4, denote the angle from the vector $\mathbf{T}_1$ to $\mathbf{T}_i, i=1,\cdots,N-3$ as $\theta_i\in(-\pi,\pi]$. Under this setting, $\theta_1=0$. Define a set $\Omega=\{\mathbf{T}|\mathbf{T}\in T_c\Gamma,\mathbf{T}\neq\mathbf{T}^1,\mathbf{T}\neq\mathbf{T}^2,\|\mathbf{T}\|=1\}$. We assume that the tangent vector $\mathbf{T}_1$ is selected so that $\mathbf{T}_1\in\Omega$. With this configuration, we propose the following proposition.

Proposition 4.4 *Consider a formation with N agents as illustrated in Fig. 4 with the assumptions that $\mathbf{T}_1\in\Omega$ and that the surface $\Gamma(\mathbf{r}_c)$ has two unique principal directions at $\mathbf{r}_c$. Then the following statements hold for the discretized Taubin's algorithm 4.1;*

(1) the algorithm provide nonsingular estimates of principal curvatures and principal directions if and only if

$$\sum_{i=1}^{N-3}\omega_i\kappa_i\sin 2\theta_i\neq 0, \tag{25}$$

where θ_i is the angle between $\mathbf{T}_i$ and $\mathbf{T}_1$, and $\theta_1=0$.

(2) $N\geq 6$ must be satisfied to avoid singularity in the estimates. If the formation is symmetric, then $N\neq 7$.

Proof for Statement (1). Choose $\mathbf{T}_1$ and the corresponding orthonormal vector $\mathbf{T}_1^\perp$ as the basis of the tangent plane, then $\mathbf{T}_i$ can be written as: $\mathbf{T}_i=\mathbf{T}_1\cos\theta_i+\mathbf{T}_1^\perp\sin\theta_i$, $i=1,2,...,N-3$. Substitute $\mathbf{T}_i$ into equation (24), we can obtain

$$\begin{aligned}\mathbf{M}_v=\sum_{i=1}^{N-3}\omega_i\kappa_i(&\mathbf{T}_1\mathbf{T}_1^T\cos^2\theta_i+\mathbf{T}_1(\mathbf{T}_1^\perp)^T\cos\theta_i\sin\theta_i\\&+\mathbf{T}_1^\perp\mathbf{T}_1^T\cos\theta_i\sin\theta_i+\mathbf{T}_1^\perp(\mathbf{T}_1^\perp)^T\sin^2\theta_i).\end{aligned} \tag{26}$$

Suppose $\hat{\mathbf{T}}^1$ is one of the estimated principal directions that can be expressed as $\hat{\mathbf{T}}^1=\mathbf{T}_1\cos\hat{\theta}+\mathbf{T}_1^\perp\sin\hat{\theta}$ where

$\hat{\theta} \in (-\frac{\pi}{2}, \frac{\pi}{2}]$ is the angle between $\hat{\mathbf{T}}^1$ and $\mathbf{T}_1$. Then according to Taubin's algorithm, we can write down the following relationship:

$$\mathbf{M}_v \hat{\mathbf{T}}^1 = \hat{\lambda}_1 \hat{\mathbf{T}}^1 = \mathbf{T}_1 \hat{\lambda}_1 \cos\hat{\theta} + \mathbf{T}_1^{\perp} \hat{\lambda}_1 \sin\hat{\theta}, \qquad (27)$$

where $\hat{\lambda}_1$ is the eigenvalue corresponding to $\hat{\mathbf{T}}^1$. On the other hand, $\mathbf{M}_v \hat{\mathbf{T}}^1 = \mathbf{M}_v(\mathbf{T}_1 \cos\hat{\theta} + \mathbf{T}_1^{\perp} \sin\hat{\theta})$. Substitute $\mathbf{M}_v$ in equation (26) into the above equation and use the relationship $\mathbf{T}_1^T \mathbf{T}_1 = (\mathbf{T}_1^{\perp})^T \mathbf{T}_1^{\perp} = 1$ and $(\mathbf{T}_1^{\perp})^T \mathbf{T}_1 = \mathbf{T}_1^T \mathbf{T}_1^{\perp} = 0$, $\mathbf{M}_v \hat{\mathbf{T}}^1$ can be calculated as

$$\begin{aligned} \mathbf{M}_v \hat{\mathbf{T}}^1 = \mathbf{T}_1 [\sum_{i=1}^{N-3} \omega_i \kappa_i (\cos^2\theta_i \cos\hat{\theta} + \frac{1}{2}\sin 2\theta_i \sin\hat{\theta})] \\ + \mathbf{T}_1^{\perp} [\sum_{i=1}^{N-3} \omega_i \kappa_i (\sin^2\theta_i \sin\hat{\theta} + \frac{1}{2}\sin 2\theta_i \cos\hat{\theta})]. \quad (28) \end{aligned}$$

Hence, comparing with equation (27), we have

$$\begin{aligned} \hat{\lambda}_1 &= \sum_{i=1}^{N-3} \omega_i \kappa_i \cos^2\theta_i + \frac{1}{2}\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i \tan\hat{\theta} \\ &= \sum_{i=1}^{N-3} \omega_i \kappa_i \sin^2\theta_i + \frac{1}{2}\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i \cot\hat{\theta}. \qquad (29) \end{aligned}$$

Suppose $\sum_{i=1}^{N-2} \omega_i \kappa_i \sin 2\theta_i \neq 0$, then the above two equations give well defined solutions for $\hat{\theta}$ that satisfy:

$$\tan^2\hat{\theta} + \frac{2\sum_{i=1}^{N-3} \omega_i \kappa_i \cos 2\theta_i}{\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i} \tan\hat{\theta} - 1 = 0. \qquad (30)$$

For each solution $\hat{\theta}$, the estimated eigenvector $\hat{\mathbf{T}}^1$ has the form of $\mathbf{T}_1 \cos\hat{\theta} + \mathbf{T}_1^{\perp} \sin\hat{\theta}$. This finishes the proof for the sufficient condition. From the relationship $\mathbf{T}_1^T \mathbf{T}_1 = 1$ and $(\mathbf{T}_1^{\perp})^T \mathbf{T}_1 = 0$, we also have

$$\mathbf{M}_v \mathbf{T}_1 = \mathbf{T}_1 \sum_{i=1}^{N-3} \omega_i \kappa_i \cos^2\theta_i + \frac{1}{2}\mathbf{T}_1^{\perp} \sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i. \qquad (31)$$

We now use proof by contradiction to show the necessity. Suppose the term $\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i$ sums to zero, then $\mathbf{M}_v \mathbf{T}_1 = \mathbf{T}_1 \sum_{i=1}^{N-3} \omega_i \kappa_i \cos^2\theta_i = \lambda_1 \mathbf{T}_1$, where λ_1 is a scalar. From equation (31), we can see that $\mathbf{T}_1$ is one of the eigenvectors of $\mathbf{M}_v$ and λ_1 is the corresponding eigenvalue. According to Taubin's algorithm, this results in $\mathbf{T}_1$ being one of the principal directions. However, $\mathbf{T}_1$ is not aligned with any principal directions since $\mathbf{T}_1 \in \Omega$. This contradiction means that Taubin's algorithm can produce estimates of principal directions only if $\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i \neq 0$.

Proof for Statement (2). Consider a symmetric formation where the angles between $\mathbf{T}_1$ and $\mathbf{T}_i, i = 1, \cdots, N-3$ can be expressed as $\theta_i = \frac{2\pi}{N-3}(i-1)$. When $N = 5$, according to our formation design, agents 1 and 2 form a line and the agent 5 is located at the center of the line, which always gives us a symmetric formation. From the relationship $\sum_{i=1}^{N-3} \omega_i \kappa_i \sin 2\theta_i = \sum_{i=1}^{N-3} \omega_i \kappa_i \sin(\frac{4\pi}{N-3}(i-1))$, we can obtain that for $N = 5$, $\omega_1 \kappa_1 \sin 0 + \omega_2 \kappa_2 \sin 2\pi = 0$. In addition, when $N = 7$, we have $\omega_1 \kappa_1 \sin 0 + \omega_2 \kappa_2 \sin\pi + \omega_3 \kappa_3 \sin 2\pi + \omega_4 \kappa_4 \sin 3\pi = 0$. The summations will be zero regardless of the labeling of the sensor platforms and the values of $\omega_i \kappa_i$, which violates the condition (25). This fact indicates that we can not deploy five or seven agents arranged in the symmetric formation to implement Taubin's algorithm.

When $N = 6$, if the assumptions of the proposition are satisfied, the estimated $\hat{\theta}$ can be solved from

$$\begin{aligned} \tan^2\hat{\theta} + \frac{2(\omega_1 \kappa_1 + \omega_2 \kappa_2 \cos(2\theta_2) + \omega_3 \kappa_3 \cos(2\theta_3))}{\omega_2 \kappa_2 \sin(2\theta_2) + \omega_3 \kappa_3 \sin(2\theta_3)} \tan\hat{\theta} \\ - 1 = 0. \qquad (32) \end{aligned}$$

Therefore, the minimum number of agents that can be utilized without producing singular estimates is six. ■

Notice that for the symmetric formation, because of the relationship: $\sum_{i=1}^{N-3} \sin\frac{4\pi}{N-3}(i-1) = 0, \forall N \geq 6$, the condition (25) in Proposition 4.4 is violated if the term $\omega_i \kappa_i$ are identical. Since we assume that for the smooth surface $\Gamma(\mathbf{r}_c)$, there exist two unique principal directions $\mathbf{T}^1$ and $\mathbf{T}^2$, we can select ω_i so that $\omega_i \kappa_i$ are not identical. For example, $\omega_i = 1, i = 1, \cdots, N-3$.

Remark 4.5 *Proposition 4.4 suggests that when we design a formation using N agents as illustrated in Fig. 4 to implement the discretized Taubin's algorithm to provide estimates of the principal directions and principal curvatures on a level surface, more than six agents should be used. In addition, we can not use seven agents in a symmetric formation to implement Taubin's algorithm.*

5 Cooperative Hessian Estimation

As seen in the state equation (5) and the measurement equation (3) of the Kalman filter, the Hessian matrix of the field at the formation center needs to be estimated in order to enable the Kalman filter. As shown in Fig. 2, $\gamma(s)$ and $\gamma_1(s_1)$ are two intersecting curves on a level surface Γ. We can write down the dynamic equations for $\gamma(s)$ and $\gamma_1(s_1)$ side by side,

$$\begin{aligned} \mathbf{x}_1' &= \kappa_n \mathbf{n} + \kappa_g \mathbf{x}_2 & \mathbf{x}_2' &= \kappa_{1n} \mathbf{n} - \kappa_{1g} \mathbf{x}_1 \\ \mathbf{x}_2' &= -\kappa_g \mathbf{x}_1 + \tau_g \mathbf{n} & \mathbf{x}_1' &= \kappa_{1g} \mathbf{x}_2 - \tau_{1g} \mathbf{n}, \\ \mathbf{n}' &= -\kappa_n \mathbf{x}_1 - \tau_g \mathbf{x}_2 & \mathbf{n}' &= -\kappa_{1n} \mathbf{x}_2 + \tau_{1g} \mathbf{x}_1. \end{aligned} \qquad (33)$$

where $'$ represents the derivative with respect to the arclength parameter s or s_1 and κ_{1n}, κ_{1g} and τ_{1g} are the normal curvature, the geodesic curvature and geodesic torsion of $\alpha_1(s_1)$, respectively.

From the fact that the gradients of the surface are always perpendicular to the tangent plane, we have the following relationships: $\nabla z(\mathbf{r}_c)\cdot\mathbf{x}_1 = 0, \nabla z(\mathbf{r}_c)\cdot\mathbf{x}_2 = 0$ and $\nabla z(\mathbf{r}_c)\cdot\mathbf{n} = \|\nabla z(\mathbf{r}_c)\|$. If we take derivatives on both sides of $\nabla z(\mathbf{r}_c)\cdot\mathbf{x}_1 = 0$ with respect to s and use the relationship $\frac{d}{ds}\nabla z(\mathbf{r}_c) = \mathbf{x}_1^T\nabla^2 z(\mathbf{r}_c)$, we can obtain

$$\mathbf{x}_1^T\nabla^2 z(\mathbf{r}_c)\mathbf{x}_1 + \|\nabla z(\mathbf{r}_c)\|\mathbf{n}\cdot(\kappa_n\mathbf{n} + \kappa_g\mathbf{x}_2) = 0. \qquad (34)$$

In the frame described in equation (33) (left) for the curve $\gamma(s)$, since $\mathbf{x}_1$ is a unit vector along the $\mathbf{x}_1$ axis, and $\mathbf{x}_1, \mathbf{x}_2$ are perpendicular to each other, from the equation (34), we have $\partial_{x_1x_1}z(\mathbf{r}_c) = -\|\nabla z(\mathbf{r}_c)\|\kappa_n$. Therefore, the estimate of the first element of Hessian matrix is given by $H_{c(11)} = \partial_{x_1x_1}z(\mathbf{r}_c) = -\|\nabla z(\mathbf{r}_c)\|\kappa_n$. Also, if we take derivatives on both sides of $\nabla z(\mathbf{r}_c)\cdot\mathbf{x}_2 = 0$ and $\nabla z(\mathbf{r}_c)\cdot\mathbf{n} = \|\nabla z(\mathbf{r}_c)\|$, similar calculations can be conducted, which give us the estimates of $H_{c(12)}$ and $H_{c(13)}$: $H_{c(12)} = -\|\nabla z(\mathbf{r}_c)\|\tau_g, H_{c(13)} = \frac{d}{ds}\|\nabla z(\mathbf{r}_c)\|$. Use the similar steps to analyze the curve $\alpha_1(s_1)$, we can estimate $H_{c(22)}, H_{c(21)}$ and $H_{c(23)}$ by $H_{c(22)} = -\|\nabla z(\mathbf{r}_c)\|\kappa_{1n}, H_{c(21)} = -\|\nabla z(\mathbf{r}_c)\|\tau_{1g}$ and $H_{c(23)} = \frac{d}{ds_1}\|\nabla z(\mathbf{r}_c)\|$. Since the field is considered to be smooth, the Hessian matrix is symmetric. Therefore, $H_{c(13)} = H_{c(31)} = \frac{d}{ds}\|\nabla z(\mathbf{r}_c)\|$ and $H_{c(23)} = H_{c(32)} = \frac{d}{ds_1}\|\nabla z(\mathbf{r}_c)\|$. In addition, from the relationship $H_{c(12)} = H_{c(21)}$, we have $\tau_g = \tau_{1g}$. Note again that if $\gamma(s)$ and $\gamma_1(s_1)$ are lines of curvature on a surface, the geodesic torsion $\tau_g = \tau_{1g} = 0$, which means $H_{c(12)} = H_{c(21)} = 0$. Note that $\kappa_n = \hat{\kappa}^1$ and $\kappa_{1n} = \hat{\kappa}^2$. With the formation designed in the previous section, the last element of the Hessian $H_{c(33)}$ can be approximated by $H_{33} = \frac{\frac{z_{N-1}-z_N}{\|\mathbf{r}_{N-1}-\mathbf{r}_N\|} - \frac{z_N-z_{N-2}}{\|\mathbf{r}_N-\mathbf{r}_{N-2}\|}}{\|\mathbf{r}_{N-1}-\mathbf{r}_{N-2}\|}$.

6 Simulation Results

We demonstrate the cooperative exploration algorithm utilizing six agents. We assume that the measurements taken and the positions are shared among all the agents. At each time instant, the agents take new measurements of the field, then the cooperative Kalman filter, the curvature estimation and the Hessian estimation are computed to find the steering control forces u and v as described in Section 3. Meanwhile, the formation shape control forces are also calculated.

In the simulation illustrated in Fig. 5 and Fig. 6, three of the six agents (two are plotted as triangles and one is plotted as a circle) lie in the tangent plane of a level surface passing through the formation center and form a symmetric triangular formation. The distance between each pair of the three agents in the plane is 0.6. The other three agents (rectangular markers) are lying in a line perpendicular to the tangent plane with the sixth agent sitting in the formation center. To satisfy the constraints discussed in Section 4.4, we control the orientation of the formation so that none of the vectors connecting an agent to the formation center aligns with any principal directions of the level surface. This is accomplished by selecting the Jacobi vectors $\mathbf{q}_i$ and $\mathbf{q}_i^0, i = 1, \cdots, N-1$ so that the vector connecting the agent one (the circle) and the formation center forms an angle $\frac{\pi}{8}$ with the estimated principal direction associated with the larger principal curvature.

The goal is to detect and track one of the lines of curvature on a desired level surface in an unknown 3D scalar field with 5% i.i.d. Gaussian noise. The unknown fields are composed of cylindrical level surfaces and ellipsoidal level surfaces. We only plot one of the level surfaces on each figure with the level value $C = 20$ and set it as the desired level value that the formation center should converge to. The lines of curvature with the larger principal curvatures for both level surfaces are shown by the circles on the level surfaces in the figure. The thick lines are the trajectories of the formation center. The initial positions of the formation center are at the position $(4.3, 0, 0)$, which are -0.2 off the desired level surfaces. The six agents converge to a constant formation while the formation center moves to the desired level surfaces, and track one of the lines of curvature.

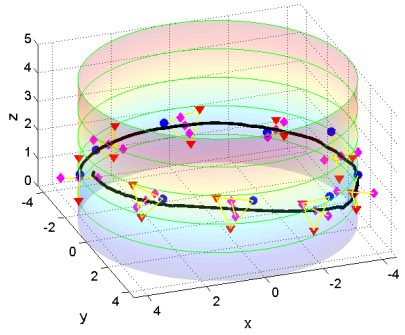


Fig. 5. Detecting and tracking a line of curvature on a cylinder by six agents. The desired level value C = 20.

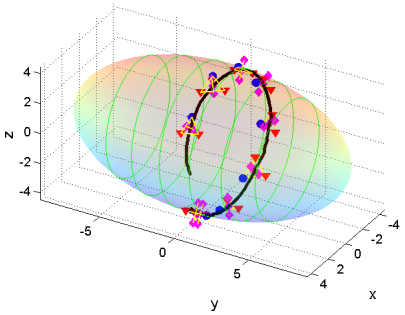


Fig. 6. Detecting and tracking a line of curvature on a ellipsoid by six agents. The desired level value C = 20.

Denote the angle between $\mathbf{T}^1$ and the inertial frame as β and the angle between $\hat{\mathbf{T}}_1$ and the inertial frame as $\hat{\beta}$. To compare the estimated principal directions with the actual principal directions, we plot $\beta - \hat{\beta}$ in Fig. 7. We can tell

that with three agents estimating the principal directions, the error is within ± 20 degree.

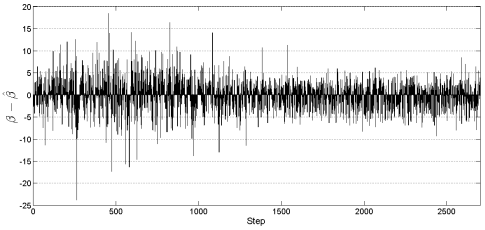


Fig. 7. Estimation error between $\hat{\beta}$ and β.

7 Conclusions

The steering control laws are able to control a formation formed by N agents to move to a desired level surface and track a class of curves in a 3D scalar field. We have shown that a discretized Taubin's algorithm, the Hessian estimation and the cooperative Kalman filter can be combined to allow a group of agents to perform cooperative exploration of 3D level surfaces by tracking lines of curvature.

References

[1] Kirillov. Alexander. *An introduction to Lie groups and Lie algebras*. Cambridge University Press, 2008.

[2] R. L. Bishop. There is more than one way to frame a curve. *The American Mathematical Monthly*, 82(3):246–251, 1975.

[3] Y. Cao, A. Fukunaga, and A. Khang. Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–27, 1997.

[4] X. Chen and F. Schmitt. Intrinsic surface properties from surface triangulation. In *Proc. European Conference on Computer Vision*, pages 739–743, Santa Margherita Ligure, Italy, 1992.

[5] M. P. do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ, 1976.

[6] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. A taxonomy for multiagent robotics. *Autonomous Robots*, 3(4):375–397, 1996.

[7] R. Ghabcheloo, A. P. Aguiar, A. Pascoal, C. Silvestre, I. Kaminer, and J. Hespanha. Coordinated path-following in the presence of communication losses and time delays. *SIAM, Journal on Control and Optimization*, 48(1):234–265, 2009.

[8] E. Hameiri and I. Shimshoni. Estimating the principal curvatures and the Darboux frame from real 3D range data. *IEEE Transactions on Systems, Man and Cybernetics*, 33(4):626–637, 2003.

[9] Z. Jin and A. L. Bertozzi. Environmental boundary tracking and estimation using multiple autonomous vehicles. In *Proc. of the 46th IEEE Conf. on Decision and Control*, pages 4918–4923, New Orleans, LA, 2008.

[10] A. Joshi, T. Ashley, Y. Huang, and A. L. Bertozzi. Experimental validation of cooperative environmental boundary tracking with on-board sensors. In *Proc. of 2009 American Control Conference*, pages 2630–2635, St. Louis, MO, 2009.

[11] E. W. Justh and P. S. Krishnaprasad. Equilibria and steering laws for planar formations. *Systems and Control Letters*, 52(1):25–38, 2004.

[12] E. W. Justh and P. S. Krishnaprasad. Natural frames and interacting particles in three dimensions. In *Proc. of the 44th IEEE Conf. on Decision and Control and the European Control Conference 2005*, pages 2841–2846. IEEE, 2005.

[13] H.K. Khalil. *Nonlinear Systems, 3rd Ed.* Prentice Hall, New Jersey, 2001.

[14] R. S. Millman and G. D. Parker. *Elements of differential geometry*. Prentice-Hall, Englewood Cliffs, NJ, 1977.

[15] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Transactions on Automatic Control*, 49(8):1292–1302, 2004.

[16] P. V. Reddy, E. W. Justh, and P. S. Krishnaprasad. Motion camouflage in three dimensions. In *Proc. of 2006 IEEE Conf. on Decision and Control*, pages 3327–3332. IEEE, 2006.

[17] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proc. of the 5th. Conf. Computer Vision*, pages 902–907, 1995.

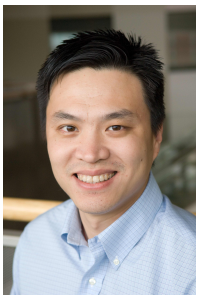
[18] W. Wu and F. Zhang. Curvature based cooperative exploration in three dimenstional scalar fields. In *Proc. of 2010 American Control Conference*, pages 2909–2915, 2010.

[19] F. Zhang. Geometric cooperative control of particle formations. *IEEE Transactions on Automatic Control*, 55(3):800–803, 2010.

[20] F. Zhang, M. Goldgeier, and P. S. Krishnaprasad. Control of small formations using shape coordinates. In *Proc. of 2003 International Conf. of Robotics and Automation*, pages 2510–2515, Taipei, Taiwan, 2003. IEEE.

[21] F. Zhang and N. E. Leonard. Cooperative control and filtering for cooperative exploration. *IEEE Transactions on Automatic Control*, 55(3):650–663, 2010.

Wencen Wu received the B.S. and M.S. degrees from the School of Electronic, Information and Electrical Engineering at Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively, and the M.S. degree from the School of Electrical and Computer Engineering at Georgia Institute of Technology, Atlanta, GA, in 2010, where she is currently pursuing a Ph.D. degree. Her research interests include cooperative exploration and control theory.



Fumin Zhang received the B.S. and M.S. degrees from Tsinghua University, Beijing, China, in 1995 and 1998, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Maryland, College Park, in 2004. He has been an Assistant Professor in the School of ECE, Georgia Institute of Technology since 2007. He was a Lecturer and Postdoctoral Research Associate in the Mechanical and Aerospace Engineering Department, Princeton University from 2004 to 2007. His research interests include marine autonomy, mobile sensor networks, and theoretical foundations for battery supported cyber-physical systems. He received the NSF CAREER Award in 2009, and the ONR YIP Award in 2010.