3-1-2023

Cooperative Filtering and Parameter Identification for Advection-Diffusion Processes Using a Mobile Sensor Network

Jie You
Rensselaer Polytechnic Institute

Ziqiao Zhang
School of Electrical and Computer Engineering

Fumin Zhang
School of Electrical and Computer Engineering

Wencen Wu
San Jose State University, wencen.wu@sjsu.edu

Cooperative Filtering and Parameter Identification for Advection-diffusion Processes Using a Mobile Sensor Network

Jie You, Ziqiao Zhang, Fumin Zhang, and Wencen Wu*

***Abstract*—This paper presents an online parameter identification scheme for advection-diffusion processes using data collected by a mobile sensor network. The advection-diffusion equation is incorporated into the information dynamics associated with the trajectories of the mobile sensors. A constrained cooperative Kalman filter is developed to provide estimates of the field values and gradients along the trajectories of the mobile sensors so that the temporal variations of the field values can be estimated. This leads to a co-design scheme for state estimation and parameter identification for advection-diffusion processes that is different from comparable schemes using sensors installed at fixed spatial locations. Utilizing state estimates from a constrained cooperative Kalman filter, a recursive least square (RLS) algorithm is designed to estimate unknown model parameters of the advection-diffusion processes. Theoretical justifications are provided for the convergence of the proposed cooperative Kalman filter by deriving a set of sufficient conditions regarding the formation shape and the motion of the mobile sensor network. Simulation and experimental results show satisfactory performance and demonstrate the robustness of the algorithm under realistic uncertainties and disturbances.**

I. INTRODUCTION

Many environmental processes are characterized by both spatial and temporal dynamics and often represented mathematically by partial differential equations (PDEs). One of the typical PDEs is the advection-diffusion equation, which has been widely used to model phenomena such as the propagation of chemical in water or air [1]. In many practical problems, parameters in the advection-diffusion equation such as the diffusion coefficient may be unknown or inaccurate. Therefore, to better understand the processes, there is a need to use parameter identification methods to refine, update, or estimate these unknown parameters [2], [3]. The parameter identification problem for PDEs has received significant recent research interest [4]–[6] with emerging applications in environmental monitoring, pollution control, and search/rescue missions [7], [8]. In particular, the dispersion of biological or chemical contaminant obeys the advection-diffusion equations. Knowledge of the diffusion coefficient would help in the estimation and prediction of the degree of contamination [9].

In the case when large numbers of static sensors are deployed in the spatial domain in question, various aspects of parameter identification of PDEs have been investigated in [4], [7], [10]–[16] and references therein. Many of these earlier research follow a general framework of inverse problems in which, given a model, it is necessary to identify the system parameters from available information about the process [16]. Although the solutions to the inverse problems of PDEs are achievable, specific inverse problems must often seek for specific solutions [16]. Among recent works, the nonlinear regression framework is used to estimate PDE parameters from noisy data [17], [18]. This nonlinear regression method requires the estimation of initial conditions of the PDEs. Furthermore, a number of contributions appear in the two-step method aiming at decreasing the computational cost of nonlinear regression [14], [19]. In the first step, all the state variables and derivatives are estimated from the noisy data by using the multivariate polynomials or nonparametric regression methods. In the second step, PDEs parameters are estimated. [14]. Although the second step can be significantly simplified, this two-step method depends heavily on the estimation accuracy of derivatives [20]. As it is shown in [21], parameter cascading method can provide more accurate PDE parameter estimates than the two-step method. Bayesian approaches are also studied for estimating parameters in linear PDEs in the literature [15].

It is rather difficult to envisage static sensor networks continuously monitoring vast spatial regions over long time horizons due to the size of the spatial domain and the high cost of installing large number of static sensors [22] [23]. For parameter identification purpose, a preferable approach would be using mobile sensor networks, which are collections of robotic agents with computational, communication, sensing, and locomotive capabilities [24]–[32]. There exist some contributions on the issue of parameter identification of PDEs using mobile sensor networks [3], [27], [33]–[35]. One of the general approaches of parameter identification is to first decide optimal locations or trajectories of sensors offline, then, formulate a least square problem and search for the parameters that minimize the error between measurements of the true state (with true parameters) and the estimated state [24], [36], [37]. This is usually referred to as performing the twin experiments in data assimilation literature [38] [39]. To find parameters that minimize the least square cost function, PDEs have to be solved using finite element methods over the entire spatial domain, and the optimal solution is obtained through numerical methods for each time step, which generates high computational load. Although these works provide a complete sensor motion along with a parameter update scheme, most of these studies develop offline schemes with few exceptions that investigate online parameter identification [2], [40], [41]. A crucial factor in responding to an emergence chemical or biological disaster is speed. Thus, it is desirable and more practical to achieve online parameter identification

The research work is supported by NSF grants CNS-1446461, CMMI-1663073, and CMMI-1917300. Jie You is with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, USA. Wencen Wu is with the Computer Engineering Department of San Jose State University. Ziqiao Zhang and Fumin Zhang are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30318, USA jyouyj@gmail.com, ziqiao.zhang@gatech.edu, fumin@gatech.edu, wencen.wu@sjsu.edu

while a mobile sensor network is exploring a field instead of performing parameter identification afterwards. For example, in chemical plume tracking, the mobile sensor network has no prior knowledge of the diffusion process, thus, it's preferable that the mobile sensor network can estimate the unknown diffusion coefficient while detecting and tracking the plume to obtain real-time information about the process. Therefore, different from existing works on offline parameter estimation for mobile sensor networks [24], [36], [37], we aim to develop an online parameter identification algorithm that estimates parameters along the trajectories of the mobile sensors, which provides benefits of reduced computational needs.

There are a number of difficulties inherent in the online parameter identification along the sensing trajectories of mobile sensor networks. First, this is an ill-posed inverse problem, which requires the identification of system parameters from the collected finite-dimensional measurements [3]. As such, one needs to assure that the unknown parameters are identifiable taking into account finite-dimensional measurements, which has been discussed in the literature [42]. For online implementation, we would prefer a recursive design so that the estimated parameters can track the measurement data and new measurements can be effectively incorporated. This recursive design becomes more difficult in a mobile sensor network scenario than the static sensor network scenario, due to a limited number of moving agents [3], [43]. With the extremely limited measurement resources in space and time, developing a proper recursive sensing and identification scheme is key to mission success. Unfortunately, the number of publications to above problems is limited so far owing to the inevitable increase in problem complexity.

In this paper, a novel cooperative filtering scheme is developed for online parameter identification of advection-diffusion processes using a mobile sensor network. We incorporate the advection-diffusion equation into the information dynamics and develop a cooperative Kalman filter. Compared to the cooperative Kalman filter in [26], the proposed filter deals with a spatial-temporal varying field instead of a static field. The proposed Kalman filter can achieve online estimation of the temporal variations of the field values along the trajectories of a mobile sensor network. Utilizing the estimates from the filters, we employ the recursive least square (RLS) method to iteratively update the estimate of the unknown parameter in the advection-diffusion equation. Furthermore, we justify a set of sufficient conditions regarding the formation shape and motion of the mobile sensor network that guarantee convergence of the proposed filter. We further provide necessary bias analysis of the proposed method. Simulation and experimental results are given to demonstrate effectiveness of the proposed approach.

In our previous conference paper [2], we designed a cooperative filtering scheme for online parameter estimation of diffusion processes using four sensing agents arranged in a symmetric formation. This work was extended in another conference paper [44] where we initialized the discussion of using the finite volume method to allow four agents in arbitrary formation to perform the parameter identification. We further extended the work in our recent article [45] where experimental results on real mobile robots measuring a CO_2 field have been achieved. In this paper, by using the finite volume method, we extend the cooperative filtering scheme [2], [45] to the case with $N \geq 4$ agents in an arbitrary formation to allow flexibility in practical scenarios. This paper also re-derived the discretized information dynamics and discovered a simpler structure that better motivates the design of a cooperative Kalman filter under state constraints, which is not used by [44], [45]. Furthermore, the new structure is leveraged to perform convergence analysis of the cooperative Kalman filter that has not been addressed in previous work. The experimental data of a real CO_2 field collected in [45] is used in this paper to verify the new algorithms, which demonstrates the applicability of the algorithms in practice.

Other than the work already mentioned, some of our earlier work explored the parameter identification problem for DPS in different theoretical directions than this paper. In [41], we developed a distributed online passive identifier to estimate the unknown parameter iteratively along the trajectory of the mobile sensor network. In [46], we proposed a multi-model structure to represent the advection-diffusion equation, which was parameterized as blended linear PDE models. These earlier works use very different models and filtering techniques, hence are complement to the effort in this paper.

The rest of the paper is organized as follows. Section II introduces the problem formulation. Section III presents the discretization and numerical approximation. Section IV derives the constrained cooperative Kalman filter for state estimation and parameter identification. Section V provides convergence analysis of the filtering scheme and bias analysis. Simulation and experiments results are given in Section VI. Conclusions and further works follow in Section VII. To increase the readability of the paper, some proofs are given in the Appendix.

II. Problem Formulation

Consider the two-dimensional (2D) scalar field $z(r,t)$ where r represents space coordinates and t represents time. Suppose the spatial domain $\Omega \subset \mathbb{R}^2$ is given such that $r \in \Omega$. The spatial gradient of the field is represented by $\nabla z(r,t)$ and the time variation is denoted as $\frac{\partial z(r,t)}{\partial t}$.

A. Environmental Model

For real-life environmental modeling, it is often necessary to accept certain restrictions for $z(r,t)$ to reduce the complexity and computational load. In this paper, we consider the restriction on the the field $z(r,t)$ where it must satisfy the two-dimensional (2D) advection-diffusion process

$$\begin{aligned} \frac{\partial z(r,t)}{\partial t} &= \theta \Delta z(r,t) + v^T \nabla z(r,t), \\ \frac{\partial \nabla z(r,t)}{\partial t} &= 0, \end{aligned} \tag{1}$$

where $\theta > 0$ is a constant unknown diffusion coefficient, Δ represents the Laplacian operator, and v is a known constant vector representing flow velocity. Note that we require that the time derivative of the spatial gradient $\frac{\partial \nabla z(r,t)}{\partial t}$ to vanish. This will simplify the model for estimation.

Assumption II.1 *The field $z(r,t)$ must satisfy the constrained advection-diffusion equation (1).*

Equation (1) can be viewed as a regularization constraint for the agents aiming to learn the underlying field $z(r,t)$. The equality constraint (1) reduced the number of possible fields that can be constructed from limited data.

Remark II.2 *The real-life spatial-temporal processes often contain significant uncertainty and are affected by many unknown factors. It is a common practice to use models based on physical principles, such as the advection-diffusion equation, and impose constraints on the model. Our filtering method can also apply to the case where $\frac{\partial \nabla z(r,t)}{\partial t}$ behaves randomly with a known mean function.*

Remark II.3 *There is no need to specify the boundary conditions for the advection-diffusion equation because our goal is to estimate $z(r,t)$ from sensor measurements. In practical applications, the exploration domain Ω is much larger than the source and sensor dimensions so that the shape of the boundary of the domain Ω does not play a role in the estimation of $z(r,t)$ from sensor measurements.*

B. Model for Mobile Sensing Agents

Consider a formation of N coordinated agents moving in the field, each of which carries a sensor that takes point measurements of the field $z(r,t)$. We consider the agents with single-integrator dynamics given by

$$\dot{r}_i(t) = u_i(t), i = 1,2,...,N, \tag{2}$$

where $r_i(t)$ and $u_i(t) \subseteq \mathbb{R}^2$ are the position and the velocity of the ith agent, respectively.

Let $r_c = [r_{c,x}, r_{c,y}]^T$ be the center of the formation formed by the mobile sensing agents at t, i.e., $r_c = \frac{1}{N}\sum_{i=1}^{N} r_i(t)$. The dynamics of the field value along the trajectory of the formation center r_c according to

$$\dot{z}(r_c,t) = \frac{\partial z(r_c,t)}{\partial r_c}\frac{dr_c}{dt} + \frac{\partial z(r_c,t)}{\partial t} = \nabla z(r_c,t)\cdot \dot{r}_c + \frac{\partial z(r_c,t)}{\partial t}, \tag{3}$$

where $\nabla z(r_c,t)$ is the spatial gradient of $z(r_c,t)$. Furthermore, the gradient $\nabla z(r_c,t)$ also evolves along the trajectory of the formation center, which satisfies

$$\dot{\nabla} z(r_c,t) = H(r_c,t)\cdot \dot{r}_c + \frac{\partial \nabla z(r_c,t)}{\partial t}, \tag{4}$$

where $H(r_c,t)$ is the Hessian matrix.

In most applications, the measurements are taken by the agents discretely over time. Let the moment when new measurements are available be t_k, where k is an integer index. Denote the position of the ith agent at the moment t_k be r_i^k and the field value at r_i^k be $z(r_i^k,k)$.

We have the following assumption for the sensing agents.

Assumption II.4 *We assume the number of agents $N \geq 4$. Each agent can measure its position r_i^k and field value $z(r_i^k,k)$. Each agent shares the measurements with all other agents.*

The measurement of the ith agent can be modeled as

$$p(r_i^k,k) = z(r_i^k,k) + n_i, \tag{5}$$

where n_i is assumed to be i.i.d. Gaussian noise.

C. Formation and Motion Control

Control laws for the velocities of the agents are required so that the mobile sensor network can move along a certain trajectory while maintaining a desired formation. We can view the entire formation as a deformable body with its shape under control. We assume that the control laws used by the agents have been designed to achieve both motion control and formation control.

Motivated by [25], [26], [47], we apply the following consensus tracking algorithm for each agent to achieve formation control: $u_i(t) = \dot{r}_i(t) = \dot{r}_i^{\mathrm{d}} - \varphi_i(r_i - r_i^{\mathrm{d}})$, where $u_i(t)$ is the control input of the ith robot, φ_i is a positive scalar, and r_i^{d} represents the desired position of the ith agent. r_i^{d} is determined by $r_i^{\mathrm{d}} = r_c + \mathbf{R}_i \cdot r_{iF}^{\mathrm{d}}$, where r_{iF}^{d} represents the desired deviation of the ith agents relative to the formation center r_c and $\mathbf{R}_i$ is the rotation matrix from body frame to inertia frame.

If the spatial gradient can be estimated, the motion control for the agents can be designed to achieve interesting behaviors. For example, we may set the velocities of the agents to be aligned with the estimated gradient direction $r_c^{k+1} = r_c^k + \tau_0 \frac{\nabla z(r_c^k,k)}{\|\nabla z(r_c^k,k)\|_2}$, where τ_0 is the speed of the sensing agent and $\nabla z(r_c^k,k)$ is the spatial gradient at the center of the formation. Then the formation can move along the spatial gradient for source seeking. For more details about the motion design, interested readers can refer to our papers [41], [48].

D. Design Goals

Now combining Equations (3), (4), and (1) together, we obtain the following equations, which we call the continuous time *information dynamics*:

$$\dot{z}(r_c,t) = \nabla z(r_c,t)\cdot(\dot{r}_c + v) + \theta\Delta z(r_c,t), \tag{6}$$

$$\dot{\nabla} z(r_c,t) = H(r_c,t)\cdot \dot{r}_c. \tag{7}$$

Our goal is to utilize measurements collected by the mobile sensors to estimate the field $z(r,t)$ and to identify the diffusion parameter θ in the information dynamics. The difficulty associated with this problem is that we relies on a relatively small number of moving sensors. The measurements are finite dimensional time series that need to be processed to estimate the field, which is an infinite dimensional object. This problem is different from the case where large number of static sensors are available to provide coverage of the spatial domain.

We solve this problem within the theoretical framework of state estimation and parameter identification. A two-step scheme, such as those in [14], [19], can be employed to solve our problem. Specifically, we *iteratively* perform the following steps.

1) Under Assumption II.1, we estimate the states $z(r_c,t)$ and $\nabla z(r_c,t)$, as well as the Hessian $H(r_c,t)$ and the Laplacian $\Delta z(r_c,t)$ based on the collected measurements in Equation (5).
2) Utilizing the estimated states, an online parameter identification algorithm will estimate the unknown constant diffusion coefficient θ.

For the first step, we have developed a cooperative filtering algorithm that convert the measurement time series to the estimates of the field modeled by Equation (1). For the second step, we have developed an recursive least square algorithm to estimate the diffusion coefficient.

III. Discretization and Numerical Approximation

We will discretize the information dynamics (6) (7) and the constraint equation (1) properly to facilitate the state estimation and parameter identification problem. The discretization involves both the spatial and the time domain. The errors associated with the descretization can be controlled to be small.

A. The Finite Volume Approximation

Suppose the current time instant is t_k. Let $r_c^k = [r_{c,x}^k, r_{c,y}^k]^T$ be the center of the formation at t_k, i.e., $r_c^k = \frac{1}{N}\sum_{i=1}^N r_i^k$. Most terms in the information dynamics can be approximated using finite difference method. The term that needs special attention is the Laplacian term $\Delta z(r_c^k, k)$. Finite difference method only works for the case when four agents are arranged in a symmetric formation [2]. Thus, we apply the finite volume method [6], [49] to approximate $\Delta z(r_c^k, k)$ with $N \geq 4$ agents in an arbitrary formation. The details of the calculations are shown in Appendix I.

Using the finite volume approximation, and let $\delta t = t_{k+1} - t_k = t_k - t_{k-1}$ be the sampling interval, we discretize the advection-diffusion PDE (1) at the formation center r_c^k as

$$\frac{z(r_c^k, k+1) - z(r_c^k, k)}{\delta t} - v^T \nabla z(r_c^k, k) = \theta \Delta z(r_c^k, k), \tag{8}$$

where $\delta t = t_{k+1} - t_k$ is the sampling interval. Then we can rewrite Equation (54) as follows:

$$\frac{z(r_c^k, k+1) - z(r_c^k, k)}{\delta t} - v^T \nabla z(r_c^k, k) = \Gamma_k \theta - e(r_c^k, k), \tag{9}$$

where Γ_k is defined in Equation (60) and $e(r_c^k, k)$ is defined in Equation (59) in Appendix I.

The sampling time δt must obey the inequalities $\delta t \leq \frac{4\theta}{|v|^2}$ and $\delta t \leq \frac{\Omega_c}{4\theta}$ for the discretization method to converge to the continous dyanmics when the volume of Ω_c is arbitrarily small [6]. Many results have shown that the convergence and accuracy of the above finite volume approximation (9) under mild technical assumptions [6], [49], [50]. Therefore, we make the following idealized assumption:

Assumption III.1 *We assume that the formation is sufficiently small, and the discretization in both space and time is sufficiently accurate so that the approximation error $e(r_c^k, k)$ is arbitrarily small e.g. $e(r_c^k, k) \approx 0$.*

The assumption is made for the convenience of theoretical analysis. Violation of this assumption will not affect the design of the filtering algorithm and the parameter estimation algorithm. The effect of nonzero $e(r_c^k, k)$ is on the accuracy of the filter and estimates. In simulation and experimental studies, we observed some bounded errors, which caused limited performance degradation shown in Fig. 2.

B. Discrete Information Dynamics

We observe from Equation (9) that both $z(r_c^k, k)$ and $z(r_c^k, k+1)$ need to be modeled by discretizing the information dynamics. We first model $z(r_c^k, k)$. The finite difference approximation of each term of (6) at time $t = t_{k-1}$ and at position $r_c = r_c^{k-1}$ give:

$$\begin{aligned} \dot{z}(r_c, t)|_{t=t_{k-1}, r_c = r_c^{k-1}} &\approx \frac{z(r_c^k, k) - z(r_c^{k-1}, k-1)}{\delta t}, \\ \nabla z(r_c, t) \cdot \dot{r}_c|_{t=t_{k-1}, r_c = r_c^{k-1}} &\approx \frac{(r_c^k - r_c^{k-1})^T \nabla z(r_c^{k-1}, k-1)}{\delta t}. \end{aligned} \tag{10}$$

Substituting Equation (10) into Equation (6) gives the information dynamics of $z(r_c^k, k)$ as

$$\begin{aligned} z(r_c^k, k) = &\left(1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c}\right) z(r_c^{k-1}, k-1) \\ &+ \frac{\hat{\theta}_k \delta t}{\Omega_c} \sum_{i=1}^N \alpha_i \cdot z(r_i^{k-1}, k-1) \\ &+ (r_c^k - r_c^{k-1} + v\delta t)^T \nabla z(r_c^{k-1}, k-1) + w_1(r_c^{k-1}, k-1), \end{aligned} \tag{11}$$

where $\hat{\theta}_k$ represents the estimate of θ at time t_k, which will be obtained from parameter identification later. $w_1(r_c^k, k)$ represents the modeling error, which accounts for positioning errors, estimation errors for the Hessian matrix, and other approximation errors including $e(r_c^k, k)$ of (59) caused by higher-order terms omitted from the finite volume scheme.

Similarly, Equation (7) can also be discretized at $t = t_{k-1}, r_c = r_c^{k-1}$ as

$$\begin{aligned} \nabla z(r_c^k, k) = &\nabla z(r_c^{k-1}, k-1) + H(r_c^{k-1}, k-1)(r_c^k - r_c^{k-1}) \\ &+ w_2(r_c^{k-1}, k-1). \end{aligned} \tag{12}$$

We define the information state as $X^{\mathrm{a}}(k+1) = [z(r_c^k, k), \nabla z(r_c^k, k)^T]^T$. We define the noise vector $\mathbf{w}^{\mathrm{a}}(k) = [w_1(r_c^{k-1}, k-1), w_2(r_c^{k-1}, k-1)]^T$. Then define the matrix

$$A_{\hat{\theta}}^{\mathrm{a}}(k) = \begin{bmatrix} 1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c} & (r_c^k - r_c^{k-1} + v\delta t)^T \\ 0 & I_{2\times 2} \end{bmatrix}, \tag{13}$$

and the input vector as

$$U^{\mathrm{a}}(k) = \begin{bmatrix} \frac{\hat{\theta}_k \delta t}{\Omega_c} \sum_{i=1}^N \alpha_i \cdot z(r_i^{k-1}, k-1) \\ H(r_c^{k-1}, k-1)(r_c^k - r_c^{k-1}) \end{bmatrix}. \tag{14}$$

The information dynamics now has the simplified form

$$X^{\mathrm{a}}(k+1) = A_{\hat{\theta}}^{\mathrm{a}}(k) X^{\mathrm{a}}(k) + U^{\mathrm{a}}(k) + \mathbf{w}^{\mathrm{a}}(k). \tag{15}$$

By applying formation control, r_i^{k-1} can be controlled to be close to r_c^{k-1}. Therefore, the field value can be locally approximated by a Taylor series up to second order as

$$\begin{aligned} z(r_i^{k-1}, k-1) \approx &z(r_c^{k-1}, k-1) + (r_i^{k-1} - r_c^{k-1})^T \nabla z(r_c^{k-1}, k-1) \\ &+ \frac{1}{2}(r_i^{k-1} - r_c^{k-1})^T H(r_c^{k-1}, k-1)(r_i^{k-1} - r_c^{k-1}), \end{aligned} \tag{16}$$

Let $Z^{\mathrm{a}}(k) = [z(r_1^{k-1}, k-1) \cdots z(r_N^{k-1}, k-1)]^T$ be the vectors of true field values. Define the matrices $C^{\mathrm{a}}(k)$ and $D^{\mathrm{a}}(k)$ as

$$C^{\mathrm{a}}(k) = \begin{bmatrix} 1 & (r_1^{k-1} - r_c^{k-1})^T \\ \vdots & \vdots \\ 1 & (r_N^{k-1} - r_c^{k-1})^T \end{bmatrix}, \tag{17}$$

and

$$D^{\mathrm{a}}(k) = \begin{bmatrix} \frac{1}{2}((r_1^{k-1} - r_c^{k-1}) \otimes (r_1^{k-1} - r_c^{k-1}))^T \\ \vdots \\ \frac{1}{2}((r_N^{k-1} - r_c^{k-1}) \otimes (r_N^{k-1} - r_c^{k-1}))^T \end{bmatrix}, \tag{18}$$

where $\otimes$ is the Kronecker product. The Taylor expansions (16) for all sensors near r_c^{k-1} can be rewritten in a vector form as

$$Z^{\mathrm{a}}(k) = C^{\mathrm{a}}(k) \cdot X^{\mathrm{a}}(k) + D^{\mathrm{a}}(k) H^{\mathrm{a}}(k), \tag{19}$$

where $H^{\mathrm{a}}(k)$ is a column vector obtained by rearranging elements of the Hessian $H(r_c^{k-1}, k-1)$.

Suppose $\hat{H}^{\mathrm{a}}(k)$ represents the estimate of the vector form Hessian $H^{\mathrm{a}}(k)$ at the center r_c^{k-1}, Equation (5) can be remodeled as

$$P^{\mathrm{a}}(k) = C^{\mathrm{a}}(k) \cdot X^{\mathrm{a}}(k) + D^{\mathrm{a}}(k) \hat{H}^{\mathrm{a}}(k) + D^{\mathrm{a}}(k) \varepsilon^{\mathrm{a}}(k) + \mathbf{n}^{\mathrm{a}}(k), \tag{20}$$

where $P^{\mathrm{a}}(k) = [p(r_1^{k-1}, k-1) \cdots p(r_N^{k-1}, k-1)]^T$ is the measurement vector, $\varepsilon^{\mathrm{a}}(k)$ represents the error in the estimation of the Hessian matrices, and $\mathbf{n}^{\mathrm{a}}(k)$ is the vector of Gaussian measurement noise n_i in Equation (5).

The next goal is to model $z(r_c^k, k+1)$. The information dynamics (6) can also be discretized at time $t = t_k$ and at position $r_c = r_c^{k-1}$ using

$$\begin{aligned} \dot{z}(r_c, t)|_{t=t_k, r_c=r_c^{k-1}} &\approx \frac{z(r_c^k, k+1) - z(r_c^{k-1}, k)}{\delta t}, \\ \nabla z(r_c, t) \cdot \dot{r}_c|_{t=t_k, r_c=r_c^{k-1}} &\approx \frac{(r_c^k - r_c^{k-1})^T \nabla z(r_c^{k-1}, k)}{\delta t}. \end{aligned} \tag{21}$$

Substituting Equation (21) into Equation (6) leads to

$$\begin{aligned} \frac{z(r_c^k, k+1) - z(r_c^{k-1}, k)}{\delta t} &= \frac{(r_c^k - r_c^{k-1})^T \nabla z(r_c^{k-1}, k)}{\delta t} \\ &\quad + v^T \nabla z(r_c^{k-1}, k) + \theta \Delta z(r_c^{k-1}, k). \end{aligned} \tag{22}$$

According to Equation (8) and Equation (9) at time t_k and position r_c^{k-1}, Equation (22) can be rewritten as

$$\begin{aligned} z(r_c^k, k+1) &= \left(1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c}\right) z(r_c^{k-1}, k) + \frac{\hat{\theta}_k \delta t}{\Omega_c} \sum_{i=1}^{N} \alpha_i \cdot z(r_i^k, k) \\ &\quad + (r_c^k - r_c^{k-1} + v \delta t)^T \nabla z(r_c^{k-1}, k) + w_1(r_c^{k-1}, k). \end{aligned} \tag{23}$$

Equation (7) can also be discretized at $t = t_k, r_c = r_c^{k-1}$ to obtain the following equation:

$$\begin{aligned} \nabla z(r_c^k, k+1) =& \nabla z(r_c^{k-1}, k) + H(r_c^{k-1}, k)(r_c^k - r_c^{k-1}) \\ &+ w_2(r_c^{k-1}, k). \end{aligned} \tag{24}$$

We define the information state as $X^{\mathrm{b}}(k+1) = [z(r_c^k, k+1), \nabla z(r_c^k, k+1)^T]^T$, the state noise vector as $\mathbf{w}^{\mathrm{b}}(k) = [w_1(r_c^k, k), w_2((r_c^k, k))]^T$, the state transition matrix as

$$A_{\hat{\theta}}^{\mathrm{b}}(k) = \begin{bmatrix} 1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c} & (r_c^k - r_c^{k-1} + v \delta t)^T \\ 0 & I_{2 \times 2} \end{bmatrix}, \tag{25}$$

and the input vector as

$$U^{\mathrm{b}}(k) = \begin{bmatrix} \frac{\hat{\theta}_k \delta t}{\Omega_c} \sum_{i=1}^N \alpha_i \cdot z(r_i^k, k) \\ H(r_c^{k-1}, k)(r_c^k - r_c^{k-1}) \end{bmatrix}. \tag{26}$$

The information dynamics now has the simplified form

$$X^{\mathrm{b}}(k+1) = A_{\hat{\theta}}^{\mathrm{b}}(k) X^{\mathrm{b}}(k) + U^{\mathrm{b}}(k) + \mathbf{w}^{\mathrm{b}}(k). \tag{27}$$

By applying formation control, r_i^k can be controlled to be close to r_c^{k-1}. Therefore, the field value can be locally approximated by a Taylor series up to second order as

$$z(r_i^k, k) \approx z(r_c^{k-1}, k) + (r_i^k - r_c^{k-1})^T \nabla z(r_c^{k-1}, k) \tag{28}$$

$$+ \frac{1}{2}(r_i^k - r_c^{k-1})^T H(r_c^{k-1}, k)(r_i^k - r_c^{k-1}). \tag{29}$$

Let $Z^{\mathrm{b}}(k) = [z(r_1^k, k) \cdots z(r_N^k, k)]^T$ be the vectors of true field values. Deifne the matrices $C^{\mathrm{b}}(k)$ and $D^{\mathrm{b}}(k)$ as

$$C^{\mathrm{b}}(k) = \begin{bmatrix} 1 & (r_1^k - r_c^{k-1})^T \\ \vdots & \vdots \\ 1 & (r_N^k - r_c^{k-1})^T \end{bmatrix}, \tag{30}$$

and

$$D^{\mathrm{b}}(k) = \begin{bmatrix} \frac{1}{2}((r_1^k - r_c^{k-1}) \otimes (r_1^k - r_c^{k-1}))^T \\ \vdots \\ \frac{1}{2}((r_N^k - r_c^{k-1}) \otimes (r_N^k - r_c^{k-1}))^T \end{bmatrix}, \tag{31}$$

The Taylor expansions (16) for all sensors near r_c^{k-1} can be rewritten in a vector form as

$$Z^{\mathrm{b}}(k) = C^{\mathrm{b}}(k) \cdot X^{\mathrm{b}}(k) + D^{\mathrm{b}}(k) H^{\mathrm{b}}(k), \tag{32}$$

where $H^{\mathrm{b}}(k)$ is a column vector obtained by rearranging elements of the Hessian $H(r_c^{k-1}, k)$.

Suppose $\hat{H}^{\mathrm{b}}(k)$ represents the estimate of the vector form Hessian $H^{\mathrm{b}}(k)$ at the center r_c^{k-1}, Equation (5) can be remodeled as

$$P^{\mathrm{b}}(k) = C^{\mathrm{b}}(k) \cdot X^{\mathrm{b}}(k) + D^{\mathrm{b}}(k) \hat{H}^{\mathrm{b}}(k) + D^{\mathrm{b}}(k) \varepsilon^{\mathrm{b}}(k) + \mathbf{n}^{\mathrm{b}}(k), \tag{33}$$

where $P^{\mathrm{b}}(k) = [p(r_1^k, k) \cdots p(r_N^k, k)]^T$ is the measurement vector, $\varepsilon^{\mathrm{b}}(k)$ represents the error in the estimation of the Hessian matrices, and $\mathbf{n}^{\mathrm{b}}(k)$ is the vector of Gaussian measurement noise n_i in Equation (5).

Remark III.2 *We can observe that the discretized information dynamics and measurement equations actually contain two sets of equations. One set on* $(z(r_c^k, k), \nabla z(r_c^k, k))$ *and another set on* $(z(r_c^k, k+1), \nabla z(r_c^k, k+1))$*. These two sets of equations appear to be uncoupled. Hence it may not be obvious why both sets are needed. We will show next that using both sets of equations will help guarantee the state constraints imposed by the advection-diffusion equation after the discretization. This will also lead to the identification of the parameter* θ_k.

C. Discretized State Constraints

The discretized advection-diffusion equation (9) will be used in two ways in this paper. First, if we assume that an estimation of the parameter θ is available as $\hat{\theta}_k$, then the states $z(r_c^k, k+1)$ and $z(r_c^k, k)$ are the "future and present" field values at a given position $r = r_c^k$. The constraint between the states $z(r_c^k, k+1)$ and $z(r_c^k, k)$ at each step is

$$z(r_c^k, k+1) - \left(1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c}\right) z(r_c^k, k) - v^T \delta t \nabla z(r_c^k, k) = \frac{\hat{\theta}_k \delta t}{\Omega_c} \sum_{i=1}^{N} \alpha_i \cdot z(r_i^k, k). \quad (34)$$

This provides an equality constraint on the states $X^{\mathrm{a}}(k)$ and $X^{\mathrm{b}}(k)$. Define $X(k) = [X^{\mathrm{a}T}(k), X^{\mathrm{b}T}(k)]^T$. The state constraint is an equality constraint induced by the discretized advection-diffusion equation, which can be rewritten in a vector form as:

$$G(k) \cdot X(k) = d(k), \quad (35)$$

where $G(k) = [(-1 + \frac{\alpha_c \hat{\theta}_{k-1} \delta t}{\Omega_c}), -v^T \delta t, 1, 0]$ and $d(k) = \frac{\hat{\theta}_{k-1} \delta t}{\Omega_c} \sum_{i=1}^{N} \alpha_i \cdot z(r_i^{k-1}, k-1)$.

Since the two systems marked by the superscripts a and b are now coupled by the state constraint. We define a set of equations without the superscripts to represent the equations for the overall state dynamics and observation equations. Let $A_{\hat{\theta}}(k) = \mathrm{diag}[A^{\mathrm{a}}_{\hat{\theta}}, A^{\mathrm{b}}_{\hat{\theta}}]$, $C(k) = \mathrm{diag}[C^{\mathrm{a}}, C^{\mathrm{b}}]$, $D(k) = \mathrm{diag}[D^{\mathrm{a}}, D^{\mathrm{b}}]$ be the relevant matrices. Let $\mathbf{w}(k) = [\mathbf{w}^{\mathrm{a}T}(k), \mathbf{w}^{\mathrm{b}T}(k)]^T$, $U(k) = [U^{\mathrm{a}T}(k), U^{\mathrm{b}T}(k)]^T$ $P(k) = [P^{\mathrm{a}T}(k), P^{\mathrm{b}T}(k)]^T$, $H(k) = [H^{\mathrm{a}T}(k), H^{\mathrm{b}T}(k)]^T$, $\varepsilon(k) = [\varepsilon^{\mathrm{a}T}(k), \varepsilon^{\mathrm{b}T}(k)]^T$ and $\mathbf{n}(k) = [\mathbf{n}^{\mathrm{a}T}(k), \mathbf{n}^{\mathrm{b}T}(k)]^T$ be the relevant vectors. Then the overall state and observation equations are:

$$\begin{aligned} X(k+1) &= A_{\hat{\theta}}(k) X(k) + U(k) + \mathbf{w}(k) \\ d(k) &= G(k) \cdot X(k) \\ P(k) &= C(k) \cdot X(k) + D(k)\hat{H}(k) + D(k)\varepsilon(k) + \mathbf{n}(k). \end{aligned} \quad (36)$$

The goal is to estimate the state $X(k)$ and the parameter θ given the measurements $P(k)$ for time up to k.

IV. State Estimation and Parameter Identification

Our solution is based on the derivation of a constrained cooperative Kalman filter. The following assumption is needed to enable the Kalman filter.

Assumption IV.1 *We assume that* $\mathbf{w}(k)$*,* $\varepsilon(k)$*, and* $\mathbf{n}(k)$ *are i.i.d Gaussian noises with zero mean. We assume* $E[\mathbf{w}(k)\mathbf{w}(k)^T] = W$*,* $E[\mathbf{n}(k)\mathbf{n}(k)^T] = R$ *and* $E[\varepsilon(k)\varepsilon(k)^T] = Q$ *are known once the positions of the sensors are known.*

Remark IV.2 *The assumption is made for theoretical convenience to enable convergence analysis of the Kalman filter. The assumption that* $w(k)$ *and* $\varepsilon(k)$ *are i.i.d Gaussian with zero mean may be unrealistic. However, this assumption is needed to justify the application of filtering techniques for state estimation. Once enough data is gathered, the estimates of* W*,* Q*, and* R *can be performed through offline system identification techniques. Therefore, the assumption about* W*,* Q*, and* R *is reasonable in many applications, for example, oceanography and meteorology. In these applications, the statistical properties of ocean fields and atmospheric fields are usually known from accumulated observational data over a long period of time. [26], [43]. Since the error covariance of* $w(k)$ *and* $\varepsilon(k)$ *are not theoretically characterized and depend on heuristics and simulations [26], we will validate the assumption through simulation.*

A. Constrained Kalman Filter

We observe that the discretized information dynamics is constrained. Extension to the Kalman filter was made by [51] to incorporate equality constraints, leading to the cooperative Kalman filter design as follows:

(1) the one-step prediction,

$$\hat{X}^-(k) = A_{\hat{\theta}}(k-1)\tilde{X}^+(k-1) + U(k-1); \quad (37)$$

where $\tilde{X}^+(k-1)$ is the current constrained state estimate and $\hat{X}^-(k)$ is a prior unconstrained state estimate.

(2) error covariance for the one-step prediction,

$$R_c^-(k) = A_{\hat{\theta}}(k-1) R_c^+(k-1) A^T_{\hat{\theta}}(k-1) + Q; \quad (38)$$

(3) optimal gain,

$$K(k) = R_c^-(k) C^T(k) [C(k) R_c^-(k) C^T(k) + D(k) Q D^T(k) + R]^{-1}; \quad (39)$$

(4) updated unconstrained estimate,

$$\hat{X}^+(k) = \hat{X}^-(k) + K(k)(P(k) - C(k)\hat{X}^-(k) - D(k)\hat{H}(k)); \quad (40)$$

(5) error covariance for the updated estimate,

$$R_c^+(k)^{-1} = R_c^-(k)^{-1} + C^T(k)[D(k) Q D^T(k) + R]^{-1} C(k); \quad (41)$$

(6) updated constrained estimate,

$$\tilde{X}^+(k) = \hat{X}^+(k) - G(k)^T [G(k) G(k)^T]^{-1} [G(k) \cdot \hat{X}^+(k) - d(k)]. \quad (42)$$

It should be noted that we derive the constrained Kalman filter (42) by directly projecting the unconstrained state estimate $\hat{X}^+(k)$ onto the constraint surface [51]. This requires the term $G(k)G(k)^T$ to be invertible. In our case, $G(k)G(k)^T = (1 - \frac{\alpha_c \hat{\theta}_k \delta t}{\Omega_c})^2 + 1 + v^T v \cdot \delta t^2$, which is nonsingular at each time step.

B. Parameter Identification

The Hessian $\hat{H}(k)$ in the term $U(k)$ can be viewed as a parameter that needs to be identified to enable the cooperative Kalman filter. By time step $k-1$, we have obtained an estimate of $\tilde{X}^+(k-1)$ from the cooperative Kalman filter. Using the computed estimates $\tilde{X}^+(k-1)$ and $U(k-1)$, before the arrival of measurements at time step k, we can obtain a prediction for $X(k)$ as $\hat{X}^-(k) = A_{\hat{\theta}}(k-1)\tilde{X}^+(k-1) + U(k-1)$. If we assume the number of sensors $N \geq 4$ and the formation is not co-linear, we have $P(k) = C(k) \cdot \hat{X}^-(k) + D(k)\hat{H}(k)$. The Hessian estimate can be solved by using the least mean square method,

$$\hat{H}(k) = \left(D(k)^T D(k)\right)^{-1} D(k)^T \left(P(k) - C(k)\hat{X}^-(k)\right). \quad (43)$$

Remark IV.3 *To enable the Hessian estimation in* (43)*, the matrix* $D(k)^T D(k)$ *must be nonsingular. Thus, the minimum number of agents that enables the Hessian estimation is four in 2D. In real applications, it is better to select some redundant agents and a nonsymmetric formation to guarantee the nonsingular property.*

Remark IV.4 *Since the sensor measurements* $p(r_i^k,k)$ *and* $p(r_i^{k-1},k-1)$ *are available in the measurement vector* $P(k)$*, one straightforward and simple way is to replace* $z(r_i^k,k)$ *and* $z(r_i^{k-1},k-1)$ *with the sensor measurements* $p(r_i^k,k)$ *and* $p(r_i^{k-1},k-1)$*, which is adopted in this paper. The other way is to design a separate one-step filter to reduce the noise of measurements [26]. Running the one-step filter makes our closed loop process more complex and increases the computation cost, which is omitted here. Interested readers can refer to [26] for more details.*

Once the state of the cooperative Kalman filter is estimated sequentially over time, we use the recursive least square method to iteratively update the estimate of θ based on the discretized model (9). An estimate of the information state $X(k+1)$ is available as $\hat{X}(k+1) = [\hat{z}(r_c^k,k), \nabla\hat{z}(r_c^k,k), \hat{z}(r_c^k,k+1), \nabla\hat{z}(r_c^k,k+1)]^T$, then by combining the terms on the left hand side of Equation (9), we define the term $\hat{Y}(r_c^k,k)$ as

$$\hat{Y}(r_c^k,k) = \frac{\hat{z}(r_c^k,k+1) - \hat{z}(r_c^k,k)}{\delta t} - v^T \nabla\hat{z}(r_c^k,k). \tag{44}$$

Next, the field value $z(r_i^k,k)$ is replaced by the measurement $p(r_i^k,k)$. Define the estimate of $\hat{\Gamma}_k$ as the estimate for Γ_k as follows:

$$\hat{\Gamma}_k = \frac{1}{\Omega_c}[\sum_{i=1}^{N} \alpha_i \cdot p(r_i^k,k) - \alpha_c \cdot \hat{z}(r_c^k,k)].$$

This leads to

$$\hat{Y}(r_c^k,k) = \hat{\Gamma}_k \theta + \eta(k), \tag{45}$$

where η_k represents the approximation error.

Assumption IV.5 *We assume* $\eta(k)$ *is a noise term with zero mean and bounded covariance matrix* R_η.

Note that the term η_k contains approximation errors from several sources of approximations. Hence it might not be Gaussian noise, and it may also be correlated in time k. Nevertheless, based on the cooperative filtering scheme, the diffusion coefficient can be directly estimated without the need of numerically solving the diffusion equation. Given an initial estimate for the diffusion coefficient, a simple application of the RLS method can iteratively update the estimate of θ. Following the canonical procedure of RLS estimation outlined in [52], we derive the following equations to update the estimate θ.

$$\hat{\theta}_k = \hat{\theta}_{k-1} + K_\theta(k)\left(\hat{Y}(r_c^k,k) - \hat{\Gamma}_k\hat{\theta}_{k-1}\right); \tag{46}$$

$$K_\theta(k) = \Lambda(k-1)\hat{\Gamma}_k^T\left(\hat{\Gamma}_k\Lambda(k-1)\hat{\Gamma}_k^T + R_\eta\right)^{-1}; \tag{47}$$

$$\Lambda(k) = \left(I - K_\theta(k)\hat{\Gamma}_k\right)\Lambda(k-1), \tag{48}$$

where $\hat{\theta}_k$ is the estimate of θ, $K_\theta(k)$ is the estimator gain matrix, $\Lambda(k)$ is the estimation error covariance matrix.

The proposed recursive cooperative filtering scheme is based on two subsystems: the cooperative Kalman filtering subsystem (Equations (37)-(42)) and the RLS subsystem in (46). In the cooperative Kalman filtering subsystem, assuming that the parameter $\hat{\theta}_k$ is constant and known, we run the cooperative Kalman filter to estimate the states based on the collected measurements. In the RLS subsystem, assuming that the estimated states can track the true values, we employ the RLS method to iteratively update the estimate of θ.

V. Convergence and Bias Analysis

In this section, we prove the convergence of the cooperative Kalman filter. Theorem 7.4 in [53] states that if the time-varying system dynamics are uniformly completely controllable and uniformly completely observable, the Kalman filter for this system converges. With this result, we will establish a set of sufficient conditions for the mobile sensors such that the uniformly complete controllability and observability of the unconstrained Kalman filter can be guaranteed.

A. Convergence of the Cooperative Kalman Filter

From Remark III.2, the two subsystems marked by superscripts a and b are uncoupled if the state constraint is not considered. In the proof of uniformly complete controllability and observability of the unconstrained Kalman filer, the system will be decoupled into two subsystems marked by superscripts a and b. We will first analyze the convergence of the Kalman filters for the two subsystems separately, and then analyze the convergence of the Kalman filter for the whole system.

Let $\Phi(k,j)$ be the state transition matrix from time t_j to t_k, where $k > j$. Then, $\Phi(k,j) = A_{\hat{\theta}}(k-1)A_{\hat{\theta}}(k-2)\cdots A_{\hat{\theta}}(j) = \Phi^{-1}(j,k)$ and we define $\Phi(k,j) = \mathrm{diag}[\Phi^{\mathrm{a}}, \Phi^{\mathrm{b}}]$, where $\Phi^{\mathrm{a}}(k,j) = A^{\mathrm{a}}_{\hat{\theta}}(k-1)A^{\mathrm{a}}_{\hat{\theta}}(k-2)\cdots A^{\mathrm{a}}_{\hat{\theta}}(j)$ and $\Phi^{\mathrm{b}}(k,j) = A^{\mathrm{b}}_{\hat{\theta}}(k-1)A^{\mathrm{b}}_{\hat{\theta}}(k-2)\cdots A^{\mathrm{b}}_{\hat{\theta}}(j)$. Since $A^{\mathrm{a}}_{\hat{\theta}}(j) = A^{\mathrm{b}}_{\hat{\theta}}(j)$ for any j, we can have $\Phi^{\mathrm{a}}(k,j) = \Phi^{\mathrm{b}}(k,j)$ for any $j < k$ and the following lemma.

Lemma V.1 *For* $\Phi(k,j)$ *as defined above and* $C(k)$ *as defined in* (20)*, we can have*

$$\begin{aligned} \Phi^{\mathrm{a}}(k,j) = \Phi^{\mathrm{b}}(k,j) &= \begin{bmatrix} \xi_{\hat{\theta}} & \phi^T \\ 0 & I_{2\times 2} \end{bmatrix}, \\ \Phi^{\mathrm{a}}(j,k) = \Phi^{\mathrm{b}}(j,k) &= \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & -\frac{\phi^T}{\xi_{\hat{\theta}}} \\ 0 & I_{2\times 2} \end{bmatrix}, \end{aligned} \tag{49}$$

and

$$\begin{aligned} C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k) &= \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix}, \\ C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k) &= \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix} \end{aligned} \tag{50}$$

where $\xi_{\hat{\theta}} = (1 - \frac{\alpha_c \hat{\theta}_{k-1} \delta t}{\Omega_c})(1 - \frac{\alpha_c \hat{\theta}_{k-2} \delta t}{\Omega_c}) \cdots (1 - \frac{\alpha_c \hat{\theta}_{j} \delta t}{\Omega_c})$,

$$\phi^T = (r_c^{k-1} - r_c^{k-2} + v\delta t)^T + \sum_{n=1}^{k-j-1} \left(\prod_{m=1}^{n} (1 - \frac{\alpha_c \hat{\theta}_{k-m} \delta t}{\Omega_c}) \right) (r_c^{k-n-1} - r_c^{k-n-2} + v\delta t)^T,$$

and $C(j)\Phi(j,k) = \mathrm{diag}[C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k), C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)]$

Let's first restate the definitions of uniformly completely controllability and uniformly complete observability, respectively (modified from Definitions in [53]).

Definition V.2 *The proposed cooperative filter is uniformly completely controllable if there exist $\tau_1 > 0$, $\lambda_1 > 0$, and $\lambda_2 > 0$ such that the controllability Grammian $\mathfrak{C}(k, k-\tau_1) = \sum_{j=k-\tau_1}^{k} \Phi(k,j) W \Phi(k,j)^T$ satisfies $\lambda_1 I_{6\times 6} \le \mathfrak{C}(k, k-\tau_1) \le \lambda_2 I_{6\times 6}$ for all $k > \tau_1$. Here W is the covariance for the state error $w(k)$.*

In the following procedures, we provide a set of sufficient conditions such that the uniformly complete controllability and observability of the proposed filter can be satisfied by showing the upper and lower bounds of the controllability and observability Grammian. In the procedure, there exist some positive real numbers $\lambda_1, \lambda_2, \cdots, \lambda_{23}$. All of these real numbers are time-independent bounds for various quantities, the values of which do not affect the correctness of our discussions. Note that, in this paper, a relation between two symmetric matrices $A_1 \le A_2$ means that for any vector s with compatible dimension, there exists $s^T A_1 s \le s^T A_2 s$. We have the following proposition for uniformly complete controllability.

Proposition V.3 *The proposed cooperative filter is uniformly completely controllable if the following conditions are satisfied:*

(Cd1) The covariance matrix W is bounded, i.e., $\lambda_3 I \le W \le \lambda_4 I$ for some constants $\lambda_3, \lambda_4 > 0$.

(Cd2) The speed of each agent is uniformly bounded, i.e., $\|r_i^j - r_i^{j-1}\| \le \lambda_5$ for all time j, for $i = 1, \cdots, N$, and for some constant $\lambda_5 > 0$.

(Cd3) The estimated parameter $\hat{\theta}_j$ is bounded, i.e., $0 \le \hat{\theta}_j < \lambda_6$. By properly selecting the sampling interval δt and formation size Ω_c, we can make $\hat{\theta}_j$ satisfy that $0 < 1 - \frac{\alpha_c \hat{\theta}_j \delta t}{\Omega_c} \le 1$ for all time j, which means $\lambda_6 = \frac{\Omega_c}{\alpha_c \delta t}$.

Proof See the Appendix.

Definition V.4 *The proposed cooperative filter is uniformly completely observable if there exist $\tau_2 > 0$, $\lambda_9 > 0$, and $\lambda_{10} > 0$ such that the observability Grammian $\mathfrak{O}(k, k-\tau_2) = \sum_{j=k-\tau_2}^{k} \Phi^T(j,k) C^T(j) [D(j) Q D^T(j) + R]^{-1} C(j) \Phi(j,k)$ satisfies $\lambda_9 I_{6\times 6} \le \mathfrak{O}(k, k-\tau_2) \le \lambda_{10} I_{6\times 6}$ for all $k > \tau_2$. Here Q and R are the covariance matrices for Hessian estimation error $\varepsilon(k)$ and the measurement noise $n(k)$, respectively.*

To prove the uniformly complete observability, we also require one elementary lemma [26]. The proof just uses basic linear algebra knowledge, and thus omitted here.

Lemma V.5 *Suppose two 2×1 vectors $a = [a_1\ a_2]^T$ and $b = [b_1\ b_2]^T$ form an angle Ψ such that $0 < \Psi < \pi$. Then the minimum eigenvalue λ_{min} of the 2×2 matrix $M = a \cdot a^T + b \cdot b^T$ is strictly positive, i.e. $\lambda_{min} > 0$*

For uniformly complete observability, the following sufficient conditions are established for a moving formation.

Proposition V.6 *The proposed Kalman filter is uniformly completely observable if (Cd2), (Cd3) and the following conditions are satisfied:*

(Cd4) The number of agents N is greater than or equal to 3.

(Cd5) The covariance matrices R and Q are bounded, i.e., $\lambda_{11} I \le R \le \lambda_{12} I$ and $0 \le Q \le \lambda_{13} I$ for some constants $\lambda_{11}, \lambda_{12}, \lambda_{13} > 0$.

(Cd6) The distance between each agent and the formation center is uniformly bounded from both above and below, i.e., $\lambda_{14} \le \|r_i^{j-1} - r_c^{j-1}\| \le \lambda_{15}$ for all j, for $i = 1, 2, ..., N$, and for some constants $\lambda_{14}, \lambda_{15} > 0$.

(Cd7) There exists a constant time difference τ_2, and for all $k > \tau_2$, there exists a time instance $j_1 \in [k-\tau_2, k]$, as well as two agents indexed by i_1 and i_2, such that $r_{i_1}^{j_1-1}, r_{i_2}^{j_1-1}, r_c^{j_1-1}$ are not colinear; and for all $k > \tau_2$, there exists a time instance $j_2 \in [k-\tau_2, k]$, such that $r_1^{j_2}, \cdots, r_N^{j_2}$ are not colinear.

Proof See the Appendix.

By applying Theorem 7.4 in [53], we guarantee the convergence of the unconstrained filter from the uniformly complete controllability and observability properties.

Now consider the state equality constraint (35), the convergence analysis for a Kalman filter under equality constraints on the states has been performed in [51]. Our problem can be addressed similarly. Equation (42) indicates that the constrained estimate $\tilde{X}^+$ can be viewed as the projection of the unconstrained state estimate $\hat{X}^+$ on to the constrained state space defined by $G(k)$ and $d(k)$. Given that $G(k)$ is full rank, according to Theorem 4 in [51], if X is the true value of the state then the following holds

$$\|X - \tilde{X}^+\| \le \|X - \hat{X}^+\|, \tag{51}$$

where $\|\cdot\|$ is the l_2 norm. This shows that the estimation error of the constrained Kalman filter is bounded by the estimate error of the unconstrained Kalman filter. Because the unconstrained Kalman filter is convergent, the convergence of the constrained Kalman filter can also be guaranteed.

Remark V.7 *Compared to the cooperative Kalman filter in [26], the proposed filter deals with a spatial-temporal varying field instead of a static field. Thus, the performance of the filter depends on the parameters of the spatial-temporal varying field. It should be noted that (Cd3) is essential, which indicates that if the estimated parameter $\hat{\theta}$ is bounded by $0 \le \hat{\theta} < \frac{\Omega_c}{\alpha_c \delta t}$, the convergence of the cooperative Kalman filter can be guaranteed. That means the cooperative Kalman filter can successfully track the states even though the estimated parameter $\hat{\theta}$ is biased or slightly different from the true parameter.*

B. Parameter Identification

First, We can show that the Hessian estimate is unbiased.

Proposition V.8 *The estimate of the Hessian term $\hat{H}(k)$ given in* (43) *is unbiased with error covariance matrix* $\left(D(k)^T D(k)\right)^{-1} D(k)^T [R - C(k)R_c^-(k)C(k)]D(k)[\left(D(k)^T D(k)\right)^{-1}]^T$.

Proof
From the analysis of the property of Kalman filter, we have $P(k) = Z(k) + n(k)$ and $\hat{X}^-(k) = X(k) + \psi_1(k)$, where $E[n(k)n(k)^T] = R$ and $E[\psi_1(k)\psi_1(k)^T] = R_c^-(k)$. Then we have

$$\begin{aligned}\hat{H}(k) &= \left(D^T D\right)^{-1} D^T \left(P - C\hat{X}^-\right)(k)\\ &= \left(D^T D\right)^{-1} D^T \left(Z - CX + n - C\psi_1\right)(k) \qquad (52)\\ &= H(k) + \left(D(k)^T D(k)\right)^{-1} D(k)^T \left(n(k) - C(k)\psi_1(k)\right).\end{aligned}$$

Since $n(k)$ and $\psi_1(k)$ have zero mean, the expectation value $E[\hat{H}(k)]$ equals $E[H(k)]$. The error covariance can be directly calculated to be

$$\left(D^T D\right)^{-1} D^T [R - CR_c^- C]D[\left(D^T D\right)^{-1}]^T(k).$$

■

Unfortunately, the recursive least square method in Equation (46) may not produce an unbiased estimate for θ. It is well known that least square methods generate biased parameter estimates when time correlated noise terms are presented (see Section 7.3 of [52]). In our case, even though the online estimate of θ may be biased, performance of the parameter identification method can be further refined offline by using the methods such as the bootstrap method [54], [55]. We will employ this method in the simulation results to improve the accuracy of the estimation of θ.

It should be noted that the convergence of the Kalman filter is not sensitive to the biases in parameter $\hat{\theta}_k$. In other words, the estimated states from the cooperative Kalman filtering can successfully track the true values even though the estimated parameter $\hat{\theta}_k$ is biased or slightly different from the true parameter.

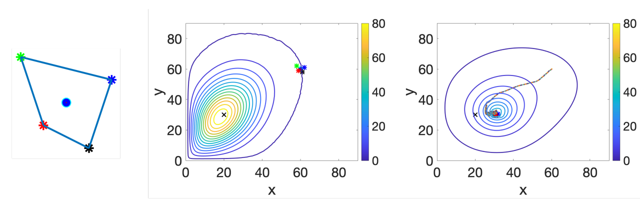
Remark V.9 *One may conjecture that the extended Kalman filter (EKF) approach [7], [52] can be used to treat unknown parameters $\hat{\theta}_k$ as an additional state variable and define an augmented system. However, this augmented system is nonlinear, since parameter $\hat{\theta}_k$ is multiplied by state variable $X(k)$ in* (36). *In this case, the corresponding state dynamics are nonlinear and time-varying, which makes it difficult to analyze the convergence of the resulting constrained cooperative Kalman filter.*

VI. Simulation and Experimental Results

A. Measuring a Simulated field

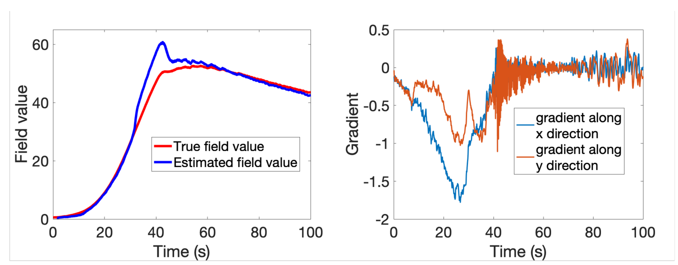
To demonstrate the performance of the proposed approach for online parameter estimation, we consider the 2D advection-diffusion equation (1) with the nominal value of $\theta = 0.066$ and the flow velocity $v = (0.05, 0)$ for a simulated domain. The initial condition is illustrated in Fig.1 (b), in which the maximum value is at point $(20, 30)$. The whole domain of PDE is a rectangular area $0 \le x \le 90$, $0 \le y \le 90$ with spatial discretization of 1. We implement an alternating direction implicit (ADI) finite volume scheme in MATLAB, with 90-by-90 spatial grid. To simulate the modelling error of the discrete representation, a Gaussian noise with the magnitude of about 1% of the noise-free field value is added to the field values. We also add 5% (in variance) Gaussian noise to measurements taken by sensors. A computational time step of $0.1s$ is chosen for the simulation, which satisfies the stability requirement of finite volume method.

In the simulation, we select the asymmetric initial locations of four sensing agents represented by the red, blue, green, and black stars as shown in Fig. 1 (a). In Fig. 1 (b) and (c), the contours represent the level curves of the field values and the blue doted line is the trajectory of agents. At each time step, the agents take measurements of the field, run the proposed cooperative Kalman filter as well as the RLS algorithm to obtain the estimates of the diffusion coefficient θ, and move along the gradient direction estimated by the cooperative Kalman filter while converging to a desired formation. The performance of the state estimation and the gradient estimation is illustrated in Figs. 2 (a) and (b).



(a) formation at t=0s (b) t=0s (c) t=100s

Fig. 1. Gradient climbing trajectory of the mobile sensor network and evolution of the field values. (a) Asymmetric formation of the mobile sensor network at beginning time. (b) Field values at beginning time. (c) Field values at final time and the trajectory of agents.



(a) (b)

Fig. 2. State estimation at the formation center along trajectory. (a) Field value estimation. (b) Gradient estimation.

Initially, we set the estimate $\hat{\theta}_0 = 2$. The online estimate of the parameter is compared to the nominal diffusion coefficient in Fig. 3, to show its accuracy. One can see that $\hat{\theta}$ converges to the nominal value with 1.95% error, confirming the effectiveness of the proposed algorithm. Even though the online RLS only can provide a generally biased result, we show that this slight bias in Fig. 3 can be further tuned offline using the bootstrap method.

The Bootstrap method is a Monte Carlo simulation based statistical technique. The basic idea of bootstrap methods for refining the bias estimation is resampling the original training samples of size n to produce M bootstrap training sets of size n, each of which is used to train a bootstrap estimate. To obtain

TABLE I
COMPARISON OF RLS AND BOOTSTRAP METHODS

Nominal value	0.3	0.4	0.5	0.6	0.7
Online RLS	0.2626	0.3755	0.4534	0.5696	0.6451
Bootstrap	0.2686	0.3873	0.4704	0.5965	0.6849
Online RLS	Standard Deviation is 15×10^{-4}				
Bootstrap	Standard Deviation is 4×10^{-4}				

an efficient bootstrap estimate, the number of resampling times M is ordinarily chosen in the range $25-200$. In this work, we set $M = 50$, which is often enough to give a good estimate [55]. Readers can refer to [55] for more details of the bootstrap methods.

To achieve a fair validation, we randomly choose 100 nominal values in the range $[0.3\ 0.7]$ and compare the performance of RLS and bootstrap methods. We tabulate the standard deviation of the errors and specify some results in Table I. One can see that with the bootstrap method [54], [55], the bias of the parameter has now been significantly reduced.

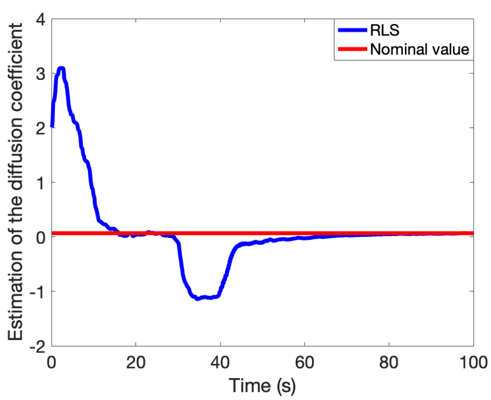
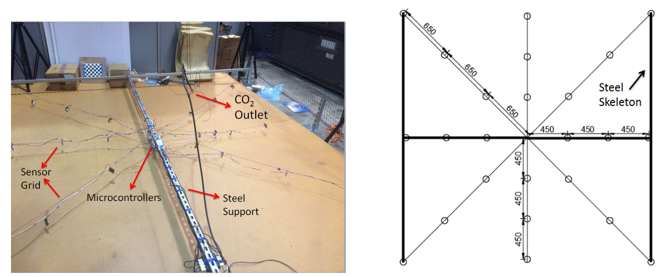


Fig. 3. Estimation of diffusion coefficient θ with the initial value 2. The blue solid line represents the estimated coefficient $\hat{\theta}$ and the red solid line is the nominal value, which is set to 0.06.

We also look into the case with bad initial guess where we set the initial estimate $\hat{\theta}_0 = 2$, $\hat{z}_0 = 85.9156$ and $\nabla\hat{z}_0 = [45.6427, 53.7009]^\intercal$ with true initial state value $z_0 = 0.5510$ and $\nabla z_0 = [-0.0365, -0.0082]^\intercal$. The estimated field value soon converges to the true field value. Here the nominal value of the parameter is set as $\theta = 5.5834$, and the estimated parameter converges to $\hat{\theta} = 5.7382$ with bias$=-0.1548$. This shows that even with bad initial guess, the state estimation and parameter estimation still converge to the true value using our proposed method.

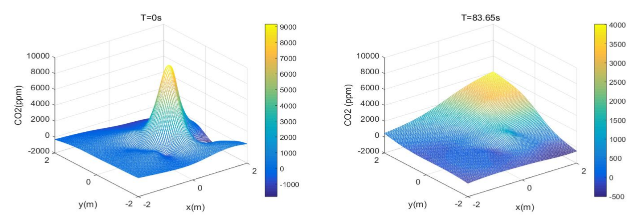
B. Experimental Data and Simulated Agent Motion

A controllable CO_2 diffusion field in a lab setup was introduced in [45]. As illustrated in Fig. 4, the CO_2 field is distributed in an area with $3.5 \times 3.5\ m^2$. A sensor grid which consists of 24 CO_2 sensors is assembled to measure the concentration of the gas over the area. The sensors are calibrated so that they all have consistent measurement values when we reproduce the experiment in the same environment. During the diffusion process of CO_2 gas, the sensor grid measures the gas concentration at the fixed locations and sends the data to MATLAB running in the central computer. MATLAB then reproduces the diffusion process by interpolating the field values collected by the sensor grid at every discrete time instant. The diffusion process obtained from the real field on November 7, 2016 is shown in Fig. 5. CO_2 begins diffusing at step t = 0s and ends at t = 120s. The computational time step is 1s. Given the measurements collected from the sensor gird, the nominal value of diffusion coefficient θ_n can be determined as $\theta_n = 0.239$. For more detatils about the experimental data collection, please refer to recent work [45].



(a) (b)

Fig. 4. The illustration of the experimental setup and the sensor grid [45].



(a) (b)

Fig. 5. Snapshots of the diffusion field collected by the sensor grid and visualized by MATLAB [45].

We verify the experimental performance for diffusion coefficient identification with four simulated sensing agents deployed in the reconstructed field based on the experimental data. To achieve a fair experimental validation, the experiment is performed using two different starting points for the agents marked by “A” and “B” in Fig. 6. Note that the field data is collected while there is no air movement. Hence our filtering equations do not contain the advection terms.

We control the sensing agents to move along the estimated gradient direction while keeping a constant formation. In Fig. 6, the contours represent the level curves of the diffusion field, the colored dots represent the four sensing agents, the black star represents the source, the orange line and purple line represent the trajectories of the center of the mobile sensor network staring from A and B, respectively. The experiment begins at step $t = 0s$ and ends at $t = 120s$. The measuring frequency of the sensors is set to 0.5Hz. As we can observe from the figure, the agents trace the gradient of the diffusion field in both experiments to find the diffusion source of the CO_2 gas, which is the point with the highest CO_2 concentration.

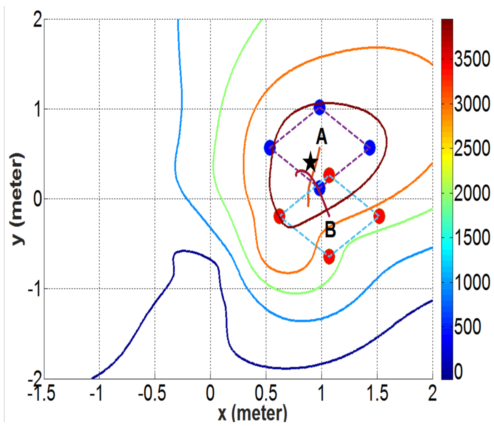


Fig. 6. The trajectories of the agents in the two experiments. The black star marks the source of the field.

While the mobile sensor network is moving towards the source, it also achieves real-time identification of the diffusion coefficient by implementing the cooperative Kalman filter and the RLS algorithm. Initially, we set the estimate $\hat{\theta}_0 = 1$. The estimation results of the diffusion coefficient are shown in Fig. 7. As we can observe from Fig. 7 that, the estimates of the parameter converge to stabilized values in both experiments, both of which are very close to the estimated nominal value. The two values differ by a small amount of 0.0212. Nevertheless, it demonstrates that the proposed algorithm is robust under realistic uncertainties and disturbances.

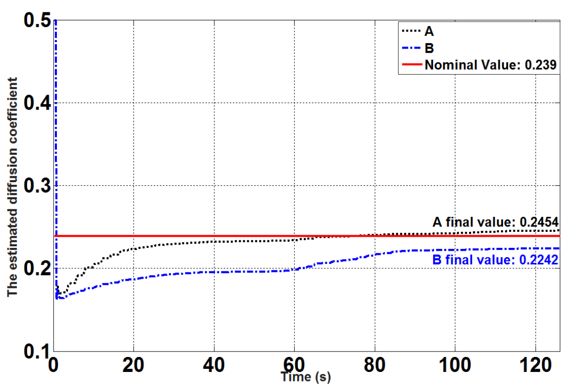


Fig. 7. The estimated diffusion coefficients staring from A and B point are shown in the dotted black line and dashed blue line, respectively. The nominal value of diffusion coefficient (the red line) is 0.239, which is estimated from measurements collected by the static sensor grid.

VII. Conclusion

This work presented a cooperative filtering and parameter estimation algorithm for advection-diffusion processes measured by a mobile sensor network. We provide an approach to discretize the advection-diffusion equation in both space and time domain leading to the information dynamics equations. Based on the information dynamics, a constrained Kalman filter is proposed for state estimation and a RLS estimation is proposed for parameter estimation. Theoretical justifications are provided for the convergence analysis of the cooperative filter. Simulation and experimental results are provided to demonstrate the efficiency of the proposed method. Future work includes extending the proposed algorithm to PDE models with spatially varying parameters.

Appendix I

Finite volume approximation of $\Delta^2 z$

We first construct a volume Ω_c around the formation center r_c^k. The constructions can be performed in different ways, such as the cell-centered scheme and the vertex-centered scheme [49], [50]. In this work, the volume Ω_c is constructed as a closed polygon that is formed by the perpendicular bisectors of the line segments $r_1^k r_c^k, r_2^k r_c^k, \cdots, r_N^k r_c^k$. We choose the midpoints of the line segments $r_1^k r_c^k, r_2^k r_c^k, \cdots, r_N^k r_c^k$. For segment $r_i^k r_c^k$, a perpendicular bisector is a line that passes through the midpoint on $r_i^k r_c^k$. These perpendicular bisectors will intersect each other and form a closed polygon with a simple loop boundary. The area enclosed by this polygon is the finite volume Ω_c. To illustrate the idea, we plot the case when $N = 4$, in Fig. 8. Ω_c is the volume that is enclosed by the polygon $A_1A_2A_3A_4$. The points M_1, M_2, M_3, M_4 are the midpoints of the line segments $r_1^k r_c^k, r_2^k r_c^k, r_3^k r_c^k, r_4^k r_c^k$, respectively.

Let $\mathbf{S}_i = A_iA_{i+1}$ where $A_{N+1} = A_1$, and let $\hat{n}_i$ be the outward unit normal vector on the boundary segment $\mathbf{S}_i$. Let the boundary of Ω_c be $\mathbf{S}$ that now contains the segments $\mathbf{S}_i$ for $i = 1, 2, .., N$. We can see that $\hat{n}_i$ is constant and aligned with $r_i^k r_c^k$ for each segment $\mathbf{S}_i$ for $i = 1, 2, ..., N$. Our construction using the perpendicular bisectors guarantees that $\hat{n}_i$ is perpendicular to the boundary on each segment $\mathbf{S}_i$.

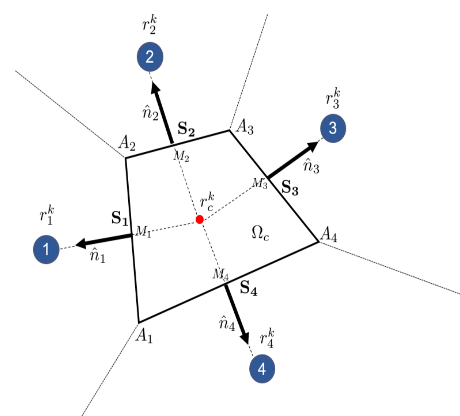


Fig. 8. Finite-volume construction for a mobile sensor network in 2D.

By applying the Green's theorem to the integration of equation $\Delta z(r,t)$ over the finite volume Ω_c, we can have the following expression [56]:

$$\int \int_{\Omega_c} \Delta z(r,t)\ d\Omega_c = \oint_{\mathbf{S}} (\nabla z(r,t))^T \hat{n} dr \tag{53}$$

The integration of (53) over a finite volume Ω_c shown in Fig. 8 results in a spatially discretized equation that holds when the volume of Ω_c is small:

$$\Delta z(r_c^k, k) = \frac{1}{\Omega_c} \left(\sum_{i=1}^{N} \int_{\mathbf{S}_i} (\nabla z(r,t))^T \hat{n}_i\ dr \right). \tag{54}$$

Next, we will derive $\nabla z(r,t)$, $r \in \mathbf{S}_i$ in equation (53) at time step k. For any given $i = 1, 2, ..., N$, with r_i^k being close to r_c^k, $z(r_i^k, k)$ can be locally approximated as follow,

$$\begin{aligned} z(r_i^k,k) - z(r_c^k,k) \approx& (\nabla z(r,k))^T (r_i^k - r_c^k) \\ &+ \int_0^1 \left(H_{r_i^k}(\xi) - H_{r_c^k}(\xi) \right) \xi d\xi, \quad r \in \mathbf{S}_i, \end{aligned} \tag{55}$$

where $H_{r_i^k}(\xi) = (r_i^k - r)^T H\left(\xi r + (1-\xi) r_i^k, k\right)(r_i^k - r)$ with $H\left(\xi r + (1-\xi) r_i^k, k\right)$ being the Hessian matrix at the point $\xi r + (1-\xi) r_i^k$, $r \in \mathbf{S}_i$. By the construction of the finite volume, we have

$$\begin{aligned} (\nabla z(r,t))^T \hat{n}_i =& (\nabla z(r,k))^T \frac{(r_i^k - r_c^k)}{\mid r_i^k - r_c^k \mid} \\ \approx& \frac{z(r_i^k,k) - z(r_c^k,k)}{\mid r_i^k - r_c^k \mid} \\ &- \frac{1}{\mid r_i^k - r_c^k \mid} \int_0^1 \left(H_{r_i^k}(\xi) - H_{r_c^k}(\xi) \right) \xi d\xi. \end{aligned} \tag{56}$$

Substituting the expression of $\nabla z(r,t) \cdot \hat{n}_i$ into $\int_{\mathbf{S}_i} \theta \nabla z(r,k) \cdot \hat{n}_i \; dr$ gives,

$$\begin{aligned} &\int_{\mathbf{S}_i} \theta \nabla z(r,k) \cdot \hat{n}_i \; dr \\ &\approx \theta \frac{\mid \mathbf{S}_i \mid}{\mid r_i^k - r_c^k \mid} \left(z(r_i^k,k) - z(r_c^k,k) \right) \\ &\quad - \frac{\theta}{\mid r_i^k - r_c^k \mid} \int_{\mathbf{S}_i} \int_0^1 \left(H_{r_i^k}(\xi) - H_{r_c^k}(\xi) \right) \xi d\xi dr, \end{aligned} \tag{57}$$

where $\mid \mathbf{S}_i \mid$ is the length of the boundary segment $\mathbf{S}_i$.

Define the coefficients α_i and α_c as follows:

$$\begin{aligned} \alpha_i &= \frac{\mid \mathbf{S}_i \mid}{\mid r_i^k - r_c^k \mid}, \\ \alpha_c &= \sum_{i=1}^{N} \frac{\mid \mathbf{S}_i \mid}{\mid r_i^k - r_c^k \mid}, \end{aligned} \tag{58}$$

Define an approximation error term as

$$e(r_c^k,k) = \frac{1}{\Omega_c} \sum_{i=1}^{N} \frac{\theta}{\mid r_i^k - r_c^k \mid} \int_{\mathbf{S}_i} \int_0^1 (H_{r_i^k}(\xi) - H_{r_c^k}(\xi)) \xi d\xi dr. \tag{59}$$

$e(r_c^k,k)$ is the sum of integration of the differences of two Hessian matrices at r_i^k and r_c^k, which is the higher order term relative to the geometric distance $\|r_i^k - r_c^k\|$. To further simplify the notations, we define

$$\Gamma_k = \frac{1}{\Omega_c} \left[\sum_{i=1}^{N} (\alpha_i z(r_i^k,k)) - \alpha_c z(r_c^k,k) \right]. \tag{60}$$

Then

$$\theta \Delta z(r_c^k,k) = \Gamma_k \theta - e(r_c^k,k) \tag{61}$$

It should be noted that the α_i and α_c coefficients are related to the shape of the formation that the mobile agents form. In a special case where four agents form a symmetric formation, $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 1$ and $\alpha_c = 4$, which agree with the coefficients obtained by the finite difference method [2].

Appendix II

Proof of Proposition V.3

Based on condition (Cd1), we obtain that the controllability Grammian satisfies

$$\lambda_3 \sum_{j=k-\tau_1}^{k} \Phi(k,j)\Phi(k,j)^T \leq \mathfrak{C}(k, k-\tau_1)$$

and

$$\mathfrak{C}(k, k-\tau_1) \leq \lambda_4 \sum_{j=k-\tau_1}^{k} \Phi(k,j)\Phi(k,j)^T$$

for any k and τ_1 such that $k > \tau_1$. Therefore, if we can find the uniform bounds for each of these semi-definite symmetric matrices, i.e., $\Phi(k,j)\Phi(k,j)^T$, the overall bound for the controllability Grammian can be obtained readily. We first apply Lemma V.1 to compute $\Phi^{\mathrm{a}}(k,j)\Phi^{\mathrm{a}}(k,j)^T$, i.e.,

$$\begin{aligned} \Phi^{\mathrm{a}}(k,j)\Phi^{\mathrm{a}}(k,j)^T &= \begin{bmatrix} \xi_{\hat{\theta}} & \phi^T \\ 0 & I_{2\times 2} \end{bmatrix} \begin{bmatrix} \xi_{\hat{\theta}} & \phi^T \\ 0 & I_{2\times 2} \end{bmatrix}^T \\ &= \begin{bmatrix} \xi_{\hat{\theta}}^2 + \|\phi\|^2 & \phi^T \\ \phi & I_{2\times 2} \end{bmatrix}. \end{aligned} \tag{62}$$

Using basic linear algebra, we can obtain the eigenvalues of matrix (62) as follows

$$\begin{aligned} \lambda_1 &= \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + \sqrt{(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2)^2 - 4\xi_{\hat{\theta}}^2} \right), \\ \lambda_2 &= 1, \\ \lambda_3 &= \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 - \sqrt{(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2)^2 - 4\xi_{\hat{\theta}}^2} \right) = \frac{\xi_{\hat{\theta}}^2}{\lambda_1}. \end{aligned}$$

It is easy to show that

$$\begin{aligned} \lambda_1 &= \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + \sqrt{(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2)^2 - 4\xi_{\hat{\theta}}^2} \right) \\ &\geq \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + \sqrt{(1 + \xi_{\hat{\theta}}^2)^2 - 4\xi_{\hat{\theta}}^2} \right) \\ &= \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + 1 - \xi_{\hat{\theta}}^2 \right) \\ &\geq 1, \end{aligned}$$

and

$$\begin{aligned} \lambda_1 &= \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + \sqrt{(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2)^2 - 4\xi_{\hat{\theta}}^2} \right) \\ &\leq \frac{1}{2}\left(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2 + \sqrt{(1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2)^2} \right) \\ &= 1 + \xi_{\hat{\theta}}^2 + \|\phi\|^2. \end{aligned}$$

Due to condition (Cd3), we can see that $0 < \xi_{\hat{\theta}} \le 1$.

$$
\begin{aligned}
&\|\phi\| \\
=&\left\|(r_c^{k-1} - r_c^{k-2} + v\delta t)^T \right.\\
&\left. + \sum_{n=1}^{k-j-1}\left(\prod_{m=1}^{n}(1 - \frac{\alpha_c \hat{\theta}_{k-m}\delta t}{\Omega_c})\right)(r_c^{k-n-1} - r_c^{k-n-2} + v\delta t)^T\right\| \\
\le&\left\|(r_c^{k-1} - r_c^{k-2} + v\delta t)^T\right\| \\
&+ \sum_{n=1}^{k-j-1}\left(\prod_{m=1}^{n}(1 - \frac{\alpha_c \hat{\theta}_{k-m}\delta t}{\Omega_c})\right)\left\|(r_c^{k-n-1} - r_c^{k-n-2} + v\delta t)^T\right\| \\
\le&\left\|(r_c^{k-1} - r_c^{k-2} + v\delta t)^T\right\| + \sum_{n=1}^{k-j-1}\left\|(r_c^{k-n-1} - r_c^{k-n-2} + v\delta t)^T\right\| \\
\le&\|r_c^{k-1} - r_c^{k-2}\| + \|v\delta t\| + \sum_{n=1}^{k-j-1}\left(\|r_c^{k-n-1} - r_c^{k-n-2}\| + \|v\delta t\|\right) \\
\le&(k-j)(\lambda_5 + \|v\delta t\|) \\
\le&\tau_1(\lambda_5 + \|v\delta t\|).
\end{aligned}
$$

Hence we can show that λ_1 is bounded both above and below,

$$1 \le \lambda_1 \le 2 + \tau_1(\lambda_5 + \|v\delta t\|),$$

and the maximum value of λ_1 is $\lambda_8 = 2 + \tau_1(\lambda_5 + \|v\delta t\|)$.

Since $\lambda_3 = \frac{\xi_{\hat{\theta}}^2}{\lambda_1}$, we can have that

$$0 < \frac{\xi_{\hat{\theta}}^2}{2 + \tau_1(\lambda_5 + \|v\delta t\|)} \le \lambda_3 \le \xi_{\hat{\theta}}^2 \le 1,$$

and the minimum value of λ_3 is $\lambda_7 = \frac{\min \xi_{\hat{\theta}}^2}{\lambda_8} > 0$.

Therefore, we can conclude that $\lambda_7 I_{3\times3} \le \Phi^{\mathrm{a}}(k,j)\Phi^{\mathrm{a}}(k,j)^T \le \lambda_8 I_{3\times3}$ for all time $j \in [k-\tau_1, k]$. Since $\Phi^{\mathrm{a}}(k,j) = \Phi^{\mathrm{b}}(k,j)$, we can also have $\lambda_7 I_{3\times3} \le \Phi^{\mathrm{b}}(k,j)\Phi^{\mathrm{b}}(k,j)^T \le \lambda_8 I_{3\times3}$ for all time $j \in [k-\tau_1, k]$. This means that for $\Phi(k,j) = \mathrm{diag}[\Phi^{\mathrm{a}}, \Phi^{\mathrm{b}}]$, $\lambda_7 I_{6\times6} \le \Phi^(k,j)\Phi^(k,j)^T \le \lambda_8 I_{6\times6}$ holds for all time $j \in [k-\tau_1, k]$. Hence, $\lambda_3\lambda_7\tau_1 I_{6\times6} \le \mathfrak{C}(k, k-\tau_1) \le \lambda_4\lambda_8\tau_1 I_{6\times6}$. Let $\lambda_1 = \lambda_3\lambda_7\tau_1$ and $\lambda_2 = \lambda_4\lambda_8\tau_1$. Thus, according to Definition V.2, we have proved the uniformly complete controllability claim. ■

Proof of Proposition V.6

Based on condition (Cd6), we first observe that every elements in $D(k)$ is bounded. Hence, from conditions (Cd5) and (Cd6), we can prove that there exists two positive constants $\lambda_{16}, \lambda_{17}$ such that $\lambda_{16} I_{N\times N} \le [D(k)QD^T(k) + R] \le \lambda_{17} I_{N\times N}$. Then, the observability Grammian satisfies $\lambda_{17}^{-1}\sum_{j=k-\tau_2}^{k}\Phi^T(j,k)C^T(j)C(j)\Phi(j,k) \le \mathfrak{O}(k, k-\tau_2) \le \lambda_{16}^{-1}\sum_{j=k-\tau_2}^{k}\Phi^T(j,k)C^T(j)C(j)\Phi(j,k)$ for any k and τ_2 such that $k > \tau_2$. Then the uniformly completely observability can be proved by finding the positive uniform upper and lower bounds for $\sum_{j=k-\tau_2}^{k}\Phi^T(j,k)C^T(j)C(j)\Phi(j,k)$ for all $k > \tau_2$.

Since $C(j)\Phi(j,k) = \mathrm{diag}[C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k), C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)]$, we can have

$$
\begin{aligned}
&\Phi^T(j,k)C^T(j)C(j)\Phi(j,k) \\
&= \mathrm{diag}\Big[\left(C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)\right)^T C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k), \\
&\qquad \left(C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)\right)^T C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)\Big].
\end{aligned}
$$

In order to find the positive uniform upper and lower bounds for $\sum_{j=k-\tau_2}^{k}\Phi^T(j,k)C^T(j)C(j)\Phi(j,k)$ for all $k > \tau_2$, we will look into subsystems marked by a and b first.

According to Lemma V.1 and the definition of formation center that $r_c^{j-1} = \frac{1}{N}\sum_{i=1}^{N} r_i^{j-1}$, we can get the matrix $\Phi^{\mathrm{a}T}(j,k)C^{\mathrm{a}T}(j)C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)$ in Equation (63) for subsystem with superscript a.

$$
\begin{aligned}
&\Phi^{\mathrm{a}T}(j,k)C^{\mathrm{a}T}(j)C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k) \\
&= \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix}^T \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j-1} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix} \\
&= \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & -\frac{N}{\xi_{\hat{\theta}}^2}\phi^T \\ -\frac{N}{\xi_{\hat{\theta}}^2}\phi & \sum_{i=1}^{N}\left(r_i^{j-1} - r_c^{j-1}\right)\left(r_i^{j-1} - r_c^{j-1}\right)^T + \frac{N}{\xi_{\hat{\theta}}^2}\phi\phi^T \end{bmatrix}.
\end{aligned}
\tag{63}
$$

Due to conditions (Cd2) and (Cd6), we can observe that each element of the matrix (63) is bounded above, i.e., $\Phi^{\mathrm{a}T}(j,k)C^{\mathrm{a}T}(j)C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k) \le \lambda_{18} I_{3\times3}$ for some constant $\lambda_{18} > 0$.

Similarly for subsystem with superscript b, we can get the matrix $\Phi^{\mathrm{b}T}(j,k)C^{\mathrm{b}T}(j)C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)$ in Equation (64).

$$
\begin{aligned}
&\Phi^{\mathrm{b}T}(j,k)C^{\mathrm{b}T}(j)C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k) \\
&= \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix}^T \begin{bmatrix} \frac{1}{\xi_{\hat{\theta}}} & (r_1^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \vdots & \vdots \\ \frac{1}{\xi_{\hat{\theta}}} & (r_N^{j} - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \end{bmatrix} \\
&= \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & \frac{1}{\xi_{\hat{\theta}}}\sum_{i=1}^{N}(r_i^j - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \frac{1}{\xi_{\hat{\theta}}}\sum_{i=1}^{N}(r_i^j - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}}) & \Sigma \end{bmatrix},
\end{aligned}
\tag{64}
$$

where $\Sigma = \sum_{i=1}^{N}(r_i^j - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})(r_i^j - r_c^{j-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T$.

Due to conditions (Cd2) (Cd6) and that ϕ and $\xi_{\hat{\theta}}$ are bounded above, we can observe that each element of the matrix (64) is bounded above, i.e., $\Phi^{\mathrm{b}T}(j,k)C^{\mathrm{b}T}(j)C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k) \le \lambda_{19} I_{3\times3}$ for some constant $\lambda_{19} > 0$.

Hence the upper bound for $\Phi^T(j,k)C^T(j)C(j)\Phi(j,k)$ exists and $\Phi^T(j,k)C^T(j)C(j)\Phi(j,k) \le \lambda_{20} I_{6\times6}$, where $\lambda_{20} = \max\{\lambda_{18}, \lambda_{19}\}$.

For the lower bound, we can first show that the matrix $\Phi^{\mathrm{a}T}(j,k)C^{\mathrm{a}T}(j)C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)$ and $\Phi^{\mathrm{b}T}(j,k)C^{\mathrm{b}T}(j)C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)$ are positive semidefinite for any $j \in [k-\tau_2, k]$. Then we can use conditions (Cd4), (Cd6) and (Cd7) to show that $\Phi^{\mathrm{a}T}(j,k)C^{\mathrm{a}T}(j)C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)$ and $\Phi^{\mathrm{b}T}(j,k)C^{\mathrm{b}T}(j)C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)$ are strictly positive definite for some time instance $j_1 \in [k-\tau_2, k]$, which means that there exists the lower bound $\lambda_{21} > 0$ such that $\lambda_{21} I_{6\times6} \le \sum_{j=k-\tau_2}^{k}\Phi(j,k)^T C(j)^T C(j)\Phi(j,k)$.

Consider any nonzero vector $x \in \mathbb{R}^3$, and for any subsystem we can find that

$$x^T \Phi^{\mathrm{y}}(j,k)^T C^{\mathrm{y}}(j)^T C^{\mathrm{y}}(j)\Phi^{\mathrm{y}}(j,k)x$$
$$= (C^{\mathrm{y}}(j)\Phi^{\mathrm{y}}(j,k)x)^T \left(C^{\mathrm{y}}(j)\Phi^{\mathrm{y}}(j,k)x\right) \geq 0, \quad y \in \{a,b\}.$$

This shows that the matrix $\Phi(j,k)^T C(j)^T C(j)\Phi(j,k)$ is positive semidefinite for any $j \in [k-\tau_2, k]$, which implies that $\sum_{j=k-\tau_2}^{k} \Phi(j,k)^T C(j)^T C(j)\Phi(j,k)$ is also positive semidefinite.

Consider the time instance j_1 given in (Cd7), and the matrix $\Phi^{\mathrm{a}}(j_1,k)^T C^{\mathrm{a}}(j_1)^T C^{\mathrm{a}}(j_1)\Phi^{\mathrm{a}}(j_1,k)$ can be reduced using row operations as follows,

$$\Phi^{\mathrm{a}}(j_1,k)^T C^{\mathrm{a}}(j_1)^T C^{\mathrm{a}}(j_1)\Phi^{\mathrm{a}}(j_1,k)$$
$$= \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & -\frac{N}{\xi_{\hat{\theta}}^2}\phi^T \\ -\frac{N}{\xi_{\hat{\theta}}^2}\phi & \sum_{i=1}^N \left(r_i^{j_1-1} - r_c^{j_1-1}\right)\left(r_i^{j_1-1} - r_c^{j_1-1}\right)^T + \frac{N}{\xi_{\hat{\theta}}^2}\phi\phi^T \end{bmatrix}$$
$$\to \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & -\frac{N}{\xi_{\hat{\theta}}^2}\phi^T \\ 0 & \sum_{i=1}^N \left(r_i^{j-1} - r_c^{j-1}\right)\left(r_i^{j-1} - r_c^{j-1}\right)^T \end{bmatrix}.$$

Since for each $i \in \{1,\cdots,N\}$ matrix $\left(r_i^{j-1} - r_c^{j-1}\right)\left(r_i^{j-1} - r_c^{j-1}\right)^T$ is positive semidefinite, $\sum_{i=1}^N \left(r_i^{j-1} - r_c^{j-1}\right)\left(r_i^{j-1} - r_c^{j-1}\right)^T$ is also positive semidefinite.

Consider the two agents $i_1, i_2 \in \{1,\cdots,N\}$ given condition (Cd7). Since $r_{i_1}^{j_1-1}, r_{i_2}^{j_1-1}, r_C^{j_1-1}$ are not colinear, the two vectors $\left(r_{i_1}^{j-1} - r_c^{j-1}\right)$ and $\left(r_{i_2}^{j-1} - r_c^{j-1}\right)$ form an angle Ψ such that $0 < \Psi < \pi$. According to Lemma V.5, the minimum eigenvalue of the 2×2 symmetric matrix $M = \left(r_{i_1}^{j-1} - r_c^{j-1}\right)\left(r_{i_1}^{j-1} - r_c^{j-1}\right)^T + \left(r_{i_2}^{j-1} - r_c^{j-1}\right)\left(r_{i_2}^{j-1} - r_c^{j-1}\right)^T$ is strictly positive and M is strictly positive definite. This means that $\sum_{i=1}^N \left(r_i^{j-1} - r_c^{j-1}\right)\left(r_i^{j-1} - r_c^{j-1}\right)^T$ is strictly positive definite and has full rank. Since $\frac{N}{\xi_{\hat{\theta}}^2} \neq 0$, the matrix $\Phi^{\mathrm{a}}(j_1,k)^T C^{\mathrm{a}}(j_1)^T C^{\mathrm{a}}(j_1)\Phi^{\mathrm{a}}(j_1,k)$ is strictly positive definite, and $\sum_{j=k-\tau_2}^{k} \Phi^{\mathrm{a}}(j,k)^T C^{\mathrm{a}}(j)^T C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)$ is strictly positive definite. Hence, there exists a lower bound $\lambda_{22} > 0$ such that $\lambda_{22} I_{3\times 3} \leq \sum_{j=k-\tau_2}^{k} \Phi^{\mathrm{a}}(j,k)^T C^{\mathrm{a}}(j)^T C^{\mathrm{a}}(j)\Phi^{\mathrm{a}}(j,k)$.

For subsystem marked by superscript b, consider the time instance j_2 given in (Cd7), and the matrix $\Phi^{\mathrm{b}}(j_2,k)^T C^{\mathrm{b}}(j_2)^T C^{\mathrm{b}}(j_2)\Phi^{\mathrm{b}}(j_2,k)$ can be reduced using row operations as follows,

$$\Phi^{\mathrm{b}}(j_2,k)^T C^{\mathrm{b}}(j_2)^T C^{\mathrm{b}}(j_2)\Phi^{\mathrm{b}}(j_2,k)$$
$$= \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & \frac{1}{\xi_{\hat{\theta}}}\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ \frac{1}{\xi_{\hat{\theta}}}\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}}) & \Sigma \end{bmatrix}$$
$$\to \begin{bmatrix} \frac{N}{\xi_{\hat{\theta}}^2} & \frac{1}{\xi_{\hat{\theta}}}\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T \\ 0 & \sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{k=1}^N (r_k^{j_2} - r_i^{j_2})^T \end{bmatrix},$$

where $\Sigma = \sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})(r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})^T$.

For the term $\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{k=1}^N (r_k^{j_2} - r_i^{j_2})^T$, exchanging index will not change the result, which implies that

$$\begin{aligned}
&\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{k=1}^N (r_k^{j_2} - r_i^{j_2})^T \\
&= \sum_{k=1}^N (r_k^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{i=1}^N (r_i^{j_2} - r_k^{j_2})^T \\
&= \frac{1}{2}\Bigg(\sum_{i=1}^N (r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{k=1}^N (r_k^{j_2} - r_i^{j_2})^T \\
&\quad + \sum_{k=1}^N (r_k^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})\sum_{i=1}^N (r_i^{j_2} - r_k^{j_2})^T\Bigg) \\
&= \frac{1}{2}\sum_{i=1}^N\sum_{k=1}^N \Bigg((r_i^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})(r_k^{j_2} - r_i^{j_2})^T \\
&\quad + (r_k^{j_2} - r_c^{j_2-1} - \frac{\phi}{\xi_{\hat{\theta}}})(r_i^{j_2} - r_k^{j_2})^T\Bigg) \\
&= -\frac{1}{2}\sum_{i=1}^N\sum_{k=1}^N \left(r_k^{j_2} - r_i^{j_2}\right)\left(r_k^{j_2} - r_i^{j_2}\right)^T.
\end{aligned}$$

Since for any $i,k \in \{1,\cdots,N\}$ $\left(r_k^{j_2} - r_i^{j_2}\right)\left(r_k^{j_2} - r_i^{j_2}\right)^T$ is positive semidefinite, $\sum_{i=1}^N\sum_{k=1}^N \left(r_k^{j_2} - r_i^{j_2}\right)\left(r_k^{j_2} - r_i^{j_2}\right)^T$ is also positive semidefinite.

According to condition (Cd7), $r_1^{j_2}, \cdots, r_N^{j_2}$ are not colinear, which implies that there exists at least two vectors $\left(r_{k_1}^{j_2} - r_{i_3}^{j_2}\right)$ and $\left(r_{k_2}^{j_2} - r_{i_4}^{j_2}\right)$ form an angle Ψ' such that $0 < \Psi' < \pi$. According to Lemma V.5, the minimum eigenvalue of the 2×2 symmetric matrix $M' = \left(r_{k_1}^{j_2} - r_{i_3}^{j_2}\right)\left(r_{k_1}^{j_2} - r_{i_3}^{j_2}\right)^T + \left(r_{k_2}^{j_2} - r_{i_4}^{j_2}\right)\left(r_{k_2}^{j_2} - r_{i_4}^{j_2}\right)^T$ is strictly positive and M' is strictly positive definite. This means that $\sum_{i=1}^N\sum_{k=1}^N \left(r_k^{j_2} - r_i^{j_2}\right)\left(r_k^{j_2} - r_i^{j_2}\right)^T$ is strictly positive definite and has full rank. Since $\frac{N}{\xi_{\hat{\theta}}^2} \neq 0$, the matrix $\Phi^{\mathrm{b}}(j_2,k)^T C^{\mathrm{b}}(j_2)^T C^{\mathrm{b}}(j_2)\Phi^{\mathrm{b}}(j_2,k)$ is strictly positive definite, and $\sum_{j=k-\tau_2}^{k} \Phi^{\mathrm{b}}(j,k)^T C^{\mathrm{b}}(j)^T C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)$ is strictly positive definite. Hence, there exists a lower bound $\lambda_{23} > 0$ such that $\lambda_{23} I_{3\times 3} \leq \sum_{j=k-\tau_2}^{k} \Phi^{\mathrm{b}}(j,k)^T C^{\mathrm{b}}(j)^T C^{\mathrm{b}}(j)\Phi^{\mathrm{b}}(j,k)$, and there exists a lower bound $\lambda_{21} = \min\{\lambda_{22}, \lambda_{23}\} > 0$ such that $\lambda_{21} I_{6\times 6} \leq \sum_{j=k-\tau_2}^{k} \Phi(j,k)^T C(j)^T C(j)\Phi(j,k)$.

Therefore, we can conclude that $\lambda_{21} I_{6\times 6} \leq \sum_{j=k-\tau_2}^{k} \Phi(j,k)^T C(j)^T C(j)\Phi(j,k)$ and $\Phi(j,k)^T C(j)^T C(j)\Phi(j,k) \leq \lambda_{20} I_{3\times 3}$ for all $j \in [k-\tau_2, k]$. Hence $\lambda_{17}^{-1}\lambda_{21} I_{3\times 3} \leq \mathfrak{O}(k, k-\tau_2) \leq \lambda_{16}^{-1}\lambda_{20}\tau_2 I_{3\times 3}$. Let $\lambda_9 = \lambda_{17}^{-1}\lambda_{21}$ and $\lambda_{10} = \lambda_{16}^{-1}\lambda_{20}\tau_2$. Thus, according to Definition V.4, we have proved the uniformly complete observability claim. ■

References

[1] R. Ghez, *Diffusion Phenomena*. Kluwer Academic/ Plenum Publishers, 2nd edition, 2001.

[2] J. You, F. Zhang, and W. Wu, "Cooperative filtering for parameter identification of diffusion processes," in *Proc. of IEEE 55th Conference on Decision and Control*, 2016, pp. 4327–4333.

[3] D. Uciński and M. Patan, "Sensor network design for the estimation of spatially distributed processes," *Int. J. Appl. Math. Comput. Sci.*, vol. 20, no. 3, pp. 459–481, 2010.

[4] M. Krstic and A. Symshlyaev, "Adaptive control for PDEs," *Annual Reviews in Control*, vol. 32, no. 2, pp. 149–160, 2008.

[5] M. Krstic and A. Smyshlyaev, "Adaptive boundary control for unstable parabolic pdes, Part I: Lyapunov design," *IEEE Transactions on Automatic Control*, vol. 53, no. 7, pp. 1575–1591, 2008.

[6] M. A. Demetriou, N. A. Gatsonis, and J. R. Court, "Coupled control-computational fluids approach for the estimation of the concentration form a moving gaseous source in a 2-D domain with a lyapunov-guided sensing aerial vehicle," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 3, pp. 853–867, 2014.

[7] L. Rossi, B. Krishnamachari, and C. C. J. Kuo, "Distributed parameter estimation for monitoring diffusion phenomena using physical models," in *Sensor and Ad Hoc Communications and Networks, IEEE SECON, First Annual IEEE Communications Society Conference*, 2004, pp. 460–469.

[8] C. H. Colburn, J. B. Cessna, and T. R. Bewley, "State estimation in wall-bounded flow systems. part 3. the ensemble kalman filter," *Journal of Fluid Mechanics*, vol. 682, pp. 289–303, 2011.

[9] S. Omatu and K. Matumoto, "Distributed parameter identification by regularization and its application to predication of air pollution," *Internation Journal of Systems Science*, vol. 22, no. 10, pp. 2001–2012, 1991.

[10] A. Symshlyaev and M. Krstic, "Adaptive boundary control for unstable parabolic PDEs - Part II: Estimation-based designs," *Automatica*, vol. 43, no. 9, pp. 1543–1556, 2007.

[11] J. M. Niedzwecki and P. Y. F. Liagre, "System identification of distributed parameter marine riser models," *Ocean Engineering*, vol. 30, pp. 1387–1415, 2003.

[12] H. Li and C. Qi, "Modeling of distributed parameter systems for application - a synthesized review from time-space separation," *Journal of Process Control*, vol. 20, pp. 891–901, 2010.

[13] L. Guo and S. A. Billings, "Identification of partial differential equation models for continuous spatio-temporal dynamical systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 8, pp. 657 – 661, 2006.

[14] J. Ramsay and B. Silverman, *Functional data analysis*. Springer, New York, 2005.

[15] O. Chkrebtii, D. Campbell, B. Calderhead, and M. Girolami, "Bayesian solution uncertainty quantification for differential equations," *Bayesian Analysis*, vol. 11, no. 4, pp. 1239–1267, 2016.

[16] V. Isakov, *Inverse Problems for Partial Differential Equations*. Applied Mathematical Sciences, Springer-Verlag, New York, 1998.

[17] D. Vallette, G. Jacobs, and J. Gollub, "Oscillations and spatiotemporal chaos of one-dimensional fluid fronts," *Phys Rev E.*, vol. 55, pp. 4274–4287, 1997.

[18] D. Coca and S. Billings, "Direct parameter identification of distributed parameter systems," *International Journal of Systems Science*, vol. 31, no. 1, pp. 11–17, 2000.

[19] X. Xun, J. Cao, B. Mallick, A. Maity, and R. Carroll, "Parameter estimation of partial differential equation models," *journal of American Statistical Association*, no. 108, pp. 1009–1020, 2013.

[20] T. Muller and J. Timmer, "Parameter identification techniques for partial differential equations," *International Journal of Bifurcation and Chaos*, vol. 14, no. 6, pp. 2053–2060, 2014.

[21] G. Frasso, J. Jaeger, and P. Lambert, "Parameter estimation and inference in dynamic systems described by linear partial differential equations," *Advances in Statistical Analysis*, vol. 100, no. 3, pp. 259–287, 2016.

[22] W. Burgard, A. Cheyer, L. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[23] A. I. Mourikis and S. I. Roumeliotis, "Performance analysis of multi-robot cooperative localization," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 666–681, 2006.

[24] D. Uciński, *Optimal measurement methods for distributed parameter system identification*. Boca Raton, FL: CRC Press, 2004.

[25] W. Wu and F. Zhang, "Robust cooperative exploration with a switching strategy," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 828–839, 2012.

[26] F. Zhang and N. E. Leonard, "Cooperative control and filtering for cooperative exploration," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 650–663, 2010.

[27] S. Martnez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.

[28] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas, "Cooperative air and ground surveillance," *IEEE Robotics & Automation Magazine*, vol. 13, no. 3, pp. 16–25, 2006.

[29] Z. Tang and U. Ozguner, "Motion planning for multitarget surveillance with mobile sensor agents," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 898–908, 2005.

[30] M. E. Campbell and W. W. Whitacre, "Cooperative tracking using vision measurements on season uavs," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 4, pp. 613–626, 2007.

[31] M. A. Demetriou, "Guidance of mobile actuator-plus-sensor networks for improved control and estaimation of distributed parameter system," *IEEE Transations on Automatic Control*, vol. 55, no. 7, pp. 1570–1584, 2010.

[32] W. Wu and F. Zhang, "A speeding-up and slowing-down strategy for distributed source seeking with robustness analysis," *IEEE Transactions on Control of Networked Systems*, vol. 3, no. 3, pp. 231–240, 2016.

[33] E. Rafajlowicz and E. Skubalska-Rafajlowicz, "On optimal space-time excitation structures for parameter estimation in linear partial differential equations," *Optimal Design of ExperimentsTheory and Application*, pp. 130–145, 2011.

[34] M. A. Demetriou and I. I. Hussein, "Estimation of spatially distributed process using mobile spatially distributed sensor network," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 266–291, 2009.

[35] A. Dogandzic and A. Nehorai, *EEG/MEG spatio-temporal dipole source estimation and array design*. New York: Marcel Dekker, 2004, no. 393-442, high-resolution and Robust Signal Processing.

[36] D. Uciński and Y. Chen, "Time-optimal path planning of moving sensors for parameter estimation of distributed systems," in *Proc. of IEEE 44th Conference on Decision and Control*, 2005, pp. 5257–5262.

[37] C. Tricaud and Y. Chen, "Optimal trajectories of mobile remote sensors for parameter estimation in distributed cyber-physical systems," in *American Control Conference*, 2010, pp. 3211–3216.

[38] E. Kalnay, *Atmospheric modeling, data assimilation, and predictability*. Cambridge University Press, 2003.

[39] G. Evensen, "The ensemble kalman filter: Theoretical formulation and practical implementation," *Ocean dynamics*, vol. 53, no. 4, pp. 343–367, 2003.

[40] V. N. Christopoulos and S. Roumeliotis, "Adaptive sensing for instantaneous gas release parameter estimation," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 4450–4456.

[41] J. You and W. Wu, "Online passive identifier for spatially distributed systems using mobile sensor networks," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 6, pp. 2151–2159, 2017.

[42] I. Sivergina, M. Polis, and I. Kolmanovsky, "Source identification for parabolic equations," *Mathematics of Control, Signals, and Systems*, vol. 16, pp. 141–157, 2003.

[43] E. Fiorelli, N. E. Leonard, P. Bhatta, and et al., "Multi-AUV control and adaptive sampling in monterey bay," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 935–948, 2006.

[44] J. You, Y. Zhang, M. Li, K. Su, F. Zhang, and W. Wu, "Cooperative parameter identification of advection-diffusion processes using a mobile sensor network," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 3230–3236.

[45] W. Wu, J. You, Y. Zhang, M. Li, and K. Su, "Parameter identification of spatial–temporal varying processes by a multi-robot system in realistic diffusion fields," *Robotica*, vol. 39, no. 5, pp. 842–861, 2021.

[46] J. You and W. Wu, "Cooperative identification of advection–diffusion processes with spatially varying coefficients based on a multi-model structure," *Cyber-Physical Systems*, vol. 3, no. 1-4, pp. 103–120, 2017.

[47] W. Ren and R. W. Beard, *Distributed consensus in multi-vehicle cooperative control, communications and control engineering series*. London: Springer-Verlag, 2008.

[48] J. You and W. Wu, "Sensing-motion co-planning for reconstructing a spatially distributed field using a mobile sensor network," in *Proc. 56th Annual Conference on Decision and Control*, 2017, pp. 3113–3118.

[49] J. Droniou, "Finite volume schemes for diffusion equations: introduction to and review of modern methods," *Mathematical Models and Methods in Applied Sciences*, vol. 24, no. 8, pp. 1575–1619, 2014.

[50] D. M. . Causon, C. G. Mingham, and L. Qian, *Introductory finite volume methods for pdes*. Bookboon, 2011.

[51] D. Simon and T. L. Chia, "Kalman filtering with state equality constraints," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 128–136, 2002.

[52] L. Ljung, *System Identification, 2nd ed.* Prentice-Hall, Englewood Cliffs, 1998.

[53] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Academic Press, New York, 1970.

[54] L. Z. Guo, S. A. Billings, and H. L. Wei, "Estimation of spatial derivatives and identification of continuous spatio-temporal dynamical systems," *Internal Journal of Control*, vol. 79, no. 9, pp. 1118–1135, 2006.
[55] B. Efron and R. J. Tibshirani, *An introduction to the Boostrap*. New York: Chapman & Hall, 1993.
[56] F. Hermeline, "A finite volume method for the approximation of diffusion operators on distorted meshes," *Journal of computational Physics*, vol. 160, no. 2, pp. 481–499, 2000.

Jie You received the B.S. at North China Electric Power University, Beijing, China, in 2011, the M.S. degree at Zhejiang University in 2014, and the Ph.D. from the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, in 2018. His research interests include mobile sensor networks and control theory.

Ziqiao Zhang received the M.S. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 2021, and the B.S. degree from Sichuan University, Chengdu, China, in 2018. Currently, she is a Ph.D. student in electrical and computer engineering of Georgia Institute of Technology, Atlanta, GA, USA. Her research interests include mobile sensor networks, cooperative filtering, and opinion dynamics.

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 joined the School of ECE, Georgia Institute of Technology in 2007, where he is the Dean's Professor. 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 unmanned systems. He received the NSF CAREER Award in 2009, and the ONR YIP Award in 2010.

Wencen Wu received the B.S. and M.S degrees from Shanghai Jiao Tong University, Shanghai, China, in 2006 and 2009, respectively, and the dual M.S. degree and Ph.D. degree from the School of Electrical and Computer Engineering at the Georgia Institute of Technology, in 2010 and 2013, respectively. She joined the Computer Engineering Department at San Jose State University in 2018. She was an Assistant Professor at the Rensselaer Polytechnic Institute from 2013 to 2018. Her research interests include robotics, systems and control theory, and machine learning applied to intelligent autonomous multi-robot systems and distributed parameter systems.