Decentralized Environmental Modeling by Mobile Sensor Networks

Kevin M. Lynch, Senior Member, Ira B. Schwartz, Peng Yang, Student Member, Randy A. Freeman, Member

Abstract—Cooperating mobile sensors can be used to model environmental functions such as the temperature or salinity of a region of ocean. In this paper we adopt an optimal filtering approach toward fusing local sensor data into a global model of the environment. Our approach is based on the use of PI average consensus estimators, whereby information from each mobile sensor diffuses through the communication network. As a result, this approach is scalable and fully decentralized, and allows changing network topologies and anonymous agents to be added and subtracted at any time. We also derive control laws for mobile sensors to move to maximize their sensory information relative to current uncertainties in the model. The approach is demonstrated by simulations including modeling ocean temperature.

Index Terms—Multi-agent systems, decentralized control, distributed control, mobile sensor networks, average consensus estimation, ocean modeling.

I. INTRODUCTION

Consider a group of $n$ communicating mobile sensors (agents) cooperatively measuring a scalar environmental field, such as environment temperature or enemy density, as a function of the spatial location $r$. Let $g : \mathbb{R}^m \rightarrow \mathbb{R}$ be the environmental function, where $m$ is the dimension of the ambient space. Let $f(r)$ be a parameterized approximation to this function,

$$g(r) \approx f(r) = \sum_{j=1}^{\ell} x^{(j)} \psi^{(j)}(r),$$

where $\{\psi^{(j)}\}$ is a finite subset of a set of basis functions for scalar-valued signals over $\mathbb{R}^m$, and each $x^{(j)} \in \mathbb{R}$ is a coefficient multiplying its associated function. Examples of $\psi^{(j)}$ are sinusoids in Fourier series, wavelets, piecewise polynomials, splines, etc. A parameterized approximation of the environmental function, then, is given by the set of coefficients $x = \{x^{(1)}, \ldots, x^{(\ell)}\}$. The job of the sensor network is to estimate a model of the environment $\hat{x} \approx x$, where each agent’s sensor takes a scalar measurement of the environment $g$ only at its own location. We call the process of maintaining such a model environmental modeling.

We require a solution to the environmental modeling problem with the following features:

- Each agent maintains a model of the environment. By allowing each agent to maintain a model of the environment, each agent can make more intelligent local control decisions. Each agent’s model should fuse measurement data from all other agents. Agents should agree on the model that best fits the data, i.e., they should reach consensus.
- The communicated data are independent of the number of agents in the system. The communicated data needed for estimation must be summarizable in some way. This rules out message-passing with unique identifiers to accumulate information from all agents.
- The mobile sensors may be heterogeneous. Different agents may have different sensing and motion capabilities.
- The communication network need not be known and is allowed to change over time. As agents move, they may break contact with some agents and establish contact with others. Also, agents may be added to or subtracted from the system. There are no centralized agents, so performance improves or degrades smoothly with the addition or subtraction of agents.

With these properties, environmental modeling is robust, fully decentralized and scalable. We can also take advantage of the mobility of the agents to improve the estimate of the modeled variable. In some cases, the motion of the agents may even influence the evolution of the environment. These two classes of tasks are illustrated in the following two scenarios.

Scenario 1: Temperature and salinity monitoring. Each agent maintains a model of the temperature and/or salinity of a fixed region of the ocean. Each agent moves to refine the group estimate by positioning its sensor so as to maximally reduce uncertainty in the estimate.

Scenario 2: Herding. Agents maintain a density model of a group of adversarial or neutral agents. The agents move in an attempt to control the other group’s density. A model of the other agents’ repulsion from the monitoring agents can be used to predict their motion.

In this paper we focus on scenarios of the first type, where the mobile sensors move to maximize the expected information they receive as a function of the current uncertainty in the model.
A. Related Work

One approach to implementing optimal data fusion is simply to communicate each distinct measurement taken by every sensor in the ad hoc network to every other sensor, possibly tagged with the sensor ID and timestamp. Using appropriate communication protocols and routing schemes, each sensor accumulates data and passes them on to its neighbors [3], [20]. Each agent then implements an optimal estimator using its accumulated information. This approach to information communication is cumbersome for large sensor networks with changing topology and without synchronized clocks, and instead we focus on strategies where the communicated information is summarized rather than simply relayed.

In this paper we make use of the information form of the Kalman filter (KF) for optimal decentralized data fusion. The utility of the information form for decentralized filtering has long been recognized [9], [16], [18], [42], [46], as the correction step of the filter involves simply summing independent computations on the measurements from the individual sensors. For each sensor to maintain a fused model, each sensor must calculate (or otherwise have access to) the necessary sums. This is straightforward when there is a centralized summing agent, or when communication between agents is all-to-all: essentially, each agent acts as a centralized summer. Alternatively, if the network has a particular hierarchical structure, or is known to be acyclic with a known local topology, the sums can be calculated in a distributed manner [16]. In our case, the communication graph is not assumed to have any particular structure, except that it is undirected.

Perhaps the most closely related formulation to that of this paper is the approach to decentralized data fusion and control for active sensor networks of Durrant-Whyte et al. [17], [28]. They describe a general Bayesian approach to decentralized data fusion based on the use of channel filters which, in the information filter case, allow the necessary sums to be calculated in a decentralized way. The correctness of these channel filters requires that the network be acyclic so that no measurements are “counted twice”; if not, then conservative data fusion could be implemented, or a decentralized algorithm could be used to find an acyclic spanning tree of the network to use for communication. Mobile sensors are controlled to gather more information about the environment based on an entropy-like measure of the measurement uncertainty. The local control decisions may be arrived at by each agent individually or by negotiation among the group, and the controls may be simple gradient-following or based on a finite lookahead (e.g., optimal receding horizon control). Distributed approaches to negotiating a common cooperative control are described in [31], [44].

In this paper we formulate the problem of modeling a scalar field using a similar estimate-and-control approach. We define a measure of the model quality and a gradient control law to move the agents to collect sensor data that improves the quality of the model. While these are relevant to any centralized or decentralized implementation of environmental modeling, we focus on the decentralized case where there is no particular structure on the communication network, agents can be added or subtracted at any time, and synchronized clocks are not available to assign timestamps to sensory data. To achieve decentralized data fusion under such weak assumptions, we use decentralized PI (proportional-integral) dynamic average consensus estimators to allow each robot to calculate the sums needed to implement the information filter [13], [14]. These PI filters cause information from all the agents to “diffuse” through the network. This approach to sharing information has the advantage of allowing the agents to exactly estimate the necessary sums (in constant or slowly-changing environments) under weak assumptions on the network topology. A disadvantage is that agents need to be in constant communication to facilitate information diffusion, even when no new sensor readings are available. Another issue is that the time constant of the diffusion process, which depends on the network topology and filter gains, means that either (1) the KF update rate is limited by the settling time of the diffusion process, or (2) if the diffusion process is not settled (the case considered in this paper), then the system implements only an approximation to the KF, where data are low-pass filtered before fusing. For slowly-changing environments, however, the expected estimates converge to those of a centralized filter.

The PI dynamic average consensus estimators utilized in this paper belong to a class of estimators that generalize static average consensus estimators [35], [36], [38], [47] to the case of changing inputs. The purpose of these dynamic average consensus estimators is to allow each agent to (approximately) track the average of the time-varying inputs to all the agents. A high-pass dynamic average consensus estimator was introduced in [41], and its use in distributed Kalman filtering is described in [40]. In this case, identical noisy sensors measure the same time-varying quantity. The distributed scheme mimics a centralized KF when average consensus is reached between KF iterations; it acts as a purely local KF if no communication occurs between Kalman iterations; and the behavior of the system varies smoothly between these two extremes in the usual case. Recognizing that high-frequency content of the measured signal typically corresponds to noise, Olfati-Saber and Shamma proposed a low-pass dynamic consentus filter [37]. This filter does not specifically compute the average of different inputs, but bounds on its tracking performance are given for the case of multiple noisy measurements of the same time-varying quantity. The distributed KF described in [33] uses both the low-pass dynamic consensus filter of [37] and a band-pass consensus filter that composes the low-pass filter with the high-pass average filter of [41]. This KF is applied to multiple noisy measurements of the same quantity.

The PI dynamic average consensus estimators used in this paper combine the benefits of being dynamic average consensus estimators (as with the high-pass average estimator of [41]) as well as being low-pass (as with the low-pass, but nonaveraging, filter of [37]). The integral action of the linear dynamics of the PI average estimators results in zero-steady state error when the inputs and the communication graph topology become fixed (possibly after an initial transient period of changing graph topology). For the high-pass estimator of [41], zero steady-state error requires extra bookkeeping to
keep track of which communication links are active. Intermittent communication noise or drops cause the high-pass filters to drift relative to the (average) solution manifold, whereas a “forgetting” factor in the PI filter results in a stable filter from communication noise to errors relative to the solution manifold, allowing this noise to be “forgotten” over time. In addition, the PI average consensus filters have a stable filter between noisy inputs and outputs, unlike the direct feed-through of high-pass average consensus filters. All of these properties make the PI average consensus filter attractive for use in decentralized KF. More details can be found in [14].

In decentralized estimate-and-control problems, as studied in this paper, agents cooperatively estimate properties of the environment and simultaneously implement control laws based on those properties. As a result, the control laws influence the estimates, and the estimates influence the control laws, making the coupled stability of these interacting systems a fundamental issue, studied in [13], [48].

Other approaches to controlled sampling and modeling of a scalar field include [11], [12], [27], [29], [32], [39], [50], [51]. In [32], vehicles move in a fixed formation along an estimated gradient, while in [12], [51], vehicles are controlled to track a level set of the field. In [50], a fixed network of sensor buoys measures the temperature of a lake, while a single mobile sensor chooses a path to attempt to minimize the mean square error (integrated over the monitored area) in the temperature reconstruction. Based on optimal interpolation theory and a spatio-temporal model of ocean temperature fluctuations, Leonard et al. [27] derive, in advance, super-elliptical paths for underwater vehicles that nearly optimally sample ocean temperatures according to a measure of the model’s mean-square error integrated over the region and time period of interest. A distributed Kriged Kalman filter is developed in [10] to estimate a spatio-temporal field. Gradient control laws are developed to move the mobile sensors to critical points of the field. In [11], [39], sensors move to cover an environment by maximizing the sum of the information gain based on the local “interestingness” of the environment. In both [30], [34], a measure of sensor information is used to drive the motion of mobile sensors, and it is observed in [34] that “flocking” behavior of mobile sensors emerges purely due to their information-seeking objective of tracking a target. Active information gathering for target tracking is also described in [8].

There is a large and rapidly growing literature on the use of multirobot systems for a number of other tasks, not reviewed here, such as surveillance, mapping, and localization, including large-scale deployments [19], [25]. A cross-section of recent work on multirobot systems can be found in special issues in the IEEE Transactions on Robotics and Automation [6] and the Proceedings of the IEEE [45]. Recent work on the related topic of networked control systems can be found in another special issue of the Proceedings of the IEEE [5].

B. Overview

Section II describes the Kalman filtering approach to estimating a parameterized model of the environment, and Section III derives the control law to improve sensor information. Section IV describes the PI average consensus estimator that allows decentralized, scalable implementation of the KF for arbitrary networks. Section V illustrates example applications, and Section VI provides conclusions and avenues for further research.

II. CENTRALIZED ENVIRONMENTAL MODELING

A. Discrete-Time Formulation

In the case that the environment is changing, we assume the environmental parameters \( x = [x^{(1)}, \ldots, x^{(\ell)}]^T \) evolve according to the linear dynamics

\[
x(k) = F(k)x(k-1) + G(k)u(k) + w(k)
\]

where \( F \in \mathbb{R}^{\ell \times \ell}, G \in \mathbb{R}^{\ell \times q}, u \in \mathbb{R}^q \). For estimation purposes, it is assumed that \( F, G, \) and the inputs \( u \) are known or can be estimated (e.g., from a linearized model of nonlinear environmental dynamics). The disturbance input \( w \in \mathbb{R}^\ell \) is a Gaussian noise with zero mean and covariance matrix \( Q(k) \).

In the absence of a good model of the dynamics of the environment, we can set \( F \) to the identity matrix and \( G \) to 0.

The environment is measured by the \( n \) agents’ scalar sensors at their locations \( p_1, \ldots, p_n \in \mathbb{R}^m \), giving the measurements \( z = [z_1, \ldots, z_n]^T \in \mathbb{R}^n \), according to

\[
z(k) = H(k)x(k) + v(k),
\]

where \( v \in \mathbb{R}^n \) is zero-mean Gaussian noise with diagonal, positive-definite covariance \( R(k) \). Row \( i \) of the \( n \times \ell \) measurement matrix \( H \) is written

\[
H_i = \Psi(p_i) = [\psi^{(1)}(p_i) \ \psi^{(2)}(p_i) \ \ldots \ \psi^{(\ell)}(p_i)],
\]

where \( \{\psi^{(1)}, \ldots, \psi^{(\ell)}\} \) are the model basis functions.

The optimal estimator for this system is a Kalman filter [4], [43]. Let \( \hat{x} \) be the optimal estimate of \( x \) and \( P \) be the \( \ell \times \ell \) covariance matrix corresponding to the uncertainty of the estimate. Defining the (Fisher) information matrix \( Y = P^{-1} \) and the information vector \( \hat{\tau} = P^{-1}\hat{x} \), the discrete-time KF iterations can be written in information form as follows.

**prediction:**

\[
\hat{Y}(k) = (F(k)Y^{-1}(k-1)F^T(k) + Q(k))^{-1}
\]

**correction:**

\[
\hat{\tau}(k) = \hat{\tau}(k) + H^T(k)R^{-1}(k)z(k)
\]

Rewriting the correction update, we get

\[
Y(k) = \hat{Y}(k) + C(k)
\]

\[
\hat{\tau}(k) = \hat{\tau}(k) + y(k),
\]
where, dropping the dependence on $k$, the measurement covariance $C$ and the measurement vector $y$ are written

$$
C = H^T R^{-1} H = \sum_{i=1}^{n} H_i^T R_i^{-1} H_i = \sum_{i=1}^{n} \Psi^T (p_i) R_i^{-1} \Psi (p_i)
$$

(9)

$$
y = H^T R^{-1} z = \sum_{i=1}^{n} H_i^T R_i^{-1} z_i = \sum_{i=1}^{n} \Psi^T (p_i) R_i^{-1} z_i,
$$

(10)

where $R_i$ is the scalar measurement variance of agent $i$’s sensor. Each agent can implement the correction equations if it has access to the $\ell \times \ell$ matrix $C$ and the $\ell$-vector $y$. The matrix $C$ consists of $\ell (\ell + 1)/2$ unique scalar sums, as the matrix is symmetric, and the vector $y$ consists of $\ell$ scalar sums.

If the sensor noise model $R_i$ is a function of the environmental value, e.g., variance proportional to the signal, then the sensor covariance $R_i$ can be written as a function of the expected measurement, $R_i (H_t \hat{x})$, in calculating $C$ and $y$ in the correction steps of the filter. Also, note that calculating $H_t$ requires exact knowledge of the location of agent $i$. The case where agents are uncertain of their location is discussed further in Section VI.

For simplicity we assume linear process dynamics and measurement functions, but the methods can be trivially extended to the nonlinear case by using extended Kalman filters.

### B. Continuous-Time Formulation

In the continuous-time case, the system dynamics and measurement equations are

$$
\dot{x}(t) = F(t)x(t) + G(t)u(t) + w(t),
$$

(11)

$$
z(t) = H(t)x(t) + v(t),
$$

(12)

where $w(t)$ and $v(t)$ are drawn from zero-mean Gaussian random processes with spectral density matrices $Q(t)$ and $R(t)$, respectively. The optimal Kalman-Bucy filter is

$$
\dot{P} = FP + PF^T + Q - PCP \quad \Psi
$$

(13)

$$
\dot{x} = F \hat{x} + Gu + P(y - C \hat{x}),
$$

(14)

where $C$ and $y$ are given in (9) and (10), respectively. We could equivalently write (13)–(14) in information form, in direct analogy with the discrete-time case (see, e.g., [17]).

### III. CONTROL FOR BETTER SENSOR DATA

Each mobile sensor moves to maximize the information in its sensor data as a function of its current location $p_i$, its sensor uncertainty $R_i$, the set of basis functions $\{\psi^{(j)}\}$ of the environmental model, and the model uncertainty as represented in the covariance matrix $P$. Let $A \subset \mathbb{R}^m$ denote the bounded region which is being modeled by the mobile sensors. The quality of the current environmental model is inversely related to the cost

$$
J = \int_A \Psi (r) P \Psi^T (r) dA,
$$

where $r \in A$ and $dA$ is a differential element of $A$. The cost $J$ is the integral of the variance of the model over the region of interest $A$ (Figure 1). This cost takes into account the scaling by the basis functions $\{\psi^{(j)}\}$ of the uncertainty represented by $P$, as well as the variance in the model. The cost $J$ is similar in spirit to the spatial component of the spatio-temporal cost defined in [27].

We assume agents are fully actuated with velocities as controls. To reduce the cost $J$, each agent chooses a motion control based simply on the gradient of $J$ with respect to its own configuration. The dependence of $J$ on agents’ configurations will be made explicit below.

### A. Discrete-Time Formulation

Each agent moves according to the dynamics

$$
p_i (k + 1) = p_i (k) + f_i (k), \quad f_i \in \mathbb{R}^m
$$

and implements the simple gradient control law

$$
f_i = -K \frac{\partial J}{\partial p_i},
$$

(15)

with a scalar gain $K > 0$ and

$$
\frac{\partial J}{\partial p_i} = \int_A \Psi (r) \frac{\partial P}{\partial p_i} \Psi^T (r) dA.
$$

(16)

We plug in $P(k + 1)$ for $P$ in (16), indicating that it is the covariance matrix after the next sensor reading that is of interest. By matrix calculus we have

$$
\frac{\partial P(k + 1)}{\partial p_i} = \frac{\partial Y^{-1}(k + 1)}{\partial p_i}
$$

$$
= -Y^{-1}(k + 1) \frac{\partial Y(k + 1)}{\partial p_i} Y^{-1}(k + 1)
$$

$$
= -Y^{-1}(k + 1) \frac{\partial Y(k + 1)}{\partial p_i} Y^{-1}(k + 1) P(k + 1).
$$

To evaluate $P(k + 1) = Y^{-1}(k + 1) = (\dot{Y}(k + 1) + C(k + 1))^{-1}$, we assume that all other agents are stationary, i.e., $C(k + 1) =$
C(k). The gradient $\partial Y / \partial p_i$ reduces to

$$\frac{\partial Y}{\partial p_i} = \frac{\partial C}{\partial p_i} = \frac{\partial H^T}{\partial p_i} R^{-1} H + H^T R^{-1} \frac{\partial H}{\partial p_i}$$

$$= 2 \frac{\partial H^T}{\partial p_i} R^{-1} H. \quad (17)$$

Here we assume that the measurement noise $R_i$ does not directly depend on $p_i$, but such a dependence could easily be added.

Writing out the control law (15) in full, we get

$$f_i = 2KR_i^{-1} \int_A \Psi(r) P \frac{\partial Y}{\partial p_i}(p_i) H(p_i) P \Psi^T(r) dA. \quad (18)$$

Expressing $p_i \in \mathbb{R}^m$ in terms of its components $[p_{i,1}, \ldots, p_{i,m}]^T$, we can write the control (18) simply in terms of its components $f_i = [f_{i,1}, \ldots, f_{i,m}]^T$,

$$f_{i,j} = 2KR_i^{-1} \int_A \Psi(r) P \frac{\partial \Psi}{\partial p_{i,j}}(p_i) (p_i) P \Psi^T(r) dA.$$

We notice that the control $f_i$ is small (the sensor is not in a hurry to reposition itself) if (a) the sensor uncertainty $R_i$ is large, (b) the covariance $P$ will be small at the next step even without sensor motion, or (c) the gradient of $\Psi(p_i)$ with respect to $p_i$ is small (i.e., the measurement matrix $H_i = \Psi(p_i)$ is not sensitive to small changes in the position). Note that the control is a function only of the environment basis functions $\{\psi^{(j)}\}$, the domain $A$ being monitored, the sensor uncertainty $R_i$, the control gain $K$, and the current $P$ and $C$ matrices. It is not directly dependent on the estimate $\hat{x}$. Accordingly, if the sensor uncertainty $R_i$ is independent of the environment, the motions of the agents could be computed in advance, as a function of the initial conditions. This is no longer the case if there is uncertainty in the agents’ locations, which must be estimated in real-time. In this case, the models of the agents’ locations as well as the measurement-dependent model of the environment impact the gradient control law. This is discussed further in Section VI.

Preventing agent-agent collision has a higher priority than information gathering, so controls commanded by the control law (15) may be modified by virtual inter-agent repulsive functions, or simply overridden based on a conservative collision-likelihood condition.

Example 1: The multi-agent system consists of agents moving on a line, $p_i \in \mathbb{R}$, monitoring an interval $[0,1] \subset \mathbb{R}$ using a quadratic model of the environment, with basis functions $\Psi(r) = [1 \quad r \quad r^2]^T$. Assuming that the expected covariance (without agent motion) at the next step is the identity matrix $P = I_3$, plugging into (18) yields agent $i$’s control

$$f_i = 2KR_i^{-1} \int_0^1 [1 \quad r \quad r^2] \begin{bmatrix} 0 & 1 & 1/p_i \ 2p_i & p_i^2 & 1 \ r^2 \end{bmatrix} \begin{bmatrix} r \ 0 \ 0 \end{bmatrix} dr$$

$$= 2KR_i^{-1} \int_0^1 [1 \quad r \quad r^2] I_3 \begin{bmatrix} 1 & 0 & 1 \ 0 & 2p_i & 2p_i^2 \ 0 & 2p_i & 2p_i^2 \end{bmatrix} \begin{bmatrix} r \ 0 \ 0 \end{bmatrix} dr$$

$$= 2KR_i^{-1} \int_0^1 (2p_i^3r^4 + 3p_i^2r^3 + 3p_i r^2 + r) dr$$

$$= 2KR_i^{-1} ((2/5)p_i^3 + (3/4)p_i^2 + p_i + 1/2),$$

a polynomial in the agent’s position with coefficients determined by $K, R_i^{-1}, \Psi(r), P$, and the region being monitored. In general, if the basis functions are monomials of the position coordinates of degree $s$ or less, the sensor locations that result in zero motion are solutions of polynomial equations of degree $2s-1$. In the example above, an agent is only stationary at $p_i = -0.756$; agents on either side of this point move away from this point. This point may not be an equilibrium, however; in general, at the next step, the $P$ matrix will change, and so will the solution to the polynomial equation.

An open question is how to choose the gain $K > 0$ for the discrete-time case. Sufficiently small values guarantee improved positioning, at the cost of slow improvement, while large values may cause unstable oscillation in the agents’ positions and a worsening in the cost $J$. The gain can be tuned offline to be appropriate for the chosen basis functions $\Psi(r)$, or adaptive step-size schemes may be chosen instead of using a fixed $K$.

B. Continuous-Time Formulation

In the continuous-time formulation we consider mobile sensors with single-integrator dynamics $\dot{p}_i = f_i$, where $p_i, f_i \in \mathbb{R}^m$.

Each agent implements a gradient control law

$$f_i(t) = -K \frac{\partial J}{\partial p_i}(t) = -K \int_A \Psi(r) \frac{\partial P}{\partial p_i}(t) \Psi^T(r) dA$$

with a scalar gain $K > 0$. Using (13), the term $\frac{\partial P}{\partial p_i}(t)$ is computed by integrating the differential equation

$$\frac{\partial P}{\partial p_i} = F \frac{\partial P}{\partial p_i} + \frac{\partial P}{\partial p_i} F^c$$

$$- \frac{\partial P}{\partial p_i} C P - P \frac{\partial P}{\partial p_i} C - \gamma \frac{\partial P}{\partial p_i}$$

from the initial condition $\frac{\partial P}{\partial p_i}(0)$. As before, the influence of the position of the mobile sensor on the quality of the estimate is evident in the term $\partial C/\partial p_i (17)$. 

IV. DECENTRALIZED IMPLEMENTATION

The environmental modeling approach outlined in Sections II and III is amenable to a direct centralized implementation. To arrive at a decentralized implementation of Kalman filtering, each agent requires access to the global information $C$ and $y$ in (9) and (10), respectively. To calculate the sensor-information gradient control laws, each agent requires the fused global covariance matrix $P$. There are a number of ways to achieve this in a decentralized way (Section I-A); in this paper we use diffusive average estimators that require no knowledge of the structure of the communication network. In Section IV-A we describe the PI average consensus estimators and their properties, and in Section IV-B we show that these properties allow the decentralized KF environment estimates to converge to that of the ideal centralized KF for

\[ ^{2}\text{The polynomials are multivariate if the monitored region has dimension } m > 1. \]
slowly-changing environments. Although both discrete-time and continuous-time formulations are possible, in this section we conduct our analysis in continuous-time.

A. PI Dynamic Average Consensus Estimators

Each agent maintains \(\ell (\ell + 3)/2\) sums of inputs, as represented by \(C\) and \(y\) in (9) and (10), respectively. Let agent \(i\)'s input to the sums be denoted by the vector \(\phi_i\), a function of its \(1 \times \ell\) measurement matrix \(H_i = \Psi(p_i)\), its sensor variance \(R_i\), and its measurement \(z_i\). To maintain these sums, each agent implements a proportional-integral (PI) average consensus estimator \([13], [14]\) and multiplies the results by \(n\), the number of agents in the system. The value of \(n\) may either be known in advance, in the case of a fixed number of mobile sensors, or estimated by a separate decentralized estimation procedure \([7], [26]\).

The linear PI average consensus estimator implemented by agent \(i\) is \(^3\)

\[
\dot{\nu}_i = \gamma(\phi_i - \nu_i) - K_P \sum_{j \in \mathcal{N}_i} (\nu_i - \nu_j) + K_I \sum_{j \in \mathcal{N}_i} (\eta_i - \eta_j) \tag{19}
\]

\[
\dot{\eta}_i = -K_I \sum_{j \in \mathcal{N}_i} (\nu_i - \nu_j), \tag{20}
\]

where \(\phi_i\) is agent \(i\)'s input vector to the sums, \(\nu_i\) is agent \(i\)'s estimate of the average of all the agents’ inputs, \(\eta_i\) is an internal estimator state, \(K_P, K_I \geq 0\) are estimator gains, \(\gamma > 0\) is a parameter governing the rate at which new information replaces old information in the dynamic averaging process, and \(\mathcal{N}_i\) indicates the set of one-hop neighbors of agent \(i\) in the communication network. Each agent communicates its estimate \(\nu_i\) and its internal estimator state \(\eta_i\) to its neighbors in the network. Each \(\nu_i(t)\) approximately tracks the true average of the inputs \(\phi_i(t)\) as the inputs and communication topology change, and if the inputs and (connected) communication topology eventually become constant, each \(\nu_i\) converges exactly to \(\frac{1}{n} \sum_{i=1}^{n} \phi_i\) (zero steady-state error).

The idea is illustrated in Figure 2. In Figure 2 (top), the inputs \(\phi_1 \ldots \phi_n\) are averaged by the centralized filter \(B\), and this average is provided to each agent, allowing an exact implementation of the KF for each agent. In contrast, Figure 2 (bottom) shows the decentralized averaging approach, where the decentralized filter \(B'\) describes the networked PI filters and produces the agents’ (different) average estimates \(\nu_i\) from the inputs \(\phi_i\). As a result, the agents maintain different environmental estimates \(\{\hat{x}_i, P_i\}\), which will be shown to converge to a centralized filter’s estimate for a slowly-changing environment.

Without loss of generality, let the inputs \(\phi_i\) be scalars. (Multiple identical filters running simultaneously give the vector version of the PI estimator.) We make two key observations that hold for any connected network of PI estimators of the form (19)–(20):

(P1) Each individual transfer function from an input \(\phi_i(t)\) to an error \(\nu_i(t) - \frac{1}{n} \sum_{i=1}^{n} \phi_i(t)\) is stable with a zero at the origin. Therefore, each error transfer function has a low-frequency magnitude response asymptote of 20 dB/dec, yielding zero steady-state error to DC inputs.

(P2) Uniform positive scaling of gains \(K_P', K_I', \gamma'\) by \(\kappa > 0\) to \(K_P = \kappa K_P', K_I = \kappa K_I', \gamma = \kappa \gamma'\) simply shifts the frequency response of these transfer functions. For \(\kappa > 1\), the frequency response is shifted to higher frequencies, and for \(\kappa < 1\) it is shifted to lower frequencies.

As an example, Figure 3 shows the singular value plots of the continuous-time MIMO transfer functions \(B'\) (the PI estimators) and \(B' - B\) (the error between the estimated averages \(\nu_i(t)\) of the time-varying inputs \(\phi_i(t)\) and the actual average of those inputs) for a network of nine agents communicating only with adjacent agents, with \(K_P = K_I = 1\). These plots show that the filter \(B'\) acts as a low-pass averaging filter with zero error for DC inputs. The effect of using decentralized averaging, therefore, is to reject high-frequency signals in a network-dependent way. Increasing the filter gains simply pushes the singular value plots in Figure 3 (right) to higher frequencies, without changing their shape, allowing the filter to better track the average of higher-frequency inputs. (In practice, communication delays or discrete-time implementations put an upper bound on filter gains that maintain stability.) Adding links to the network (keeping filter gains constant) both changes the shape of the frequency responses (while keeping the same asymptotic magnitude slopes) and shifts them to higher frequency.

For slowly-changing environments, the primary effect of using the filter \(B'\) instead of \(B\) is to color the sensing and process noise (assumed white in the KF). Theorems 1, 2, and 3 further establish the properties of PI dynamic average consensus estimators that make them useful for decentralized Kalman filtering. Theorem 1 bounds the estimator error as a function of the rate of change of the input for a fixed network; Theorem 2 bounds the error for a changing network and constant inputs; and Theorem 3 establishes the stability of the coupled estimators for arbitrarily fast time-varying networks.

\(^3\)The discrete-time version of the PI estimator is obtained by replacing \(\dot{\nu}_i\) and \(\dot{\eta}_i\) by \(\Delta \nu_i\) and \(\Delta \eta_i\), respectively. In the case of a discrete-time estimator, the sample time need not be the same as that of the discrete-time KF.
Our analysis uses the concept of the (possibly timedependent) communication network Laplacian \( \mathcal{L}(t) = \text{diag}(A(t)1) - A(t) \), where 1 is a vector of ones and \( A(t) \) is the symmetric adjacency matrix with \( a_{ij}(t) = a_{ji}(t) = 1 \) if agents \( i \) and \( j \) communicate and \( a_{ij}(t) = a_{ji}(t) = 0 \) if they do not [15]. We assume the network is always connected, i.e., there is a path on the communication graph connecting each pair of agents. In the case of \( n \) agents, this means \( \text{rank}(\mathcal{L}) = n - 1 \) and that all eigenvalues, except one singulare eigenvalue at the origin, are positive and real. The unit eigenvector associated with the zero eigenvalue is \( v_e = 1/\sqrt{n} \). We also define \( S \in \mathbb{R}^{n \times (n-1)} \) such that \( [v_e, S] \) is an orthogonal matrix. The Laplacian \( \mathcal{L} \) of any connected network satisfies \( \mathcal{S}^T \mathcal{L} \mathcal{S} \geq \varepsilon I \), and the largest such value of \( \varepsilon \) (i.e., the second-smallest eigenvalue of \( \mathcal{L} \), \( \lambda_2(\mathcal{L}) \)) is called the algebraic connectivity of the graph.

For simplicity in notation, let each agent's vector input \( \phi_i \) to the networked PI average estimators be a scalar \( s_i \in \mathbb{R} \), with \( s = [s_1, \ldots, s_n]^T \). Then the vector PI estimators (19)–(20) consist of \( \ell(\ell + 3)/2 \) independent PI estimators. With this notation, the networked PI estimators (19) and (20) can be written collectively as

\[
\begin{bmatrix}
\nu(t) \\
\eta(t)
\end{bmatrix} = \begin{bmatrix}
-\gamma I - L_P(t) & L_I(t) \\
-L_I(t) & 0
\end{bmatrix} \begin{bmatrix}
\nu(t) \\
\eta(t)
\end{bmatrix} + \begin{bmatrix}
\gamma I \\
0
\end{bmatrix} s(t),
\]

(21)

where \( L_P(t) = \kappa L \) and \( L_I(t) = K_I L(t) \). We examine the error dynamics

\[
e_v(t) = \nu(t) - \frac{11^T}{n} s(t),
\]

(22)

as \( t \to \infty \), where \( \beta > 0 \) is a function of \( K_P', K_I', \gamma', \lambda_2(L) \), and \( \lambda_{\max}(L) \), as defined in the proof. In particular, for any \( \rho_1 \geq 0 \) and any \( \varepsilon \geq 0 \), we can choose \( \kappa \) sufficiently large such that \( e_v(t) \) converges to a ball of size \( \varepsilon \).

**Proof:** Consider the state coordinate change

\[
\nu(t) = \begin{bmatrix}
v_e \\
S \chi(t)
\end{bmatrix}, \quad \chi(t) = \begin{bmatrix} v_e \cr S \sigma(t) \end{bmatrix}, \quad \sigma(t) = \begin{bmatrix} v_e \cr S \eta(t) \end{bmatrix}.
\]

(23)

(24)

In the new state coordinates \( \chi, \sigma \), the system (21) becomes

\[
\dot{\chi}(t) = \begin{bmatrix}
-\gamma & 0 \\
-\gamma I - S^T L_P S & \gamma S^T
\end{bmatrix} \chi(t) + \begin{bmatrix}
\gamma v_e^T \\
\gamma S^T
\end{bmatrix} \eta(t),
\]

(25)

(26)

The first element \( \sigma_1 \) of the vector \( \sigma \) represents an uncontrollable, unobservable state with dynamics \( \dot{\sigma}_1(t) = 0 \). If we drop this state and define \( \sigma_2(t) = [\sigma_2(t) \ldots \sigma_n(t)]^T = S^T \eta(t) \), then we can write the remaining dynamics as

\[
\begin{bmatrix}
\dot{\chi}(t) \\
\dot{\sigma}_2(t)
\end{bmatrix} = A_1 \begin{bmatrix}
\chi(t) \\
\sigma_2(t)
\end{bmatrix} + A_2 s(t),
\]

(28)

(29)

where

\[
A_1 = \begin{bmatrix}
-\gamma & 0 & 0 \\
0 & -\gamma I - S^T L_P S & S^T \chi(t) \\
0 & -S^T L_I S & 0
\end{bmatrix} = \begin{bmatrix}
-\gamma & 0 & 0 \\
0 & -S^T L_I S & 0
\end{bmatrix},
\]

(30)

\[
A_2 = \begin{bmatrix}
\gamma v_e^T \\
\gamma S^T
\end{bmatrix}, \quad A_3 = [ v_e \cr S \cr 0 ], \quad A_4 = -\frac{11^T}{n}.
\]

Because \( S^T L_I S \) is invertible, we may define the vectors

\[
\bar{\chi}(t) = \begin{bmatrix}
\nu_e^T s(t) \\
\vdots \\
0
\end{bmatrix}, \quad \bar{\sigma}_2(t) = -\gamma(S^T L_I S)^{-1} S^T s(t)
\]

(31)

which have the property that

\[
\begin{bmatrix}
A_1 & A_2 \\
A_3 & A_4
\end{bmatrix} \begin{bmatrix}
\bar{\chi}(t) \\
\bar{\sigma}_2(t)
\end{bmatrix} = 0.
\]

(32)

Next we shift the origin in the state space by defining

\[
\zeta(t) = \begin{bmatrix}
\chi(t) \\
\sigma_2(t)
\end{bmatrix} - \begin{bmatrix}
\bar{\chi}(t) \\
\bar{\sigma}_2(t)
\end{bmatrix}
\]

(33)

which from (32) satisfies

\[
\dot{\zeta}(t) = A_1 \zeta(t) - \begin{bmatrix}
\bar{\chi}(t) \\
\bar{\sigma}_2(t)
\end{bmatrix}
\]

(34)

\[
e_v(t) = A_3 \zeta(t).
\]

(35)
To examine the evolution of \( \zeta(t) \), we split it into the components \( \zeta_1(t) \) and \( \zeta_2(t) = [\zeta_2(t) \ldots \zeta_{2n-1}(t)]^T \). The dynamics of \( \zeta_1(t) \) are simply

\[
\dot{\zeta}_1(t) = -\gamma \zeta_1(t) - v_1^e \hat{s}(t).
\]

By the bound \(|\hat{s}(t)| \leq \rho_1\), \( \zeta_1(t) \) converges exponentially to a ball of radius

\[
\frac{\rho_1}{\gamma} = \frac{\rho_1}{\kappa \gamma'}
\]

as \( t \to \infty \).

The dynamics of \( \zeta_2(t) \) are given by

\[
\dot{\zeta}_2(t) = D\zeta_2(t) - \begin{bmatrix} 0 \\
\hat{s}_1(t) & 0 & \ldots & 0
\end{bmatrix}
\]

where

\[
D = \begin{bmatrix}
-\gamma I - S^T L_\rho S & S^T L_1 S \\
-S^T L_1 S & 0
\end{bmatrix},
\]

the \((2n-2) \times (2n-2)\) lower right block of \( A_1 \). Differentiating (31), the term \( \dot{\hat{s}}_1(t) \) obeys

\[
|\dot{\hat{s}}_1(t)| \leq \frac{\rho_1}{K_1^2 \lambda_2(L)}.
\]

This derives from \(|\dot{s}(t)| \leq \rho_1\).

By Lemma 1 (see Appendix), the spectrum of \(-D\) satisfies the bound

\[
\text{Re}[\text{spectrum}(-D)] \geq \kappa \beta
\]

where

\[
\beta = \min_\lambda \frac{1}{2} \text{Re} \left[ \gamma' + K_p \lambda - \sqrt{(\gamma' + K_p \lambda)^2 - 4(K_1^2 \lambda_2(L))^2} \right] > 0
\]

with \( \lambda \in \{\lambda_2(L), \lambda_{\text{max}}(L)\} \). Plugging the bounds (39) and (40) into (37), we see that \( \zeta_2(t) \) converges exponentially to a ball of radius

\[
\frac{\rho_1}{\kappa K_1^2 \beta \lambda_2(L)}
\]

as \( t \to \infty \). Combining with the bound (36) and plugging into (35), the result follows.

Theorem 1 is another statement of the properties (P1) and (P2) of networks of PI average consensus estimators. For a bounded rate of change of the inputs, the eventual estimator error can be made arbitrarily small by sufficiently large estimator gains.\(^4\) For the particular case of constant inputs, we have the following corollary.

**Corollary 1:** Consider a fixed, connected network of \( n \) agents implementing PI average consensus estimators of the form (21), where \( K_p, K_1, \gamma > 0 \). For any constant set of inputs \( s \) and any initial states \( \nu(t_0), \eta(t_0) \), the estimator states \( \nu(t) \) and \( \eta(t) \) converge to constant vectors and \( e_v(t) \) converges exponentially to zero as \( t \to \infty \).

The proof follows directly from the proof of Theorem 1.

If the network is switching, each switch introduces a transient to the estimator error. If the input signal is constant, however, these errors become small, as shown in Theorem 2.

\(^4\)In the case of communication delays or a discrete-time implementation, however, the gains must respect upper bounds to ensure stability.

---

**Theorem 2:** Consider a time-varying (but always connected) network of \( n \) agents. The agents implement PI average consensus estimators of the form (21), with gains \( K_p = \kappa K_p', K_1 = \kappa K_1', \gamma = \kappa \gamma' \), where \( K_p, K_1, \gamma > 0 \). For any constant input \( s \) and any \( \beta > 0 \), we can choose \( \kappa > 0 \) sufficiently large that \( e_v(t) \) converges to a ball at the origin of radius \( \beta |s| \) as \( t \to \infty \).

**Proof:** We begin with (28) (where \( A_1(t) \) is now time-varying through the Laplacians), and we split \( \chi(t) \) into components \( \chi_1(t) \) and \( \chi_2(t) = [\chi_2(t) \ldots \chi_{2n-1}(t)]^T \). The dynamics can now be written

\[
\dot{\chi}_1(t) = -\gamma \chi_1(t) + \gamma v_1^e \hat{s}
\]

\[
\begin{bmatrix} \chi_1(t) \\ \dot{\chi}_2(t) \end{bmatrix} = D(t) \begin{bmatrix} \chi_1(t) \\ \dot{\chi}_2(t) \end{bmatrix} + \begin{bmatrix} \gamma S^T s \\ 0 \end{bmatrix},
\]

where \( D(t) \) is defined in (38), but is now time-varying through the Laplacians. This allows us to write the error (29) as

\[
e_v(t) = v_1^e (\chi_1(t) - v_1^e s) + S \chi_2(t).
\]

We examine \( \mu(t) \) and \( S \chi_2(t) \) separately as \( t \to \infty \) to arrive at the result.

The dynamics (43) implies

\[
\chi_1(t) = e^{-\gamma(t-t_0)} [\chi_1(t_0) - v_1^e s] + v_1^e s
\]

\[
e^{-\gamma(t-t_0)} v_1^e [\nu(t_0) - s] + v_1^e s.
\]

We can then write

\[
\mu(t) = e^{-\gamma(t-t_0)} \frac{11^T}{n} [\nu(t_0) - s],
\]

which converges exponentially to 0 as \( t \to 0 \).

Next, we apply Lemma 2 (see Appendix) to (40) to show that \( \chi_2(t) \), and therefore \( S \chi_2(t) \), converges to a ball at the origin of radius proportional to \( |s|/\kappa \), where the constant of proportionality is independent of \( \kappa \). This relies on the fact that \( c_1 I \leq S^T L(t)S \leq c_2 I \) for some \( c_2 > c_1 > 0 \) for all time \( t \) (these bounds are easily determined based on the size of the network \( n \)), and by noticing that large values of \( \alpha \) and \( \zeta \) in Lemma 2 are possible for sufficiently large \( \kappa \).

The network of PI estimators is input-to-state stable (ISS), excluding a single uncontrollable and unobservable scalar state. Therefore, even if the inputs \( s(t) \) and the network Laplacian \( L(t) \) are arbitrarily fast time-varying, a bound on \( |s(t)| \) implies a bound on the error \( |e_v(t)| \).

**Theorem 3:** Consider a time-varying (but always connected) network of \( n \) agents. The agents implement PI average consensus estimators of the form (21), with gains \( K_p, K_1, \gamma > 0 \). Excluding a single uncontrollable and unobservable scalar state which remains constant, the remaining dynamics of (21) are ISS.

**Proof:** The proof follows directly from the dynamics (43)–(44), modified so that \( s(t) \) is time-varying, and again applying Lemma 2 to (44).

In summary, the network of PI dynamic average consensus estimators creates a stable filter from inputs to estimator errors, which also has small tracking error when the inputs and communication network are changing slowly. While the
analysis of this section is for a network with a fixed number of agents $n$, errors introduced by adding or subtracting agents are transient, so PI dynamic average consensus estimators are useful for ad hoc evolving sensor networks.

B. Application of PI Estimators to Environment Modeling

Each mobile sensor simultaneously implements (a) the PI average consensus estimator based on the inputs $\phi_i$ (including measurements), (b) an environmental modeling KF estimator based on the outputs of the PI estimators, and (c) the information-gradient control law (15). In a discrete-time implementation, each agent may execute measurements, average consensus estimator iterations, KF iterations, and applications of the control law at different timesteps, typically with the average estimators running at the highest rate. If this rate is sufficiently high, the PI estimators converge to small errors between Kalman iterations, and the individual decentralized KFs become essentially equivalent to a centralized KF.

We would like to understand the effect of using decentralized estimates of the sums $\bar{C}$ and $\bar{y}$ in (9) and (10) instead of centralized sums. As in Section IV-A, the analysis is conducted in continuous time. We first consider the case of stationary agents with a fixed communication network.

**Theorem 4:** Let the $n$ agents be stationary with a constant connected communication network, and let $F = 0$ in the continuous-time process dynamics (11). Then for any PI average consensus estimator gains $K_P$, $K_I$, $\gamma > 0$, $P_i(t) - P(t)$ converges to zero as $t \to \infty$, where $P(t)$ is the centralized KF covariance and $P_i(t)$ is agent $i$’s decentralized KF covariance estimate. If we further assume that the process inputs are $u = 0$, then agent $i$’s expected environmental estimator error $E[\hat{x}_i(t) - \hat{x}(t)]$ converges to zero as $t \to \infty$.

**Proof:** Let $\hat{C}_i(t)$ denote agent $i$’s PI filter estimate of the centrally computed $C(t)$ in (9). Because $H(t)$ and $R(t)$ are constant for stationary agents, by Corollary 1, each $\hat{C}_i(t)$ converges exponentially to the constant $C = C$. Because $F = 0$ and the model process noise covariance $Q(t)$ is constant, the individual covariance filter (13) of each agent

$$\dot{P}_i(t) = Q - P_i(t)C\hat{C}_i(t)P_i(t)$$

converges to

$$\dot{\hat{P}}_i(t) = Q - P_i(t)C\hat{C}_i(t)P_i(t),$$

with a steady-state value equivalent to that of the centralized filter $\dot{P}(t) = Q - P(t)CP(t).$\(^5\)

If we further assume $u = 0$, then the process evolves according to

$$\dot{x}(t) = w(t), \quad w(t) \sim \mathcal{N}(0, Q),$$

yielding

$$x(t) = x_0 + \int_0^t w(\tau) d\tau = x_0 + g(t).$$

\(^5\)Due to PI average estimator transients, the individual estimates $\hat{C}_i(t)$ may not always be positive semidefinite. When this occurs, to ensure filter convergence, the agent can simply substitute a matrix $C_i(t)$ in its filter equation, obtained by projecting $\hat{C}_i(t)$ onto the convex set of positive semidefinite matrices.

The centralized measurement vector $y(t)$ in (10) is therefore

$$y(t) = H^T R^{-1} H x_0 + H^T R^{-1} (H \varphi(t) + v(t)).$$

Because the decentralized PI average estimators are low-pass linear filters with zero DC error, the approximate measurement vector $\hat{y}_i(t)$ computed by agent $i$ tends to $\hat{y}_i(t) = H^T R^{-1} H x_0 + \varphi_i(t)$ as $t \to \infty$, where $\varphi_i(t)$ is a colored, low-pass filtered version of $H^T R^{-1} (H \varphi(t) + v(t))$. The properties of this low-pass filter are network dependent, but because the filter is linear and the noises $\varphi(t)$ and $v(t)$ are zero mean, we have

$$E[\hat{y}_i(t) - y(t)] = E[\hat{y}_i(t)] - E[y(t)] \to H^T R^{-1} H x_0 - H^T R^{-1} H x_0 = 0 \quad (47)$$

as $t \to \infty$.

Agent $i$’s decentralized estimate $\hat{x}_i(t)$ evolves according to

$$\dot{\hat{x}}_i(t) = P_i(t)(\hat{y}_i(t) - \hat{C}_i(t)\hat{x}_i(t)), \quad (48)$$

where the agent’s estimates $P_i(t)$ and $\hat{C}_i(t)$ were shown above to converge to the centralized values while $E[\hat{y}_i(t)]$ converges to $E[y(t)]$. Therefore, $E[\hat{x}_i(t) - \hat{x}(t)] \to 0$ as $t \to \infty$.

Theorem 4 addresses the case where the environment is driven only by process noise and the sensors are stationary with a fixed communication graph. By the analysis of Section IV-A however, the tracking properties of Theorem 4 degrade smoothly in the case of slowly moving agents, slowly changing communication topologies, and a slowly changing environment. Examples are given in Section V.

V. EXAMPLES

The examples below apply the discrete-time framework to modeling of one-dimensional and two-dimensional domains.

A. Polynomial Basis on a One-Dimensional Domain

Figure 4 shows nine agents modeling an interval $[0, 1]$ of a one-dimensional environment. The agents are stationary and spaced at equal intervals of 0.1, with agent 1 at 0.1, agent 2 at 0.2, etc. The actual environment is described by the equation $g(r) = x^{(1)} + x^{(2)}r + x^{(3)}r^2$, where $x^{(1)} = 1.5, x^{(2)} = -5.5, x^{(3)} = 7$. The agents use the non-orthogonal basis function subset $\{1, r, r^2\}$. Each sensor’s measurement variance is $R_i = 0.1$. The environment is static, so the process model has $F = I_3$ (the $3 \times 3$ identity matrix) and $u = 0$, with zero process noise. The agents’ filters use $0.1I_3$ to model the process noise covariance. Each agent communicates only with adjacent agents and uses a discrete-time PI estimator with $K_P = 0.1, K_I = 0.01$ and $\gamma = 0.05$ to estimate $C$ and $y$. At each iteration, each agent performs one measurement, one update of its $\hat{C}_i$ and $\hat{y}_i$ consensus estimates, and one KF iteration. This guarantees that the consensus estimators have not converged before the KF iteration is executed.

Figure 4 shows the evolution of the environmental model of agent 2 at four different times, beginning from $\hat{r}_2(0) = 0, \bar{y}_2(0) = 0.01I_3$. Each snapshot also shows the variance $\Psi(r)P_2\Psi^T(r)$ in the region of interest, $r \in [0, 1]$. Agent 2’s
model initially fits the environment well in its own vicinity, but not at a distance. As information diffuses from more distant sensors, its model at a distance improves. Despite the transient effects of average consensus estimation, the steady-state estimation of the static environment is identical to that of the centralized estimator. For easy visualization in this example, each sensor exactly measures the environment.

The effect of the consensus estimator dynamics on decentralized Kalman filtering is illustrated in Figure 5. The sensing problem is identical to that in Figure 4, except now the process noise is $Q = 0.1I_3$. The performance of a centralized KF is compared to the evolution of agent 2’s decentralized KF estimate using PI average consensus estimators with three different communication topologies: each agent communicates with all agents (top right), each agent communicates only with agents within a distance of 0.45 (bottom left), and each agent communicates only with adjacent agents (bottom right). Each iteration consists of one consensus estimate update and one KF iteration. While a good estimate of the model is obtained eventually in all cases, the consensus convergence dynamics become more apparent as the communication network becomes sparser and the time constant of information diffusion becomes larger. The low-pass filtering of the measurements due to the diffusion dynamics is evident when comparing the model of the centralized KF to the models of the decentralized KFs.

Figure 6 shows environment estimates for the case where the environment evolves according to $F(k) = G(k) = I_3$, $u(k) = [0, -0.3\sin(0.3k), 0]^T$, and zero process noise. The result is a sinusoidally-varying $x_2$. The agents continue to use $u = 0$ and $Q = 0.1I_3$ for the prediction phase of their local KFs. Figure 6 compares the model of a centralized KF to agent 2’s model using PI average consensus estimators and a communication network with links only between adjacent agents. Despite the use of $u = 0$ in the prediction phase of the KF, the centralized KF’s estimate tracks the environment well (Figure 6 (top left)). Due to the time constant of diffusion of the average consensus estimators, however, agent 2’s estimate cannot keep up with the environment (Figure 6 (top right)). This performance limitation is not unique to diffusive information sharing—any information-sharing scheme (including message-passing) results in performance limitations when information must undergo multi-hop routing, with its associated delays. To reduce the possibility of KF divergence and inconsistency when the environment is changing too quickly to be tracked, safeguards should be implemented to detect when the statistics of the local sensor measurements are inconsistent with the model. In this case, a variety of remedies are available, including artificially adding noise to the model estimate. (See, for example, [4].) Further investigation of this phenomenon is a topic for future work. In general, tracking of a changing environment improves with decreased input frequency, higher network connectivity, or more (and faster) PI filter iterations between KF iterations. This is demonstrated in Figure 6 (bottom), where the PI average estimators are allowed more iterations between KF iterations. The average estimators are nearly converged by the time of the KF iteration, resulting in performance approximating that of a centralized KF.

Figure 7 demonstrates the model improvement obtained by allowing the agents to reposition themselves using the control law (15). In this example, a centralized KF is used to focus on the effect of motion rather than the transient consensus estimator dynamics. Assuming an initial information matrix $Y(0) = 0.1I_3$ and information vector $\tilde{r}(0) = 0$ for all agents, and setting $R_i = 0.5$ and $Q = 0.01I_3$, Figure 7 (top) shows the motion of the agents and the corresponding evolution of the objective function for a control gain $K = 2$ as compared to the objective function for stationary agents. Figure 7 (bottom) shows the spreading out of nine agents that are initially equally
spaced in the interval \([0, 0.1]\) with \(K = 0.1\). Here the utility of motion is even more apparent, as the initial distribution of the sensors is poor for monitoring the interval \([0, 1]\). Note that the agents do not necessarily remain in the interval \([0, 1]\), as the effect of each of the basis functions \(\{1, r, r^2\}\) is global, not localized to the interval. In part for this reason, basis functions that are localized in space, e.g., wavelets, splines, or radial basis functions, are more appropriate choices for environmental models.

**B. Localized Basis Functions on a Two-Dimensional Domain**

One application of decentralized environmental modeling is modeling of marine environments. Current efforts to make improved predictions in marine weather forecasting require ocean metrology updates from remote, possibly mobile, sensing devices [1], [27]. The framework described in this paper can be applied to develop a scalable, decentralized approach to sensor fusion and control of mobile sensors. In this example, we describe an approach to measuring thermal isoclines using spatio-temporal data from the Navy Coastal Ocean Model (NCOM) [2], [24]. NCOM, developed at the Naval Research Lab at Stennis Space Center, uses information from satellite altimetry and sea surface measurements as well as predictions by the Navy Operational Global Atmospheric Prediction System to drive a simulation of ocean salinity and temperature.

Figure 8 shows a temperature image of the surface of the Gulf of Mexico. This is a two-dimensional slice of a three-dimensional snapshot-in-time of ocean temperatures as a function of latitude, longitude, and depth, taken from NCOM.

We used a randomly-placed virtual sensor network of 100 nodes to perform decentralized temperature modeling of a fixed region of the Gulf. Normalizing the length of the edges of the square region to be 1, each node is capable of communicating with others that are within a distance less than 0.3. The resulting network is indicated in Figure 8. The 26 basis functions used to describe the environment are \(\{\psi^{(1)}(u, v), \ldots, \psi^{(26)}(u, v)\}\), where \((u, v)\) are coordinates on the ocean surface and each \(\psi^{(j)}, j = 2 \ldots 26\), is an isotropic Gaussian radial basis function with a standard deviation of 0.3. The centers of the Gaussians are on a \(5 \times 5\) square grid in the modeled region. The influence of the Gaussians at a distance is small, leading to better localization properties than global polynomials. Alternative basis functions could be derived from piecewise splines or multiresolution orthogonal wavelets.

Agents implement discrete-time consensus estimation with
In this paper, we have assumed that each agent measures its own position \( p_i \) exactly. Instead, each agent may maintain an estimate \( \{ \hat{p}_i, \Sigma_i \} \) of its own position, consisting of first and second moments, determined by a separate estimation process (e.g., based on noisy GPS data or measurements of landmarks). Uncertainty in the position \( p_i \) introduces uncertainty in the linear transformation \( H_i = \Psi(p_i) \) in the measurement equation, on top of the sensor noise variance \( R_i \).

While it is possible to directly apply the Kalman filtering approach described above by replacing \( \Psi(p_i) \) with \( \Psi(\hat{p}_i) \), this does not adequately capture the effect of the uncertainty \( \Sigma_i \) in the agent’s position. One problem is that this approach underestimates the uncertainty in the measurement, and therefore the model covariance \( P \) will shrink faster than is justified by the measurement data. A conservative solution to this problem is to assume a larger sensor noise \( R_i \). Another problem, however, is that the measurement-corrected information vector \( \hat{r} \) will be biased by the fact that \( \Psi^T R_i^{-1} z_i \) evaluated at \( (\hat{p}_i, \hat{x}_i) \) is not generally the same as the expected value of this expression evaluated over the joint probability density function (pdf) represented by \( \{ \hat{p}_i, \Sigma_i \} \) and \( \{ \hat{x}_i, P_i \} \).

A better solution is to have each agent replace their inputs to the average consensus estimator, \( H_i^T R_i^{-1} H_i + H_i^T R_i^{-1} z_i \), with the expected quantities \( \mathbb{E}[\Psi^T (p_i) R_i^{-1} (x) \Psi(p_i)] \) and \( \mathbb{E}[\Psi^T (p_i) R_i^{-1} (x) z_i] \), respectively, over the pdfs given by \( \{ \hat{x}_i, P_i \} \) and \( \{ \hat{p}_i, \Sigma_i \} \). Here the sensor covariance \( R_i \) is potentially a function of the environment \( x \). These expected quantities can be calculated by means of the unscented transform, originally developed to achieve higher-order accuracy in the nonlinear transformation of pdfs, beyond the accuracy obtained by linearizing the transformation at the mean of the estimate [21]–[23]. The idea is to choose a representative set of “sigma points” \( \mathcal{V} \) from the joint pdf describing the agent’s position, environmental model, and sensor noise. This set \( \mathcal{V} \) has the same mean and covariance as the joint pdf.

We calculate the weighted expected covariance matrix and weighted expected measurement vector over all \( v \in \mathcal{V} \) to replace \( H_i^T R_i^{-1} H_i + H_i^T R_i^{-1} z_i \) in (9) and (10). For details on choosing sigma points and weights, see [23].

This improved filtering in the case of uncertainty in agent position can still be implemented in a decentralized fashion. Information-gradient-following control can be derived as before, by replacing the term \( \partial C / \partial p_i \) with

\[
\frac{\partial}{\partial p_i} \mathbb{E}[\Psi^T (p_i) R_i^{-1} \Psi(p_i)].
\]

Other issues for future work are summarized below. A number of these arise from the coupling of motion control with the Kalman filter and the dynamics of the consensus estimator.

- Determine average consensus estimator gains \( K_P, K_I, \gamma \), as a function of the network topology, that ensure both filter stability to communication time delays as well as fast convergence. Agents can employ different gains to improve the stability and performance measures. Centralized solutions to simplified versions of this problem are presented in [47], [49].
- Extend the analysis of PI dynamic average consensus estimators to the case where the network is connected...
over time intervals, without necessarily being connected at any instant.

- Determine bounds on motion control gains $K$ guaranteeing stability of the coupled system considering the latency effects of communication delays and the PI consensus estimator dynamics.
- Project the control calculated by (15) onto the subset of controls that satisfies auxiliary conditions, such as agent-agent collision avoidance, and incorporate any vehicle underactuation or nonholonomic constraints.
- Use a priori models of the environment, as well as specifications on the allowable environmental monitoring errors, to automatically choose an environmental parameterization.
- Incorporate motion costs in the objective function, as a tradeoff between information acquisition and energy usage.
- Implement safeguards to detect and prevent decentralized KF divergence and inconsistency when the environment changes faster than can be supported by the time constant of decentralized average consensus.
- Extend the estimate-and-control approach to control of the environmental variable, as in herding of adversarial or neutral agents.

**Appendix**

**Lemma 1:** Consider a block matrix

$$D = \begin{bmatrix} A & -B \\ B & 0 \end{bmatrix}$$

where $A, B \in \mathbb{R}^{n \times n}$ are real, symmetric, and positive definite (i.e., $A$ and $B$ have positive real eigenvalues), and $A$ and $B$ commute ($AB = BA$). Let $\lambda_{A,\min}$ and $\lambda_{B,\min}$ be the minimum eigenvalues of $A$ and $B$, respectively, and $\lambda_{A,\max}$ and $\lambda_{B,\max}$ be the maximum eigenvalues. Then any eigenvalue $\lambda$ of $D$ satisfies the following bounds:

1. $\Re(\lambda) \geq \min \Re \left( \frac{1}{2} \left( \lambda_A - \sqrt{\lambda_A^2 - 4\lambda_B^2} \right) \right)$

where $\lambda_A \in \{\lambda_{A,\min}, \lambda_{A,\max}\}$

2. $\Re(\lambda) \leq \Re \left( \frac{1}{2} \left( \lambda_{A,\max} + \sqrt{\lambda_{A,\max}^2 - 4\lambda_{B,\min}^2} \right) \right)$

3. $\Im(\lambda) \leq \Im \left( \frac{1}{2} \sqrt{\lambda_{A,\min}^2 - 4\lambda_{B,\min}^2} \right)$

In particular, bound (i) implies $\Re(\lambda) > 0$.

**Proof:** The eigenvalues $\lambda$ of $D$ satisfy

$$\det \begin{bmatrix} A - \lambda I & -B \\ B & -\lambda I \end{bmatrix} = 0.$$ 

Because $B$ and $-\lambda I$ commute, this determinant can be simplified to

$$\det((A - \lambda I)(-\lambda I) - B(-B)) = \det[\lambda^2 I - \lambda A + B^2].$$

(49)

Because $A$ and $B$ commute, we can define a single similarity transform

$$\hat{A} = S^T AS, \quad \hat{B} = S^T BS$$

such that $\hat{A}$ and $\hat{B}$ are diagonal, with the eigenvalues of $A$ and $B$ along the diagonal. Applying this similarity transform to each term in the determinant of (49), we get

$$\det[\lambda^2 I - \lambda A + B^2] = \det[\lambda^2 S^T IS - S^T AS + S^T B^2 S] = \det[\lambda^2 I - \lambda \hat{A} + \hat{B}^2].$$

Because the matrix is diagonal, each root of

$$\det[\lambda^2 I - \lambda \hat{A} + \hat{B}^2] = 0$$

satisfies

$$\lambda^2 - \hat{\alpha}_{ii} \lambda + \hat{\beta}_{ii}^2 = 0$$

for some $i \in \{1 \ldots n\}$, and therefore has the form

$$\lambda = \frac{1}{2} \left( \hat{\alpha}_{ii} \pm \sqrt{\hat{\alpha}_{ii}^2 - 4\hat{\beta}_{ii}^2} \right) \cdot$$

(50)

The proof is completed by considering all possible combinations of one eigenvalue of $A$ and one eigenvalue of $B$ in (50).

**Lemma 2:** Let $A, B : \mathbb{R} \rightarrow \mathbb{R}^{p \times p}$ be piecewise-continuous and such that

$$\|A(t)\| \leq \varepsilon_1, \quad A(t) + A^T(t) \leq -2\varepsilon_2 I, \quad \text{and}$$

$$2\rho_1 I \leq B(t) + B^T(t) \leq \rho_2 I$$

(51)

for all $t \in \mathbb{R}$ and for some constants $\varepsilon_1, \varepsilon_2, \rho_1, \rho_2 > 0$. If we define the positive constant

$$\sigma \triangleq \frac{\rho_1 \varepsilon_2}{\varepsilon_1 + \rho_1 \rho_2} \cdot$$

(52)

then $\sigma \leq \sqrt{2}/4$ and the matrices

$$F(t) = \begin{bmatrix} A(t) & B^T(t) \\ -B(t) & 0 \end{bmatrix}, \quad P = \begin{bmatrix} I & -\sigma I \\ -\sigma I & I \end{bmatrix}, \quad \text{and}$$

$$Q = \begin{bmatrix} \varepsilon_2 I & 0 \\ 0 & \sigma \rho_1 I \end{bmatrix}$$

(53)

satisfy

$$(1 - \sigma)I \leq P \leq (1 + \sigma)I$$

(54)

$$PF(t) + F^T(t)P + Q \leq 0$$

(55)

for all $t \in \mathbb{R}$. Furthermore, for any $t_0 \in \mathbb{R}$, the state $\chi(t)$ of the system

$$\chi(t) = F(t)\chi(t) + G\nu(t), \quad G = \begin{bmatrix} I \\ 0 \end{bmatrix}$$

(56)

with bounded, piecewise-$C^0$ input $\nu : \mathbb{R} \rightarrow \mathbb{R}^p$ satisfies

$$|\chi(t)|^2 \leq \frac{1 + \alpha}{1 - \sigma} \left( e^{-\alpha(t-t_0)}|\chi(t_0)|^2 + \frac{1}{\alpha^2} \right)$$

(57)

for all $t \geq t_0$, where $\alpha$ can be any constant chosen such that $0 < \alpha < \min\{\varepsilon_2, \rho_1\}$, and

$$\alpha \triangleq \frac{1}{1 + \sigma} \min\{\varepsilon_2 - \varepsilon_1, \sigma(\rho_1 - \varepsilon_1)\} > 0.$$  

(58)

In particular, the system (56) is ISS.

**Proof:** The inequalities (51) imply $\varepsilon_2 \leq \varepsilon_1$ and $2\rho_1 \leq \rho_2$, which means

$$\sigma \leq \frac{\rho_1 \varepsilon_2}{\varepsilon_1^2 + 2\rho_1^2} \leq \frac{\sqrt{2}}{4} \cdot$$

(59)
where the rightmost inequality follows from the fact that the function \( x \rightarrow x/(1 + 2x^2) \) has a maximum value of \( \sqrt{2}/4 \).

The inequalities (54) are easily verified, and

\[
P F(t) + F^T(t)P + Q = \begin{bmatrix} A + A^T + \sigma(B + B^T) & -\sigma A^T \\ -\sigma A & -\sigma(B + B^T) + \sigma \rho_1 I \end{bmatrix} \leq -\sigma \cdot \left[ \begin{bmatrix} \left( \frac{1}{\sigma} \varepsilon_2 - \rho_2 \right) I \\ A(t) \end{bmatrix} \rho_1 I \right] \leq 0
\]

because (52) implies that \( R(\sigma, t) \geq 0 \) for all \( t \in \mathbb{R} \). Next,

\[
PGG^TP = \begin{bmatrix} I & -\sigma I \\ -\sigma I & \sigma^2 I \end{bmatrix} = P - \begin{bmatrix} 0 & 0 \\ 0 & (1 - \sigma^2)I \end{bmatrix}
\]

and

\[
Q - \frac{c}{1 + \sigma} PGG^TP = \begin{bmatrix} \varepsilon_2 I & 0 \\ 0 & \sigma \rho_1 + \varsigma(1 - \sigma) \end{bmatrix} \leq \begin{bmatrix} -\varsigma/P \\ 1 + \sigma \end{bmatrix} P \geq \min \{ \varepsilon_2, 1 \} I - \frac{\varsigma}{1 + \sigma} P + \frac{\varsigma}{1 + \sigma} \min \{ \varepsilon_2, \sigma \rho_1 + \varsigma(1 - \sigma) \} P
\]

It follows that \( V(t) = \chi^t(t) P \chi(t) \) satisfies

\[
V(t) = \chi^t(t) \begin{bmatrix} PF(t) + F^T(t)P & \chi(t) + 2 \chi^t(t)PGPG^TP \chi(t) \\ 2\chi^t(t)PGPG^TP \chi(t) & \chi(t) + 2 \chi^t(t)PGPG^TP \chi(t) \end{bmatrix} \leq \chi^t(t) \begin{bmatrix} 1 + \sigma \|
u(t)\|^2 \\ 1 + 2\sigma \|
u(t)\|^2 \end{bmatrix} \leq -\alpha V(t) + \frac{1 + \sigma}{\varsigma} \|
u(t)\|^2 \leq -\alpha V(t) + \frac{1 + \sigma}{\varsigma} \|
u\|^2_\infty
\]

Hence from the comparison principle we obtain

\[
V(t) \leq e^{-\alpha(t-t_0)} \left[ V(t_0) - \frac{1 + \sigma}{\alpha \varsigma} \|
u\|^2_\infty \right] + \frac{1 + \sigma}{\alpha \varsigma} \|
u\|^2_\infty
\]

from which (57) follows.

Note that to obtain the tightest bound in (57), we would choose \( \varsigma \) to maximize \( \alpha \varsigma \).

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their detailed comments and suggestions for improving the paper, and Eugene Kushnirsky for his assistance with Lemma 1.

REFERENCES


