Davide Spinello1
Department of Mechanical Engineering,
University of Ottawa,
Ottawa, ON K1N 6N5, Canada
e-mail: dspinello@uottawa.ca

Daniel J. Stilwell
The Bradley Department of Electrical &
Computer Engineering,
Virginia Polytechnic Institute & State University,
Blacksburg, VA 24060
e-mail: stilwell@vt.edu

1Introduction

Cooperative sensing of an object is one of the many proposed capabilities for mobile sensor networks [1–3]. Mobility, coupled with data fusion algorithms and motion control algorithms, enables a team of mobile sensors to adjust their relative geometry in real-time to enhance sensing performance. These ideas have been applied to a variety of domains, including target tracking [4–15], formation and coverage control [16–27], rendezvous and deployment [28], and environmental tracking and monitoring [29–32]. A recent, comprehensive review can be found in Ref. [3], and several technical aspects are discussed in the book [33] from the same authors. We extend these ideas to the problem of localizing an acoustic source using a team of mobile sensors with limited communication capabilities, as it occurs, for example, in underwater applications with autonomous vehicles. We address the challenge of limited communication by developing a fully decentralized motion planning algorithm for applications in which communication between sensor nodes is infrequent. A simplified version of the algorithm proposed here has been implemented in a platoon of autonomous underwater vehicles equipped with arrays of hydrophones to measure the bearing angle with respect to an acoustic source [34], cooperatively estimating the state of a static source from noisy measurements. The motion control algorithm is based on the constraint that within finite time intervals all agents mutually exchange suitable information. Neighbor rules are therefore not explicitly considered as the algorithmic framework is motivated by applications with a limited number of agents involved and with low reliability of data sharing operations.

Cooperative object localization with limited communication is addressed in Refs. [8,35] by coupling estimation and motion control algorithms with consensus filters to achieve asymptotic agreement between agents. A similar idea appears in Ref. [30] for environmental tracking applications. Different cooperative tasks in networked systems with limited and/or time varying communication have been investigated in the framework of consensus and synchronization [36–45]. In this work, we seek to develop approaches that are effective when communication between agents is highly restricted, for example in underwater environments. We consider a new class of distributed motion control algorithms developed to operate with only occasional communication between the vehicles. To enable coordination despite limited communication, we embed a local observer on board each agent to estimate states of the other agents in the team. The local observer used herein generalizes the observer proposed in Ref. [46] to sensor networks with limited communication. Each agent implements a centralized gradient-based control law, but with locally estimated variables in the place of state variables that would be communicated directly from other agents in a centralized implementation. Asymptotic convergence of the estimators allows for the implementation of a system which is asymptotically equivalent to the one proposed in Ref. [5].

In works that address control of sensor motion in mobile sensor networks, the estimation problem is commonly solved by assuming that the observation noise is independent of the process state, see, for example, Refs. [5,7,8,47]. However, as pointed out in Ref. [48], this assumption is not realistic for classes of sensors commonly used in applications, such as bearing-only and range-only sensors. In order to account for state dependent measurement noise in the illustration presented herein, we utilize a generalized extended Kalman filter that is proposed in Ref. [49]. By using the same approach, an analogous class of extended Kalman filters has been developed in Ref. [50].

The rest of the paper is organized as follows. In Sec. 2, we formulate the problem in which a team of mobile sensors estimates a dynamical process’ state form noisy measurements. The main contribution of the paper is in Sec. 3, where we present a class of distributed estimators that solve the motion control problem for the team of mobile sensors in the case of infrequent communication between agents. We focus on the task of cooperative target localization. However, the methodology is not restricted to this class of problems. In Sec. 4, we briefly describe the extended Kalman filter equations derived in Ref. [49] for nonlinear estimation with state dependent observation noise. The extended Kalman filter is coupled with the motion control algorithm to solve the cooperative target localization problem. In Sec. 5, we illustrate the methodology through simulations, by considering an example with three mobile sensors cooperatively localizing an acoustic source with a circulant communication structure. Conclusions are summarized in Sec. 6.

2Problem Description

We consider a group of \( N \) mobile agents with states \( q_i \in \mathbb{R}^7, i = 1, \ldots, N \). The agents acquire noisy measurements of a
stochastic dynamical process, mutually exchange information and maintain local estimates of the state of the process that in turn determines the motion of the agents. Therefore, we consider a coupled motion control and estimation problem, with framework and notations defined below.

2.1 Target State Transition. The $N$ agents (mobile sensors) are modeled as point masses with states $q_i \in \mathbb{R}^i, i = 1, \ldots, N$. The objective of the team of $N$ agents is to configure themselves spatially in order to obtain the best estimate of the state $x \in \mathbb{R}^x$ of a dynamical process, whose evolution is described by a nonlinear discrete-time stochastic state transition equation, see, for example, Ref. [51]

$$ x[k + 1] = f(x[k]) + \nu[k] \quad (1) $$

where $f : \mathbb{R}^x \to \mathbb{R}^x$ is the possibly nonlinear state transition function, and $\nu \in \mathbb{R}^x$ is an additive process noise. We assume that $\nu$ is a sequence of independent random variables with normal probability distribution $\mathcal{N}(0, Q)$, therefore satisfying the relations

$$ E[\nu[k]] = 0 \quad \forall k \quad (2a) $$

$$ E[\nu[k]\nu^T[l]] = Q[k]\delta_{kl} \quad \forall k, l \quad (2b) $$

where $E[\cdot]$ is the expectation operator, $Q$ is the target’s noise covariance matrix, $\delta_{kl}$ is the Kronecker delta.

2.2 Observation Model. Each sensor generates measurements of the system (1) at discrete time instants. The observation made at time $k$ by sensor $i$ is

$$ z_i[k] = h_i(q_i[k], x[k]) + v_i(q_i[k], x[k]) \quad (3) $$

where $z_i[k] \in \mathbb{R}^s$, with $p \leq s$, $h_i$ is the observation function of the model, and the noise $v_i$ is a sequence of independent random variables with normal probability distribution $\mathcal{N}(0, \sigma_i(q_i, x))$, where $\sigma_i \in \mathbb{R}^{s \times \rho}$ is the symmetric positive-definite observation noise covariance matrix. We emphasize that $\sigma_i$ depends on the state $x$. We have chosen to explicitly show state-dependent noise since it arises in many applications, including the underwater acoustic sensor application that motivates this example, and because in a gradient-descent framework, correctly accounting for state-dependent noise in computing gradients can make a dramatic difference in performance.

The measurement $z_i[k]$ can be treated as the realization of a multivariate normal distribution described by the conditional moments

$$ E[z_i[k]|x[k]] = h_i(q_i[k], x[k]) \quad (4a) $$

$$ E[(z_i[k] - h_i(q_i[k], x[k]))(z_i[k] - h_i(q_i[k], x[k]))^T] = \sigma_i(q_i[k], x[k]) \quad (4b) $$

We assume that noise terms are time- and space-uncorrelated, that is

$$ E[v_i(q_i[k], x[k])v_i^T(q_i[l], x[l])] = \sigma(q_i[k], x[k])\delta_{kl}\delta_{ij} \quad \forall k, l, \forall i, j \quad (5) $$

Additionally, we assume the following cross-correlation independence condition [51]

$$ E[\nu[k]\nu_i^T(q_i[l], x[l])] = 0, \quad \forall k, l \quad (6) $$

2.3 Group Trajectories. We introduce the collection of states

$$ \mathbb{R}^{N} \ni q = (q_1, \ldots, q_N)^T \quad (7) $$

The objective of the team of $N$ agents is encoded by the cost function

$$ \mathbb{R}^{N} \ni q \mapsto J(q) \in \mathbb{R} \quad (8) $$

which is required to satisfy the following conditions [52]

Non-negativity:

$$ J(q) \geq 0 \quad \forall q \in \mathbb{R}^{N} \quad (9a) $$

Lipschitz continuity of $\nabla J$:

The function $J$ is continuously differentiable and

$$ \|\nabla J(q_1) - \nabla J(q_2)\| \leq \kappa_p\|q_1 - q_2\| \quad (9b) $$

for all $q_1, q_2 \in \mathbb{R}^{N}$, and some positive constant $\kappa_p$.

A standard approach to solve this class of problems is to apply a gradient descent algorithm to actively control the agents. For a detailed discussion of gradient descent algorithms and convergence properties, see Ref. [52, Section 3.2]. Individual control inputs $u_i$ for the states $q_i$ are obtained by

$$ u_i(q) = -\Gamma_i\nabla_J(q) \quad (10) $$

where $\Gamma_i \in \mathbb{R}^{s_i \times s_i}$ is a control gain matrix and $\nabla_J(q)$ is the gradient in the direction $q_i$. Note that the control inputs depend on the collection of states $q$, and therefore individual actions are affected by the behavior of the group.

We consider the discrete-time framework. The evolution equation for the state $q_i$ of agent $i$ is therefore given by the gradient-descent update, see Ref. [52]

$$ q_i[k + 1] = q_i[k] + Tu_i(q_i[k]) \quad (11) $$

where $T$ is the updating time interval and $k$ is an integer that labels the time instant. Similarly, the gradient descent update for the collection of states $q$ is given by

$$ q[k + 1] = q[k] + Tu(q[k]) \quad (12a) $$

$$ u(q) = -\Gamma\nabla J(q) \quad (12b) $$

where $\Gamma \in \mathbb{R}^{s \times N}$ and $\nabla J \in \mathbb{R}^{s}$ are defined by

$$ \Gamma = \text{diag}(\Gamma_1, \ldots, \Gamma_N) \quad (13a) $$

$$ \nabla J = (\nabla q_1 J \cdots \nabla q_N J) \quad (13b) $$

We define the output (instrumented measurements) of agent $i$ to be $y_i = \gamma(q_i) \in \mathbb{R}^{q_i}$, where $\gamma : \mathbb{R}^x \to \mathbb{R}^{q_i}$ is a nonlinear function. By introducing the function

$$ g(q) = (\gamma(q_1) \cdots \gamma(q_N))^T \quad (14) $$

we write the output of the group of agents at time $k$ as

$$ y[k] = g(q[k]) \in \mathbb{R}^{qN} \quad (15) $$

3 Full-State Observer

3.1 Full State Estimators. Note that $J$ in Eq. (8) incorporates information from all vehicles at each time step, not just from those vehicles that communicate during that time step. When communication only occurs intermittently, sensor $i$ does not have all the necessary information to compute $u_i$. To overcome this problem, we embed a local observer on each sensor to maintain a local estimate of external sensor states. In general, observer-based decentralized control is fraught with difficulties since there is no separation principle for decentralized systems. However, in the
possible snapshot of the networked systems considered here are each node has a loop and therefore degree \( \frac{1}{2} \).

Let \( \hat{q}_i \) be the estimation of agent \( i \) embedded in agent \( i \). The collection of estimates embedded in agent \( i \) is

\[
\hat{q}_i := (\hat{q}_{i1}, \hat{q}_{i2}, \ldots, \hat{q}_{iN})^T \in \mathbb{R}^N
\]

(16)

The estimate of the state of agent \( j \) embedded in agent \( i \) is updated using knowledge of control gains and output measurement (when communicated) of agent \( j \) through the estimation algorithm

\[
\hat{q}_i[k + 1] = \hat{q}_i[k] + T(u_i(\hat{q}_i[k]) + K_i(\hat{q}_i[k])a_{ij}[k]y_j[k] - y_i[k])
\]

(17)

where \( K_i \in \mathbb{R}^{dN \times d^2} \) is a gain matrix that depend in general on the estimate \( \hat{q}_i \), and \( y_j \) is output from agent \( j \) that is occasionally communicated to agent \( i \). The estimated output is given by

\[
y_j = \gamma(\hat{q}_j)
\]

and \( a_{ij} \in \mathbb{R}^{dN \times d^2} \) is defined as

\[
a_{ij} = \begin{cases} 
I_d & \text{if agent } j \text{ communicates to sensor } i \\
0_d & \text{otherwise}
\end{cases}
\]

(19)

where \( \ell > 0 \) is a scalar constant such that Eq. (19) is well defined in terms of physical dimensions. We assume that \( a_{ij} \in \mathbb{R}^d \) for all \( k \), implying that every agent has always access to the most updated version of its state, or otherwise that the estimators \( \hat{q}_i \) coincide with the true states \( q_i \), for \( i = 1, \ldots, N \). Therefore, the underlying communication graph has no isolated nodes, since each node has a loop and therefore degree \( \geq 2 \). Symbols and a possible snapshot of the networked systems considered here are schematically represented in Fig. 1.

We collect the gain matrices \( K_i \) in diagonal form and define the \( sN \times dN \) (\( N^2 \) blocks) matrix

\[
K_i(\hat{q}_i) = \text{diag}(K_{i1}(\hat{q}_1), \ldots, K_{iN}(\hat{q}_N))
\]

(20)

and write the estimation algorithm embedded in agent \( i \) as

\[
\hat{q}_i[k + 1] = \hat{q}_i[k] + T(u_i(\hat{q}_i[k]) + K_i(\hat{q}_i[k])A_i[k](g(\hat{q}_i[k]) - g(q_i[k])))
\]

(21)

where \( g : \mathbb{R}^{dN} \to \mathbb{R}^{dN} \) is defined in Eq. (14), the \( sN \) input vector \( u \) is defined in Eq. (12b), and

\[
A_i := \text{diag}(a_{i1}, a_{i2}, \ldots, a_{iN}) \in \mathbb{R}^{dN \times dN}
\]

(22)

For the case of a complete and undirected communication network we have \( A_i = I_d \).

By collecting all the state estimates, the outputs and the inputs into the following vectors

\[
\hat{q}_i = (\hat{q}_{i1}, \hat{q}_{i2}, \ldots, \hat{q}_{iN})^T \in \mathbb{R}^{dN}
\]

(23)

\[
g(\hat{q}) = (g(\hat{q}_1), g(\hat{q}_2), \ldots, g(\hat{q}_N))^T \in \mathbb{R}^{dN}
\]

(24)

\[
u(\hat{q}) = (u(\hat{q}_1), u(\hat{q}_2), \ldots, u(\hat{q}_N))^T \in \mathbb{R}^{dN}
\]

(25)

and by defining the block diagonal \( sN^2 \times dN^2 \) (\( N^2 \) blocks) gain matrix on the matrices in Eq. (20)

\[
K(\hat{q}) = \text{diag}(K_{11}(\hat{q}_1), \ldots, K_{NN}(\hat{q}_N))
\]

(26)

we write the collective estimation algorithm as

\[
\hat{q}[k + 1] = \hat{q}[k] + T(u(\hat{q}[k]) + K(\hat{q}[k])A[k](g(\hat{q}[k]) - I_N \otimes g(\hat{q}[k])))
\]

(27)

where \( I_N \in \mathbb{R}^{N} \) is the vector with all entries equal to one, \( \otimes \) is the Kronecker product operator, see Ref. [53], and

\[
A = \text{diag}(A_1, A_2, \ldots, A_N) \in \mathbb{R}^{d^2 \times d^2}
\]

(28)

The estimator (27) is an asymptotic observer for the system (12) if \( \hat{q}_i[k] \to I_N \otimes q_i[k] \) as \( k \to \infty \). As an immediate consequence of the block-diagonal structure of Eq. (27), the following statement holds:

**Proposition 3.1.** The algorithm (27) is an asymptotic observer for the system (12) if and only if each estimator \( \hat{q}_i[k] \) in Eq. (21), with \( i = 1, \ldots, N \), is an asymptotic observer for Eq. (12). That is, \( \hat{q}_i[k] \to q_i[k] \) as \( k \to \infty \).

### 3.2 Estimator Gains

In order to characterize the estimator gains, we introduce the \( sN \times dN \) matrix

\[
G(\hat{q}) = \begin{pmatrix}
\nabla_{\hat{q}_1}^c(\hat{q}_1) & 0_{sxd} & 0_{sxd} & 0_{sxd} \\
0_{sxd} & \nabla_{\hat{q}_2}^c(\hat{q}_2) & 0_{sxd} & 0_{sxd} \\
\vdots & \vdots & \ddots & \vdots \\
0_{sxd} & 0_{sxd} & \nabla_{\hat{q}_N}^c(\hat{q}_N)
\end{pmatrix}
\]

(29)

If the gains \( K_{ij} \) in Eq. (17) are set to

\[
K_{ij}(\hat{q}_i) = -\Gamma \nabla_{\hat{q}_i}^c(\hat{q}_i)
\]

(30)

then the estimation algorithm (21) can be expressed as

\[
\hat{q}_{i[k + 1]} = \hat{q}_{i[k]} + T w(\hat{q}_i[k], q_i[k])
\]

(31)

where

\[
w(q_i, q_i) = u(\hat{q}_i) - \Gamma G(\hat{q}_i)A_i[k](g(\hat{q}_i) - g(q_i))
\]

(32)

with \( \Gamma \) defined in Eq. (13a) and \( A_i[k] \) in Eq. (22). The basic idea of the gains introduced in Eq. (30) is to correct the estimators \( \hat{q}_i \) in a direction along which the squared error function...
Thus, by setting the observer gains as in Eq. (30), the updating implies that the estimator (21) evolves as the gradient flow of the matrix $A_i$ using definition (29), we have

$$F : \mathbb{R}^{N_s} \times \mathbb{R}^{N_s} \ni (\mathbf{q}, \mathbf{q}) \mapsto \mathbf{F}(\mathbf{q}, \mathbf{q}) \in \mathbb{R}$$

such that

$$w(\mathbf{q}, \mathbf{q}) = -\mathbf{\Gamma} \nabla \mathbf{F}(\mathbf{q}, \mathbf{q})$$

implying that the estimator (21) evolves as the gradient flow of the cost function $F$.

**Proof.** Let

$$F(\mathbf{q}, \mathbf{q}) = \mathbf{J}(\mathbf{q}) + \frac{1}{2}(g(\mathbf{q}) - g(\mathbf{q}))^T A_i (g(\mathbf{q}) - g(\mathbf{q}))$$

where $\mathbf{J}$ is the cost function associated to the system (12). The function $F$ is therefore non-negative because the diagonal matrix $A_i$ is semipositive definite (diagonal entries are either 0 or $\ell$). By using definition (29), we have

$$\nabla \mathbf{F}(\mathbf{q}, \mathbf{q}) = \nabla \mathbf{J}(\mathbf{q}) + G_i (\mathbf{q}) A_i (g(\mathbf{q}) - g(\mathbf{q}))$$

Thus, by setting the observer gains as in Eq. (30), the updating direction $w$ in Eq. (31) is given by

$$w(\mathbf{q}, \mathbf{q}) = -\mathbf{\Gamma} \nabla \mathbf{F}(\mathbf{q}, \mathbf{q})$$

with the $j$th entry of the $sN$ vector $\nabla \mathbf{F}$ given by

$$[\nabla \mathbf{F}]_j(\mathbf{q}, \mathbf{q}) = \nabla_{q_j} \mathbf{J}(\mathbf{q}) + \nabla_{q_i} \mathbf{g}(\mathbf{q}) a_{ij} (\mathbf{q}) - \gamma(\mathbf{q})$$

Note that, in view of Eq. (36), the physical dimensions of $\ell$ in Eq. (19) equal the ratio between the physical dimensions of $J$ and the square of the physical dimensions of the output function $\gamma$.

### 3.3 Convergence Analysis

Convergence of nonlinear observers is very difficult to prove in general. In Ref. [46], a local convergence result has been obtained for a time invariant communication network with the output injection term designed as the gradient descent of a quadratic error term. Here, we consider a time-varying communication network to handle the case of in-frequent communication.

#### 3.3.1 Gradient Descent Algorithms

For completeness and readability, we present a brief account of the results in Ref. [52, Section 3.2] concerning gradient optimization algorithms, adapted to the case studied herein. Proofs are given in Appendix B.

Let $F : \mathbb{R}^{N_s} \times \mathbb{R}^{N_s}$ be a scalar function, satisfying

1. (Positive definiteness):

   $$F(\mathbf{q}, \mathbf{q}) \geq 0 \quad \forall \mathbf{q}, \mathbf{q} \in \mathbb{R}^{N_s}$$

2. (Lipschitz continuity of $\nabla F$). For every $k$, the function $F$ is continuously differentiable with respect to its arguments, and there exists constants $k, \ell$ such that

   $$||\nabla \mathbf{F}(\mathbf{q}, \mathbf{q}) - \nabla \mathbf{F}(\mathbf{\tilde{q}}, \mathbf{\tilde{s}})|| \leq k||\mathbf{q} - \mathbf{\tilde{q}}||$$

A key consequence of assumption (41) is the descent Lemma.

**Lemma 3.3.** (Descent Lemma, adapted from Ref. [52, Proposition A.2.2]) Under assumption (41) we have

$$F(\mathbf{q} + \mathbf{\epsilon}, \mathbf{q} + \mathbf{\epsilon}) \leq F(\mathbf{q}, \mathbf{q}) + \nabla \mathbf{F}(\mathbf{q}, \mathbf{q})^T \mathbf{\epsilon} + \frac{1}{2} k||\mathbf{\epsilon}||^2$$

The following statement follows from Proposition 3.4 and it is used in the proof of Proposition 3.6. The proof is given in Appendix B.
3.3.2 Global Results. The quadratic error function (33) has a minimum at $E_i = 0$ with multiplicity given by the dimension of the kernel of the matrix $A_i$. For a complete communication network in which each sensor broadcasts its information to all the other sensors at each updating time, the matrix $A_i$ is full rank for each $k$ and therefore the global minimum $E_i = 0$ is achieved for $g(q) = g(q)$. For more general communication structures in which the matrix $A_i[k]$ is time varying and not full rank, $E_i = 0$ does not in general imply asymptotic convergence of the output. In this case, asymptotic convergence of the output requires some structure on the communication network.

Here, we consider $a_{ij}$ to be proportional to the identity when a link between $i$ and $j$ exists; global asymptotic results are not affected by replacing the identity with a positive definite matrix, so that $A_i$ is semidefinite and $E_i$ is still a seminorm. However, in the spirit of modeling the existence of intermittent communication links between pairs of nodes, we consider $a_{ij}$ as defined in Eq. (19). In the control law, the multiplication by a generic positive definite control feedback gain matrix is equivalent to the introduction of a generalized semidefinite matrix $A_i$.

Our approach is to assume that there is a finite time during which all agents eventually communicate to all other agents, and then we show convergence to zero of the error in the output estimate. The utility of this approach is that it allows us to consider network topologies that are time-varying and perhaps never connected in a frozen-time sense [54].

Let $\tau_j[k]$ be an integer that labels the closest past time instant, with respect to $k$, at which agent $i$ has received information from agent $j$. In this work, we consider communication events that are characterized by finite time intervals within which each agent sends at least one data packet to all the other agents in the group, since we are interested in modeling small groups of mobile vehicles where only the time dependence of communication events is relevant [34]. More general interactions that include local neighbor rules and numerosity are addressed, among others, in Ref. [40,55,56]. Communication events are deterministically characterized in the framework of partially asynchronous networks as introduced by Ref. [52]. This is formalized by the assumption that, for all $i$ and $j$, and for all $k \geq 0$, there exists a finite integer $k \geq 1$ such that

$$\max\{0, k - \tilde{k} + 1\} \leq \tau_j[k] \leq k$$

Assumption (55) means that within a finite time interval, each agent in the network shares his information at least once. The partial asynchronous characterization is deterministic and it is a simple way to characterize the evolution of communication events. Such evolution can be generalized for example by considering $\tau_j$ to be the element of an evolving random variable with given probability distribution, therefore defining a stochastically evolving process. In this case, the convergence of the error $E_i$ has to be addressed in terms of appropriate moments [40].

**Proposition 3.6.** Consider the cost function (36) and the sequence $\{q[k] \}$ generated by algorithm (31). Under the hypothesis (55) on the communication network, the estimated output $g(q[k])$, for all $i \in \{1, \ldots, N\}$, asymptotically converges to the true output $g(q[k])$, that is

$$\lim_{k \to \infty} g(q[k]) = g(q[k])$$

**Proof.** From assumptions (9) on function $J$ and from definition (36), function $F$ satisfies conditions (40) and (41). Therefore, the result in Proposition 3.4 holds and we have

$$\lim_{k \to \infty} \nabla_q F(q[k], q[k]) = 0$$

which implies that the asymptotic value of the estimator $q_i$ minimizes $F$. Moreover, from Proposition 3.5, we have

$$\lim_{k \to \infty} E_i(q[k], q[k]) = 0$$

and therefore the gradient descent iterations (31) decrease the error function $E_i$ in Eq. (33) along a direction such that the asymptotic value of the state $q_i$ corresponds to a minimum of $E_i$. Furthermore, since $E_i$ is a semipositive definite quadratic form, the set of all the minima is given by $E_i = 0$, that is, the sequence $\{q_i[k]\}$ associated to gradient descent iterations (31) is such that

$$\lim_{k \to \infty} E_i(q[k], q[k]) = 0$$

From assumption (55), for $j = 1, \ldots, N$ there exists a strictly increasing sequence of time instants $\tau_j[k]$ such that

$$\lim_{k \to \infty} \tau_j[k] = \infty$$

Therefore the subsequence $E_i(q[j\tau_j[k]], q[j\tau_j[k]])$ has the same limit as the sequence $E_i(q[k], q[k])$ (Bolzano-Weierstrass theorem, see, for example, Ref. [57]), that is

$$\lim_{k \to \infty} E_i(q[k], q[k]) = 0$$

For every $k$ we have $a_{ij}[\tau_j[k]] = 0$ (see Eq. (19)); from Eqs. (33) and (14), we can write

$$\ell^2 \|q_j[\tau_j[k]] - q_j[\tau_j[k]]\|^2 \leq E_i(q[k], q[k])$$

for $j = 1, \ldots, N$. Since $\|q_j[\tau_j[k]] - q_j[\tau_j[k]]\|^2$ is a positive definite quadratic form, Eqs. (63) and (57) imply (56).

The following Corollaries follow directly from Proposition 3.6.

**Corollary 1.** If the output function $g$ distinguishes the states in the sense of Definition D.1, then the system (31) is an observer for the system (12).

**Corollary 2.** If the output function $g$ is linear in the state, then the system (31) is trivially an observer for the system (12).

3.3.3 Local Result. In this section we address local asymptotic stability of the error dynamics by linearizing the steepest descent discrete time updates for the estimators $q_i$ around the nominal true state $q$. We introduce the estimation error

$$e_i[k] = q_i[k] - q[k]$$

The discrete-time evolution equation for the error is therefore given by

$$e_i[k + 1] = e_i[k] - T(w(q[k] + e_i[k], q[k]) - a(q[k]))$$

where $E_i$ is the scalar function defined in Eq. (33).
where $w$ and $u$ are defined in Eqs. (38) and (12b). Note that $w(q[k], q[k]) = u(q[k])$. We study the local convergence properties of the error by linearizing (65) about (70). Summing terms in Eq. (72), we obtain

$$e_i[k + 1] = (I_{N} - TD_i[k])e_i[k] + O\left(||e_i[k]||^2\right)$$

(66)

where $D_i[k]$ is the $sN \times sN$ matrix given by

$$D_i[k] = TT\left(\nabla_x \nabla_{\theta} J(q[k]) + G(q[k])A_i[k]G^T(q[k])\right)$$

$$= TT\nabla_x \nabla_{\theta} F(q[k], q[k])$$

(67)

By introducing the transfer function

$$\Phi_i[k, j] = (I_{N} - TD_i[k - 1])\cdots (I_{N} - TD_j[j])$$

(68)

we rewrite the linearized error dynamics as

$$e_i[k + 1] = \Phi_i[k + 1, 0]e_i[0]$$

(69)

**Proposition 3.7.** Assume that $\nabla_x \nabla_{\theta} J(q[k]) \geq 0$ for all $k$. If the initial error is bounded, that is, there exists a finite constant $\eta > 0$ such that $||e_i[0]|| \leq \eta$, and the following relation holds for all $k$

$$Tz_{\text{max}}(\Gamma)^k \hat{\lambda}_{\text{max}}(D_i[k]) < 1$$

(70)

where $\hat{\lambda}_{\text{max}}$ denotes the maximum eigenvalue of its argument, then

$$\lim_{k \to \infty} e_i[k] = 0.$$

**Proof.** From Eq. (67) and the assumption $\nabla_x \nabla_{\theta} J(q[k]) > 0$ for all $k$, it follows that $D_i[k] > 0$ for all $k$. Let

$$\tilde{\lambda} = \max_k \hat{\lambda}_{\text{max}}(D_i[k])$$

which is positive since $D_i[k]$ are positive definite. From Eq. (68)

$$||e_i[k]|| \leq ||e_i[0]|||\Phi_i[k, 0]||$$

$$\leq \theta \prod_{j=0}^k \|I_{N} - TD_j[j]\|$$

$$\leq \eta\left(1 - T\tilde{\lambda}\hat{\lambda}_{\text{max}}(\Gamma)\right)^k$$

(71)

Let $\psi = 1 - T\tilde{\lambda}\hat{\lambda}_{\text{max}}(\Gamma)$ which is smaller than 1 from assumption (70). Summing terms in Eq. (72), we obtain

$$\sum_{k=0}^{\infty} ||e_i[k]|| \leq \eta \sum_{k=0}^{\infty} \phi^k = \frac{\eta}{1 - \psi} < \infty$$

(72)

which implies $\lim_{k \to \infty} e_i[k] = 0$. □

The local stability results are based on the existence of the Hessian matrix $D_i[k]$ defined in Eq. (67). The relation (70) establishes inverse proportionality between the largest eigenvalue of the Hessian and the largest feedback gain, and it gives an operational condition for selecting gains and updating time steps that guarantee asymptotic convergence of the estimators $q_i$ to the true state $q$ (state error $e_i$). The weaker conditions posed in the previous section ensure instead only the asymptotic convergence of the estimated output to the true output (output error $E_i$). We note that the partial asynchronous structure (55) implies that the sequence of Hessians $\{D_i[\tau_j[k]]\}_{k \geq 0}$ exists provided that $F$ is twice differentiable.

4 Application to Cooperative Target Estimation

4.1 Sensor Model. In what follows we consider range-only sensors [7,8]. Therefore, $p = 1$ and the observation function $h_i$ is given by the following expression that takes scalar values

$$h_i(q_i, x) = ||Lx - q_i||$$

(74)

where $|| \cdot ||$ is the two-norm, and $L : \mathbb{R}^n \to \mathbb{R}^2$ is a $2 \times n$ matrix that maps the state $x$ to the target position in the plane. For example, if $x$ includes the position and the velocity, then $n = 4$ and

$$L = \begin{pmatrix} 1 & 0 \end{pmatrix} I_2$$

(75)

For a static target, $x$ represents the position of the target and $L = I_2$.

Following Refs. [5,6], we assume the sensor noise variance $\sigma$ to be given by

$$\sigma_i(q_i, x) = b_0 + b_1(b_2 - ||Lx - q_i||)^2$$

(76)

The expression in Eq. (76) corresponds to the assumption that there exist an optimal sensing distance $b_2$ at which the observation noise is minimum. The sensor model defined by Eqs. (74) and (76) can be seen as a first approximation of a ultrasound sensor model, see Ref. [7] and references therein.

4.2 Control Law. The Fisher information matrix encodes the amount of information related to a set of measurements in estimating the state, see, for example, Ref. [51]. Under the assumption that the estimation error is normally distributed, the Fisher information matrix equals the inverse of the Cramer-Rao lower bound, see Ref. [59]. Since the error covariance matrix is bounded from below by the Cramer-Rao lower bound, the minimization of the inverse of the Fisher information matrix is equivalent to reducing the uncertainty in the estimation. The minimization of the uncertainty is achieved by generating trajectories for the mobile sensors starting from a scalar measure associated to the estimation error covariance, see, for example, Refs. [5,7,8]. For a comparison between different scalar measures used for optimal trajectories generation, see Refs. [60,61].

Let $\hat{x}$ be an estimate of the target state. The cost function $J$ in Eq. (8) is given by

$$J(\hat{x}, q) = \frac{\text{det} S(\hat{x}, q)}{S^{-1} = \sum_{j=1}^{N} U_j}$$

(77a)

(77b)

where $U_j$ is the Fisher information matrix associated to sensor $j$ measurement, defined in Eq. (92c). Note that $S$ is the sum over all $j$. In Refs. [5,7], the formation control problem has been addressed by considering a fully connected and undirected sensor network, whereas in Ref. [8] consensus estimators have been introduced in order to account for different network topologies. Here, we apply the method proposed in Sec. 3 for addressing the case in which sensors share data infrequently.

We introduce the local polar coordinates (see Fig. 2)

$$r_i = ||Lx - q_i||, \quad \beta_i = \arctan\left(\frac{(Lx - q_i)^T e_i}{(Lx - q_i)^T e_i}\right)$$

(78)

where $e_1 = (1)^T$ and $e_2 = (0)^T$ are unit basis vectors for a global Cartesian coordinates system. From definitions (78), the function $J$ can be expressed as a function of polar variables $r_i, \beta_1, \ldots, r_N, \beta_N$. The gradient of the cost function in local polar coordinates is given by Refs. [5,8]

$$\nabla J(r_1, \beta_1, \ldots, r_N, \beta_N) = \sum_{i=1}^{N} \left(\frac{\partial J}{\partial r_i} e_i + \frac{\partial J}{\partial \beta_i} e_i\right)$$

(79)

where $e_i$ and $e_\beta$ are basis vectors for the local polar coordinates, with $||e_i|| = 1$ and $||e_\beta|| = 1/r_i$, see, for example, Ref. [62]. The
derivatives of the cost function can be computed by using the following standard matrix calculus identities [53]

\begin{align}
\frac{\partial A^{-1}}{\partial \tau} &= -A^{-1} \frac{\partial A}{\partial \tau} A^{-1} \\
\frac{\partial}{\partial \tau} \det A &= \det A \text{tr} \left( A^{-1} \frac{\partial A}{\partial \tau} \right)
\end{align}

(80a)

(80b)

where \( A \) is a nonsingular matrix depending on the scalar parameter \( \tau \). From Eq. (77), we have

\begin{align}
\frac{\partial J}{\partial r_i} &= \text{tr} \left( S^+ \frac{\partial S}{\partial r_i} \right) = -\text{tr} \left( A^{+} \frac{\partial A}{\partial r_i} \right) \\
\frac{\partial J}{\partial p_i} &= \text{tr} \left( S^+ \frac{\partial S}{\partial p_i} \right) = -\text{tr} \left( A^{+} \frac{\partial A}{\partial p_i} \right)
\end{align}

(81a)

(81b)

We introduce the vectors \( p_i = Lx - q_i \). From definitions (78), we have

\begin{equation}
p_i = r_i (\cos \beta_i e_1 + \sin \beta_i e_2)
\end{equation}

(82)

Time deriving we obtain

\begin{equation}
\dot{p}_i = \dot{r}_i (\cos \beta_i e_1 + \sin \beta_i e_2) + r_i \dot{\beta}_i (-\sin \beta_i e_1 + \cos \beta_i e_2)
\end{equation}

(83)

Let \( \dot{r}_i \) and \( \dot{\beta}_i \) be velocity inputs for \( r_i \) and \( \beta_i \). Individual control laws for kinematically actuated vehicles are generated by considering the gradient flow associated to \( J \), with individual velocity inputs equal to the projection of the gradient of the cost function into the local basis \( (e_1, e_2, e_3) \). By taking the inner products in Eq. (79), we have the gradient flows

\begin{align}
\dot{r}_i &= -\Gamma_r \frac{\partial J}{\partial r_i} \\
\dot{\beta}_i &= -\Gamma_\beta \frac{1}{r_i^2} \frac{\partial J}{\partial \beta_i}
\end{align}

(84a)

(84b)

where \( \Gamma_r > 0 \) and \( \Gamma_\beta > 0 \) are control gains. Writing Eq. (83) in discrete time form and substituting from Eq. (84), we obtain the following individual discrete-time evolution equations in Cartesian coordinates

\begin{align}
p_i[k+1] &= p_i[k] + Tu_i(p_i[k]) \\
u_i(p) &= -\mathcal{O}(p_i) \Gamma_i \left( \frac{\partial J}{\partial r_i} \frac{1}{r_i^2} \frac{\partial J}{\partial \beta_i} \right)^T(p)
\end{align}

(85a)

(85b)

where we introduced the rotation matrix

\begin{equation}
\mathcal{O} = \begin{pmatrix} \cos \beta_i & -\sin \beta_i \\ \sin \beta_i & \cos \beta_i \end{pmatrix}
\end{equation}

(86)

the gain matrix \( \Gamma_i = \text{diag} (\Gamma_r, \Gamma_\beta) \), and the collection of states \( p = (p_1, \ldots, p_N)^T \).

Remark. Consider the system (12) with output \( y = g(p) = (h_1(p_1), \ldots, h_N(p_N)) \), where \( h_i \) is the measurement function in Eq. (74). For this output function, we have

\begin{equation}
g(r_1, \ldots, r_N, \beta_1, \ldots, \beta_N) = g(r_1, \ldots, r_N, \beta_1 + \theta, \ldots, \beta_N + \theta)
\end{equation}

(87)

for any \( \theta \). Therefore, for this input and output maps, the states \( (r_1, \ldots, r_N, \beta_1, \ldots, \beta_N) \), \( (r_1, \ldots, r_N, \beta_1 + \theta, \ldots, \beta_N + \theta) \) are not distinguishable in the sense of Definition D.1 in Appendix D, and therefore the corresponding system is not globally observable. However, the input \( u \) in Eq. (12b) asymptotically drives the sensors to a configuration that minimizes the cost function \( J \). According to the analysis in Appendix C, minima of \( J \) have the property

\begin{equation}
J(r_1, \ldots, r_N, \beta_1, \ldots, \beta_N) = J(r_1, \ldots, r_N, \beta_1 + \theta, \ldots, \beta_N + \theta)
\end{equation}

(88)

Therefore the estimators (31) are such that, asymptotically, \( \min J(p_i) = \min J(p) \).

4.3 Nonlinear Estimation With State-Dependent Noise. In this section, we briefly illustrate the extended Kalman filter developed in Ref. [49] that is used in the simulations presented in Sec. 5.

4.3.1 Modified Extended Kalman Filter. The objective is to estimate the state \( x[k] \) using measurements \( z[k] \). Typically, this class of target tracking problems is solved by applying an extended Kalman filter, see, for example, Refs. [5,7,58], which assumes that the measurement noise is independent from the state of the system. However, in our case the sensor noise statistics are dependent on the state of the system, which violates the assumptions required for a Kalman filter. To correctly address the state-dependent noise in the measurements, we utilize a generalized extended Kalman filter described in Ref. [49] to generate an estimate \( \hat{x}_i \) for the source state and the related error covariance \( P_i \in \mathbb{R}^{w_{x_i}, w_{x_i}} \).

As with the extended Kalman filter, the modified algorithm consists of prediction and update steps. For sensor \( i \), let

\begin{equation}
Z_i[k] = \{z_i[k], \ldots, z_i[0]\}
\end{equation}

(89)

be the set of measurements taken up to time \( k \). The state estimate and the error covariance matrix at time \( k \) given the measurements up to time \( l \) are given by

\begin{align}
\hat{x}_i[k|l] &= \mathbb{E}[x_i[k|Z[l]]] \\
P_i[k|l] &= \mathbb{E}[x_i[k] - \hat{x}_i[k|l])(x_i[k] - \hat{x}_i[k|l])^T | Z[l]]
\end{align}

(90a)

(90b)

It is assumed that there exists a state estimate \( \hat{x}_i[k-1|k-1] \) at time \( k - 1 \) and associated error covariance \( P_i[k-1|k-1] \). The prediction step is given by, see Ref. [51] for details...
\[ \dot{x}_i[k] = f(x_i[k-1]+[P_i^{-1}[k-1]]x_i[k-1]) \quad (91a) \]

\[ P_i[k-1] = \nabla_i f(x_i[k-1]+[P_i^{-1}[k-1]]) \times P_i[k-1][k-1] \nabla_i^T f(x_i[k-1]+[P_i^{-1}[k-1]]) + Q[k] \quad (91b) \]

where \( \nabla_i \) is the gradient with respect to \( x_i \).

By using a maximum likelihood approach coupled with the Gauss-Newton algorithm (see Ref. [49]), the local state and error covariance update equations are

\[ x_i[k] = x_i[k-1] - [P_i^{-1}[k-1]]x_i[k-1] + R_i[k]^{-1} s_i[k] \quad (92a) \]

\[ s_i[k] = \frac{-z_i}{\sigma_i} \nabla_i h_i + \frac{1}{2\sigma_i} \left( \frac{1 - \frac{z_i^2}{\sigma_i}}{\sigma_i} \right) \nabla_i^T \sigma_i \nabla_i h_i \]

\[ R_i[k] = \frac{1}{\sigma_i} \nabla_i^T h_i \nabla_i h_i + \frac{z_i}{2\sigma_i} \left( \nabla_i^T \sigma_i \nabla_i h_i + \nabla_i^T \sigma_i \nabla_i h_i \right) + \frac{1}{4\sigma_i} \left( \frac{z_i^2}{\sigma_i} + \frac{1}{\sigma_i} \right) \nabla_i^T \sigma_i \nabla_i \sigma_i \]

\[ P_i^{-1}[k] = P_i^{-1}[k-1] + U_i[k] \quad (92c) \]

\[ U_i[k] = \left[ \frac{1}{\sigma_i^2} \nabla_i^T h_i \nabla_i h_i + \frac{1}{2\sigma_i^2} \nabla_i^T \sigma_i \nabla_i \sigma_i \right]_{i,[k-1]} \]

where \( \xi_i = z_i - h_i \). The terms in Eq. (92) are defined in the Appendix. Note that if \( \sigma_i \) did not depend on \( x \) then the gradient term \( \nabla_i \sigma_i \) would be zero and Eq. (92) would reduce to the standard extended Kalman filter update equations. The matrix \( U \) in Eq. (92e) is the Fisher information term in the error covariance update. Update filter equations analogous to Eq. (92) have been derived in Ref. [50] by using the Newton-Raphson algorithm coupled with a maximum likelihood approach.

4.3.2 Data Fusion. Note that a generalized extended Kalman filter (92) is implemented on each sensor. Whenever a sensor receives data from another sensor, the external information is fused to obtain a target state estimate that accounts for the shared data. In Ref. [50], data fusion equations are derived by considering the joint probability distribution of measurements taken from different sensors. Using the maximum likelihood approach, one obtains equations analogous to Eq. (92) that account for shared measurements and predictions.

Let \( I_i[k] \) be the set of indices of all sensors that communicate with vehicle \( i \) at time \( k \), and \( |I_i[k]| \) the cardinality of the set \( I_i[k] \). Also note that \( i \in I_i[k] \), for all \( k \), meaning that each vehicle has always access to its most updated data. The update algorithm is then, see Ref. [50] for details

\[ \dot{x}_i[k] = \dot{x}_i[k-1] - [P_i^{-1}[k-1]]x_i[k-1] + R_i[k]^{-1} s_i[k] \quad (93a) \]

\[ P_i[k]^{-1} = P_i[k-1]^{-1} + U_i[k] \quad (93b) \]

\[ U_i[k] = \sum_{j \in I_i[k]} U_j[k] \quad (93c) \]

\[ R_i[k] = \sum_{j \in I_i[k]} R_j[k] \quad (93d) \]

\[ s_i[k] = \sum_{j \in I_i[k]} s_j[k] \quad (93e) \]

where \( j \neq i \), the initial estimation error is

\[ q_i = q_i + (5 \text{ m}, 5 \text{ m}) \quad (96) \]

5 Simulation Results

In order to illustrate the proposed methodology, we consider a stationary target. The state \( x[k] \) is the 2-vector specifying the position in the plane, and the kinematical model in Eq. (1) specializes to

\[ f(x[k]) = x[k] \quad (94) \]

We assume \( Q[k] = 0 \). The target position is \( (0, 0) \). We consider a group of three homogeneous sensors. Following Ref. [5], the sensor parameters in Eq. (76) are \( b_h = 0.1528 \text{ m}^2 \) and \( b_t = 0.0008 \). The optimal sensing distance is set \( b_s = 15 \text{ m} \). Since \( a_t < 1 \), minima of the cost function for the control law are attained for \( r = b_s \), see Appendix C. The extended Kalman filter is initialized with \( \dot{x}[0] = (-5 \text{ m}, 5 \text{ m}) \), and \( P_i[0] = 10 \text{ m}^2 I_2 \) for \( i = 1, 2, 3 \). Coordinates are measured with respect to a global Cartesian reference frame.

The initial states of the three mobile sensors are \( (-20 \text{ m}, -15 \text{ m}), (-30 \text{ m}, 0), \) and \( (15 \text{ m}, 30 \text{ m}) \). The updating time interval in Eq. (31) is \( \ell = 1 \text{ s} \). Gains in Eq. (84) are \( K_r = 20 \text{ m}^2 \text{s}^{-1} \) and \( K_b = 25 \text{ m}^2 \text{s}^{-1} \), and in Eq. (19) we set \( \ell = 0.005 \text{ m}^2 \). The communication network is circulant, therefore satisfying condition (55).

As a first example, we consider the output to be equal to the state, that is \( y_i = q_i \). Therefore, in Eq. (30), we have

\[ \nabla_i y = \begin{pmatrix} \cos \beta_i & \sin \beta_i \\ -\sin \beta_i & \cos \beta_i \end{pmatrix} \quad (95) \]

For\( i 
eq j \), the initial estimation error is

\[ q_i = q_i + (5 \text{ m}, 5 \text{ m}) \quad (96) \]

Trajectories generated by algorithm (31) are shown in Fig. 3, with empty circles representing the final position of the moving agents and the solid circle representing the true position of the target. Consistently with the analysis in Appendix D, the sensors converge to a symmetric configuration at the optimal sensing distance from the target, with angular spacing satisfying conditions (B22). The time history of the quantities \( \sum_{i=1}^3 \cos \beta_i[k] \) and \( \sum_{i=1}^3 \sin \beta_i[k] \) is plotted in Fig. 4, with solid and dashed lines, respectively. It is clear that those quantities asymptotically tend to zero, satisfying the optimal configuration conditions (121).

The time history of the functions \( J(q_i) \) in Eq. (36) for the three sensors is shown in Fig. 5. We observe that these are individual measures of the error in the target state location. The final configuration is such that the three functions evaluate at the same value, which is the minimum of \( F \) in Eq. (36) since the error \( E_i \) converges to zero, see Proposition 3.5. For the sensor starting at position \( (-20 \text{ m}, -15 \text{ m}) \), Fig. 6 shows, with dashed and dotted lines, the time history of the errors of the estimators \(|q_{ij} - q_i|\), for

Fig. 3 Trajectories of sensors estimating the position of a static target, with output equal to the sensors position

Fig. 3 Trajectories of sensors estimating the position of a static target, with output equal to the sensors position

Fig. 3 Trajectories of sensors estimating the position of a static target, with output equal to the sensors position

Fig. 3 Trajectories of sensors estimating the position of a static target, with output equal to the sensors position

Fig. 3 Trajectories of sensors estimating the position of a static target, with output equal to the sensors position
\[ j = 2, 3. \] The solid line refers to \(|\hat{q}_1 - q_1|\), which is obviously zero for all \(k\). Consistently with Corollary 2, the error in the state estimation is asymptotically zero. Similar results hold for the errors associated to the other two sensors.

For the same geometry and values of the system parameters, we consider a second example in which the motion control algorithm is applied to generate trajectories with the output equal to the measurement function (74), that is \(y_j = r_j\), for \(j = 1, 2, 3\). In this case, we have

\[ r_j = r_j(\theta_j) = 10(\theta_j) \quad (97) \]

Trajectories generated by the motion control algorithm (31) are shown in Fig. 7, with empty circles representing the final position. As in the previous case, in the final configuration the sensors are placed at the optimal sensing distance. From Fig. 8, it is clear that in the final configuration the cost functions related to the three sensors evaluate at the same minimum.

Figure 9 shows, with dashed and dotted lines, the time history of the errors of the output \(|h(\hat{p}_{11}) - h(p_1)|\), for \(j = 2, 3\). The solid line refers to \(|h(\hat{p}_{11}) - h(p_1)|\), which is obviously zero for all \(k\). Consistently with Proposition 3.6, the error in the state output estimation is asymptotically zero. Similar results hold for the errors associated to the other two sensors.
In order to illustrate a scenario of coordination with relatively rare communication sharing, we simulate the same system (with $\ell = 0.01$) of three mobile sensors where each agent broadcasts its position every twenty updating time steps. Functions $J(\dot{q}_i)$ are plotted in Figure 10 that reveals the expected agreement between cost functions despite the relatively rare information sharing events. The disagreement $|\dot{q}_j - \dot{q}_i|$, $j = 1, 2, 3$, is plotted in Fig. 11. We note that the convergence is slower due to the lower frequency of information sharing events. Time histories of the individual state estimates of the target are plotted in Fig. 12. Although the convergence of the state estimates is not enforced, in the case presented here it emerges asymptotically.

For the case of information sharing events every twenty time steps and a group of ten mobile sensors we plot in Fig. 13 the trajectories of the agents (solid lines with final states represented by circles) along with the trajectories (dashed lines with final states represented by triangles) generated by the same control algorithm with complete communication network topology. Mobile sensors are initially placed randomly in the box $[-60, -10] \times [-60, -10]$; it is clear that the distributed observer mimics the complete communication network topology despite relatively rare communication sharing events, and it represents a robust implementation that accounts for data packets drop and communication failures in general.

We note that in a frozen-time sense the motion coordination algorithm is scalable since one can design the communication network to have a maximum fixed total number of communication links at every time step and satisfy the partial asynchronous constraint. However, it is not scalable over the finite time interval set to satisfy the partial asynchronous evolution, since this condition poses the existence of $N^2$ communication links within such finite interval.

### 6 Conclusions

We proposed a class of distributed motion control algorithms for coordination of platoons of mobile agents with limited communication capabilities. This class of distributed motion control algorithms is designed to operate with occasional interagents communication. Coordination is enabled by embedding a local observer on board each agent in order to estimate the states of the other agents in the group, and a centralized gradient descent based control law is implemented but with local states rather than the global ones that would be communicated directly in a centralized architecture. We have shown that if for each updating time instant there exists a finite time interval within which all the agents in the team communicate at least once, global asymptotic agreement between communicated and estimated quantities is achieved. Moreover, by communicating the states the evolution of the estimators corresponds to the evolution of a Luenberger observer. We also derived local conditions for asymptotic convergence by linearizing the estimators’ evolutions about the trajectories of the actual states and by analyzing the corresponding error evolution.

We applied the proposed class of algorithms to the distributed cooperative estimation of a noisy dynamical process by a platoon of mobile sensor nodes with state dependent measurement noise. We considered a circulant directed communication network structure. Our method has been illustrated through simulation results in which we show asymptotic agreement among estimated and actual outputs whenever communication involves a nonlinear function of the states, and asymptotic agreement among states whenever the states are directly shared.

The evolution of the communication network adopted here is among the simplest one can use to describe intermittent communication links among agents. A possible generalized framework would include the characterization of communication links as stochastic processes, with subsequent convergence dictated by the
Appendix A: Expressions for the Gradients in the Kalman Filter Update Equations

From Eqs. (74) and (76), the Cartesian coordinates of the gradients with respect to \( x \) of the observation function \( h_i \) and of the noise variance \( \sigma_i \) are, respectively, given by

\[
\nabla_v h_i = \frac{1}{||Lx - q||} \left( (Lx - q)^T e_1 \right) L \tag{A1a}
\]

\[
\nabla_\sigma \sigma_i = \frac{2a_i(||Lx - q|| - a_2)}{||Lx - q||} \left( (Lx - q)^T e_1 \right) L \tag{A1b}
\]

In terms of the polar coordinates (78), the Cartesian forms of the gradients in Eq. (A1) are given by

\[
\nabla_v h_i = \left( \cos \beta_i, \sin \beta_i \right) L \tag{A2a}
\]

\[
\nabla_\sigma \sigma_i = 2a_i (r_1 - a_2) \left( \cos \beta_i, \sin \beta_i \right) L \tag{A2b}
\]

Appendix B: Gradient Descent Algorithms

B.1 Proof of Lemma 3.3. Let \( \hat{\xi}, \hat{\zeta} \) be scalar parameters and let \( \psi(\hat{\xi}, \hat{\zeta}) := F(\hat{q}, \xi, \hat{q}, \zeta) \). The chain rule yields

\[

\frac{\partial \psi}{\partial \xi} (\hat{\xi}, \hat{\zeta}) = \nabla_\xi F(\hat{q}, \xi, \hat{q}, \zeta)^T \hat{q} \tag{B1a}
\]

\[

\frac{\partial \psi}{\partial \zeta} (\hat{\xi}, \hat{\zeta}) = \nabla_\zeta F(\hat{q}, \xi, \hat{q}, \zeta)^T \hat{q} \tag{B1b}
\]

Consider the curve \( r := (\xi(\lambda), \zeta(\lambda))^T \), parameterized by the scalar \( \lambda \in [0,1] \). Therefore

\[

\hat{\xi}(\lambda) = \hat{x}, \quad \hat{\zeta}(\lambda) = \hat{\lambda}, \quad \frac{dr}{d\lambda} = (1,1)^T \tag{B2}
\]

Direct application of the gradient theorem for vector calculus and the use of Eq. (B1) give

\[

F(\hat{q}, \xi, \hat{q}, \zeta) - F(q, q) = \psi(1,1) - \psi(0,0)
\]

\[
= \int_0^1 \nabla_\xi \psi^T \frac{dr}{d\lambda} d\lambda = \int_0^1 \left( \frac{\partial \psi}{\partial \xi} + \frac{\partial \psi}{\partial \zeta} \right) d\lambda
\]

\[
= \int_0^1 \left( \nabla_\xi F(\hat{q}, \xi, \hat{q}, \zeta)^T \frac{d\lambda}{d\xi} \right) q + \int_0^1 \left( \nabla_\zeta F(\hat{q}, \xi, \hat{q}, \zeta)^T \frac{d\lambda}{d\zeta} \right) q
\]

\[
\int_0^1 \left( \nabla_q F(\hat{q}, \xi, \hat{q}, \zeta)^T \right) q + \int_0^1 \left( \nabla_q F(\hat{q}, \xi, \hat{q}, \zeta)^T \right) q d\lambda
\]

**B.2 Proof of Proposition 3.4.** Using Eqs. (31) and (12), the Descent Lemma (Lemma 3.3), and assumptions (43)–(46) we obtain

\[
F(\hat{q}, k + 1, q, k + 1) \leq F(\hat{q}, k, q) + T \nabla_q F(\hat{q}, k, q)^T w(\hat{q}, k, q)
\]

\[
+ \frac{T}{2} \left( \kappa ||w(\hat{q}, k, q)||^2 + \kappa ||u(q)||^2 \right)
\]

where \( \hat{\mu} = \max \{\hat{\mu}, \mu\} \), and \( \kappa = \max \{\kappa, \kappa\} \). Let

\[
\beta = T \left( \frac{\hat{\mu}}{2\kappa} \right)
\]

which is positive for

\[
0 < \beta < \frac{2\mu}{\kappa}
\]

We have one inequality (B5) for each \( k = 0,1,\ldots \); by adding these inequalities and by using the non-negativity condition (40) we obtain

\[
0 \leq F(\hat{q}, k + 1, q, k + 1) \leq F(\hat{q}, 0, q) - \beta \sum_{j=0}^k \left( ||w(\hat{q}, j, q)||^2 + ||u(q)||^2 \right)
\]
\[
\sum_{j=0}^{\infty} \left( \|w(\hat{q}_j, q_j)\|^2 + \|u(q_j)\|^2 \right) \leq \frac{1}{\beta} F(\hat{q}_0, q_0) < \infty \tag{B9}
\]

Therefore
\[
\sum_{j=0}^{\infty} \|w(\hat{q}_j, q_j)\|^2 < \infty \tag{B10a}
\]

\[
\sum_{j=0}^{\infty} \|u(q_j)\|^2 < \infty \tag{B10b}
\]

which imply
\[
\lim_{k \to \infty} w(\hat{q}_k, q_k) = 0 \tag{B11a}
\]

\[
\lim_{k \to \infty} u(q_k) = 0 \tag{B11b}
\]

and, by using condition (43) and (44)
\[
\lim_{k \to \infty} \nabla_{\hat{q}} F(\hat{q}_k, q_k) = 0 \tag{B12a}
\]

\[
\lim_{k \to \infty} \nabla_{q} J(q_k) = 0 \tag{B12b}
\]

**B.3 Proof of Proposition 3.5.** From Eqs. (38) and (36), we have
\[
w(\hat{q}_k, q_k) = u(\hat{q}_k) + v(\hat{q}_k, q_k) \tag{B13}
\]

where \( v = -\Gamma \nabla E_i \). Therefore, the following equality holds for all \( k \) (law of cosines)
\[
\|w(\hat{q}_k, q_k)\|^2 = \|u(\hat{q}_k)\|^2 + \|v(\hat{q}_k, q_k)\|^2 + 2v(\hat{q}_k, q_k)u(\hat{q}_k) \tag{B14}
\]

From (B10a) and from the assumption that \( \|u\| \) and \( \|v\| \) are bounded it follows that
\[
\sum_{j=0}^{\infty} \|v(\hat{q}_j, q_j)\|^2 < \infty \tag{B15}
\]

which implies \( \lim_{k \to \infty} V(\hat{q}_k, q_k) = 0 \) and, by using Eq. (53)
\[
\lim_{k \to \infty} \nabla_{\hat{q}} E_i(\hat{q}_k, q_k) = 0 \tag{B16}
\]

**Appendix C: Investigation of Minima of the Function J**

Under the usual constraints for observability for the Kalman filter, the target error estimates reach a steady state. Given the properties of gradient descent algorithms discussed in Sec. 4.2, the sensors will reach a configuration such that the cost function reach a minimum, provided that the step size is small enough. In order to examine the nature of minima, we consider the explicit form of the cost function \( J \).

Consider the range-only sensor model described in Sec. 2.2. By using definitions (92), (76), and (A2) we have
\[
U_i = \begin{pmatrix}
\cos^2 \beta_i & \cos \beta_i \sin \beta_i \\
\cos \beta_i \sin \beta_i & \sin^2 \beta_i
\end{pmatrix} \tag{B17a}
\]

\[
z_i = \frac{1}{\sigma_i} + \frac{b_i^2(r_i - b_i)^2}{\sigma_i^2} = \frac{\sigma_i + b_i(\sigma_i - b_i)}{\sigma_i^2} \tag{B17b}
\]

\[
S^{-1} = \begin{pmatrix}
\sum_{i=1}^{N} z_i \cos^2 \beta_i & \sum_{i=1}^{N} z_i \cos \beta_i \sin \beta_i \\
\sum_{i=1}^{N} z_i \cos \beta_i \sin \beta_i & \sum_{i=1}^{N} z_i \sin^2 \beta_i
\end{pmatrix} \tag{B18}
\]

In order to compute the determinant of \( S \), we use the relation \( \det S = (\det S^{-1})^{-1} \). Therefore
\[
\det S^{-1} = \det \left( \sum_{i=1}^{N} U_i = \begin{pmatrix}
\sum_{i=1}^{N} z_i \cos^2 \beta_i & \sum_{i=1}^{N} z_i \cos \beta_i \sin \beta_i \\
\sum_{i=1}^{N} z_i \cos \beta_i \sin \beta_i & \sum_{i=1}^{N} z_i \sin^2 \beta_i
\end{pmatrix} \right)
\]

\[
= \sum_{i=1}^{N} z_i^2 \cos^2 \beta_i \sin^2 \beta_i + \sum_{i \neq j} z_i z_j \cos \beta_i \sin \beta_i \sin \beta_j
\]

\[
= \frac{1}{2} \sum_{i=1}^{N} z_i z_i \sin^2 (\beta_i - \beta_j) \tag{B19}
\]

where we used the well known trigonometric relation \( \sin 2 \beta \sin \beta = \sin (\beta - \beta) \). Therefore, for this class of sensors with the modified extended Kalman filter in Eq. (92) we have the same result as in Ref. [7, Lemma 2.2].

Maxima of Eq. (B19) correspond to maximum values of \( z_i \) and of \( \sin^2 (\beta_i - \beta_j) \). We find the extrema of \( z_i \) as the solution of the equation \( \partial z_i / \partial r_i = 0 \). We have
\[
\frac{\partial z_i}{\partial r_i} = \frac{2a_i(a_2 - r_i)}{\sigma_i^2} \left( a_0(1 - 2a_1) + a_1(1 + 2a_1)(a_2 - r_i) \right) \tag{B20}
\]

which is equal to zero for
\[
r_i = b_2 \quad \tag{B21a}
\]

\[
r_i = r_i^* := b_2 + \sqrt{\frac{b_0(2b_1 - 1)}{b_1(2b_1 + 1)}} \tag{B21b}
\]

There are three cases, depending on the term \( 2b_1 - 1 \):

1. For \( b_1 = 1/2 \) there are three coincident maxima achieved for \( r_i = b_2 \), which corresponds also to minimum measurement noise.
2. For \( b_1 < 1/2 \) the only admissible (real) minimum is achieved again for \( r_i = b_2 \).
3. For \( a_1 > 1/2 \), there are three real extrema, provided that the right-hand side of (B21b) is positive. In this case, \( r_i = b_2 \) is a local minimum (and therefore a local maximum for the cost
Appendix D: Observability Notions for Nonlinear Systems

We introduce the definition of observability for nonlinear systems from Ref. [63].

Definition D.1. Let \( q[k] \) be the solution of system (12) under the application of the input \( u \) in \( [k_0, k] \). A pair of initial states \( (q[k_0], q'[k_0]) \) is distinguishable for the system (12) if

\[
\forall u \in \mathbb{R}^q, \forall k \geq k_0, \quad g(q[k]) \neq g(q'[k])
\]

(B23)

Definition D.2. A system in the form Eq. (12) is observable if for every pair of initial states \( (q[k_0], q'[k_0]) \) there exists an input \( u \) such that \( (q[k], q'[k]) \) are distinguishable.

An input that distinguishes every pair of initial states in \( [k_0, k] \) is called a universal input on \([k_0, k]\). A non universal input is called a singular input. Unlike linear systems, observable nonlinear systems may admit singular inputs.

References


[11] For the optimal relative angular position, we have that for \( N = 2 \) the minimum of \( \beta_1 - \beta_2 = \pi/2 \), with the intuition of having the two sensors pointing on orthogonal directions. For \( N \geq 3 \), the same result as Ref. [7, Proposition 2.3] holds, that is a set \( \{ \beta_1, \ldots, \beta_N \} \) is a critical point for Eq. (118) if any two vectors \( \{ \cos 2\beta_i, \sin 2\beta_i \} \), \( i = 1, \ldots, N \), are linearly independent.

\[
\sum_{i=1}^{N} \cos 2\beta_i = 0 \quad \text{and} \quad \sum_{i=1}^{N} \sin 2\beta_i = 0 \quad (B22)
\]

function \( J \), while \( r_1^* \) in (B21b) are maxima for \( x_k \), as illustrated qualitatively in Fig. 14. Therefore, for those sensors with initial distance to the target greater than \( b_2 \), the final position is such that the distance to the target equal \( r_1^* \); similarly, if the initial distance to the target is less that \( b_2 \), the final position is such that the distance to the target equals \( r_0 \).

Fig. 14 Plot of the function \( \partial \omega_1/\partial r \) for \( b_1 = 1/2 \).
Diffusion-Based EM Algorithm for Distributed Estimation in Unreliable Sensor
150–160.
ability and Robust Design for Leader-Following Linear Multi-Agent Systems,”
Automatica, 49(7), pp. 2199–2205.
[46] Shimizu, K., 2000, “Nonlinear State Observers by the Gradient Descent Meth-
od,” Proceedings of the IEEE International Conference on Control Applica-
tions, pp. 616–622.
Process., 78, pp. 61–78.
1358–1366.
Dependent Gaussian Observation Noise,” Tech. Rep. 2, VaCAS.
[51] Bar-Shalom, Y., and Fortman, T. E., 1988, Tracking and Data Association,
Academic Press, San Diego, CA.
[52] Bertsekas, D. P., and Tsitsiklis, J. N., 1989, Parallel and Distributed Compu-
ton, NJ.
Linear Switched Systems Using Exponential Splitting,” SHAM J. Control
Control 48(6), pp. 988–1001.
[56] Ren, W., and Beard, R. W., 2005, “Consensus Seeking in Multiagent Systems
Active Sensing Using Consensus Filters,” Proceedings of the IEEE Interna-
tional Conference on Robotics and Automation.
sensor Systems, CRC Press, Boca Raton, FL.
[61] Mihaylova, L., Lefebvre, T., Bruyninckx, H., Gadeyne, K., and Shutter, J. D.,
Numerical Methods and Applications, pp. 316–324.