Distributed Input and State Estimation Using Local Information in Heterogeneous Sensor Networks

Dzung Tran¹, Tansel Yucelen¹*, Selahattin Burak Sarsilmaz¹ and Sarangapani Jagannathan²

¹Laboratory for Autonomy, Control, Information, and Systems (LACIS), Department of Mechanical Engineering, University of South Florida, Tampa, FL, United States, ²Embedded Control Systems and Networking Laboratory (ECSNL), Department of Electrical and Computer Engineering, Missouri University of Science and Technology, Rolla, MO, United States

A new distributed input and state estimation architecture is introduced and analyzed for heterogeneous sensor networks. Specifically, nodes of a given sensor network are allowed to have heterogeneous information roles in the sense that a subset of nodes can be active (that is, subject to observations of a process of interest) and the rest can be passive (that is, subject to no observation). Both fixed and varying active and passive roles of sensor nodes in the network are investigated. In addition, these nodes are allowed to have non-identical sensor modalities under the common underlying assumption that they have complimentary properties distributed over the sensor network to achieve collective observability. The key feature of our framework is that it utilizes local information not only during the execution of the proposed distributed input and state estimation architecture but also in its design in that global uniform ultimate boundedness of error dynamics is guaranteed once each node satisfies given local stability conditions independent from the graph topology and neighboring information of these nodes. As a special case (e.g., when all nodes are active and a positive real condition is satisfied), the asymptotic stability can be achieved with our algorithm. Several illustrative numerical examples are further provided to demonstrate the efficacy of the proposed architecture.

Keywords: heterogeneous sensor networks, active and passive node roles, distributed input and state estimation, stability analysis, numerical examples

1. INTRODUCTION

As technological advances have boosted the development of integrated microsystems that combine sensing, computing, and communication on a single platform, we are rapidly moving toward a future in which large numbers of integrated microsensors will have the capability to operate in both civilian and military environments. Such large-scale sensor networks will support applications with dramatically increasing levels of complexity, including situational awareness, environment monitoring, scientific data gathering, collaborative information processing, and search and rescue; to name but a few examples. One of the important areas of research in sensor networks is the development of distributed estimation algorithms for dynamic information fusion. Because these algorithms are reliable to possible loss of a subset of nodes and communication links and they are flexible in the sense that nodes can be added and removed by making only local changes to the sensor network.
There are two common ways to do distributed dynamic information fusion. Specifically, one classical way include decentralized data fusion, for example, see Makarenko and Durrant-Whyte (2004), Cunningham et al. (2013), and Hollinger et al. (2015), where these methods have been shown to work well in practice for many applications without formal stability guarantees. Unlike these methods, system-theoretic dynamic information fusion involve equations of motion to describe time behavior of the information fusion process and they also offer stability guarantees (e.g., Olfati-Saber (2005), Olfati-Saber and Shamma (2005), and Spanos et al. (2005)). The contribution of this paper builds on system-theoretic dynamic information fusion approaches.

Although distributed estimation algorithms have had strong appeal owing to their reliability and flexibility as outlined above, a critical roadblock to achieve correct dynamic information fusion with these algorithms is heterogeneity. Heterogeneity in sensor networks is unavoidable in real-world applications. To elucidate this fact, consider a target estimation problem as a motivating example. Specifically, nodes of a given sensor network can have heterogeneous information roles in the target estimation problem such that a subset of nodes can be subject to observations of this target (active nodes) and the rest can be subject to no observation (passive nodes). Thus, only active nodes have to be taken into account during the dynamic information fusion process. In addition, note that nodes of a sensor network can also have non-identical sensor modalities; for example, a subset of nodes can sense the target position and others can sense the target velocity. This case also needs to be considered in the dynamic information fusion process.

Dealing with these classes of heterogeneity in sensor networks to achieve correct and reliable dynamic information fusion is a challenging task using distributed estimation algorithms. Toward this end, notable contributions in the literature include Olfati-Saber (2005, 2007), Olfati-Saber and Shamma (2005), Spanos et al. (2005), Freeman et al. (2006), Demetriou (2009), Bai et al. (2010), Taylor et al. (2011), Ustebay et al. (2011), Chen et al. (2012), Millán et al. (2013), Mu et al. (2014), Yucelen (2014), Yucelen and Peterson (2014, 2016), Casbeer et al. (2015), Peterson and Yucelen (2015, 2016), and Peterson et al. (2015, 2016). Specifically, the authors of Olfati-Saber (2005), Olfati-Saber and Shamma (2005), Spanos et al. (2005), Freeman et al. (2006), Demetriou (2009), Bai et al. (2010), Taylor et al. (2011), and Chen et al. (2012) propose dynamic consensus algorithms that are suitable for sensor networks with all nodes being active. However, as discussed above, a subset of nodes in a sensor network can be passive that they may not be able to sense a process of interest and collect information. While the authors of Ustebay et al. (2011), Mu et al. (2014), and Casbeer et al. (2015) present methods that cover specific applications when a subset of nodes are passive (and the remaining nodes are active), their results are in the context of static consensus and, hence, they are not suitable in their presented form for dynamic data-driven applications.

To address this challenge, the authors of Yucelen (2014), Yucelen and Peterson (2014, 2016), Peterson and Yucelen (2015, 2016), and Peterson et al. (2015, 2016) introduce the concept of sensor networks with active and passive nodes in the context of dynamic consensus. However, nodes of the considered class of sensor networks are implicitly assumed to have identical sensor modalities since each node is modeled using single integrator dynamics. Finally, the authors of Olfati-Saber (2007) and Millán et al. (2013) consider dynamic information fusion for sensor networks having non-identical sensor modalities, where the former contribution requires each node to be active via sensing some states of a process of interest. While this is implicitly not assumed in the latter contribution, global information is required during the distributed algorithm design in terms of guaranteeing global asymptotic stability (although the proposed algorithm can be executed by solely relying on local information exchange between neighboring nodes).

The contribution of this paper is to introduce and analyze a new distributed input and state estimation architecture for heterogeneous sensor networks. Specifically, nodes of a given sensor network are allowed to have heterogeneous information roles in the sense that a subset of nodes can be active (that is, subject to observations of a process of interest) and the rest can be passive (that is, subject to no observation). Both fixed and varying active and passive roles of sensor nodes in the network are investigated. In addition, these nodes are allowed to have non-identical sensor modalities under the common underlying assumption that they have complimentary properties distributed over the sensor network to achieve collective observability (see, for example, Olfati-Saber (2007), Millán et al. (2013), and references therein). The key feature of our framework is that it utilizes local information not only during the execution of the proposed distributed input and state estimation architecture but also in its design unlike the results in Millán et al. (2013); that is, global uniform ultimate boundedness of error dynamics is guaranteed once each node satisfies given local stability conditions independent from the graph topology and neighboring information of these nodes. Several illustrative numerical examples are further provided to demonstrate the efficacy of the proposed architecture. Note that the material in this paper was partially presented in Tran et al. (2017).

2. NOTATION AND MATHEMATICAL PRELIMINARIES

The notation used in this paper is fairly standard. Specifically, \(\mathbb{R}\) denotes the set of real numbers, \(\mathbb{R}^n\) denotes the set of \(n \times 1\) real column vectors, \(\mathbb{S}^{n \times m}_{+}\) denotes the set of \(n \times m\) real matrices, \(\mathbb{S}^{n \times n}_{+}\) (resp., \(\mathbb{S}^{n \times n}_{-}\)) denotes the set of \(n \times n\) positive-definite (resp., positive-semidefinite) real matrices, \(\mathbf{0}_n\) denotes the \(n \times 1\) vector of all zeros, \(\mathbf{I}_n\) denotes the \(n \times 1\) vector of all ones, and \(\mathbf{I}_n\) denotes the \(n \times n\) identity matrix. In addition, we write \((\cdot)^t\) for transpose, \((\cdot)^{-1}\) for generalized inverse, \(\lambda_{\min}(A)\) and \(\lambda_{\max}(A)\) for the minimum and maximum eigenvalue of the Hermitian matrix \(A\), respectively, \(\lambda_i(A)\) for the \(i\)-th eigenvalue of \(A\), where \(A\) is Hermitian and the eigenvalues are ordered from least to greatest value, diag(\(a\)) for the diagonal matrix with the vector \(a\) on its diagonal, \([x]_i\) for the entry of the vector \(x\) on the \(i\)-th row, and \([A]_{ij}\) for the entry of the matrix \(A\) on the \(i\)-th row and \(j\)-th column. Note that, throughout the paper, we use \(A > 0\) (resp., \(A \geq 0\)) to indicate \(A \in \mathbb{S}^{n \times n}_{+}\) (resp., \(A \in \mathbb{S}^{n \times n}_{+}\)).
We now recall some basic notions from graph theory and refer to textbooks Mesbahi and Egerstedt (2010) and Godsil et al. (2001) for details. Specifically, graphs are broadly adopted in the sensor networks literature to encode interactions between nodes. An undirected graph \( G \) is defined by a set \( V_G = \{1, \ldots, N\} \) of nodes and a set \( E_G \subset V_G \times V_G \) of edges. If \((i,j) \in E_G\), then the nodes \( i \) and \( j \) are neighbors and the neighboring relation is indicated with \( i \sim j \). The degree of a node is given by the number of its neighbors. Letting \( d_i \) be the degree of node \( i \), then the degree matrix of a graph \( G \), \( D(G) \in \mathbb{R}^{N \times N} \), is given by
\[
D(G) \triangleq \text{diag}(d_i), \quad d = [d_1, \ldots, d_N]^T.
\]

A path \( i_0i_1 \ldots i_L \) is a finite sequence of nodes such that \( i_{k-1} \sim i_k \), \( k = 1, \ldots, L \), and a graph \( G \) is connected if there is a path between any pair of distinct nodes. The adjacency matrix of a graph \( G \), \( A(G) \in \mathbb{R}^{N \times N} \), is given by
\[
[A(G)]_{ij} = \begin{cases} 1, & \text{if } (i,j) \in E_G, \\ 0, & \text{otherwise}. \end{cases}
\]

The Laplacian matrix of a graph, \( L(G) \in \mathbb{R}^{N \times N} \), playing a central role in many graph-theoretic treatments of sensor networks, is given by
\[
L(G) \triangleq D(G) - A(G).
\]

The spectrum of the Laplacian of an undirected and connected graph can be ordered as
\[
0 = \lambda_1(L(G)) < \lambda_2(L(G)) \leq \ldots \leq \lambda_N(L(G)) < \infty,
\]
with \( 1 \) as the eigenvector corresponding to the zero eigenvalue \( \lambda_1(L(G)) \) and \( L(G)1_N = 0_N \) and \( e^{L(G)}1_N = 1_N \). Throughout this paper, we assume that the graph \( G \) of a given sensor network is undirected and connected.

To develop the main results of this paper, the following lemmas are necessary.

**Lemma 1** (Proposition 8.1.2, Bernstein (2009)). Let \( A \in \mathbb{R}^{n \times n} \) and \( B \in \mathbb{R}^{n \times m} \). If \( A \geq 0 \) and \( B > 0 \), then \( A + B > 0 \).

**Lemma 2** (Proposition 8.2.4, Bernstein (2009)). Let \( A \in \mathbb{R}^{n \times n} \), \( B \in \mathbb{R}^{n \times m} \), \( C \in \mathbb{R}^{m \times m} \), and
\[
X = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix}.
\]

Then, the following statements are equivalent:
(i) \( X \geq 0 \).
(ii) \( A \geq 0 \), \( C + B^TA^{1/2}B \geq 0 \), \( (I - AA^{1/2})B = 0 \).
(iii) \( C \geq 0 \), \( A - BC^{1/2}B^T \geq 0 \), \( (I - CC^{1/2})B^T = 0 \).

Finally, \( C_0\Omega \) is defined as a polytope or a bounded polyhedron, which is the intersection of a finite number of halfspace and hyperplanes (Boyd and Vandenberghe, 2004). For the following lemma, let \( P \in \mathbb{R}^{n \times n} \), \( A(t) \in \mathbb{R}^{n \times n} \), \( C_0\Omega \triangleq \text{Co}\{A_1, \ldots, A_L\} \), and \( A(t) \in \text{Co}\{A_1, \ldots, A_L\} \) where \( A(t) \) denotes the convex hull and \( A_i \in \mathbb{R}^{n \times n} \) are the vertices of the polytope.

**Lemma 3** (Boyd et al., 1994). If \( P > 0 \), \( A_1^TP + PA_1 \leq 0 \) holds, then \( A_1^TP + PA_1 \geq 0 \) holds.

By letting \( P = I_n \), it follows from Lemma 3 that \( A_1^TP + A_1 \leq 0 \) holds, when \( A_1^TP + A_1 \leq 0 \) holds. If, in addition, \( A(t) \) is symmetric, then it further follows that \( A(t) \leq 0 \) holds, if \( A_1 \leq 0 \).

### 3. DISTRIBUTED INPUT AND STATE ESTIMATION FOR ACTIVE-PASSIVE SENSOR NETWORKS WITH FIXED NODE ROLES

In this section, we introduce and analyze a distributed input and state estimation architecture for heterogeneous sensor networks, where the active and passive role of each node is fixed. For this purpose, consider a system of interest with the (open-loop or closed-loop) dynamics given by
\[
\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0,
\]
where \( x(t) \in \mathbb{R}^n \) denotes an unmeasurable process state vector, \( w(t) \in \mathbb{R}^p \) denotes an unknown bounded input (e.g., command) to this process with a bounded time rate of change, \( A \in \mathbb{R}^{n \times n} \) denotes the Hurwitz system matrix necessary for internal process stability, and \( B \in \mathbb{R}^{n \times p} \) denotes the system input matrix.

Next, consider a sensor network with \( N \) nodes exchanging information among each other using their local measurements according to an undirected and connected graph \( G \). In the sense of Yucelen (2014), Yucelen and Peterson (2014, 2016), Peterson and Yucelen (2015, 2016), and Peterson et al. (2015, 2016), if a node \( i \), \( i = 1, \ldots, N \), is subject to observations of the process equation (5) given by
\[
y_i(t) = C_i x(t),
\]
where \( y_i(t) \in \mathbb{R}^{m_i} \) and \( C_i \in \mathbb{R}^{m_i \times n} \) denote the measurable process output and the system output matrix for node \( i \), \( i = 1, \ldots, N \), respectively, then we say that it is an active node. Similarly, if a node \( i \), \( i = 1, \ldots, N \), has no observations, then we say that it is a passive node. Notice that the above formulation allows for non-identical sensor modalities since \( C_i \) of active nodes can be different. Here, as standard in the literature, we assume that each active node has complimentary properties distributed over the sensor network to guarantee collective observability (see, for example, Olfati-Saber (2007), Millán et al. (2013), and references therein), although the pairs \((A, C_i)\), \( i = 1, \ldots, N \), may not be locally observable. In mathematical sense, collective observability condition means that the pair \((A, C)\) is observable, where \( C = [C_1^T, C_2^T, \ldots, C_N^T]^T \) (e.g., see Millán et al. (2013)).

Here, we are interested in the problem of distributively estimating the unmeasurable state \( x(t) \) and the unknown input \( w(t) \) of the process given by equation (5) using a sensor network, where active nodes are subject to the observations given by equation (6).

For this purpose, the rest of this section is divided into two parts, where we first introduce the proposed distributed estimation architecture and then analyze its stability in detail using tools and methods from system theory.

#### 3.1. Proposed Distributed Estimation Architecture

For node \( i, i = 1, \ldots, N \), consider the distributed estimation algorithm given by
\[
\hat{x}_i(t) = (A - \gamma P_i^{-1})\hat{x}_i(t) + Bw_i(t) + g_{i}y_i(t) - C_i\hat{x}_i(t)
\]
\[
- \alpha P_i^{-1}\sum_{j \sim i} (\hat{x}_i(t) - \hat{x}_j(t)), \quad \hat{x}_i(0) = \hat{x}_0,
\]
\[ \dot{\hat{w}}(t) = g_iJ_i(y(t) - C_i\hat{x}_i(t)) - (\sigma_iK_i + \gamma I_p)\hat{w}(t) \]
\[ - \alpha \sum_{i\neq j} (\hat{w}_i(t) - \hat{y}_j(t)), \quad \hat{w}_0(0) = \hat{w}_0, \quad (8) \]

where \( \hat{x}_i(t) \in \mathbb{R}^n \) is a local state estimate of \( x(t) \) for node \( i \), \( \hat{w}_i(t) \in \mathbb{R}^p \) is a local input estimate of \( w(t) \) for node \( i \), \( L_i \in \mathbb{R}^{n \times m}, J_i \in \mathbb{R}^{p \times m} \) and \( K_i \in \mathbb{S}^n_+ \) are design matrices of node \( i \), and \( \alpha, \gamma \), and \( \sigma_i \in \mathbb{R} \) are positive design coefficients for node \( i \). Here, \( g_i = 1 \) for active nodes and otherwise \( g_i = 0 \). In addition, \( P_i > 0 \) is the consensus gain satisfying the linear matrix inequality given by

\[ R_i = \begin{bmatrix}
    \bar{A}_i^T P_i + P_i \bar{A}_i & -P_iB + g_iC_i^T J_i \\
    -B_i^T P_i + g_iJ_iC_i & -2\sigma_iK_i
\end{bmatrix} \leq 0, \quad (9) \]

where \( \bar{A}_i \triangleq A - g_iL_iC_i \). \quad (10)

**Remark 1.** The local condition given by equation (9) for node \( i \), \( i = 1, \ldots, N \), plays a central role in the stability analysis presented in the next section. Specifically, if the proposed input and state estimation architecture given by equations (7) and (8) satisfies the local condition given by equation (9) for each node, then the global uniform ultimate boundedness of error dynamics is guaranteed for the overall sensor network. In addition, note that the local condition given by equation (9) is well-posed. To see this, for example, let \( P_i \) satisfy the linear matrix inequality given by \( \bar{A}_i^T P_i + P_i \bar{A}_i < 0, i = 1, \ldots, N \). Then, it can be readily shown that there can exist a sufficiently large \( \sigma_i \), \( i = 1, \ldots, N \), such that equation (9) holds. As a special case, if all nodes are active and a well-known positive real condition \( P_iB = C_i^T J_i \) holds (see, for example, Corless and Tu (1998), Akhenak et al. (2004), Chen and Chowdhury (2007), Bowong and Tewa (2008), Mohamed et al. (2012), and references therein), then it can be easily seen that equation (9) holds even for small values of \( \sigma_i \), \( i = 1, \ldots, N \). From this standpoint, it should be also mentioned that equation (9) relaxes this condition \( P_iB = C_i^T J_i \) similar in spirit to how the authors of Kim (2011), Kim et al. (2011, 2015), Yucelen (2011), Yucelen et al. (2011), and Yucelen et al. (2015) relax similar conditions. Finally, once again, for the special case when all nodes are active, if \( \mathcal{H}(s) \triangleq I_{C_i}(s) - \bar{A}_i^{-1}B + \sigma_iK_i \) is passive, \( i = 1, \ldots, N \), then equation (9) is feasible and vice versa (Boyd et al., 1994).

### 3.2. Stability Analysis

Let \( \tilde{x}_i(t) \triangleq x(t) - \hat{x}_i(t) \) and \( \tilde{w}_i(t) \triangleq \hat{w}_i(t) - w(t) \). Then, based on equations (7) and (8),

\[ \dot{\tilde{x}}_i(t) = Ax(t) + Bw(t) - (A - \gamma P_i^{-1})\tilde{x}_i(t) - B\tilde{w}_i(t) \]
\[ - g_iL_i(y(t) - C_i\hat{x}_i(t)) + \alpha P_i^{-1} \sum_{i\neq j} (\tilde{x}_i(t) - \tilde{x}_j(t)) \]
\[ = (A - g_iL_iC_i)\tilde{x}_i(t) - B\tilde{w}_i(t) + \alpha P_i^{-1} \sum_{i\neq j} (\tilde{x}_i(t) - \tilde{x}_j(t)) \]
\[ + \gamma P_i^{-1}\tilde{x}_i(t) \]
\[ = \tilde{A}_i\tilde{x}_i(t) - B\tilde{w}_i(t) - \alpha P_i^{-1} \sum_{i\neq j} (\tilde{x}_i(t) - \tilde{x}_j(t)) \]
\[ - \gamma P_i^{-1}(\tilde{x}_i(t) - x(t)), \quad \tilde{x}_i(0) = \tilde{x}_0, \quad (11) \]

\[ \dot{\tilde{w}}_i(t) = g_iJ_iC_i\tilde{x}_i(t) - \sigma_iK_i(\tilde{w}_i(t) + w(t)) - \alpha \sum_{i\neq j} (\tilde{w}_i(t) - \tilde{w}_j(t)) \]
\[ - \gamma(\tilde{w}_i(t) + w(t)) - \tilde{w}_i(t), \quad \tilde{w}_0(0) = \tilde{w}_0. \quad (12) \]

Now, considering the aggregated vectors given by

\[ \tilde{x}(t) \triangleq [\tilde{x}_1^T(t), \tilde{x}_2^T(t), \ldots, \tilde{x}_N^T(t)]^T \in \mathbb{R}^{Nn}, \quad (13) \]
\[ \tilde{w}(t) \triangleq [\tilde{w}_1^T(t), \tilde{w}_2^T(t), \ldots, \tilde{w}_N^T(t)]^T \in \mathbb{R}^{Np}, \quad (14) \]

we can write the error dynamics as

\[ \dot{\tilde{x}}(t) = \begin{bmatrix} \tilde{A}_1 & 0 & \cdots & 0 \\ 0 & \tilde{A}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \tilde{A}_N \end{bmatrix} \tilde{x}(t) - \begin{bmatrix} \tilde{B} & 0 & \cdots & 0 \\ 0 & \tilde{B} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \tilde{B} \end{bmatrix} \tilde{w}(t) - \alpha \begin{bmatrix} \tilde{L}_{11}P_1^{-1} & \tilde{L}_{12}P_1^{-1} & \cdots & \tilde{L}_{1N}P_1^{-1} \\ \tilde{L}_{21}P_2^{-1} & \tilde{L}_{22}P_2^{-1} & \cdots & \tilde{L}_{2N}P_2^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{L}_{NN}P_N^{-1} & \tilde{L}_{N2}P_N^{-1} & \cdots & \tilde{L}_{NN}P_N^{-1} \end{bmatrix} \tilde{x}(t) - \gamma \begin{bmatrix} \tilde{P}_1^{-1} \\ \vdots \\ \tilde{P}_N^{-1} \end{bmatrix} \tilde{x}(t), \quad (15) \]

\[ \dot{\tilde{w}}(t) = \begin{bmatrix} g_1J_1C_1 & 0 & \cdots & 0 \\ 0 & g_2J_2C_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_NJ_NC_N \end{bmatrix} \tilde{w}(t) - \begin{bmatrix} \sigma_1K_1 & 0 & \cdots & 0 \\ 0 & \sigma_2K_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_NK_N \end{bmatrix} \tilde{w}(t) - \alpha \begin{bmatrix} \tilde{L}_{11}I_p & \tilde{L}_{12}I_p & \cdots & \tilde{L}_{1N}I_p \\ \tilde{L}_{21}I_p & \tilde{L}_{22}I_p & \cdots & \tilde{L}_{2N}I_p \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{L}_{NN}I_p & \tilde{L}_{N2}I_p & \cdots & \tilde{L}_{NN}I_p \end{bmatrix} \tilde{w}(t) - \gamma \begin{bmatrix} I_p & 0 & \cdots & 0 \\ 0 & I_p & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I_p \end{bmatrix} \tilde{w}(t), \quad (16) \]

where \( \tilde{L}_{ij} \in \mathbb{R} \) is the entry in the \( i \)-th row and \( j \)-th column of the Laplacian matrix.

The error dynamics now can be written a compact form as

\[ \dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) - (I_{N \times N} \otimes B)\tilde{w}(t) - P^{-1}(F \otimes I_p)\tilde{x}(t) \]
\[ + \gamma P^{-1}(I_{N \times N} \otimes I_p)\tilde{x}(t), \quad (17) \]
\[ \dot{\tilde{w}}(t) = M\tilde{x}(t) - K(\tilde{w}(t) + I_{N \times I_p}w(t)) - (F \otimes I_p)\tilde{w}(t) \]
\[ - \gamma(I_{N \times I_p}w(t) - I_{N \times I_p}\tilde{w}(t), \quad (18) \]
where
\[ A \triangleq \text{diag}([A_1, A_2, \ldots, A_N]), \]
\[ M \triangleq \text{diag}([g_{i1}C_1, g_{i2}C_2, \ldots, g_{iN}C_N]), \]
\[ K \triangleq \text{diag}([\sigma_1K_1, \sigma_2K_2, \ldots, \sigma_NK_N]), \]
\[ F \triangleq \alpha L(G) + \gamma I_N, \]
\[ P \triangleq \text{diag}([P_1, P_2, \ldots, P_N]), \]
with \( L(G) \) being the Laplacian matrix. Note that \( P > 0 \) readily follows from \( P_i > 0 \).

**Theorem 1.** Consider the process given by equation (5) and the distributed input and state estimation architecture given by equations (7) and (8). Assume equation (9) holds and nodes exchange information using local measurements subject to an undirected and connected graph \( G \). Then, the error dynamics given by equations (17) and (18) are uniformly bounded.

_Proof._ Consider the Lyapunov function candidate given by
\[ V(x, \dot{w}) = \dot{x}^T P \ddot{w} + \ddot{w}^T \dot{w}. \] (24)

Note that \( V(0, 0) = 0 \) and \( V(x, \dot{w}) > 0 \) for all \( (x, \dot{w}) \neq (0, 0) \). Taking time-derivative of \( V(x, \dot{w}) \) along the trajectories of equations (17) and (18) yields
\[ \dot{V}(\cdot) = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(P(1_N \otimes B)\ddot{w}(t)
\]
\[ - 2\dot{x}^T(t)(F \otimes I_p)\ddot{w}(t)
\]
\[ - 2\ddot{x}^T(t)(\dot{K} + \gamma I_{2p})(1_N \otimes I_p)\ddot{w}(t)
\]
\[ = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(\dot{K}\ddot{w}(t)
\]
\[ - 2\dot{x}^T(t)(P(1_N \otimes B) - M T)\ddot{w}(t)
\]
\[ - 2\ddot{x}^T(t)(F \otimes I_p)\ddot{w}(t)
\]
\[ - 2\ddot{x}^T(t)(\dot{K} + \gamma I_{2p})(1_N \otimes I_p)\ddot{w}(t)
\]
\[ = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(\dot{K}\ddot{w}(t)
\]
\[ + \ddot{x}^T(t)(\ddot{w}(t))
\]
\[ + 2 \gamma (1_N \otimes I_p)w(t)\ddot{w}(t)
\]
\[ = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(\dot{K}\ddot{w}(t)
\]
\[ + \ddot{x}^T(t)(\ddot{w}(t))
\]
\[ + 2 \gamma (1_N \otimes I_p)w(t)\ddot{w}(t)
\]
\[ = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(\dot{K}\ddot{w}(t)
\]
\[ + \ddot{x}^T(t)(\ddot{w}(t))
\]
\[ + 2 \gamma (1_N \otimes I_p)w(t)\ddot{w}(t)
\]
\[ = \dot{x}^T(t)(\dot{A}_iP + PA\dot{x}(t)) - 2\ddot{x}^T(t)(\dot{K}\ddot{w}(t)
\]
\[ + \ddot{x}^T(t)(\ddot{w}(t))
\]
\[ + 2 \gamma (1_N \otimes I_p)w(t)\ddot{w}(t)
\]
\[ = z^T(t)R_Az(t) + z^T(t)R_Az(t) + 2\ddot{x}^T(t)\phi.
\] (25)

where
\[ z(t) \triangleq [\dot{x}^T(t), \ddot{w}(t)]^T, \]
\[ R_A \triangleq \begin{bmatrix} \dot{A}_iP + PA & -P(1_N \otimes B) + M T \\ -(1_N \otimes B)^T P + M & -2\dot{K} \end{bmatrix}, \]
\[ R_B \triangleq \begin{bmatrix} -2(F \otimes I_p) & 0 \\ 0 & -2(F \otimes I_p) \end{bmatrix}, \]
\[ R \triangleq R_A + R_B \]
\[ \phi \triangleq \begin{bmatrix} \gamma(1_N \otimes I_p)x(t) \\ -(\dot{K} + \gamma I_{2p})(1_N \otimes I_p)w(t) - (1_N \otimes I_p)\ddot{w}(t) \end{bmatrix}, \]
\[ \phi \triangleq \sqrt{\frac{\gamma^2}{2}\|1_N \otimes I_p\|^2_x^2 + \|\dot{K} + \gamma I_{2p}\|_2^2\|1_N \otimes I_p\|^2_w^2 + \|1_N \otimes I_p\|^2_w^2}, \]
\[ \phi \triangleq \sqrt{N\gamma^2\|x(t)\|_2^2 + \|\dot{K} + \gamma I_{2p}\|_2^2\|w(t)\|_2^2 + \|\dot{w}(t)\|_2^2}, \]
\[ \phi \triangleq \sqrt{\frac{\gamma^2}{2}\|x(t)\|_2^2 + \|\dot{K} + \gamma I_{2p}\|_2^2\|w(t)\|^2_w + \|\dot{w}(t)\|^2_w}. \] (37)
Now, one can write
\[
\dot{V}(\cdot) = z^T(t)Rz(t) + 2z^T(t)\phi \\
\leq \lambda_{\text{max}}(R)\|z(t)\|_2^2 + 2\|z(t)\|_2 \phi \\
\leq (1-\theta)\lambda_{\text{max}}(R)\|z(t)\|_2^2 + \theta\lambda_{\text{max}}(R)\|z(t)\|_2^2 + 2\|z(t)\|_2 \phi,
\]
(38)
with \(\lambda_{\text{max}}(R) < 0\) and \(\theta \in (0,1)\). Letting \(\mu_1 \triangleq -\frac{2\phi}{\lambda_{\text{max}}(R)} > 0\) and \(\Omega_1 \triangleq \{z(t) : \|z(t)\|_2 \leq \mu_1\}\), it follows that \(\dot{V}(\cdot) \leq (1-\theta)\lambda_{\text{max}}(R)\|z(t)\|_2^2 < 0\) outside the compact set \(\Omega_1\) and, hence, the error dynamics given by equations (17) and (18) are uniformly ultimately bounded by Theorem 4.18 of Khalil (2002).

The following corollary to the above theorem is now immediate.

**Corollary 1.** Consider the process given by equation (5) and the distributed input and state estimation architecture given by equations (7) and (8). Assume that equation (9) holds and nodes exchange information using local measurements subject to an undirected and connected graph \(G\). Then, for all \(z(0) \in \mathbb{R}^{(n+p)}\), there exists \(T = T(z(0), \mu_1) \geq 0\) such that
\[
\|\tilde{x}(t)\|_2 \leq \xi_1 \triangleq \sqrt{\frac{\lambda_{\text{max}}(P)}{\lambda_{\text{min}}(P)}} \max\{\|z(0)\|_2\} e^{(1-\theta)\lambda_{\text{max}}(R)/2\lambda_{\text{max}}(P)}t, \quad \forall t \geq 0,
\]
(39)
\[
\|\tilde{w}(t)\|_2 \leq \xi_1, \quad \forall t \geq 0,
\]
(40)
where
\[
P = \begin{bmatrix} P & 0 \\ 0 & 1_{Np} \end{bmatrix},
\]
(41)
and
\[
\|\tilde{x}(t)\|_2 \leq \psi_1 \triangleq \sqrt{\frac{\lambda_{\text{max}}(P)}{\lambda_{\text{min}}(P)}} \mu_1, \quad t \geq T,
\]
(42)
\[
\|\tilde{w}(t)\|_2 \leq \xi_1 \triangleq \sqrt{\lambda_{\text{max}}(P)} \mu_1, \quad t \geq T.
\]
(43)

**Proof.** Note that
\[
\dot{V}(\cdot) = z^T(t)R \tilde{x}(t) + \tilde{w}^T(t) \tilde{w}(t) \\
= [z^T(t) - \tilde{w}^T(t)] \begin{bmatrix} P & 0 \\ 0 & 1_{Np} \end{bmatrix} \begin{bmatrix} \tilde{x}(t) \\\ \tilde{w}(t) \end{bmatrix} \\
= z^T(t)Pz(t).
\]
(44)

Let \(c_1 \triangleq \lambda_{\text{min}}(P), c_2 \triangleq \lambda_{\text{max}}(P), \) and \(c_3 \triangleq -(1-\theta)\lambda_{\text{max}}(R)\). From equation (44), we have
\[
c_1 \|z(t)\|_2^2 \leq \dot{V}(\cdot) \leq c_2 \|z(t)\|_2^2.
\]
(45)

In addition, \(\dot{V}(\cdot) \leq -c_3\|z(t)\|_2^2\) for all \(\|z(t)\|_2 \geq \mu_1\). By Theorem 4.5 of Khalil (2014), since the domain \(D = \mathbb{R}^{(n+p)}\), for every initial state \(z(0)\), the bound of the overall system is
\[
\|z(t)\|_2 \leq \sqrt{\frac{c_1}{c_3}} \max\{\|z(0)\|_2 e^{-(\gamma/2c_3)t}, \mu_1\} = \xi_1, \quad \forall t \geq 0.
\]
(46)

Using the fact that \(\|\tilde{x}(t)\|_2 \leq \|z(t)\|_2 \) and \(\|\tilde{w}(t)\|_2 \leq \|z(t)\|_2\), equations (39) and (40) follow immediately.

In the proof of Theorem 1, we show that \(V(\cdot)\) cannot grow outside the compact set \(\Omega_1\), thus equation (42) follows from \(\lambda_{\text{min}}(P)\)
\[
\|\tilde{x}(t)\|_2 \leq V(\tilde{x}(t)) \leq \lambda_{\text{max}}(P)\|z(t)\|_2 \leq \lambda_{\text{max}}(P)\mu_1.
\]
(47)
Identically, equation (43) follows from \(\|\tilde{w}(t)\|_2 \leq V(\tilde{x}(t), \tilde{w}(t)) \leq \lambda_{\text{max}}(P)\|z(t)\|_2 \leq \lambda_{\text{max}}(P)\mu_1^2\). The proof is now complete. 

**Remark 2.** While this paper shows the uniform ultimate boundedness of the error dynamics, the provided results equations (42) and (43) can be used to tune the design parameters to achieve acceptable performance criteria. The uniform ultimate boundedness can be considered as a result of the considered complex problem that we address here, which we can recap the main points as:

(i) The proposed algorithm only utilizes local information for designing agent-wise dynamics to achieve the stability, unlike existing results in Millán et al. (2013).
(ii) With regard to the considered problem in this paper, for the first time, we allow a subset of nodes to be passive (that is, subject to no observation).
(iii) The sensing capability of active nodes can be different among sensors.
(iv) Not only the states of the process are unknown but also the inputs are unknown.
(v) We do not assume a common positive real condition, e.g., \(PB = C_1^T H_1\), which in practice may not be easy to satisfy.
(vi) The inputs are not constant.

If we relax some of these conditions, the asymptotic stability can be obtained with a version of the proposed algorithm (see Appendix A in Data Sheet in Supplementary Material).

To summarize, the nature of the distributed estimation problem subjected to (i)–(vi) is challenging. In order to solve this problem using only local information and without the positive real condition, the condition equation (9) is required by the nature of the problem. In addition, we need the assumption that \(A\) is Hurwitz to make equation (9) feasible especially for passive nodes. Furthermore, when the input \(w(t)\) is time-varying, adding leakage terms is unavoidable to prove the stability (e.g., see Ioannou and Sun (2012)).

**Remark 3.** Since the ultimate bounds given by equations (42) and (43) depend on the design parameters of the proposed distributed input and state estimation architecture, they can be used as design metrics such that the design parameters can be judiciously selected to make equations (42) and (43) small. However, unlike the stability of our framework that is guaranteed once each node satisfies the local condition given by equation (9), such a performance characterization requires global information. However, one can further analyze the effect of each specific design parameter to these ultimate bounds and make conclusions without possibly requiring global information, which will be considered as a future research direction.

The following remarks discuss how to choose our design parameters while Appendix C in Data Sheet in Supplementary Material summarizes their effect for interested readers.

**Remark 4.** Note that the terms “\(-\gamma P^{-1} \tilde{x}(t)\)” and “\(-(\sigma K_i + \gamma_1 \hat{p}) \tilde{w}(t)\)” appearing, respectively, in equations (7) and (8)
are often referred as leakage terms. If the gains \( \gamma P_i^{-1} \) and \( \sigma_i K_i + \gamma I \) respectively, multiplying these terms are not small, then they may result in poor overall system performance (see, for example, Volyanskyy et al. (2009), Yucelen and Haddad (2013), and references therein) and, hence, it is of common practice to choose these multiplier gains \( \gamma \) and \( \sigma_i \) to be small. However, as noted in Remark 1, \( \sigma_i \) may not be chosen as small unless all nodes are active and the condition \( P_i B = C_i^T I_i \) holds. Therefore, we cast equation (9) as an optimization problem given by

\[
\begin{align*}
\text{minimize} & \quad \sigma_i, \\
\text{subject to} & \quad (9),
\end{align*}
\]

for all nodes \( i = 1, \ldots, N \). In addition, it should be noted that since the matrix \( K \) appear in the numerator of the ultimate bound, \( \sigma_i \) and \( K_i \) should be chosen such that the norm \( \|\sigma_i K_i\|_2 \) is small.

**Remark 5.** To elucidate the effect of design parameters to the ultimate bound given by equation (42), we consider, for example, a system with 4 sensors (1 and 3 are active nodes, and 2 and 4 are passive nodes) tracking a target with dynamics

\[
\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & -0.25 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} w(t),
\]

where \( w(t) = \sin(0.25t) \). Node 1 is subject to \( C_1 = [1 \ 0] \) and node 3 is subject to \( C_3 = [0 \ 1] \). We design \( \sigma_i \) by solving the linear matrix inequality equation (9). As a result, with \( I_i = K_i \) and \( K_1 = K_2 = K_3 = K_4 = 50 \), we have \( \sigma_1 = 0.03, \sigma_2 = \sigma_3 = 0.05 \) and \( \sigma_4 = 0.03 \) with \( P_1 = \begin{bmatrix} 26.31 & -2.67 \\ -2.67 & 4.01 \end{bmatrix}, P_2 = P_4 = \begin{bmatrix} 1.54 & 0.08 \\ 0.08 & 1.62 \end{bmatrix} \) and \( P_3 = \begin{bmatrix} 4.60 & 4.02 \\ 4.02 & 25.62 \end{bmatrix} \). We then vary \( \sigma \) and \( \gamma \) to see the effect of these parameters to the ultimate bound \( \psi_1 \) given by equation (42). Figure 1 shows the effect of the variation in \( \alpha \) and \( \gamma \) to equation (42). From the figure, we can see that one can pick a small value for \( \gamma \) and a large value for \( \alpha \) to reduce the ultimate bound.

### 3.3. Illustrative Numerical Example

We now present several numerical examples to illustrate the results given earlier in this section. For this purpose, consider a process composed of two decoupled systems with the dynamics given by equation (5), where

\[
A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega^2_{n1} & -2\omega_n \xi_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega^2_{n2} & -2\omega_n \xi_2 \end{bmatrix},
\]

\[
B = \begin{bmatrix} 0 & 0 \\ \omega^2_{n1} & 0 \\ 0 & 0 \\ 0 & \omega^2_{n2} \end{bmatrix},
\]

\( \omega_{n1} = 1.2, \xi_1 = 0.9, \omega_{n2} = 0.5, \) and \( \xi_2 = 0.6 \). This process, for example, can represent a linearized vehicle model with the first and third states corresponding to the positions in the \( x \) and \( y \) directions, respectively, while the second and fourth states corresponding to the velocities in the \( x \) and \( y \) directions, respectively. The initial conditions are set to \( x_0 = [-3, 0.5, 2.5, 0.25] \). In addition, we consider the input is given by

\[
w(t) = \begin{bmatrix} 2.5 \sin(t) \\ 4 \cos(1.2t) \end{bmatrix},
\]

To maintain the readability of the paper, the values of \( L_i, \sigma_i, P_i \) in the following examples are put in Appendix B in Data Sheet in Supplementary Material.

**Example 1.** For the first example, we consider a sensor network with 12 nodes exchanging information over an undirected and connected graph topology, where there are 4 active nodes and 8 passive nodes as shown in Figure 2. Each node's sensing capability is represented by equation (6) with the output matrices

\[
C_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},
\]

\[
D_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},
\]

\[
H_i = \begin{bmatrix} 0.03 & 0.03 & 0.03 & 0.03
\end{bmatrix},
\]

\[
\gamma = 1, \quad \sigma = 0.03, \quad \alpha = 0.03, \quad \psi_1 = [1 \ 1 \ 1 \ 1], \quad \psi_2 = [1 \ 1 \ 1 \ 1]
\]

\[
A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega^2_{n1} & -2\omega_n \xi_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega^2_{n2} & -2\omega_n \xi_2 \end{bmatrix},
\]

\[
B = \begin{bmatrix} 0 & 0 \\ \omega^2_{n1} & 0 \\ 0 & 0 \\ 0 & \omega^2_{n2} \end{bmatrix},
\]

\[
(50)
\]

\[
(51)
\]

\[
(52)
\]

**FIGURE 1** | Effect of \( \gamma \in [0.2, 2] \) and \( \alpha \in [0.25, 1, 2.5, 5, 10, 50] \) to the ultimate bound \( \psi_1 \) in equation (42), where the arrow indicate the direction \( \alpha \) is increased.
for the odd index nodes and
\[
C_i = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\] (54)

for the even index nodes. In addition, all nodes are subject to zero initial conditions and we set \( I_i = K_i = \text{diag}([100; 100]) \), \( \alpha = 50 \), and \( \gamma = 0.1 \). For the observer gain \( L_i \), the odd index nodes are subject to [equation (A.6) in Supplementary Material] while the even index nodes are subject to [Equation (A.7) in Supplementary Material].

By solving the linear matrix inequality equation (9) for each node, \( \sigma_1 \) and \( P_i > 0 \) are obtained as \( \sigma_1 = \sigma_5 = \sigma_6 = \sigma_7 = \sigma_8 = \sigma_9 = \sigma_{10} = \sigma_{11} = \sigma_{12} \) where \( \sigma_1, \sigma_2, \) and \( \sigma_3 \) are subject to equations (A.8), (A.9), and (A.10) in Supplementary Material, respectively. In addition, \( P_1, P_2, \) and \( P_{12} \) are subject to equations (A.11), (A.12), and (A.13) in Supplementary Material, respectively. Note that \( P_1 = P_5 = P_6 = P_7 \), \( P_2 = P_4 = P_8 = P_9 = P_{10} = P_{11} = P_{12} \), where \( P_1, P_2, \) and \( P_{12} \) are the same as equations (A.11), (A.12), and (A.13) in Supplementary Material, respectively. Other parameters and gains are also kept the same. Figures 6 and 7 show the performance of the sensor network for the proposed distributed estimation architecture. In addition, in order to compare the performance of Example 1 and Example 2, the state and input error norms of both examples are plotted in Figures 8 and 9, respectively. The transient responses are captured in the figures approximately during the first 2 or 3 s, and it can be seen that Example 2 converges faster than Example 1 in state estimation, yet it encounters overshoot in input estimation. In addition, we can roughly approximate the average of both state and input error norms are reduced by a factor of 2 in Example 2 compared to Example 1. In general, Examples 1 and 2 show that the steady-state performance is improved by increasing the number of active nodes in the sensor network.

**Example 2.** In this example, we increase the number of active nodes in the sensor network to 8 as depicted in Figure 5. The sensing capability of each agent is the same as in Example 1. Note that, because of the change in the number of active nodes, the design parameters are adjusted accordingly as \( \sigma_1 = \sigma_3 = \sigma_5 = \sigma_7, \sigma_2 = \sigma_4 = \sigma_6 = \sigma_8, \sigma_9 = \sigma_{10} = \sigma_{11} = \sigma_{12} \) where \( \sigma_1, \sigma_2, \) and \( \sigma_3 \) are subjected to equations (A.14), (A.15), and (A.16) in Supplementary Material, respectively. In addition, \( P_1 = P_3 = P_5 = P_7, P_2 = P_4 = P_8, P_9 = P_{10} = P_{11} = P_{12} \), where \( P_1, P_2, \) and \( P_{12} \) are the same as equations (A.11), (A.12), and (A.13) in Supplementary Material, respectively. Other parameters and gains are also kept the same.

![Communication graph of the sensor network in Example 1 with 4 active nodes and 8 passive nodes](image1)

**FIGURE 2** Communication graph of the sensor network in Example 1 with 4 active nodes and 8 passive nodes (lines denote communication links, squares denote active nodes, and circles denote passive nodes) Tran et al. (2017).

![State estimates of the sensor network in Example 1 with 4 active nodes and 8 passive nodes under the proposed architecture equations (7) and (8) (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes)](image2)

**FIGURE 3** State estimates of the sensor network in Example 1 with 4 active nodes and 8 passive nodes under the proposed architecture equations (7) and (8) (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes) Tran et al. (2017).

Examples 2 and 3 show that the steady-state performance is improved by increasing the number of active nodes in the sensor network. In addition, in order to compare the performance of Example 1 and Example 2, the state and input error norms of both examples are plotted in Figures 8 and 9, respectively.
4. DISTRIBUTED INPUT AND STATE ESTIMATION FOR ACTIVE-PASSIVE SENSOR NETWORKS WITH VARYING NODE ROLES

We now generalize the results of the previous section to the case when the active and passive role of each sensor node is varying over time. For this purpose, once again, we consider a process given by equations (5). In addition, if a node in the sensor network is active for some time instant, then it is subject to the observations of the process given by equation (6) on that time instant, otherwise it is a passive node and has no observation. Note that a node is assumed to be smoothly changed back and forth between active and passive mode (i.e., \( g(t) \) is a smooth function on the interval \([0, 1]\)). The proposed algorithm is discussed in Section 4.1, followed by the stability analysis (Section 4.2), and a numerical example is presented to illustrate the efficacy of the methods (Section 4.3).

4.1. Proposed Distributed Estimation Architecture

For node \( i, i = 1, \ldots, N \), consider the distributed estimation algorithm given by

\[
\dot{\hat{x}}_i(t) = (A - \gamma P_i^{-1}) \hat{x}_i(t) + B \hat{w}_i(t) + g_i(t)L_i(y_i(t) - C_i \hat{x}_i(t)) - \alpha P_i^{-1} \sum_{j \neq i} (\hat{x}_j(t) - \hat{x}_i(t)), \quad \hat{x}_i(0) = \hat{x}_0, \tag{58}
\]

\[
\dot{\hat{w}}_i(t) = g_i(t)J_i(y_i(t) - C_i \hat{x}_i(t)) - (\sigma_i K_i + \gamma I_p) \hat{w}_i(t) - \alpha \sum_{j \neq i} (\hat{w}_j(t) - \hat{w}_i(t)), \quad \hat{w}_i(0) = \hat{w}_0, \tag{59}
\]

where \( \hat{x}_i(t) \in \mathbb{R}^n \) is a local state estimate of \( x(t) \) for node \( i \), \( \hat{w}_i \in \mathbb{R}^p \) is a local input estimate of \( w(t) \) for node \( i \), \( L_i \in \mathbb{R}^{n \times m} \), \( J_i \in \mathbb{R}^{p \times m} \), and \( K_i \in \mathbb{R}^{p \times p} \) are design matrices of node \( i \), and \( \alpha, \gamma, \) and \( \sigma_i \) are positive design coefficients for node \( i \). Note that the

where \( C_1 = C_5 = C_9, \quad C_2 = C_4 = C_6 = C_8 = C_{10} = C_{12}, \) and \( C_3 = C_7 = C_{11} \). Note that for the odd index nodes, the pair \( (A, C_i) \) is not observable. We also choose \( J_i = K_i = \text{diag}([100; 100]) \), \( \alpha = 50 \), and \( \gamma = 0.1 \).

Here, the observer gain \( L_i \) is chosen such that \( L_1 = L_5 = L_9 \), \( L_2 = L_4 = L_6 = L_8 = L_{10} = L_{12} \), and \( L_3 = L_7 = L_{11} \) where \( L_1, L_2, \) and \( L_3 \) are subject to equations (A.17), (A.18), and (A.19) in Supplementary Material, respectively. By solving the linear matrix inequality equation (9) for each node, \( \sigma_i \) and \( P_i > 0 \) are obtained as \( \sigma_1 = \sigma_5, \sigma_2 = \sigma_4 = \sigma_6, \sigma_8, \sigma_3 = \sigma_7 \), and \( \sigma_9 = \sigma_{10} = \sigma_{11} = \sigma_{12} \) where \( \sigma_1, \sigma_2, \sigma_3, \) and \( \sigma_9 \) are subject to equations (A.20), (A.21), (A.22), and (A.23) in Supplementary Material, respectively. In addition, \( P_1 = P_3, \quad P_2 = P_4 = P_5 = P_8, \quad P_3 = P_7, \) and \( P_6 = P_9 = P_{10} = P_{11} = P_{12} \) where \( P_1, P_2, P_3, \) and \( P_{12} \) are subject to equations (A.24), (A.12), (A.25), and (A.13) in Supplementary Material, respectively. Figures 10 and 11 show that under the proposed distributed estimation architecture, nodes are able to closely estimate the process states and inputs, although some active nodes are not able to fully observe the process.
FIGURE 6 | State estimates of the sensor network in Example 2 with 8 active nodes and 4 passive nodes under the proposed architecture equations (7) and (8) (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes) Tran et al. (2017).

FIGURE 7 | Input estimates of the sensor network in Example 2 with 8 active nodes and 4 passive nodes under the proposed architecture equations (7) and (8) (the dash lines denote the inputs of the actual process and the solid lines denote the input estimates of nodes) Tran et al. (2017).

FIGURE 8 | State error norms of the sensor networks in Example 1 and Example 2.
FIGURE 9 | Input error norms of the sensor networks in Example 1 and Example 2.

FIGURE 10 | State estimates of the sensor network in Example 3 with 12 active nodes under the proposed architecture equations (7) and (8) (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes) Tran et al. (2017).

FIGURE 11 | Input estimates of the sensor network in Example 3 with 12 active nodes under the proposed architecture equations (7) and (8) (the dash lines denote the inputs of the actual process and the solid lines denote the input estimates of nodes) Tran et al. (2017).
parameter \( g(t) \) in this section is time-varying and \( g(t) \in [0, 1] \). In addition, \( P_i > 0 \) is the consensus gain satisfying the two linear matrix inequalities given by

\[
\begin{align*}
R_{1i} &\triangleq \begin{bmatrix} A^TP_i + P_iA & -P_iB \\ -B^TP_i & -2\sigma_i K_i \end{bmatrix} \leq 0, \\
R_{2i} &\triangleq \begin{bmatrix} (A - L_i C_i)^TP_i + P_i(A - L_i C_i) & -P_iB + C_i^TP_i \\ -B^TP_i + I_iC_i & -2\sigma_i K_i \end{bmatrix} \leq 0, \quad i = 1, \ldots, N.
\end{align*}
\]  

(60)  

(61)

### 4.2. Stability Analysis

Let \( \hat{x}_i(t) \triangleq x(t) - \hat{x}_i(t) \) and \( \hat{w}_i(t) \triangleq \hat{w}_i(t) - w(t) \). Then, similar to equations (11) and (12), one can write

\[
\begin{align*}
\dot{\hat{x}}_i(t) &= \hat{A}_i(t)\hat{x}_i(t) - B\hat{w}_i(t) - \alpha P_i^{-1} \sum_{j \neq i} \hat{x}_j(t) - \hat{x}_i(0), \\
\dot{\hat{w}}_i(t) &= g_i(t)J_iC_i\hat{x}_i(t) - \gamma_iK_i(\hat{w}_i(t) + w(t)) - \alpha \sum_{j \neq i} \hat{w}_j(t) - \hat{w}_i(0),
\end{align*}
\]

where

\[
\hat{A}_i(t) \triangleq A - g_i(t)L_iC_i.
\]

(62)  

(63)  

(64)

Therefore, similar to Section 3.2, the compact form of the error dynamics are given by

\[
\begin{align*}
\dot{\hat{x}}(t) &= \hat{A}(t)\hat{x}(t) - (I_N \otimes B)\hat{w}(t) - P^{-1}(F \otimes I_n)\hat{x}(t) + \gamma P^{-1}(I_N \otimes I_n)\hat{x}(t), \\
\dot{\hat{w}}(t) &= M(t)\hat{x}(t) - K(\hat{w}(t) + (I_N \otimes I_p)w(t)) - (F \otimes I_p)\hat{w}(t) - \gamma(1_N \otimes I_p)w(t) - (1_N \otimes I_p)\hat{w}(t),
\end{align*}
\]

where

\[
\begin{align*}
\hat{A}(t) &\triangleq \text{diag}([\hat{A}_1(t), \hat{A}_2(t), \ldots, \hat{A}_N(t)]), \\
M(t) &\triangleq \text{diag}([g_1(t)J_1C_1, g_2(t)J_2C_2, \ldots, g_N(t)J_NC_N]),
\end{align*}
\]

(65)  

(66)  

(67)  

(68)

and \( \hat{K}, F, \) and \( P \) are the same as equations (21), (22), and (23), respectively.

**Theorem 2.** Consider the process given by equation (5) and the distributed input and state estimation architecture given by equations (58) and (59). Assume equations (60) and (61) hold and nodes exchange information according to the undirected connected graph \( \mathcal{G} \). Then, the error dynamics given by equations (65) and (66) are uniformly ultimately bounded.

**Proof.** Consider the Lyapunov function candidate given by equation (24). Following the steps from the proof of Theorem 1, differentiating equation (24) along the trajectories of equations (65) and (66) yields

\[
\dot{V}(\cdot) = z^T(t)\dot{R}_A(t)z(t) + z^T(t)\dot{R}_Bz(t) + 2z^T(t)\phi,
\]

(69)

where \( z(t), \dot{R}_B \) and \( \phi \) are defined in equations (26), (28), and (30), respectively. In addition, \( \dot{R}_A(t) \) is given by

\[
\dot{R}_A(t) \triangleq \begin{bmatrix} \hat{A}(t)TP + PA(t) & -P(1_N \otimes B) + M(t) \\ -P(1_N \otimes B)P + M(t) & -2K \end{bmatrix}.
\]

(70)

Note that for this varying case of active and passive node roles, \( R_i \) in equation (9) becomes

\[
\begin{align*}
R_i(t) &= \begin{bmatrix} (A - g_i(t)L_iC_i)^TP_i + P_i(A - g_i(t)L_iC_i) & -P_iB + g_i(t)C_i^TP_i \\ -B^TP_i + I_iC_i & -2\sigma_i K_i \end{bmatrix} \\
&= \begin{bmatrix} A^TP_i + P_iA & -P_iB \\ -B^TP_i & -2\sigma_i K_i \end{bmatrix} + \begin{bmatrix} g_i(t)J_iC_i & 0 \\ 0 & I_iC_i \end{bmatrix}.
\end{align*}
\]

(71)

Since \( g_i(t) \in [0, 1] \), \( R_1 \) in equation (60) and \( R_2 \) in equation (61) corresponds to \( g_i(t) = 0 \) and \( g_i(t) = 1 \) in equation (71), respectively. Therefore, \( R_1 \) and \( R_2 \) are the vertices of the polytope. By Lemma 3, when the linear matrix inequalities equations (60) and (61) hold, \( R_i(t) \leq 0 \) for all \( g_i(t) \in [0, 1] \). Consequently, using the same argument as in the proof of Theorem 1, we have \( R_A(t) \leq 0 \). Hence, equation (69) becomes

\[
\dot{V}(\cdot) \leq \lambda_{\max}(R_B)||z(t)||_2^2 + 2||z(t)||_2\phi,
\]

(72)

\[
\leq (1 - \theta)\lambda_{\max}(R_B)||z(t)||_2^2 + \theta\lambda_{\max}(R_B)||z(t)||_2^2 + 2||z(t)||_2\phi,
\]

with \( \lambda_{\max}(R_B) < 0 \) and \( \theta \in (0, 1) \). Letting \( \mu_2 \triangleq \frac{-2\phi}{\theta\lambda_{\max}(R_B)} > 0 \) and

\[
\Omega_2 \triangleq \{z(t) : ||z(t)||_2 \leq \mu_2\},
\]

it follows that \( \dot{V}(\cdot) \leq (1 - \theta)\lambda_{\max}(R_B)||z(t)||_2^2 < 0 \) outside the compact set \( \Omega_2 \) and, hence, the error dynamics given by equations (62) and (63) are uniformly ultimately bounded from Theorem 4.18 in Khalil (2002).

**Corollary 2.** Consider the process given by equation (5) and the distributed input and state estimation architecture given by equations (58) and (59). Assume that equations (60) and (61) hold and nodes exchange information according to the undirected connected graph \( \mathcal{G} \). Then, for all \( z(0) \in \mathbb{R}^{n+i+p} \), there exists \( T = T(z(0), \mu_2) \geq 0 \) such that...
\[ \| \hat{x}(t) \|_2 \leq \xi_2 = \sqrt{\frac{\lambda_{\max}(\bar{P})}{\lambda_{\min}(\bar{P})}} \max\{\| z(0) \|_2, \mu_2 \} \times e^{(1-\theta)\lambda_{\max}(R_b)/2\lambda_{\max}(P)t}, \forall t \geq 0, \]  
\[ (73) \]

\[ \| \hat{w}(t) \|_2 \leq \xi_2, \forall t \geq 0, \]  
\[ (74) \]

where
\[ \bar{P} = \begin{bmatrix} P & 0 \\ 0 & 1_{Np} \end{bmatrix}, \]  
\[ (75) \]

and
\[ \| \hat{x}(t) \|_2 \leq \psi_2 = \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} \mu_2, \quad t \geq T, \]  
\[ (76) \]

\[ \| \hat{w}(t) \|_2 \leq \zeta_2 \triangleq \sqrt{\lambda_{\max}(\bar{P})} \mu_2, \quad t \geq T. \]  
\[ (77) \]

**Proof.** Same theoretical steps follow from the proof of Corollary 1 and, hence, the proof is omitted here. \[ \blacksquare \]

### 4.3. Illustrative Numerical Example

In this section, we present numerical examples to illustrate the results discussed in Sections 4.1 and 4.2. For this purpose, we consider a process as the vehicle model depicted in Section 3.3 with the dynamics given by equation (5), where \( A \) and \( B \) are defined in equations (50) and (51), respectively. To maintain the readability of the paper, the values of \( L_i, \sigma_j, P_i \) in the following

**FIGURE 13** | State estimates of the active-passive sensor network in Example 4 with 12 nodes under the proposed architecture equations (58) and (59) and satisfying the linear matrix inequalities equations (60) and (61) (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes).

**FIGURE 14** | Position estimates (first and third states of the process) of the active-passive sensor network in Example 4 with 12 nodes under the proposed architecture equations (58) and (59) and satisfying the linear matrix inequalities equations (60) and (61) (the dash line denote the trajectory of the actual process (i.e., the combination of the first and third states) and the solid lines denote the state estimates of nodes). Here, AN stands for the active nodes.
examples are put in Appendix B in Data Sheet in Supplementary Material.

**Example 4.** For this example, the initial conditions of the process are set to $x_0^T = [-3, 0.5, 2.5, 0.25]$. In addition, we consider the input is given by

$$w(t) = \begin{bmatrix} 2.5 \sin(t) \\ 3.5 \cos(1.2t) \end{bmatrix} \quad (78)$$

We now consider an active-passive sensor network with 12 nodes exchanging information over an undirected and connected graph topology as presented in Figure 12, where the active and passive roles of each node is varying overtime. Specifically, the sensors are distributed over an area, and each sensor position is shown in Figure 12. Suppose that each sensor sensing range is a circle with the radius $r = 3$. Recall that the first and third states of the process (or the vehicle) correspond to the positions in the $x$-axis and $y$-axis directions, respectively. If the vehicle's position is within a sensor sensing range, then that sensor becomes smoothly active. On the other hand, if the vehicle's position is out of the sensor sensing range, then it becomes smoothly passive. Note that, for the transition of $g_i(t)$, we use the function $g_i(t) = e^{-\beta t}$ when node $i$ is switching from 1 to 0, and $g_i(t) = 1 - e^{-\beta t}$ when node $i$ is switching from 0 to 1, where $\beta$ is a positive constant. We adapt this transition from Figure 2D of Zavlanos et al. (2011). The network has two types of sensors, and each node's sensing capability is represented by equation (6) with the output matrices

$$C_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (79)$$

**FIGURE 15** | Input estimates of the active-passive sensor network in Example 4 with 12 nodes under the proposed architecture equations (58) and (59) and satisfying the linear matrix inequalities equations (60) and (61) (the dash lines denote the inputs of the actual process and the solid lines denote the input estimates of nodes).

**FIGURE 16** | State estimates of the active-passive sensor network in Example 4 with 12 nodes under the proposed architecture equations (58) and (59) with the decrease in $\sigma_i, i = 1, \ldots, 12$ (the dash lines denote the states of the actual process and the solid lines denote the state estimates of nodes).
for the odd index nodes and
\[
C_i = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\] (80)
for the even index nodes. Note that the pair \((A, C_i)\) is observable for all \(i = 1, \ldots, 12\) in this example and, therefore, the collective observability assumption is satisfied. All nodes are subjected to zero initial conditions and we set \(J_i = K_i = \text{diag}([100; 100]), \alpha = 50, \gamma = 0.1\). For the observer gain \(L_i\), the odd index nodes are subject to equation (A.26) in Supplementary Material while the even index nodes are subject to equation (A.27) in Supplementary Material.

By solving the linear matrix inequalities equations (60) and (61) simultaneously for each node, \(\sigma_i\) and \(P_i > 0\) are obtained as \(\sigma_1 = \sigma_3 = \sigma_5 = \sigma_7 = \sigma_9 = \sigma_{11} = 0.001\) and \(\sigma_2 = \sigma_4 = \sigma_6 = \sigma_8 = \sigma_{10} = \sigma_{12} = 0.001\) where \(\sigma_1\) and \(\sigma_2\) are subject to equations (A.28) and (A.29) in Supplementary Material, respectively. In addition, \(P_1\) and \(P_2\) are subject to equations (A.30) and (A.31) in Supplementary Material, respectively. Note that \(P_1 = P_3 = P_5 = P_7 = P_9 = P_{11}\) and \(P_2 = P_4 = P_6 = P_8 = P_{10} = P_{12}\).

Under the proposed distributed estimation architecture equations (58) and (59), nodes are able to closely estimate the process states as shown in Figure 13. Specifically, Figure 14 illustrates that the sensor network is able to estimate the trajectory of the vehicle (the first and third states of the process), while the input estimated in Figure 15 is not as good as the case for fixed node roles presented in Section 3.3, this can be explained by the conservatism of the solution of the linear matrix inequalities equations (60) and (61).

Example 5. For this example, the initial conditions of the process are set to \(x_0^T = [-2, 0.5, 2.5, 0.25]\). In addition, we consider the input is given by
\[
w(t) = \begin{bmatrix} 1.5 \sin(0.5t) \\ 3 \cos(0.6t) \end{bmatrix}.
\] (81)

We now consider an active-passive sensor network with 13 nodes labeled, respectively, as 1a, 1b, 2a, 2b, 3a, 3b, 4a, 4b, 5, 6, 7, 8, and 9, exchanging information over an undirected and connected graph topology as presented in Figure 18, where the active and passive role of each node is varying overtime. Specifically, the
sensors are distributed over an area with each pair of nodes Xa and Xb (where X = 1, 2, 3, 4 and denoted as diamond in Figure 18) is grouped at the same location such that when Xa is active (or passive), so is Xb and vice versa; Xa and Xb are neighbors of each other and have the same set of neighbors.

Suppose that each sensor sensing range is a circle with the radius \( r = 3.5 \). Note that, for the transition of \( g_i(t) \), we use the same functions as Example 4. The network has six types of sensors, and each node’s sensing capability is represented by equation (6) with the output matrices

\[
C_{1a} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix},
\]

\[
C_{2b} = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\]

\[
C_{3a} = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix},
\]

\[
C_5 = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix},
\]

\[
C_6 = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}.
\]
In addition, $C_{1a} = C_{2a}$, $C_{1b} = C_{3b}$, $C_{2b} = C_{4b}$, $C_{3a} = C_{4a}$, $C_5 = C_7 = C_9$, and $C_6 = C_8$. Note that the pairs $(A, C_{Xa})$ and $(A, C_{Xb})$ where $X = 1, 2, 3, 4$ are not observable, but with the setup of the problem (nodes $Xa$ and $Xb$ are either active or both passive simultaneously), the collective observability condition is guaranteed. All nodes are subjected to zero initial conditions and we set $J_i = K_i = \text{diag}(25; 25)$, $\alpha = 75$, and $\gamma = 0.01$.

The observer gains $L_i$ are set to $L_{1a} = L_{2a}$, $L_{1b} = L_{3b}$, $L_{2b} = L_{4b}$, $L_{3a} = L_{4a}$, $L_5 = L_7 = L_9$, and $L_6$ where the gains $L_{1a}$, $L_{1b}$, $L_{2b}$, $L_{3a}$, $L_5$, and $L_6$ are subject to equations (A.32), (A.33), (A.34), (A.35), (A.36), and (A.37) in Supplementary Material, respectively. By solving the linear matrix inequalities equations (60) and (61) simultaneously for each node, $\sigma_i$ and $P_i > 0$ are obtained as $\sigma_{1a} = \sigma_{2a}$, $\sigma_{1b} = \sigma_{3b}$, $\sigma_{2b} = \sigma_{3b}$, $\sigma_{3a} = \sigma_{4a}$, $\sigma_5 = \sigma_7 = \sigma_9$ and $\sigma_6 = \sigma_8$ where $\sigma_{1a}$, $\sigma_{1b}$, $\sigma_{2b}$, $\sigma_{3a}$, $\sigma_5$ and $\sigma_6$ are subject to equations (A.38), (A.39), (A.40), (A.41), (A.42), and (A.43) in Supplementary Material, respectively. In addition, $P_{1a}$, $P_{1b}$, $P_{2b}$, $P_{3a}$, $P_5$, and $P_6$ are subject to equations (A.44), (A.45), (A.46), (A.47), (A.48), and (A.49) in Supplementary Material, respectively. Note that $P_{1a} = P_{2a}$, $P_{1b} = P_{3b}$, $P_{2b} = P_{4b}$, $P_{3a} = P_{4a}$, $P_5 = P_7 = P_9$, and $P_6 = P_8$.

Under the proposed distributed estimation architecture equations (58) and (59), nodes are able to closely estimate the process states as shown in Figure 19. Specifically, Figure 20 illustrates that the sensor network is able to estimate the trajectory of the vehicle (the first and third states of the process), while the input estimated in Figure 21 is still not as good as the case for fixed node roles presented in Section 3.3. Again, this can be explained by the conservatism of the solution of the linear matrix inequalities equations (60) and (61). If we had the flexibility to reduce $\sigma_i$ to small values, for example, $\sigma_i = 0.001$, while keeping other
parameters the same, the performance of the input and state estimate would become much better as shown in Figures 22 and 23, respectively. Note that with the choice of \( \sigma_i = 0.001 \) for this example, the linear matrix inequalities equations (60) and (61) are no longer satisfied. Once again, numerical methods to reduce such conservatism in linear matrix inequality computations for equations (60) and (61) and/or relax the linear matrix inequality condition will be investigated as a future research.

5. CONCLUSION

A distributed input and state estimation architecture was investigated for heterogeneous sensor networks having nodes with both fixed and varying active and passive information processing roles and non-identical sensor modalities. It was shown that the proposed framework utilizes local information not only during the execution of the proposed estimation algorithm but also in its design; that is, global uniform ultimate boundedness of error dynamics is guaranteed once each node satisfies given local stability conditions independent from the graph topology and neighboring information of these nodes. Several numerical examples illustrated the efficacy of the proposed architectures. Future research will include applications of the proposed framework to dynamic data-driven sensor network scenarios to guide and control autonomous vehicles and we will also consider extensions to time-varying graph topologies. It should be also mentioned especially for the results in Section 4 that structural sensor network construction to always guarantee collective observability is another interesting future research direction that will be considered by the authors.

AUTHOR CONTRIBUTIONS

DT and TY were the technical leads of this paper, where they developed the proposed distributed estimation architecture. Specifically, DT played a key role in establishing the technical proofs of the proposed architecture, including how the analysis can be performed in a local fashion and TY played a key role in fleshing out the technical tools and methods needed to accomplish the documented research. SS and SJ assisted DT and TY in all technical and numerical aspects of this paper. In addition to theory development in terms of establishing the technical proofs, DT also numerically illustrated the proposed distributed estimation architecture.

ACKNOWLEDGMENTS

The authors wish to thank the reviewers for their constructive suggestions.

FUNDING

This research has been supported by the Dynamic Data-Driven Applications Systems Program of the Air Force Office of Scientific Research [grant number FA9550-17-1-0303].

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://journal.frontiersin.org/article/10.3389/frobt.2017.00030/full#supplementary-material.

REFERENCES

Akhenak, A., Chadli, M., Maquin, D., and Ragot, J. (2004). “State estimation of uncertain multiple model with unknown inputs,” in IEEE Conference on Decision and Control, Nassau.

Bai, H., Freeman, R. A., and Lynch, K. M. (2010). “Robust dynamic average consensus of time-varying inputs,” in Conference on Decision and Control, Atlanta, 3104–3109.

Bernstein, D. S. (2009). Matrix Mathematics: Theory, Facts, and Formulas. Princeton, NJ: Princeton University Press.
