# OUTformation: Distributed Data-Gathering with Feedback under Unknown Environment and Communication Delay Constraints

SooJean Han[1*], Michelle Effros[1], Richard M. Murray[1]

*Abstract*— **Towards the informed design of large-scale distributed data-gathering architectures under real-world assumptions such as nonzero communication delays and unknown environment dynamics, this paper considers the effects of allowing feedback communication from the central processor to external sensors. Using simple but representative state-estimation examples, we investigate fundamental tradeoffs between the mean-squared error (MSE) of the central processor's estimate of the environment state, and the total power expenditure per sensor under more conventional architectures without feedback (*INformation*) versus those with broadcast feedback (*OUTformation*). The primary advantage of enabling feedback is that each sensor's understanding of the central processor's estimate improves, which enables each sensor to determine when and what parts of its current observations to transmit. We use theory to demonstrate conditions in which OUTformation maintains the same MSE as INformation with less power expended on average, and conditions in which OUTformation obtains less MSE than INformation at additional power cost. These performance tradeoffs are also considered under settings where environments undergo less variation, and sensors implement random backoff times to prevent transmission collisions. Our results are supported via numerical studies, which show that the properties derived in theory still hold even when some of the simplifying assumptions are removed.**

## I. Introduction

In previous literature, a variety of architectures for distributed data-gathering in large-scale cyberphysical network applications has been proposed with the aim of reducing unnecessary transmissions of redundant information and excess computation; see [1], [2] for surveys. However, many methods consider only structural variations in the communication among lower-level nodes such as external sensors (e.g., hierarchical clustering [3], [4]), and have rarely challenged the imposition that data flows only in a single direction from sensors on the network's edge to the higher-level processor at the network's center. Some architectures allowing *feedback communication* in the reverse direction, from the central processor to the external sensors, are developed under a diversity of simplifying assumptions. In some feedback schemes for redundant transmission reduction (e.g. [5]), communication delays are omitted; this disregards issues pertaining to outdated data, which is important to distributed state estimation in real-world large-scale networks. In others (e.g. [6]), communication delays are included, but the approach for redundant transmission reduction assumes full knowledge of the environment dynamics.

While in practice, the best choice of architecture to use for distributed data-gathering varies by setting, imposing simplifying but impractical assumptions makes it difficult to properly motivate feedback, and make fair comparisons between architectures which employ feedback and those which do not. Feedback can provide at least the following two advantages in large-scale distributed systems. First, by nature of the distributed architecture, data gathering and data processing are sequential actions; having a larger delay between the sensor transmission times and central processor receipt times can potentially lead to large estimation errors. Second, for large-scale systems, a large number of sensors may be needed to survey the environment. But independent sensors surveying a shared environment are likely to collect data which exhibits redundancies, and transmitting redundant information to the central processor is inefficient.

With these motivations in mind, the contributions of our paper are as follows. We employ a modular framework for distributed state estimation under two important real-world constraints: nonzero communication delays and limited knowledge of the environment dynamics. Because the type of feedback we consider broadcasts data "out" from the central processor to the sensors, we refer to this architecture as the *OUTformation architecture* for data-gathering, as opposed to the *INformation architecture* where all information flows from the sensors "in" to the central processor. We use a modular framework representation of the INformation and OUTformation architectures to take a first step towards characterizing their relative performances with respect to two specific metrics: 1) the mean-squared error (MSE) of the central processor's estimate of the environment state, and 2) the total power expenditure of each sensor. The importance of such a characterization arises from allowing users to determine which type of architecture is more suitable for use based on hardware specifications and limited knowledge of the environment, as well as providing essential insights to make parameter design choices for optimal performance in both MSE and power expenditure.

We theoretically analyze simple but insightful case studies demonstrating fundamental tradeoffs between the two performance metrics, and consider how the tradeoffs vary under different environment statistics and parameter design choices. In particular, we show that the main advantage of enabling feedback is improving each sensor's estimate of the central processor's estimate, which allows each sensor to make more informed decisions about when and what to transmit,

[1]SooJean Han, Michelle Effros, and Richard M. Murray are with the Division of Engineering and Applied Science at California Institute of Technology, Pasadena CA.

[*]Corresponding author. Email: `soojean@caltech.edu`

This paper is based on work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1745301.

potentially reducing both 1) its uplink power expenditure, and 2) the MSE by minimizing the delays after which the central processor receives its transmissions. We employ numerical simulations to break some of the simplifying assumptions made in our theoretical analysis, and show that the properties derived in theory still hold, which suggests that the implications of OUTformation architectures extend beyond the framework treated rigorously by the theory.

## II. INFORMATION AND OUTFORMATION

### A. Problem Setup

We consider environments which evolve as a random walk given by

$$\mathbf{x}((c+1)\Delta t) = \mathbf{x}(c\Delta t) + \sum_{i=1}^{n} \delta_i(c)\mathbf{d}(c)\mathbf{e}_i \tag{1}$$

Here, $c \in \mathbb{N}$, $\Delta t \in \mathbb{R}^+$ is the environment's transition period, and $\mathbf{e}_i$ denotes the $i$th standard basis vector of $\mathbb{R}^n$. For each $i \in \{1, \cdots, n\}$ and $c \in \mathbb{N}$, random variable $\delta_i(c)$ takes value 1 with probability $p/2$, $-1$ with probability $p/2$, and 0 with probability $1-p$ for some $p \in (0,1)$, and $\mathbf{d}(c) \in \mathbb{R}^n$ such that $d_i(c) \sim \mathcal{U}[\underline{d}, \overline{d}]$ is a uniformly-distributed random variable stepsize between constant values $0 < \underline{d} < \overline{d}$.

We focus on architectures where a single central processor communicates with $M \in \mathbb{N}$ independent external sensors. The environment parameters $\Delta t, p, \underline{d}, \overline{d}$ are unknown to the central processor. Each sensor $j \in \{1, \cdots, M\}$ employs a linear measurement equation perturbed by additive, white Gaussian noise $\mathbf{w}_j(t) \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_n)$ where $I_n \in \mathbb{R}^{n \times n}$ is identity:

$$\mathbf{y}_j(at_j) = C_j\mathbf{x}(at_j) + \mathbf{w}_j(at_j) \tag{2}$$

Here, $C_j \in \{0,1\}^{m_j \times n}$, $m_j \in \mathbb{N}, m_j < n$, such that each row contains exactly one 1 and each column contains at most one 1. Each sensor $j$ makes one observation of the environment with *sampling period* $t_j$ timesteps, and $a \in \mathbb{N}$ is any number.

Over some experiment duration $[0, T_{\text{sim}}), T_{\text{sim}} \in \mathbb{R}^+$, the central processor's objective is to use the transmissions of each sensor to construct an estimate $\hat{\mathbf{x}}(t)$ of the environment state $\mathbf{x}(t)$. Each sensor $j$'s objective is to determine when and which components $y_{jk}(t), k \in \{1, \cdots, m_j\}$ of $\mathbf{y}_j(t)$ it should transmit; we show that this is contingent upon how accurately the sensor can track the central processor's estimate $\hat{\mathbf{x}}(t)$.

**Assumption 1** (Framework Assumptions)**.** Each sensor $j$ operates on limited power sources (e.g. batteries). Communications in both uplink and downlink channels are noiseless and able to precisely encode/decode real numbers. To prevent potential collisions of transmissions, each sensor $j$ schedules *random backoff times* $B_{jk}(t) \sim F_B(\cdot)$ under the same cdf $F_B(s) \triangleq \mathbb{P}(B(t) \leq s), F_B(\Delta t) = 1$ for each component $k \in \{1, \cdots, m_j\}$ and time $t$; the $B_{jk}(t)$ are distributed independently of each other and of the noise process $\mathbf{w}_j(t)$. The central processor performs *fusion* by averaging over the most recent sensor observations it received to construct an estimate $\hat{x}_i(t)$ for each component $i \in \{1, \cdots, n\}$.

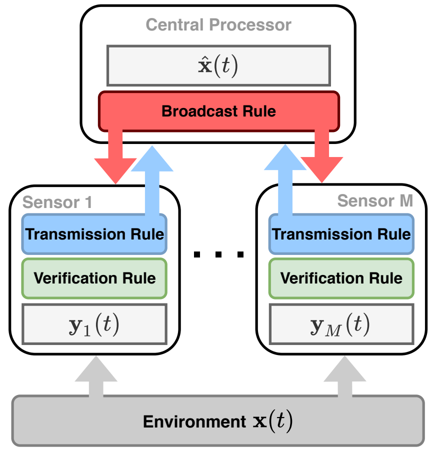



**Fig. 1:** A modular representation of an OUTformation architecture for distributed state estimation under unknown environments and communication delay constraints, using a single central processor and a network of $M$ external sensors. For each sensor, the verification rule (Definition 4) is highlighted in green, and the transmission rule (Definition 5) is in blue. The broadcast rule (Definition 3) is in red; INformation architectures are depicted by removing the red broadcast arrows.

**Remark 1.** We emphasize that our goal is not to model any particular distributed data-gathering architecture in fine detail, but to provide insights into the distinctions of architectures with and without feedback from the central processor to the sensors. Thus, while more complex functionalities (e.g., optimized random backoff strategies [7], consensus-based fusion [8], change-detection [9]) may be used than those described in Assumption 1, we focus specifically on functionalities which need to be implemented differently based on whether an architecture has enabled feedback or not.

### B. Modular Framework

Under Assumption 1, we distinguish between traditional INformation and OUTformation architectures in the following way. The presence or absence of broadcast feedback (which pushes data "outward" from the central processors to the sensors) changes how accurately each sensor can track the central processor's estimate $\hat{\mathbf{x}}(t)$.

**Definition 1** (INformation and OUTformation)**.** Under an *INformation architecture*, each sensor's estimate of $\hat{x}_i(t), i \in \{1, \cdots, n\}$ is its own previous transmission to the central processor. Under an *OUTformation architecture*, each sensor's estimate of $\hat{x}_i(t)$ is the more recent of either its own previous transmission or the latest broadcast received. A modular representation of the distributed data-gathering OUTformation architecture, which we use throughout the paper, is illustrated in Figure 1.

**Notation 1.** Parameters are assigned the superscript $(I)$ for INformation architectures, and $(O)$ for OUTformation architectures. Let $\chi$ be a placeholder variable such that $\chi = I$ for an INformation architecture, and $\chi = O$ for OUTformation.

We henceforth let $\hat{\mathbf{x}}^{(\chi)} \in \mathbb{R}^n$ be the central processor's estimate under architecture $\chi \in \{I, O\}$.

**Notation 2.** For each sensor $j \in \{1, \cdots, M\}$ and component $k \in \{1, \cdots, m_j\}$, we let $i_k \in \{1, \cdots, n\}$ correspond to the "full state vector component" from which observation $y_{jk}(t)$ is made, i.e. $x_{i_k}(t) \triangleq C_{j,(k,\cdot)}\mathbf{x}(t)$, where $C_{j,(k,\cdot)}$ denotes the $k$th row of $C_j$ from (2).

**Definition 2** (Shared vs. Unshared Components)**.** For each sensor $j \in \{1, \cdots, M\}$, let $k \in \{1, \cdots, m_j\}$ be a component and $i_k \in \{1, \cdots, n\}$ be the corresponding full-vector component (see Notation 2). Then $k$ is said to be a *shared* component if $x_{i_k}(t)$ is sensed by other sensors, and an *unshared* component if $x_{i_k}(t)$ is sensed by only $j$. Define the set $\mathcal{U}_j \subseteq \{1, \cdots, m_j\}$ to be the set of sensor $j$'s unshared components, and $\mathcal{S}_j \triangleq \mathcal{U}_j^c$ to be the set of sensor $j$'s shared components.

**Definition 3** (Broadcast Rule)**.** OUTformation architectures employ the following *broadcast rule* (red in Figure 1): a component $i \in \{1, \cdots, n\}$ is scheduled to be broadcast if $\hat{x}_i^{(O)}(t) \neq \hat{x}_i^{(O)}(s)$, where $s < t$ is the time of the previous broadcast. All scheduled values $\hat{x}_i^{(O)}(t)$ are collected the form the dimension-reduced *broadcast vector* $\hat{\mathbf{x}}^{(b)}(t) \in \cup_{i=1}^n \mathbb{R}^i$ and sent to the sensors at once. Because there is no benefit to sensors receiving updates for components they do not observe, broadcasts are made only for components which are shared in the sense of Definition 2.

**Definition 4** (Verification Rule)**.** For $\chi \in \{I, O\}$ as in Notation 1 and each sensor $j \in \{1, \cdots, M\}$, we define $T_j^{(\chi)}$, a multiple of the sampling period $t_j$ from (2), to be the *verification period* under architecture $\chi$. At time $t \triangleq aT_j^{(\chi)}$, $a \in \mathbb{N}$, the *verification rule* $\beta_{jk}^{(\chi)}$ schedules the transmission of $y_{jk}(t)$, $k \in \{1, \cdots, m_j\}$ for time $t + B_{jk}(t)$ if $\beta_{jk}^{(\chi)}(t) = 1$, where $B_{jk}(t)$ is the random backoff time from Assumption 1. Under an INformation architecture, we use:

$$\beta_{jk}^{(I)}(t) = \mathbb{1}\left\{|y_{jk}(s) - y_{jk}(t)| \geq \epsilon\right\} \tag{3}$$

while under an OUTformation architecture, we use:

$$\beta_{jk}^{(O)}(t) = \begin{cases} \mathbb{1}\{|\hat{x}_{i_k}^{(b)}(t) - y_{jk}(t)| \geq \epsilon\} & \text{if } s^* \geq s \\ \beta_{jk}^{(I)}(t) & \text{else} \end{cases} \tag{4}$$

Here, $s < t$ is the time of sensor $j$'s previous transmission of component $k$, and $s^* < t$ is the time the broadcast value $\hat{x}_{i_k}^{(b)}(t)$ was created. Threshold parameter $\epsilon > 0$ is chosen by design. The special case of *event-triggered verification* occurs when the verification rule is checked with every observation sensor $j$ generates, i.e. $T_j^{(I)} = T_j^{(O)} = t_j$.

**Definition 5** (Transmission Rules)**.** Let $a \in \mathbb{N}$, and let $\chi \in \{I, O\}$ be the placeholder variable defined in Notation 1. For each sensor $j$, let $T_j^{(\chi)}$ be the verification period from Definition 4. At time $t \triangleq aT_j^{(\chi)} + B_{jk}(t)$ such that $\beta_{jk}^{(\chi)}(t - B_{jk}(t)) = 1$, the *transmission rule* $\gamma_{jk}^{(\chi)}$ transmits $y_{jk}(t - B_{jk}(t))$, for component $k \in \{1, \cdots, m_j\}$, if $\gamma_{jk}^{(\chi)}(t, B_{jk}(t)) = 1$, where $B_{jk}(t)$ is the random backoff time from Assumption 1. Under an INformation architecture, we use:

$$\begin{aligned} \gamma_{jk}^{(I)}(t, B_{jk}(t)) = \mathbb{1}\{\beta_{jk}^{(I)}(t - B_{jk}(t)) = 1\} \wedge \\ \mathbb{1}\left\{|y_{jk}(s) - y_{jk}(t - B_{jk}(t))| \geq \epsilon\right\} \end{aligned} \tag{5}$$

while under an OUTformation architecture, we use:

$$\begin{aligned} &\gamma_{jk}^{(O)}(t, B_{jk}(t)) = \mathbb{1}\{\beta_{jk}^{(O)}(t - B_{jk}(t)) = 1\} \wedge \\ &\begin{cases} \mathbb{1}\{|\hat{x}_{i_k}^{(b)}(t) - y_{jk}(t - B_{jk}(t))| \geq \epsilon\} & \text{if } s^* \geq s \\ \mathbb{1}\left\{|y_{jk}(s) - y_{jk}(t - B_{jk}(t))| \geq \epsilon\right\} & \text{else} \end{cases} \end{aligned} \tag{6}$$

Here, $s < t - B_{jk}(t)$ is the time of sensor $j$'s previous transmission of component $k$, and $s^* < t$ is the time the broadcast value $\hat{x}_{i_k}^{(b)}(t)$ was created. Threshold parameter $\epsilon > 0$ is chosen by design.

**Notation 3.** Under Assumption 1, the three types of architectures we compare are specified below:

1) *Pure INformation* architecture $IN_0$: each sensor $j$ implements verification rule (3) with verification period $T_j^{(I)}$, and transmission rule (5) with zero threshold $\epsilon = 0$.
2) *Absolute-Difference INformation* architecture $IN(\epsilon)$: each sensor $j$ implements verification rule (3) with verification period $T_j^{(I)}$, and transmission rule (5) with fixed threshold $\epsilon > 0$.
3) *OUTformation* architecture $OUT(\epsilon)$: each sensor $j$ implements verification rule (4) with verification period $T_j^{(O)}$, transmission rule (6) with fixed threshold $\epsilon > 0$, and the central processor uses the broadcast rule from Definition 3.

**Definition 6** (Rates per Component)**.** For sensor $j$ under architecture $\chi \in \{I, O\}$, let $U_{jk}^{(\chi)}(s:t), D_{jk}^{(\chi)}(s:t) \in \mathbb{N}$ be the cumulative *number of transmissions* and *number of broadcasts received*, respectively, for component $k \in \{1, \cdots, m_j\}$ over the interval of time $[s, t)$. The *communication delay* incurred per component $k$ is denoted $\Delta t^{(u)} \in \mathbb{R}^+$ for uplink transmission, $\Delta t^{(d)} \in \mathbb{R}^+$ for downlink broadcast. For each sensor, the *rate of power* expended per component $k$ is $P_U \in \mathbb{R}^+$ for transmission and $P_D \in \mathbb{R}^+$ for receipt.

**Definition 7** (Performance Metrics)**.** The *mean-squared error (MSE)* of the environment state vector over experiment duration $[0, T_{\text{sim}})$ under architecture $\chi \in \{I, O\}$ is given by

$$\frac{1}{T_{\text{sim}}} \sum_{t=0}^{T_{\text{sim}}} \sum_{i=1}^{n} \left| x_i(t) - \hat{x}_i^{(\chi)}(t) \right|^2 \tag{7}$$

The *total amount of power expenditure per sensor* $j$ under architecture $\chi$ is given by:

$$R_j^{(\chi)}(0:T_{\text{sim}}) = P_U U_j^{(\chi)}(0:T_{\text{sim}}) + P_D D_j^{(\chi)}(0:T_{\text{sim}}) \tag{8}$$

where the power expenditure rates $P_U$,$P_D$ are from Definition 6, and $U_j^{(\chi)}(s:t) \triangleq \sum_{k=1}^{m_j} U_{jk}^{(\chi)}(s:t)$, likewise $D_j^{(\chi)}(s:t) \triangleq \sum_{k=1}^{m_j} D_{jk}^{(\chi)}(s:t)$.

## III. Theoretical Analysis

With the setup described above, we now characterize the tradeoff space between the MSE and sensor power expenditure metrics (see Definition 7) for the three architectures in Notation 3. In this section, we introduce theory to analyze $IN(\epsilon)$ and $OUT(\epsilon)$ under the following simplified version of the original problem setup in Section II-A. Later, in Section IV, we numerically study all three architectures for the original setup of Section II-A.

**Setting 1** (Two Sensors over One-Component Change Environment)**.** We consider the true environment dynamics (1) when the vector of indicators $[\delta_1(c), \cdots, \delta_n(c)]^T$ can only take values from the set $\{\mathbf{0}_n, \mathbf{e}_1, \cdots, \mathbf{e}_n, -\mathbf{e}_1, \cdots, -\mathbf{e}_n\}$ for all $c \in \mathbb{N}$, where $\mathbf{0}_n$ denotes the zero vector in $\mathbb{R}^n$. Both types of architectures operate with $M = 2$ sensors, where each sensor $j \in \{1, 2\}$ has $m_j \in \mathbb{N}$ components. Here, $m' < \min(m_1, m_2)$ components are shared in the sense of Definition 2 ($\mathcal{S} \triangleq \mathcal{S}_1 = \mathcal{S}_2, |\mathcal{S}| = m'$) and the remaining $m_j - m'$ are unshared ($|\mathcal{U}_j| = m_j - m'$). The sensors have the same sampling rate $\tau \triangleq t_1 = t_2$ (see (2)), and use event-triggered verification $T_j^{(I)} = T_j^{(O)} = \tau$ (see Definition 4), where $\tau$ is such that $\Delta t/\tau \triangleq H \in \mathbb{N}$. Because there are only two sensors in the network, sensor 1 transmits immediately ($B_{1k}(t) = 0$) while sensor 2 performs random backoff according to the cdf $F_B$ prescribed in Assumption 1 such that $F_B(\Delta t - \Delta t_u) = 1$.

**Notation 4.** For each sensor $j \in \{1, 2\}$ and each component $k \in \{1, \cdots, m_j\}$, let $p_{jk}^{(\chi)}(t, B_{jk}(t)) \triangleq \mathbb{P}(\gamma_{jk}^{(\chi)}(t, B_{jk}(t)) = 1)$ be the probability that component $k$ is transmitted uplink by sensor $j$ at time $t$ under architecture $\chi$, using (5) for $\chi = I$ and (6) for $\chi = O$. Furthermore, for simplicity of expression, every interval $[c\Delta t, (c+1)\Delta t)$, where $c \in \mathbb{N}$ and $\Delta t$ is from (1), is referred to as *environment interval c.*

**Lemma 1** (Power from Unshared Components)**.** For sensor $j \in \{1, 2\}$ under Setting 1, the expected total power expended by unshared components over environment interval $c \in \mathbb{N}$ is equivalent under $IN(\epsilon)$ or $OUT(\epsilon)$, and given by

$$\begin{aligned}\mathbb{E}[R_{j,\mathcal{U}_j}^{(\chi)}(c\Delta t : (c+1)\Delta t)] = P_U \sum_{k \in \mathcal{U}_j} \\ \Bigg(\sum_{h=0}^{H}(1 - F_B(\Delta t - h\tau - \Delta t^{(u)}))p_{jk}^{(I)}(\tau_{c-1}, B_{jk}(\tau_{c-1})) \\ + \sum_{h=1}^{H-1} F_B(\Delta t - h\tau - \Delta t^{(u)})p_{jk}^{(I)}(\tau_c, B_{jk}(\tau_c))\Bigg) \quad (9)\end{aligned}$$

where $\tau_a \triangleq a\Delta t + h\tau$ for any $a \in \mathbb{N}$, and $R_{j,\mathcal{U}_j}^{(\chi)}(c\Delta t : (c+1)\Delta t)$ is the part of the power (8) contributed by unshared components.

*Proof.* Under Definition 3, no broadcasts made for unshared components and (6) is equivalent to (5) when $k \in \mathcal{U}_j$. Hence, $IN(\epsilon)$ and $OUT(\epsilon)$ have the same uplink communications for unshared components, and their contributed total power expenditures are equivalent. For any $a \in \mathbb{N}$, let $X_{jk}^{(\chi)}(a, h)$, $h \in \{0, \cdots, H-1\}$, be the indicator denoting whether or not $y_{jk}(a\Delta t + h\tau)$ is received during environment interval $c$ under architecture $\chi \in \{I, O\}$. By Assumption 1, $y_{jk}((c-1)\Delta t + h\tau)$ is received during the interval with probability $1 - F_B(\Delta t - h\tau - \Delta t^{(u)})$, and $y_{jk}(c\Delta t + h\tau)$ with probability $F_B(\Delta t - h\tau - \Delta t^{(u)})$. The result follows by definition of $p_{jk}^{(\chi)}(t, B_{jk}(t))$ in Notation 4 and $\mathbb{E}[U_{jk}^{(\chi)}(c\Delta t : (c+1)\Delta t)] = \sum_{h=0}^{H} \mathbb{E}[X_{jk}(c-1, h)] + \sum_{h=1}^{H-1} \mathbb{E}[X_{jk}(c, h)]$. □

Because both $IN(\epsilon)$ and $OUT(\epsilon)$ have the same uplink communications for unshared components, the central processor has the same amount of data about the environment at each time $t$. Thus, we also have the following result.

**Lemma 2** (MSE from Unshared Components)**.** Using the setup of Lemma 1, suppose $\hat{\mathbf{x}}^{(\chi)}(t)$ is constructed via the fusion rule from Assumption 1 under $IN(\epsilon)$ when $\chi = I$ and $OUT(\epsilon)$ when $\chi = O$. Then

$$\sum_{i \in \mathcal{N}} \left| x_i(t) - \hat{x}_i^{(I)}(t) \right|^2 = \sum_{i \in \mathcal{N}} \left| x_i(t) - \hat{x}_i^{(O)}(t) \right|^2 \quad (10)$$

where $\mathcal{N} \triangleq \{i_k \in \{1, \cdots, n\} \,|\, k \in \mathcal{U}_1 \cup \mathcal{U}_2\}$ with $i_k$ denoting the full state vector component corresponding to component $k$ (see Notation 2).

The more interesting comparison arises for shared components $k \in \mathcal{S}$. For simplicity, we present the comparison over a specific chosen interval of communications $[T_0, T_f) \subset [0, T_{\text{sim}})$ such that $IN(\epsilon)$ and $OUT(\epsilon)$ begin from the same performance metric values at time $T_0$. By deriving conditions of performance improvement for $OUT(\epsilon)$ over $IN(\epsilon)$ during $[T_0, T_f)$, we remark the implication that this improvement cascades over the longer interval $[T_0, T_{\text{sim}})$.

**Setting 2** (Interval of Communications I)**.** Under Setting 1, choose the specific interval $[T_0, T_f)$, $T_0 \triangleq c\Delta t$ and $T_f \triangleq (c+1)\Delta t$ such that the following occurs. The environment evolves such that $[\delta_1(c), \cdots, \delta_n(c)] = \mathbf{e}_{i_k}$ for full state vector component $i_k \in \{1, \cdots, n\}$ defined in Notation 2 for a component $k \in \mathcal{S}$ which is shared in the sense of Definition 2. The sample path of observations $\mathcal{Y}_k \triangleq \{y_{jk}(t), t \in [0, (c+1)\Delta t), j \in \{1, 2\}\}$ for $k$ is such that for $\chi \in \{I, O\}$, $\beta_{jk}^{(\chi)}(c\Delta t) = 1$ and $\beta_{jk}^{(\chi)}(c\Delta t + h\tau) = 0$ for all $h \leq H-1$; here, $H \in \mathbb{N}$ is defined in Setting 1, and $\beta_{jk}^{(\chi)}$ is the verification rule from Definition 4.

**Theorem 1** (Power from Shared Components)**.** Suppose Setting 2 holds for shared component $k \in \mathcal{S}$ over environment interval $c \in \mathbb{N}$ defined in Notation 4. Then the expected difference between the power expended under $IN(\epsilon)$ and $OUT(\epsilon)$ is given by

$$\begin{aligned}&\mathbb{E}[R^{(I)}(c\Delta t : (c+1)\Delta t) - R^{(O)}(c\Delta t : (c+1)\Delta t)] \quad (11)\\ &= (P_U - P_D)(1 - F_B(\Delta t^{(u)} + \Delta t^{(d)}))\mathbb{P}(|W_1 - W_2| \geq \epsilon)\end{aligned}$$

where $W_1, W_2 \sim \mathcal{N}(0, \sigma^2)$ with $\sigma$ defined in (2), $\Delta t^{(u)}$, $\Delta t^{(d)}$, $P_U$, $P_D$ are the communication delays and power expenditure rates in Definition 6, $F_B$ is the random backoff time distribution from Setting 1, and $R^{(\chi)}(c\Delta t:(c+1)\Delta t)$ is the power expended over both sensors.

*Proof.* By the random backoff strategy of Setting 1 and communications of Setting 2, sensor 1 transmits $y_{1k}(c\Delta t)$ at time $c\Delta t$ while sensor 2 observes $y_{2k}(c\Delta t)$ and schedules to transmit at time $c\Delta t + B_{2k}(c\Delta t)$. Sensor 2's scheduled transmission is cancelled if only the central processor broadcasts $y_{1k}(c\Delta t)$ before time $c\Delta t + B_{2k}(c\Delta t)$, which occurs with probability $1 - F_B(\Delta t^{(u)} + \Delta t^{(d)})$ under Setting 1, and if $|y_{1k}(c\Delta t) - y_{2k}(c\Delta t)| \geq \epsilon$, which occurs with probability $\mathbb{P}(|W_1 - W_2| \geq \epsilon)$ by (2). The result follows from the fact that $P_U - P_D$ is the difference between transmitting and broadcasting an extra component (see Definition 6). □

**Theorem 2** (MSE from Shared Components)**.** Suppose Setting 2 holds for shared component $k \in \mathcal{S}$ over environment interval $c \in \mathbb{N}$. Let $\hat{\mathbf{x}}^{(\chi)}(t)$ be constructed via the fusion rule from Assumption 1, and suppose that $\hat{x}_{i_k}(c\Delta t) \triangleq \hat{x}^{(I)}_{i_k}(c\Delta t) = \hat{x}^{(O)}_{i_k}(c\Delta t)$. Then the probability that the MSE contributed by component $i_k$, defined in Notation 2, under $OUT(\epsilon)$ is less than that under $IN(\epsilon)$ is given by:

$$\begin{aligned}&(1 - F_B(\Delta t^{(u)} + \Delta t^{(d)}))\\&\quad * \mathbb{P}\left(\frac{1}{2}W_1 W_2 + W_2^2 - \frac{3}{4}W_1^2 > 0,\ |W_1 - W_2| < \epsilon\right)\end{aligned} \tag{12}$$

where $\Delta t^{(u)}, \Delta t^{(d)}$ are the communication delays in Definition 6, $F_B$ is the random backoff time distribution from Setting 1, and $W_1, W_2 \sim \mathcal{N}(0, \sigma^2)$ with $\sigma$ from (2).

*Proof.* For simplicity, we exclude the factor of $1/T_{\text{sim}}$ in (7) throughout our proof. Under Setting 2, the MSE of component $i_k$ during environment interval $c$ under $IN(\epsilon)$ is:

$$\begin{aligned}&(\hat{x}_{i_k}(c\Delta t) - x_{i_k}(c\Delta t))^2 \Delta t^{(u)}\\&\quad + (y_{1k}(c\Delta t) - x_{i_k}(c\Delta t))^2 (c\Delta t + B_{2k}(c\Delta t))\\&\quad + \left(\frac{1}{2}(y_{1k}(c\Delta t) + y_{2k}(c\Delta t)) - x_{i_k}(c\Delta t)\right)^2\\&\qquad * (\Delta t - B_{2k}(c\Delta t) - \Delta t^{(u)})\end{aligned} \tag{13}$$

Under $OUT(\epsilon)$, the MSE is the same as (13) if sensor 2 transmits $y_{2k}(t)$; this occurs when

$$\begin{aligned}&\{B_{2k}(c\Delta t) < \Delta t^{(u)} + \Delta t^{(d)}\} \cup\\&\{B_{2k}(c\Delta t) \geq \Delta t^{(u)} + \Delta t^{(d)}, |y_{1k}(c\Delta t) - y_{2k}(c\Delta t)| \geq \epsilon\}\end{aligned} \tag{14}$$

When sensor 2 does not transmit, the MSE is:

$$\begin{aligned}&(\hat{x}_{i_k}(c\Delta t) - x_{i_k}(c\Delta t))^2 \Delta t^{(u)}\\&\quad + (y_{1k}(c\Delta t) - x_{i_k}(c\Delta t))^2((c+1)\Delta t - \Delta t^{(u)})\end{aligned} \tag{15}$$

From (2), $(1/2)(y_{1k}(t) + y_{2k}(t)) - x_{i_k}(t) = (1/2)(W_1 + W_2)$ for any $t$ and two independent $W_1, W_2 \sim \mathcal{N}(0, \sigma^2)$; likewise, $y_{1k}(t) - x_{i_k}(t) = W_1$. Hence, the difference between (15) and (13) yields $(\Delta t - B_{2k}(c\Delta t) - \Delta t^{(u)})((1/4)(W_1 + W_2)^2 - W_1^2)$, which is positive under Setting 1 if $(1/4)(W_1 + W_2)^2 - W_1^2 > 0$. Combining this with (14) yields our result. □

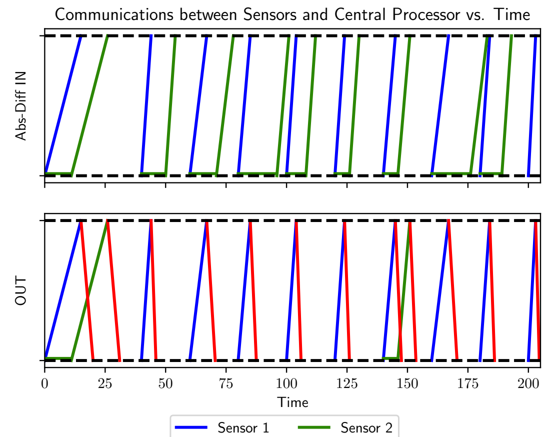



**Fig. 2:** A simple empirical illustration of the results of Theorems 1 and 2 when $\sigma = 0$ over environment (1): time-evolution of the communications between a central processor and $M = 2$ sensors under Absolute-Difference INformation architecture $IN(\epsilon)$ [Top] and OUTformation architecture $OUT(\epsilon)$ [Bottom] and event-triggered verification. Sensor 1 transmissions are in blue, sensor 2 in green, and central processor broadcasts in red. Under $OUT(\epsilon)$, if the central processor broadcasts an update from sensor 1, sensor 2 may cancel its scheduled backoff transmission (e.g., $t = 40, 60, \cdots, 120, 160, \cdots$). The slopes of each line vary because communication delays are proportional to the number of components transmitted/broadcast (see Definition 6).

**Remark 2** (Implications of Theorems 1 and 2)**.** In Figure 2, we demonstrate the cascading effect of the performance improvements derived in Theorems 1 and 2 by considering a longer interval of time than Setting 2. For significance, the original setting of multiple simultaneous component changes (see Section II-A) with no measurement noise ($\sigma = 0$) is plotted. For a general number $M \geq 2$ of sensors with small $\sigma$ in (2), multiple sensors may decrease in both uplink and downlink power at the expense of additional downlink power for one sensor, and Theorem 1 suggests an overall decrease in total power expenditure using $OUT(\epsilon)$ over $IN(\epsilon)$. Consequently, Theorem 2 and Figure 2 suggest that the difference in MSE performance between $OUT(\epsilon)$ and $IN(\epsilon)$ is always zero when $\sigma = 0$ in (2); when $\sigma \neq 0$, $OUT(\epsilon)$ yields less MSE than $IN(\epsilon)$ with probability (12).

An interesting MSE advantage of $OUT(\epsilon)$ over $IN(\epsilon)$ arises when we use larger verification periods $T^{(I)}_j > t_j$, defined in Definition 4. Again, we begin by considering a specific chosen interval of communications $[T_0, T_f) \subset [0, T_{\text{sim}})$, then remark the implications of cascading improvement over the longer interval $[T_0, T_{\text{sim}})$.

**Setting 3** (Interval of Communications II)**.** We consider the original dynamics (1) with nonzero random backoff time for both sensors (see Assumption 1). For sampling time $t_j$ defined in (2) and verification periods defined in Definition 4, let $T^{(I)}_j = T^{(O)}_j = T_j > t_j$ for sensor $j \in \{1, 2\}$. Choose interval $\mathcal{I} \triangleq [T_0, T_f)$, $T_0 \triangleq (a_2 - 1)T_2$ and $T_f \triangleq (a_1 + 1)T_1$ for some $a_1, a_2 \in \mathbb{N}$ such that $(a_2 - 1)T_2 < a_1 T_1 <$

$a_2T_2 < (a_1+1)T_1$. The environment (1) evolves such that for each sensor $j \in \{1,2\}$, 1) one unshared component $k_j \in \mathcal{U}_j$ changes value once by magnitude $d_j$ during $((a_j-1)T_j, a_jT_j]$, and 2) one shared component $k' \in \mathcal{S}$ changes value once by magnitude $d'$ during $((a_2-1)T_2, a_1T_1]$ and remains constant during $(a_1T_1, a_2T_2]$; here, $d_1, d_2, d' \sim \mathcal{U}[\underline{d}, \overline{d}]$ via the stepsize distribution from (1) and, using Notation 2, no other components $i \in \{1, \cdots, n\}/\{i_{k_1}, i_{k_2}, i_{k'}\}$ change value over $\mathcal{I}$. The sample path of observations $\mathcal{Y} \triangleq \{\mathbf{y}_j(t), t \in [0, (a_2-1)T_2) \cup \mathcal{I}, j \in \{1,2\}\}$ is such that $\beta^{(\chi)}_{2k_2}((a_2-1)T_2) = 1$, $\beta^{(\chi)}_{1k_1}(a_1T_1) = 1$, and $\beta^{(\chi)}_{jk'}(a_jT_j) = 1$ for $j \in \{1,2\}$, where $\beta^{(\chi)}_{jk}$ is the verification rule from Definition 4.

**Theorem 3** (MSE with Longer Verification Period)**.** Consider the interval defined in Setting 3. Let $\mathbf{x}^{(\chi)}(t)$ be constructed via the fusion rule from Assumption 1 such that $\hat{x}_{i_k}(T_0) \triangleq \hat{x}^{(I)}_{i_k}(T_0) = \hat{x}^{(O)}_{i_k}(T_0)$ where $T_0$ and $k \in \{k_1, k_2, k'\}$ are defined in Setting 3. Then the probability that the MSE contributed by component $i_{k_2}$ under $OUT(\epsilon)$ is less than that under $IN(\epsilon)$ during interval $\mathcal{I}$ is given by:

$$\begin{aligned}&(1 - G_{B_1,B_2}(a_1T_1 - a_2T_2 + 2\Delta t^{(u)} + \Delta t^{(d)})* \\ &\mathbb{P}\left(W_2^2 - (d_2 + W_1)^2 \leq 0, -\epsilon - d' < W_2' - W_1' < \epsilon - d'\right)\end{aligned} \tag{16}$$

Here, $G_{B_1,B_2}(s) \triangleq \mathbb{P}(B_1 - B_2 \leq s)$ for two independent $B_1, B_2 \sim F_B(\cdot)$ with $F_B$ defined in Assumption 1, $W_1, W_1', W_2, W_2' \sim \mathcal{N}(0, \sigma^2)$ are independent with $\sigma > 0$ from (2), and the communication delays $\Delta t^{(u)}, \Delta t^{(d)}$ are in Definition 6.

*Proof.* As in the proof of Theorem 2, we exclude the factor of $1/T_{\text{sim}}$ from (7). Under $OUT(\epsilon)$ in Setting 3, sensor 2 receives a broadcast update $y_{1k'}(a_tT_1)$ about component $k'$ and only transmits for component $k_2$; this occurs when

$$\begin{aligned}\{B_2(a_2T_2) - B_1(a_1T_1) \geq \\ a_1T_1 - a_2T_2 + 2\Delta t^{(u)} + \Delta t^{(d)}\} \cap \\ \{|y_{2k'}(a_2T_2) - y_{1k'}(a_1T_1)| < \epsilon\}\end{aligned} \tag{17}$$

Under $IN(\epsilon)$, sensor 2 transmits both components $k'$ and $k_2$; the central processor receives $y_{2k'}(a_2T_2)$ with an additional uplink delay of $\Delta t^{(u)}$ compared to $OUT(\epsilon)$. Thus the difference in MSE of component $i_{k_2}$ between $OUT(\epsilon)$ and $IN(\epsilon)$ during interval $\mathcal{I}$ is:

$$\begin{aligned}\Delta t^{(u)}((y_{2k_2}(a_2T_2) - x_{i_{k_2}}(c\Delta t))^2 \\ - (\hat{x}_{i_{k_2}}(a_1T_1) - x_{i_{k_2}}(c\Delta t))^2)\end{aligned} \tag{18}$$

where $c \in \mathbb{N}$ is such that $[a_2T_2 + B_2(a_2T_2) + \Delta t^{(u)}, a_2T_2 + B_2(a_2T_2) + 2\Delta t^{(u)}] \subset [c\Delta t, (c+1)\Delta t)$, $\Delta t$ defined in (1). By Setting 3 and (2), we have $y_{2k_2}(a_2T_2) - x_{i_{k_2}}(c\Delta t) = W_2$, $\hat{x}_{i_{k_2}}(a_1T_1) - x_{i_{k_2}}(c\Delta t) = d_2 + W_1$, and $y_{2k'}(a_2T_2) - y_{1k'}(a_1T_1) = d' + (W_2' - W_1')$ for independent $W_1, W_1', W_2, W_2' \sim \mathcal{N}(0, \sigma^2)$. Combined with (17), the result follows. $\square$

**Remark 3** (Implications of Theorem 3)**.** In Figure 3, we demonstrate the cascading effect of the MSE advantage derived in Theorem 3 by considering a longer interval of time than Setting 3. For presentation clarity, we choose prime-valued $T_j^{(\chi)}$ and zero backoff time. For comparison, we include the communications and MSE for the pure INformation architecture $IN_0$ (see Notation 3); $IN_0$ transmits the most number of components uplink, consequently incurring the longest transmission delays and largest MSE among the three architectures. By transmission rule (6), $OUT(\epsilon)$ always transmits the same number of components or less than $IN(\epsilon)$. Less components are likely to be transmitted for small measurement noise $\sigma$ relative to the distribution of the stepsize $\mathbf{d}(c), c \in \mathbb{N}$ in (1), since one sensor cancels backoff

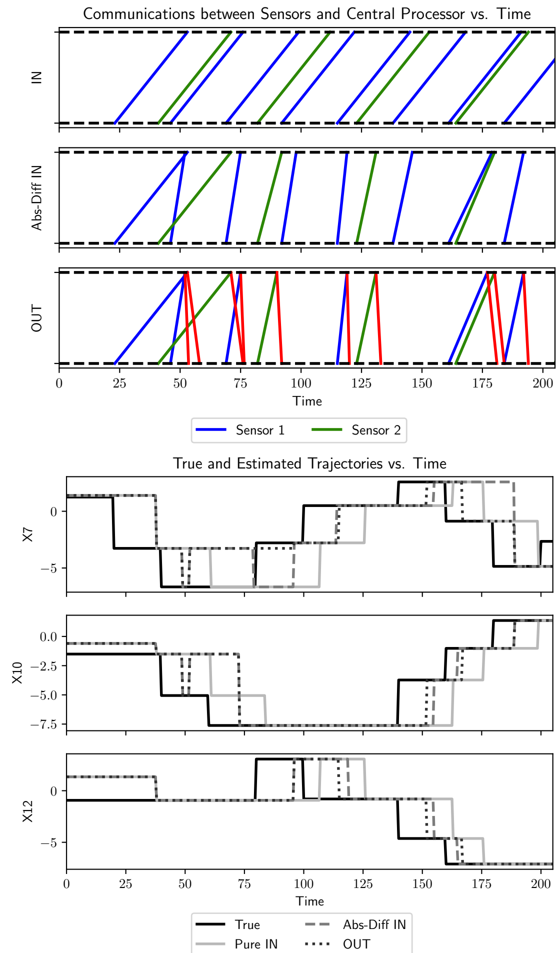
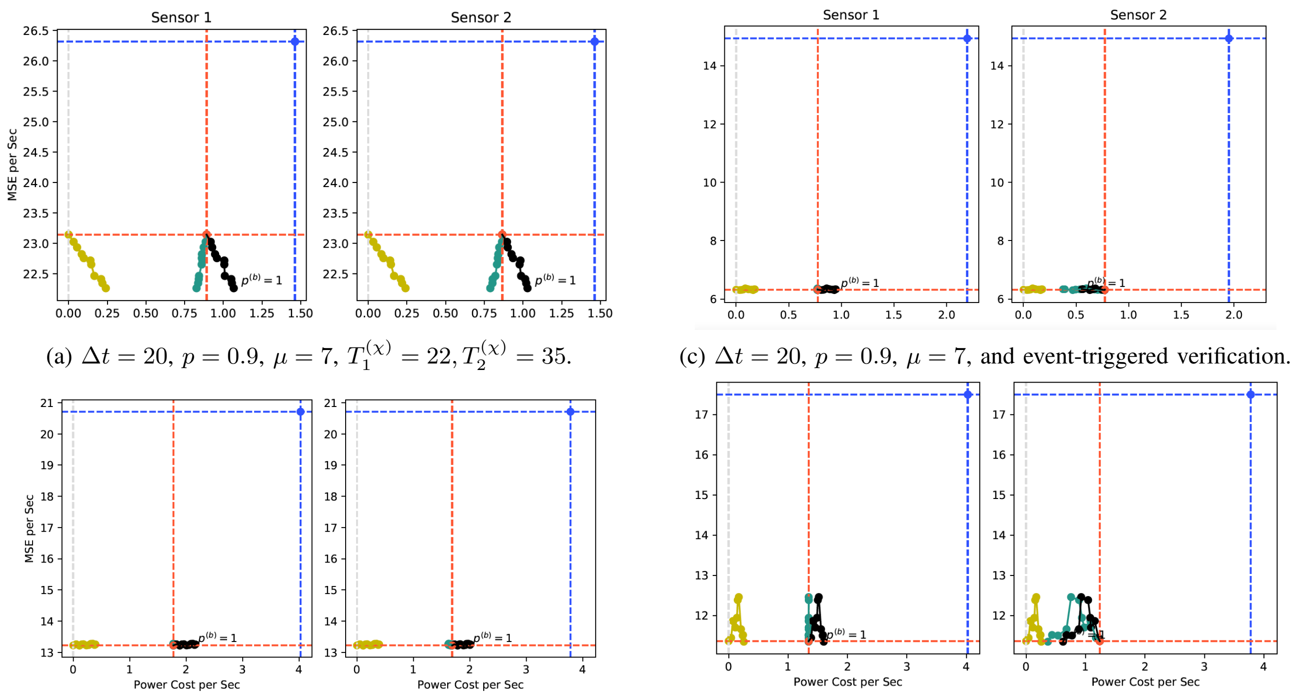



**Fig. 3:** [Top] A version of Figure 2 under original dynamics (1) with longer verification periods $T_1^{(\chi)} = 23, T_2^{(\chi)} = 41$, $\chi \in \{I, O\}$. All three architectures $IN_0$, $IN(\epsilon)$, and $OUT(\epsilon)$ of Notation 3 are plotted. Note three times where the uplink communications of $OUT(\epsilon)$ are less than $IN(\epsilon)$: 1) at $t = 92, 138$, where sensor 1 cancels its scheduled transmission and 2) at time $t = 161$, where the slope of sensor 1's uplink transmission is steeper. This suggests that updates from sensor observations are received more quickly on average under $OUT(\epsilon)$ than $IN(\epsilon)$. [Bottom] The evolution of components 7, 10, 12 of $\mathbf{x}(t)$ and $\hat{\mathbf{x}}^{(\chi)}(t)$ over time. Since the central processor receives transmissions more quickly, $\hat{\mathbf{x}}^{(O)}(t)$ tracks $\mathbf{x}(t)$ more accurately than $\hat{\mathbf{x}}^{(I)}(t)$.




(a) $\Delta t = 20$, $p = 0.9$, $\mu = 7$, $T_1^{(\chi)} = 22, T_2^{(\chi)} = 35$.

(c) $\Delta t = 20$, $p = 0.9$, $\mu = 7$, and event-triggered verification.

(b) $\Delta t = 45$, $p = 0.9$, $\mu = 7$, $T_1^{(\chi)} = 22, T_2^{(\chi)} = 35$.

(d) $\Delta t = 20$, $p = 0.9$, $\mu = 20$, and event-triggered verification.

**Fig. 4:** Plots of the MSE per timestep against power expenditure per timestep for both sensors 1 and 2, averaged over 40 Monte-Carlo simulations of environment (1). Each row of subfigures differ by their values of $\Delta t$ and $p$ (see (1)), mean random backoff time $\mu$, and frequency of verification (see Definition 4). The architectures compared are described in Notation 3. The blue dot corresponds to $IN_0$, and the orange dot for $IN(\epsilon)$, with dashed lines to emphasize their visibility. For $OUT(\epsilon)$, a constant broadcast probability $p^{(b)}$ is ranged over $\{0, 0.1, \cdots, 1\}$, and we make separate curves for the uplink, downlink, and the total (uplink + downlink) power expenditures; gold dots for downlink, green for uplink, and black for the combined total. As in Remark 3, $IN_0$ performs the worst in both metrics for all four cases. The green and black dots for $OUT(\epsilon)$ with $p^{(b)} = 0$ coincide with $IN(\epsilon)$ because no broadcasts are made. Under environments with fast variation (small $\Delta t$) relative to the verification periods $T_j^{(\chi)}$, $OUT(\epsilon)$ detects changes more quickly than $IN(\epsilon)$, providing a reduction in MSE at additional downlink power (see (a) and Remark 3); this tradeoff becomes dampened when $\Delta t$ is large, and the MSE performance rarely varies with power expenditure (see (b)). Under event-triggered verification, $OUT(\epsilon)$ maintains similar MSE as $IN(\epsilon)$ by having one sensor reduce power expenditure at the cost of the other sensor (see (c) and Remark 2); longer mean backoff time causes increasingly outdated data to be received, which increases the MSE (see (d)).

transmissions if broadcast updates from the other sensor contain similar component values (e.g., $t = 92, 138, 161$ in Figure 3). Because $OUT(\epsilon)$ incurs the least uplink delays on average among the three architectures, the central processor under $OUT(\epsilon)$ detects changes in the environment state most quickly; this is also captured by Theorem 3, where the event $\{W_2^2 - (d_2 + W_1)^2 \leq 0\}$ occurs with high probability. On the other hand, for large $\sigma$ or a suboptimal choice of threshold $\epsilon$ relative to the distribution of the stepsize, the event $\{-\epsilon - d' < W_2' - W_1' < \epsilon - d'\}$ occurs with low probability.

## IV. Numerical Analysis

In this section, we supplement the theoretical insights derived from the previous Section III by empirically plotting the tradeoff space under a variety of dynamics and parameter choices. In particular, we implement the original setup of Section II-A, consider longer intervals than those specified in Settings 2 and 3, and remove specific constraints on the sample path of observations $\mathcal{Y} \triangleq \{\mathbf{y}_j(t), t \in [0, T_{\text{sim}}), j \in \{1, 2\}\}$ and state estimates $\hat{\mathbf{x}}^{(\chi)}, \chi \in \{I, O\}$ imposed in Theorems 1, 2, and 3. We consider the effects of varying three parameters which demonstrate the most interesting comparisons between the performance of the three architectures from Notation 3: 1) the variation in the environment $\Delta t$ (see (1)), 2) the mean $\mu$ of the random backoff time distribution $F_B$ (see Assumption 1), and 3) the frequency of verification (see Definition 4). For $OUT(\epsilon)$, we additionally vary over $p^{(b)} \in [0, 1]$, a constant probability in which the central processor makes a broadcast upon receiving a transmission. We choose communication delays $\Delta t^{(d)} < \Delta t^{(u)}$ and power expenditure rates $P_U < P_D$ (see Definition 6).

The results are demonstrated in Figure 4. With a larger number $M \geq 2$ of sensors, one may be inclined to believe that $IN_0$ obtains the lowest MSE under an averaging fusion rule (see Assumption 1) by the law of large numbers, because $IN_0$ transmits the largest number of independent, redundant noisy observations. However, when communication delays and random backoff times are involved, a larger number of transmissions will take a longer delay to be received by the central processor, causing an increase in MSE. This effect was demonstrated in Remark 3.

For the values chosen to generate Figure 4 (a), $OUT(\epsilon)$ is greater than $IN(\epsilon)$ in power expenditure while $IN(\epsilon)$ is greater than $OUT(\epsilon)$ in MSE for all $p^{(b)} \in (0, 1]$. This effect was demonstrated in Remark 3; $OUT(\epsilon)$ tracks changes in the true environment more quickly than $IN(\epsilon)$, but with greater downlink power expenditure. In Figure 4 (a), this tradeoff is shown explicitly: increasing downlink power (in gold) allows for decreasing uplink power (in green) for both sensors. This suggests that for $OUT(\epsilon)$ applied to settings similar to Figure 4 (a), a balance between total power expenditure and MSE may be obtained by choosing $p^{(b)}$ appropriately. On the other hand, in Figure 4 (b), transmissions are more likely to be received by the central processor while the environment state has not changed (i.e., $T_j^{(\chi)} + \mu < \Delta t, \chi \in \{I, O\}$). Essentially, if the environment variation is slow relative to the sensors' verification periods, then $OUT(\epsilon)$ maintains approximately the same MSE as $IN(\epsilon)$ on average, aside from small variations due to noise $\sigma$, and wastes power by broadcasting unnecessarily.

The tradeoff space in Figure 4 (c) demonstrates that Remark 2 holds more generally for slower-changing environments (large $\Delta t$) and small random backoff time relative to the environment (i.e., $\mu < \Delta t$). For $OUT(\epsilon)$ under small $\sigma$, sensor 1 expends power for both transmitting and receiving broadcasts, while sensor 2 expends mostly downlink power for receiving broadcast updates. Because both sensors operate on event-triggered verification, $OUT(\epsilon)$ is, on average, the same as $IN(\epsilon)$ in the MSE; any variations such that $OUT(\epsilon)$ obtains more or less MSE than $IN(\epsilon)$ is a result of noise. This suggests that for $OUT(\epsilon)$ applied to settings similar to Figure 4 (c), a balance for the total power expenditure may be obtained by choosing $p^{(b)}$ appropriately, at little change in MSE. When $\mu \geq \Delta t$, as in Figure 4 (d), the power expenditure follows the same trends as when $\mu < \Delta t$, but the overall MSE tends to be larger. This is expected because $\mu > \Delta t$ implies that on average, sensor 2's observations about $\mathbf{x}(c\Delta t)$ are received when the true state is already changed to $\mathbf{x}((c+1)\Delta t)$.

## V. Conclusion

We provided a theoretical and numerical characterization of the tradeoff space for architectures with broadcast feedback (OUTformation) and architectures without feedback (INformation) using two performance metrics: the mean-squared error of the central processor's estimate of the environment state and the total power expenditure per sensor. Our study was motivated towards enabling users to make informed design choices in distributed data-gathering architectures for large-scale network environments under constraints such as nonzero communication delays and limited knowledge of the environment dynamics. We found that under an event-triggered verification rule (see Definition 4), OUTformation architectures expend less uplink power on average than INformation, and yields similar MSE when variation in the environment is large relative to mean backoff time (see Theorem 1, Remark 2, Figure 4 (c) and (d)). Under a periodic verification rule, OUTformation architectures enable a reduced MSE on average compared to INformation, but with additional downlink power that potentially increases the total power expenditure (see Theorem 2, Theorem 3, and Remark 3). This suggests that by varying parameters (i.e., the broadcast probability $p^{(b)}$ described in Section IV), the OUTformation architecture can attain a better tradeoff between the MSE and the total power expenditure (see Figure 4 (a) and (b)). In conclusion, the main advantage of feedback is that each sensor's understanding of the central processor's state estimate improves. Theoretical and numerical analysis of performance tradeoffs for architectures with and without feedback when other parameters (e.g., threshold $\epsilon$, number of sensors) are varied is a natural subject of future work.

## References


[1] F. Castanedo, "A review of data fusion techniques," *The Scientific World Journal*, vol. 2013, pp. 1–19, Oct 2013.

[2] M. Liggins, C.-Y. Chong, I. Kadar, M. Alford, V. Vannicola, and S. Thomopoulos, "Distributed fusion architectures and algorithms for target tracking," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 95–107, 1997.

[3] S. Bandyopadhyay and E. J. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," in *IEEE INFOCOM 2003.*, vol. 3, 2003, pp. 1713–1723.

[4] O. Younis and S. Fahmy, "Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach," in *IEEE INFOCOM 2004.*, vol. 1, 2004.

[5] B. Chen, W.-A. Zhang, L. Yu, G. Hu, and H. Song, "Distributed fusion estimation with communication bandwidth constraints," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1398–1403, 2015.

[6] X. Ge and Q.-L. Han, "Distributed event-triggered $H_\infty$ filtering over sensor networks with communication delays," *Information Sciences*, vol. 291, pp. 128 – 142, 2014.

[7] G. Nutt and D. Bayer, "Performance of CSMA/CD Networks Under Combined Voice and Data Loads," *IEEE Transactions on Communications*, vol. 30, no. 1, pp. 6–11, 1982.

[8] R. Olfati-Saber and J. Shamma, "Consensus filters for sensor networks and distributed sensor fusion," in *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005, pp. 6698–6703.

[9] P. Fearnhead and G. Rigaill, "Changepoint detection in the presence of outliers," *Journal of the American Statistical Association*, vol. 114, no. 525, pp. 169–183, 2019.