Energy-Efficient Temporal Sensing: An Age of Sample Based Approach

Vini Gupta and Swades De

***Abstract*—In this paper an energy-efficient temporal sensing framework is proposed for a wireless sensor node (SN) monitoring a temporal process. A concept of age of sample (AoS) is introduced, and expressions for the AoS and average AoS functions are derived that capture freshness of sensed samples (or inter-sample time). To incorporate the effect of process variations, weighted AoS (WAoS) and its average functions are developed. The framework solves a multi-objective optimization problem (MOP) that optimizes this average WAoS function and energy efficiency of the SN to select a few temporal sensing instances of a sensing window while maintaining a predefined sensing quality. An upper bound on the average WAoS is derived which makes the MOP solvable. Using these few measurements, the process signal across entire sensing window is estimated by leveraging inherent temporal correlation. The idea of adapting sensing window according to the changing correlation of the process is also presented, which addresses non-stationary aspect of monitored process. Simulation studies on real data-sets illustrate that, on comparison with the closest existing scheme, the proposed scheme provides 30.1% gain in sensing quality while consuming nearly same sensing energy. Further, it consumes 22.4% lesser sensing energy to maintain nearly same sensing quality.**

***Index Terms*—Age of sample, energy efficiency, sensing window, sensing quality, temporal correlation, multi-objective optimization**

I. Introduction

In the era of Internet-of-Things (IoT), various real- and non-real-time monitoring applications, such as smart healthcare [1], environment sensing (pollution monitoring) [2], industrial process monitoring [3], etc., rely mainly on wireless sensor networks (WSNs). Such networks contain battery-constrained sensor nodes (SNs) that limits their continuous sensing operations. To alleviate this problem, intelligence is imparted at the node-level to perform energy-efficient sensing. This energy-efficiency is realized by exploiting temporal correlation-induced sparsity present in a slowly-varying monitored process. This allows sensing of the process at a few time instances and using these few measurements to estimate entire observation window without significant loss in sensing quality.

A. Related Works

Early technique, presented in survey [4], altered duty cycle $\left(\frac{\text{on time}}{\text{(on + off) time}}\right)$ of a SN either randomly or based on some logic. This saves SN's energy and extends its lifetime. Likewise, the work in [5] dynamically altered the duty-cycle of a mote using a machine learning-based preamble sampling scheme. Further, some variants of random sampling approaches are discussed in works [6]–[9]. Herein, for each time instance k in a window of total K instances, a few sensors are randomly activated. Thus, a sensor is activated only at a few temporal instances out of total K instances.

Subsequently, several works were reported in literature that exploit temporal variations of the to-be-sensed processes to adapt the sampling rate at the nodes. In [7], sampling rate is exponentially increased and additively decreased based on some heuristic. In contrast, in [10], a linear increase and an exponential decrease-based approach is employed to adapt the sampling rate while simultaneously ensuring the required sensing quality. Further, context-aware adaptive sampling schemes were proposed in [11] for sensing dependent parameters in a landslide monitoring scenario. Recently, various multi-sensing approaches that adapt the sampling rate by exploiting temporal, spatial, and cross-correlation among various monitoring processes were proposed in the work [12].

Analogous to the IoT monitoring application, the authors in [13] proposed a cloud resource monitoring approach that used certain user-defined policies as threshold for current stream variability to adjust monitoring intensity. Recently, an adaptive monitoring (AdaM) framework for IoT devices was proposed in the work [14]. It dynamically adjusts temporal sampling rate of the sensing device based on the confidence of the model (probabilistic moving average estimation model) used to estimate current evolution of metric stream.

The temporal sampling problem is also addressed from communication perspective in works [15]–[19]. The sampling schemes suggested therein, such as zero-wait [17], [19], sample-at-change [19], uniform sampling, threshold policy in signal variation [17], aimed to optimize the freshness of information that a receiver has about the status of a source. To capture the freshness, age of information (AoI) metric is used in [17] while mutual information is employed in [18].

B. Research Gap and Motivation

The duty cycling and random sampling-based techniques do not guarantee the required sensing quality, as highlighted in [20]. Further, the sampling rate adaptation logic proposed in random sampling-based work [7] may not provide significant energy saving due to multiplicative increase in the sampling rate when the process variation increases above a heuristically chosen threshold value.

As suggested in [13], the sampling frequencies should be adapted such that the changes in the statistical characteris-

The authors are with the Department of Electrical Engineering and Bharti School of Telecommunications, Indian Institute of Technology Delhi, New Delhi 110016, India (e-mail: vini.gupta@ee.iitd.ac.in; swadesd@ee.iitd.ac.in).

tics of the monitored time series (i.e., the sensed process) are accommodated. The AdaM framework [14] accounts for changes in the statistical properties of the process. However, the window used for sampling period ($T \in [T_{min}, T_{max}]$) is kept fixed for all scenarios. Intuitively, the duration of data stream/process with *high* correlation should have a *larger* window and vice-versa. Besides this, most of the prior works do not incorporate measurement noise in the system model. Thus, the thresholds set therein for sampling rate adjustment may not work for noisy measurement scenarios. Further, the works [13] and [21] give insight on the inability of the error measure to guarantee an accurate quality measure for dynamic processes. However, the approaches [6], [7], [14] considered some form of the error measure only to achieve energy-efficient temporal sampling.

The AoI-based monitoring system in [15] assumed a constant inter-sample time which may not suit well for monitoring both slowly and rapidly varying processes. The sampling policies in [16], [17], [19] aimed to achieve the timeliness/freshness of status updates at the receiver without considering energy efficiency aspect of the battery-constrained sources (e.g. a sensor node). Further, in the threshold-based [17] and sample-at-change [19] approaches, the sampling decision (sense or not) at a time requires exact knowledge of the process signal at that time, which can be obtained only after performing sampling. Thus, there is no point in making the sampling decision when sensing has to be performed for the same. Also, this will require continuous sampling which is inefficient from both the energy consumption and communication perspective.

To fill these research gaps, this work proposes a novel age of sample (AoS)-based energy-efficient temporal sensing framework for a wireless sensor node. The AoS at any time t accounts for the time elapsed since the sensing time of the most recent sample of the monitored process (i.e., $t - t_{recent}$). The framework considers the temporal variation of the process and jointly optimizes the average AoS function and energy efficiency to activate the SN during a subset of time instances in a sensing window such that the required sensing quality is maintained. With the newly-introduced AoS function the sensing instants within a sensing window are optimized. From sensing perspective, this has potential applicability for real-time monitoring/sensing. Further, the developed framework adjusts the sensing window based on the correlation statistics of the non-stationary monitored process. It also provides an error-bounded estimate of the process signal across the entire sensing window using a few sensed temporal measurements. It is also notable that the measurement noise is considered in the selection of temporal sensing instances and signal estimation task as well.

The AoI-based freshness of data is modeled using channel-induced data transmission delay from source to receiver in [16]. In this context, the work [22] utilized the speed of the reader and distance traveled by it to collect data from the sensors of a cluster. Different from the AoI-based freshness of data update at the receiver, the proposed AoS-based sensing deals with freshness of the sampling process at the source node. To the best of our knowledge, the optimization of time/gap between the sensing operations within a sensing window (quantified using AoS) has never appeared in the literature. In contrast to the AoI functions, the proposed weighted AoS function depends on the variations of the monitored process.

C. Contributions and Significance

Key contributions of this work are as follows:

1) A concept of AoS function for quantifying the staleness of a sensed sample of an environmental process is introduced. Generalized AoS and average AoS functions are derived. A weighted AoS function is also defined to capture the effect of variations of the process on the proposed AoS-based temporal sensing framework.
2) A multi-objective optimization problem (MOP) is proposed that jointly optimizes the weighted AoS function and node's energy efficiency subject to a constraint on the sensing error. It selects a few sensing instances of a sensing window which contains multiple time instances.
3) An upper bound on the average weighted AoS function is derived that makes the proposed MOP tractable/solvable.
4) A logic is developed to estimate the time instants at which the statistics (mean value is considered here) of the process changes. The length of sensing window is adapted at this point according to the correlation present in the monitored process. This captures non-stationarity aspect as well of the monitored process.
5) The proposed framework is tested on real data-sets of two pollutants monitored at different sites. Simulation results demonstrate that the proposed framework provides a better average sensing quality compared to the closest existing competitive scheme while consuming nearly same sensing energy. Further, to provide same sensing quality as that of the competitive scheme, the proposed framework consumes lesser sensing energy. Unlike the existing literature, this work adapts the sensing window length according to the correlation of the process. Heat-maps of the actual process and the one estimated using the temporal sensing measurements are also presented to highlight the obtained sensing performance.

The proposed framework has immense utility in various IoT applications such as smart healthcare, environment monitoring, border surveillance, hazardous gas monitoring, etc.

D. Organization and Notation Conventions

The layout of the paper is as follows. The system model for data acquisition and estimation are presented in Section II. Section III presents the proposed concept of the AoS. Temporal sensing framework is described in Section IV, followed by simulation results and concluding remarks, respectively in Sections V and VI.

In this work, a real-valued matrix of size $M \times K$ and a real vector of size $K \times 1$ are denoted by $\mathbf{V} \in \mathbb{R}^{M \times K}$ and $\mathbf{v} \in \mathbb{R}^{K \times 1}$ respectively. The k^{th} element and l_2-norm of vector $\mathbf{v}$

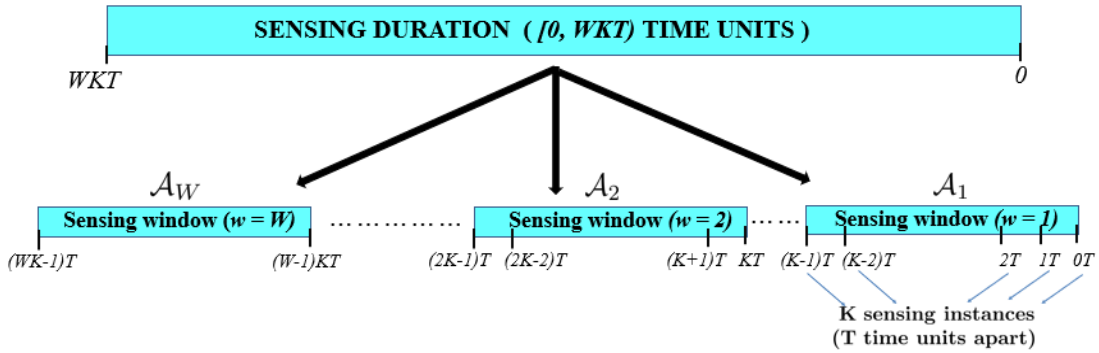

Fig. 1: Set-up for sensing.

are respectively represented by $v(k)$ and $\|\mathbf{v}\|$. The $(m,k)^{th}$ element, m^{th} row, and trace of $\mathbf{V}$ are represented by $\mathbf{V}(m,n)$, $V(m,:)$, and $\mathrm{Tr}\{\mathbf{V}\}$ respectively. The operators $(.)^T$, $\mathrm{diag}(\cdot)$, and $|.|$ denote the transpose of the vector/matrix, the standard diagonalization operation on a vector, and the cardinality of a set respectively. The superscript (k) and subscript w refer to the vector/matrix argument corresponding to k^{th} time instance of w^{th} sensing window respectively.

II. System Model: Data Acquisition and Estimation

Consider a SN monitoring a temporally varying process. Let the temporal sensing instances be T time units apart. A sensing window $\mathcal{A}_w$ that contains total K time instances for sensing is considered. $\mathcal{A}_w = \{(w-1)KT, (w-1)KT+T, \cdots, wKT-T\}$, $w = 1, \cdots, W$, with W being total number of sensing windows in which entire sensing duration (WKT time units) is divided as shown in Fig. 1. Note that the time duration of $w = 1^{th}$ sensing window ($\mathcal{A}_1$) is $[0, KT)$ with sensing instances $\{0, T, \cdots, (K-1)T\}$. The time $[(K-1)T, KT)$ is included because before the next sensing operation at KT time in the next window, the AoS will be governed by the current sensing window ($w = 1$).

Let the signal corresponding to the monitored process sensed by the SN during the K time instances of the window w is represented by temporal signal vector $\mathbf{z}_w = [z_w(1), \cdots, z_w(K)]^T \in \mathbb{R}^{K\times 1}$. Due to the temporal correlation present in the process, it can be monitored/sensed using a few time instances. Let a fusion center (FC) selects $M_w\,(\leq K)$ time instances for sensing by the SN during the sensing window w. Let the active/sleep status of the SN during the different temporal instances of window $\mathcal{A}_w$ be denoted by $a_w^{(k)} = 1/0,\ \forall\ 0 \leq k \leq K-1$. These constitute a binary activity set $\mathcal{M}_w$ such that $|\mathcal{M}_w| = \sum_{k=0}^{K-1} a_w^{(k)} = M_w$. For instance, if the SN senses at time instances 0T, (K-2)T, and (K-1)T, then $a_1^{(0)} = 1, a_1^{(K-2)} = 1$, and $a_1^{(K-1)} = 1$, which results in $\mathcal{M}_1 = \{1, 0, \cdots, 1, 1\}$ as shown in Fig. 2. The activity set $\mathcal{M}_w$ is decided by the FC before the window w's duration starts as discussed in Section IV and conveyed to the SN. The SN performs sensing operation during each of the active time instances and simultaneously sends the measurements to the FC. Note that the FC, however, performs temporal sensing instances selection after every K time instances, i.e. for a complete upcoming sensing window. To represent system model for the sensing window w, let $\mathbf{A}_w \in \mathbb{R}^{M_w\times K}$ denotes binary temporal sensing matrix which captures active/sleep status of the SN during different time instances of the sensing window $\mathcal{A}_w$. Each row corresponds to a distinct active time instance (i.e. $a_w^{(\cdot)} = 1$). The row m representing $(n-1)^{th}$ active time instance, is given by,

$$\mathbf{A}_w(m,k) = \begin{cases} 1 \text{ (active)} & , k = n \ \text{ s.t. } a_w^{(n-1)} = 1 \\ 0 \text{ (sleep)} & , k \in \{1,\ldots,K\}\backslash\{n\} \end{cases}.$$

Let the temporal measurement vector $\mathbf{y}_w \in \mathbb{R}^{M_w\times 1}$ contains the signal sensed during the active time instances $\in \mathcal{M}_w$. Thus, the data acquisition model for the w^{th} sensing window is given by,

$$\mathbf{y}_w = \mathbf{A}_w\mathbf{z}_w + \mathbf{n}_w, \tag{1}$$

where $\mathbf{n}_w \in \mathbb{R}^{M_w\times 1}$ represents white Gaussian measurement noise vector with independent and identically distributed components having zero mean and variance σ^2. Upon reception of all M_w measurements, the FC estimates complete temporal signal vector $\mathbf{z}_w$ and run temporal instances selection problem to convey temporal sensing schedule for the next sensing window to the SN. These temporal instances selection and estimation tasks are repeated for each sensing window.

Due to the temporal correlation inherent in the process, a sparse representation of the system model is derived using discrete Fourier transform (DFT)-based sparsification matrix $\mathbf{B} \in \mathbb{R}^{K\times K}$ as follows,

$$\mathbf{y}_w = \mathbf{A}_w\mathbf{B}\mathbf{x}_w + \mathbf{n}_w, \tag{2}$$

where $\mathbf{x}_w \in \mathbb{R}^{K\times 1}$ represents corresponding sparse vector. In the work [23], the above binary sensing matrix (called partial canonical identity (PCI) matrix therein) is shown to be highly incoherent with the DFT matrix and thus, the corresponding sparse vector can be reconstructed using sparse recovery schemes such as compressed sensing (CS) [24] used there. Motivated by the superior estimation performance of the sparse Bayesian learning (SBL) scheme [25] in ill-posed estimation scenarios, it is used to obtain estimate $\widehat{\mathbf{x}}_w \in \mathbb{R}^{K\times 1}$ of the sparse vector using the vector $\mathbf{y}_w$ and dictionary matrix $\mathbf{A}_w\mathbf{B} \in \mathbb{R}^{M_w\times K}$. Estimate of the temporal signal vector is then obtained as $\widehat{\mathbf{z}}_w = \mathbf{B}\widehat{\mathbf{x}}_w \in \mathbb{R}^{K\times 1}$.

It is worthwhile to mention here that the original sampling period T is decided based on Nyquist rate as follows: $f_s >=$

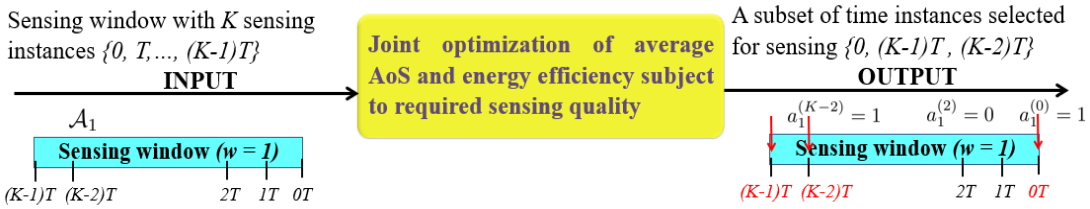

Fig. 2: Temporal sensing instances selection at the FC.

$2f_{max}$ which gives $T <= \frac{1}{2f_{max}}$, where $f_s = \frac{1}{T}$ is sampling frequency and f_{max} is maximum frequency of the measured signal. If T is set as $\frac{1}{2f_{max}}$ and the proposed framework selects a few time instants (say $M_w < K$) of the sensing window of length K for activation, then the time between two consecutive samples (i.e., inter-sample time) may evaluate to a value greater than T within this window. Although this will violate the Nyquist rate, but the beauty of sparse signal recovery used for sensing window's signal estimation is that it still recovers signal with good sensing quality. Note that the sparse signal estimation is possible because of the sparse representation of the signal owing to the inherent temporal correlation.

Also, note the assumption that the SN is configured to sense only at integer multiples of T time units if activated. Such an SN will sense the signal every T time instance for the conventional periodic sampling case. While for the proposed temporal sensing framework (Section IV), the inter-sample time of the SN could be a non-zero integer multiple of T.

III. Age of Sample (AoS) Function

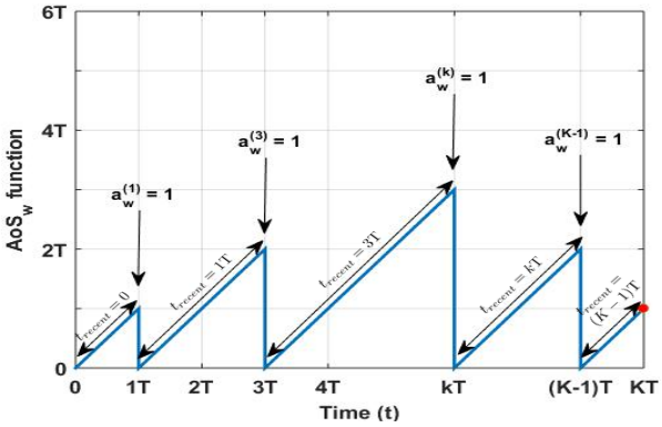

Fig. 3: Sample AoS pattern.

This section formulates an age of sample (AoS) function which represents time elapsed between the current time t and the sensing time of the most recently sensed sample of the process t_{recent}. Thus, at any time t, the AoS$(t) = t - t_{recent}$. It is a measure of freshness of the sensed sample. Smaller value of the AoS function signifies lesser inter-sample time and vice-versa. At any temporal instance kT activated for sensing, the AoS(kT) evaluates to zero because $t = t_{recent} = kT$. During absence of sensing, the AoS increases linearly in time. Fig. 3 shows pattern of the AoS function for a sensing window $\mathcal{A}_w$ with active sampling instances $a_w^{(1)}, a_w^{(3)}, a_w^{(k=6)}$, and $a_w^{(K-1=8)}$, i.e. $\mathcal{M}_w = \{0,1,0,1,0,0,1,0,1\}$. Note that, irrespective of whether the signal sampled at $t = 0$ (i.e. $a_w^{(0)} = 1/0$), $t_{recent} = 0$ until the first sample sensed after 0^{th} time instance.

The generalized expression of the AoS at any time $t \in [0, KT)$ of the w^{th} window, i.e., $\text{AoS}_w(t)$ in terms of the elements $a_w^{(\cdot)}$ of the activity set $\mathcal{M}_w$ is given by,

$$\begin{aligned}\text{AoS}_w(t) &= t - \max\left\{ia_w^{(i)}T, 0 \le i \le (j-1)\right\}, \\ & t \in [(j-1)T, jT), j = 1, \cdots, K. \end{aligned} \quad (3)$$

Note that, for a particular j, the $\text{AoS}_w(t)$ in (3) with $t \in [(j-1)T, jT)$ deals with the temporal sensing decision at a new $(j-1)T$ instance $\left(\text{i.e. } a_w^{(j-1)} = 1 \text{ or } 0\right)$.

Since the $\text{AoS}_w(t)$ depends on the active/sleep status of the SN during different sensing instances which are decided based on some intelligence/logic, it is not straightforward to characterize its distribution. Thus, the time average of the AoS_w function, denoted by $\overline{\text{AoS}_w}$, is computed as the area under the sawtooth function in Fig. 3.

$$\begin{aligned}\overline{\text{AoS}_w} &= \frac{1}{KT}\int_0^{KT} \text{AoS}_w(t)\,dt \\ &= \frac{1}{KT}\sum_{j=1}^{K}\left[\int_{(j-1)T}^{jT}\left(t - \max\left\{ia_w^{(i)}T, 0 \le i \le j-1\right\}\right)dt\right] \\ &= \frac{1}{KT}\left[\int_0^{KT} t dt - \sum_{j=1}^{K}\int_{(j-1)T}^{jT}\max\left\{ia_w^{(i)}T, 0 \le i \le j-1\right\}dt\right] \\ &= \frac{KT}{2} - \frac{1}{K}\sum_{j=1}^{K}\max\left\{ia_w^{(i)}T, 0 \le i \le j-1\right\} \end{aligned} \quad (4)$$

Minimizing the $\overline{\text{AoS}_w}$ function will only minimize the inter-sample time. It evaluates to a minimum value of $\frac{T}{2}$ when the process signal is sensed at every time instance (i.e., $a_w^{(k)} = 1, \forall k$). However, it does not answer when to decrease/increase the inter-sample time. Intuitively, the temporal sensing should be governed by the variations of the underlying process. If the process variation between two prospective sampling instances is high, then the sampling frequency should be high, i.e., the inter-sample time should be low resulting in a lower value of the AoS function. To capture this notion, weighted AoS function (WAoS_w) is defined in (5).

$$\begin{aligned}\text{WAoS}_w(t) = t - \max\left\{\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T,\ 0 \le i \le (j-1)\right\}, \\ t \in [(j-1)T, jT), j = 1, \cdots, K, \end{aligned} \quad (5)$$

where $\delta_{(j-1)T-iT}$ represents average variations of the process between time interval $[iT, (j-1)T]$. It is computed using the

$$\max\left\{\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T, 0\leq i\leq (j-1)\right\} \;\geq\; \frac{1}{j}\sum_{i=0}^{j-1}\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T \tag{8a}$$

$$\Rightarrow -\max\left\{\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T, 0\leq i\leq (j-1)\right\} \;\leq\; -\frac{1}{j}\sum_{i=0}^{j-1}\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T \tag{8b}$$

$$\Rightarrow \frac{KT}{2}-\frac{1}{K}\sum_{j=1}^{K}\max\left\{\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T, 0\leq i\leq (j-1)\right\} \;\leq\; \frac{KT}{2}-\frac{1}{K}\sum_{j=1}^{K}\left(\frac{1}{j}\sum_{i=0}^{j-1}\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T\right) \tag{8c}$$

$$\Rightarrow \overline{\text{WAoS}_w} \;\leq\; f_{UB}\left(a_w^{(\cdot)}\right). \tag{8d}$$

process signal estimated during previous sensing windows. Since a particular j in the AoS_w function (3) deals with the sensing decision at $(j-1)\,T$ time instance, the variations of the past instances are considered with respect to $(j-1)\,T$ inside the max function in (5). The corresponding average function $\overline{\text{WAoS}_w}$ is computed on similar lines as done for the function $\overline{\text{AoS}_w}$ and is given by,

$$\overline{\text{WAoS}_w}=\frac{KT}{2}-\frac{1}{K}\sum_{j=1}^{K}\max\left\{\left(\delta_{(j-1)T-iT}\right) ia_w^{(i)}T, 0\leq i\leq (j-1)\right\}. \tag{6}$$

Activating the SN at time instances with higher $\delta_{(\cdot)}$ will minimize the $\text{WAoS}_w(t)$ and $\overline{\text{WAoS}_w}$ functions, resulting in frequent sensing (or lesser inter-sample time). Note that the $\overline{\text{AoS}_w}$ function is non-negative while the $\overline{\text{WAoS}_w}$ function may give a negative value due to the incorporated weight $\delta_{(\cdot)}$. In order to minimize the former one based on the process variations, the latter function is minimized in the proposed framework discussed in the Section IV). Further, to make the optimization problem (Section IV-A) of the proposed framework solvable/tractable, an upper bound on the WAoS function $\left(f_{UB}\left(a_w^{(\cdot)}\right)\right)$, derived in Lemma 1, is used.

Lemma 1. *The average weighted AoS function $\overline{WAoS_w}$ is bounded above by a function $f_{UB}\left(a_w^{(\cdot)}\right)$, given by,*

$$f_{UB}\left(a_w^{(\cdot)}\right)=\frac{KT}{2}-\frac{1}{K}\sum_{j=1}^{K}\left(\frac{1}{j}\sum_{i=0}^{j-1}\left(\delta_{(j-1-i)T}\right) ia_w^{(i)}T\right) \tag{7}$$

Proof: This follows from the fact that $\max\{a,b,c,d\}\geq average\,(a,b,c,d)$. Thus, using this fact in the equation (6)*, the upper bound* (7) *is proved in (8a-d).*

IV. Temporal Sensing Framework

This section develops the temporal sensing framework that jointly optimizes the inter-sample delay and energy efficiency of the SN. It adapts the sensing window length based on changed statistics of the underlying monitored process.

A. AoS-based Optimization Problem

Intuitively, activating the SN at all the sampling instances will give minimum average AoS (i.e., inter-sample time) at the cost of huge energy expenditure. While activation at a fewer sampling instances will save the SN's energy at the expense of increased average AoS. Thus, there exists a trade-off between the age and energy. Optimizing this crucial trade-off is necessary especially when there is redundancy in measurements of the temporal process and the SN has limited energy reserve. An optimization problem with multiple contradicting objectives, developed in (9), is run by the FC for selection of a few temporal sensing instances to monitor varying process in the upcoming w^{th} sensing window $\mathcal{A}_w$. It trades-off between the $\overline{\text{WAoS}_w}$ and the energy efficiency of the SN subject to the constraint on required sensing accuracy.

$$\textbf{(P1)}\quad \underset{a_w^{(k)},\ \forall k}{\text{minimize}}\quad \left[\overline{\text{WAoS}_w}, \text{E}_{cons}\right]$$

$$\textbf{(C11)}\qquad \text{s.t.}\quad a_w^{(k)}\in\{0,1\},\ k=0,\cdots,K-1, \tag{9}$$

$$\textbf{(C12)}\qquad \text{Tr}\left\{\left(\frac{1}{\sigma^2}\mathbf{B}^T\mathbf{A}_w^T\mathbf{A}_w\mathbf{B}+\mathbf{\Gamma}_w^{-1}\right)^{-1}\right\}\leq\gamma.$$

The function $\overline{\text{WAoS}_w}$ of **(P1)**, given in (6), represents delay in sensing or average age of consecutive samples which are sensed based on the underlying process variations. Minimizing $\overline{\text{WAoS}_w}$, minimizes the inter-sample time. The function E_{cons} of a SN represents the number of temporal sampling instances at which the SN is activated out of the total K instances of a sensing window, i.e., $\text{E}_{cons}=\frac{1}{K}\sum_{w=1}^{W}a_w^{(k)}=\frac{M_w}{K}$. As the sensing operation consumes sensing energy (E_s) of the SN, minimizing the E_{cons} function by selecting/activating a few sensing instances will minimize the sensing energy consumption of the SN, thereby, improving its energy efficiency. Non-convex constraint **(C11)** captures active(1)/ sleep(0) status of the SN at (kT) time instance. Constraint **(C12)** sets a maximum limit γ on the Bayesian Cramér-Rao bound (BCRB). Note that the mean squared error $\left(\frac{1}{K}\left\|\mathbf{z}_w-\widehat{\mathbf{z}}_w\right\|^2\right)$ is unknown during selection of temporal sensing instances due to the unknown temporal signal vector $\mathbf{z}_w$ whose some components will be sensed based on the solution of the MOP. Thus, the BCRB is used to characterize this error similar to the work [26]. The BCRB expression for the linear system model (2), derived in Appendix A, is given by $\text{Tr}\{\left(\frac{1}{\sigma^2}\mathbf{B}^T\mathbf{A}_w^T\mathbf{A}_w\mathbf{B}+\mathbf{\Gamma_w^{-1}}\right)^{-1}\}$. The entity $\mathbf{A}_w^T\mathbf{A}_w\in\mathbb{R}^{K\times K}$ is a binary diagonal matrix with K diagonal elements-$a_w^{(0)},\cdots,a_w^{(K-1)}$.

The MOP (9) is difficult to solve due to the non-convex constraint **(C11)** and structure of the $\overline{\text{WAoS}_w}$ function. This constraint is relaxed to a convex box constraint $a_w^{(\cdot)}\in[0,1]$. Now, the resulting optimization problem may not

Algorithm 1 Exhaustive search-based solution

1: Generate $\sum_{m=1}^{K} C\binom{K}{m}$ patterns of active temporal sensing instances.
2: **for** each pattern **do**
3: Assign $a_w^{(k)}, \forall k$, as $0/1$.
4: Compute BCRB (Sec. IV-A).
5: **if** BCRB $\leq \gamma$ **then**
6: Compute $f_{UB}\left(a_w^{(\cdot)}\right)$ (7) and E_{cons} (Sec. IV-A).
7: **else** $break$.
8: **end if**
9: **end for**
10: Pick sets of $a_w^{(\cdot)}$ corresponding to minimum E_{cons}.
11: From the above selected sets with minimum E_{cons}, select a set with minimum $f_{UB}\left(a_w^{(\cdot)}\right)$.
12: Set $M_w = \sum_{k=0}^{K-1} a_w^{(k)}$.
Output: Activity set $\mathcal{M}_w = \{a_w^{(k)},\ k = 0, \cdots, K-1\}$ with minimum E_{cons} and minimum f_{UB}, M_w.

have a tractable solution as it minimizes a concave problem (because of $-\max\{\cdots\}$ in the expression (6)). However, using the upper bound $f_{UB}\left(a_w^{(\cdot)}\right)$ (7) on $\overline{\mathrm{WAoS}_w}$ results in a solvable convex MOP given in (10).

$$\textbf{(P2)} \quad \underset{a_w^{(k)},\ \forall k}{\text{minimize}} \quad \left[f_{UB}\left(a_w^{(\cdot)}\right), \mathrm{E}_{cons}\right]$$

$$\textbf{(C21)} \quad \text{s.t. } a_w^{(k)} \in [0,1],\ k = 0, \cdots, K-1, \tag{10}$$

$$\textbf{(C22)} \quad \mathrm{Tr}\left\{\left(\frac{1}{\sigma^2}\mathbf{B}^T\mathbf{A}_w^T\mathbf{A}_w\mathbf{B} + \mathbf{\Gamma}_w^{-1}\right)^{-1}\right\} \leq \gamma.$$

Note that minimizing $f_{UB}\left(a_w^{(\cdot)}\right)$ of **(P2)** minimizes $\overline{\mathrm{WAoS}_w}$. The above relaxed MOP can be solved using the well-known scalarization approach [27]. This approach associates scalar weights with each objective function of the MOP and converts it into a single objective optimization problem. The scalar weights $\in [0,1]$ and their sum equals 1. However, deciding reasonable scalar weights for each objective function could be pursued as a separate research problem as done in [10] by devising a separate algorithm for this. Another approach to solve the relaxed MOP is to use $\epsilon-$constraint method [27] in which one of the objective function is set as a constraint. Since the focus of the current work is to study the trade-off between the AoS and energy efficiency, exhaustive search algorithm, given in Algorithm 1, is used to solve the proposed MOP (10) with binary constraint **(C11)**. The algorithm generates all possible activity sets and assigns corresponding $a_w^{(k)}, \forall k$. The sets that do not obey the BCRB constraint (11) are discarded. For the remaining sets, the values of f_{UB} and E_{cons} functions are computed. Further, the sets with minimum E_{cons} (i.e., energy consumption) are retained. From these sets, a set with minimum f_{UB} is chosen as the activity set $\mathcal{M}_w$. The values $\left(a_w^{(k)} = 0/1, \forall\ k\right)$ of this set respectively give sleep/active status of the SN during K sampling instants of the w^{th} sensing window. The number of active instants M_w is obtained by summing the entities $a_w^{(\cdot)}$ of the solution set $\mathcal{M}_w$.

Complexity of the Algorithm 1 increases with increase in the length of sensing window, i.e. K. Intuitively, monitoring a process with high temporal correlation requires a higher value of K. For the data-set considered in simulation, K as high as 12 is used for high temporal correlation case. Since this algorithm is run separately at each SN of a WSN, its complexity does not increase exponentially with the size of the network (i.e. the number of SNs in the network). An alternate solution is to use scalarization approach with equal weights for each objective function in (10). Since f_{UB} and E_{cons} are linear functions of $a_w^{(k)}$, the resulting objective function $(\lambda f_{UB} + (1-\lambda)\mathrm{E}_{cons})$ is convex in $a_w^{(k)}$. Also, the constraints **(C21)** is a convex set and **(C22)** is a convex function [10]. Thus, the scalarized problem can be solved using CVX [28].

The proposed framework considers sensing energy consumption, while communication energy consumption and privacy issues will be considered in future work. The efficacy of the proposed framework in energy saving can be best observed for SNs that consume higher sensing energy compared to the communication energy. Note that, using the proposed optimization problem, the energy is saved in both sensing and communication of the sparsely sampled process signal.

B. Sensing Window Adaptation for Non-Stationary Data

Most of the real-time data are non-stationary whose statistical properties (such as mean, correlation) change over time. Using a constant length of the sensing window may result in poor signal estimation performance over time. Portion of real-time data with high correlation can have a longer sensing window compared to the portion with low correlation. More redundancy in highly correlated data can better aid the estimation of signal for a longer duration using a few sensed measurements compared to the data with low correlation (or less redundancy). Thus, adapting the length of the sensing window as per the changed statistics of the data is mandatory to maintain same/required sensing quality.

An online mechanism is developed that runs at the FC to determine the time instance at which the sensing window length needs to be altered due to the change in statistics/distribution of the observed process. The mechanism computes mean (or average) of the previously estimated data in batches. Each batch contains data corresponding to 10 time instances is considered in simulation. A larger batch size will also do. If the mean changes significantly, i.e., more than $(\pm 0.5 \times \mathrm{mean}_{tr})$ in the next few consecutive windows (considered 5 here), then it indicates change in mean statistics of the process. Note that mean_{tr} is the mean of data/process signal computed during training time. This is followed by a retraining period to adapt the sensing window length K. The training/retraining assumes complete knowledge of the monitored process signal for a few consecutive time instances ($\geq 1.5 \times$ previous K; considered ~ 20 in the simulations). For the case when K needs to be increased, these many training instances help in making a better choice of K by analyzing the trade-off between energy utilization and sensing quality with prospective higher values of K. During the training/retraining period, the temporal sensing instances selection and sensing window's signal estimation are carried out keeping the BCRB constraint same throughout. Also, correlation

Algorithm 2 AoS-based temporal sensing framework

Initialization: DFT matrix $\mathbf{B}, \gamma, K, W, \text{mean}_{tr}, data = [\,], flag = false, meanchangeloc = [\,], temploc = [\,], count = 0$, $startloop = 1, endloop = W$.
while $startloop \leq endloop$ **do**
  for $w = 1 : W$ **do**
    Obtain $\mathcal{M}_w$ and M_w by solving the MOP using Alg. 1.
    Inform the SN about active sampling instances $a_w^{(\cdot)} = 1$.
    Collect the measurements $\mathbf{y}_w$ from the SN.
    Compute binary sensing matrix $\mathbf{A}_w$ (Sec. II).
    Obtain the estimate $\widehat{\mathbf{z}}_w$ using the SBL scheme [25].
    $data = [data;\ \widehat{\mathbf{z}}_w]$.
    Calculate mean in batch of 10 instances of recently estimated data/process signal (if available). Assign it to mean_b.
    if $\text{mean}_b \notin [0.5 \times \text{mean}_{tr}, 1.5 \times \text{mean}_{tr}]$ **then**
      if $flag == false$ **then**
        Set $flag = true, count = count + 1$.
        Set $temploc = wK$.
      else if $flag == true$ **then**
        Set $count = count + 1$.
      end if
    else Set $flag = false, count = 0, temploc = [\,]$.
    end if
    if $flag == true$ and $count == 5$ **then**
      Set $meanchangeloc = temploc$.
      $break$.
    end if
  end for
  if $(startloop < endloop)$ and $(flag == true)$ **then**
    $temp = (WK) - (((w+4)K) + (\#\text{retrain instances}))$.
    Retrain to adapt K.
    Set $flag = false, count = 0, temploc = [\,]$.
    $W = floor(temp/K)$.
    $startloop = 1, endloop = W$.
  end if
end while

among the data is computed. If the correlation decreased (increased) compared to that of the previous training data, then decrease (increase) K till the sensing error range coincides with that obtained for previous training/retraining. Since the actual signal $\mathbf{z}_w$ is also known during these periods, sensing error, i.e., normalized root mean-squared error (nRMSE) $= \sqrt{\frac{\frac{1}{K}\|\mathbf{z}_w - \widehat{\mathbf{z}}_w\|^2}{\|\mathbf{z}_w\|^2}}$, can be easily calculated.

Note that the mean change detection test is considered as it is simple computation-wise. Further, in future, if this computation is moved to the SN level in distributed setting, then it is preferable to opt a test with less computation owing to limited energy resource and computation capability of the SN. Other tests such as Welch's t-test [29] and chi-square test could also be used to detect change in the process statistics.

Flow of the proposed temporal sensing framework run by the FC is outlined in Algorithm 2. This algorithm first invokes Alg. 1 to obtain the activity set $\mathcal{M}_w$ and M_w. The FC then informs the SN about the active sampling instances $\left(a_w^{(\cdot)} = 1\right)$ of the w^{th} sensing window. Thereafter, upon collection of the measurements $\mathbf{y}_w$ corresponding to the $a_w^{(\cdot)} = 1$ from the SN, the signal estimate $\widehat{\mathbf{z}}_w$ is computed. This is followed by the detection of change in the mean statistic of the monitored process. If the change is detected, then the retraining phase starts and sensing window length is adapted. Otherwise, the process repeats for next sensing window using same value of K.

V. Simulation Results

This section illustrates the energy efficacy, adaptability, and practicability of the proposed AoS-based temporal sensing framework over the existing ones in [8], [10], and [14].

Real data-sets of two gaseous pollutants NO_2 and SO_2 collected from two sites (site-1 and site-2) [30] are used for performance comparison. The data is divided into sensing windows containing K sampling instances T time units apart. Although, the data of granularity 15 minutes is used, but for the sake of convenience T is set as 1 unit. The average variation of the process δ_{rT} is computed as average of $|\widehat{z}_{w-1}(p) - \widehat{z}_{w-1}(q)|, \forall p, q \in \{1, \cdots, K\}, (p-q)T = rT$. Total number of sensing windows $W = \left\lfloor \frac{\text{data-set length}}{K} \right\rfloor$. The noise variance is set as $\sigma^2 \approx 10^{-5}$ similar to the work [26]. The process signal estimate corresponding to a sensing window (obtained using the SBL scheme) are averaged over 50 Monte-Carlo iterations. The performance metrics used in the plots are- 1. average sensing energy utilization $\left(\overline{\text{SEU}}\right) = \frac{1}{W}\sum_{w=1}^{W}\frac{M_w}{K}$. It represents sensing energy consumed per sampling instance per sensing window. Lesser $\overline{\text{SEU}}$ indicates better energy efficiency of the SN. 2. $\overline{\text{AoS}} = \frac{1}{W}\sum_{w=1}^{W}\overline{\text{AoS}_w}$, and 3. average nRMSE $\left(\overline{\text{nRMSE}}\right) = \frac{1}{W}\sum_{w=1}^{W}\sqrt{\frac{\frac{1}{K}\|\mathbf{z}_w - \widehat{\mathbf{z}}_w\|^2}{\|\mathbf{z}_w\|^2}}$. Note that, the BCRB is used as a constraint on the sensing quality in the MOP (9) because the nRMSE cannot be computed at the FC due to unknown $\mathbf{z}_w$. The $\overline{\text{nRMSE}}$ plots are used just to highlight that acceptable nRMSE values (≤ 0.2) (suggested in [31]) are obtained using the controlling parameter BCRB. Further, the proposed framework is simulated in MATLAB.

A. Choice of Sensing Window Length K

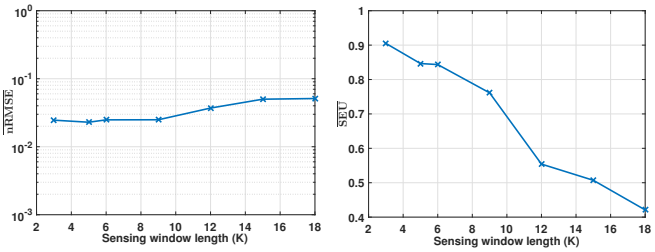

Fig. 4: Sensing window length estimation for NO_2 parameter of site-1.

In the proposed framework, the sensing decisions and signal estimation tasks are performed for each sensing window. Thus, finding appropriate length K of the window is of interest. In Figs. 4(a)-(b), $\overline{\text{nRMSE}}$ and $\overline{\text{SEU}}$ are shown for the NO_2 signal sensed at the site-1. The data corresponding to 500 sampling instances from April 12, 2020 to April 17, 2020 are used and the upper limit on the value of BCRB is set as $\gamma = 0.02$. It can be observed that, although the $\overline{\text{SEU}}$ decreases with increasing K, the $\overline{\text{nRMSE}}$ increases significantly beyond $K = 12$. Hence, the sensing window length is chosen as $K = 12$ for this data-set. Further, the sensing performance can be better observed

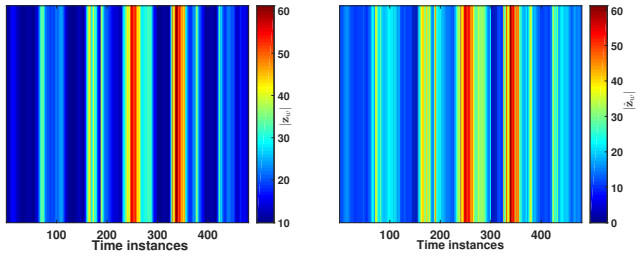

Fig. 5: Heat-maps of (a) actual and (b) estimated NO_2 signal at the site-1.

from the obtained similar variations of the actual and estimated process signals shown in heat-maps Figs. 5(a)-(b). Likewise, the window length K is set as 12 for the SO_2 data-set of the site-1 as shown in Figs. 6(a)-(b). Note that the decreasing trend in $\overline{\text{SEU}}$ is observed because with increasing K, the BCRB constraint (10) is satisfied with lesser M_w. The increasing trend on $\overline{\text{nRMSE}}$ is seen because correlation among samples of a window decreases as time lag (or K) increases which may increase error in sensing window's signal estimation.

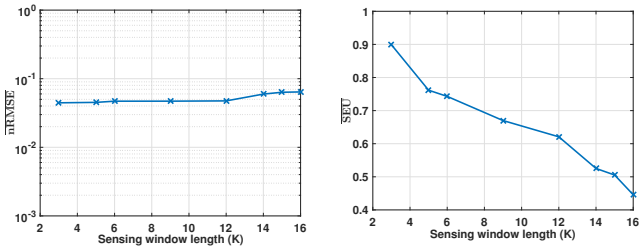

Fig. 6: Sensing window length estimation for SO_2 parameter of site-1.

It is important to mention that, using the training data, the upper limit of BCRB (γ) is set such that the obtained nRMSE is less than or equal to an acceptable threshold (e.g., nRMSE ≤ 0.2) even in the worst case. The nRMSE expected in the upcoming sensing windows' time instances (i.e., non-training period) will depend on this set BCRB constraint. At no point, the framework is trying to increase K until the nRMSE of value 0.2 is achieved while choosing the window length K. The controlling parameter BCRB plays a significant role in this scenario and it can be set as per different application requirements.

Note that the exhaustive search-based solution (Alg. 1) worked well for K upto 18. Beyond this, it was not needed to increase K because the BCRB constraint (11) violated for the considered data. [1]

Due to the observation of similar performance trend of sensing the NO_2 and SO_2 pollutants, the subsequent studies consider NO_2 pollution data only as it contains both high and low correlation data.

Remark 1. *With increase in the sensing window length K while maintaining the same upper limit on the BCRB, the $\overline{\text{SEU}}$ decreases at the expense of increasing $\overline{\text{nRMSE}}$.*

[1]If required to further increase K, then equal weights can be assigned to each objectives such that the sum of weights is 1. Thereafter, the scalarized MOP can be solved using CVX [28] solver to analyze the performance.

Remark 2. *The sensing window length K is chosen such that, with increasing K, the sensing energy consumption decreases without deteriorating the sensing quality.*

B. Performance Comparison With State-of-the-Art Schemes

TABLE I: Performance comparison of the proposed framework with the existing schemes using NO_2 data subset-16 of site-2

Strategy	$\overline{\text{nRMSE}}$	$\overline{\text{AoS}}$	$\overline{\text{SEU}}$
Proposed ($\gamma = 0.1$)	0.068345	0.656626	0.7449799
Proposed ($\gamma = 0.45$)	0.0845349	0.699799	0.625502
Proposed ($\gamma = 0.6$)	0.0947235	0.723895	0.582329
Proposed ($\gamma = 0.4$)	0.079142	0.648594	0.686746
RS-CS [8], [24]	0.097819	0.791532	0.75
Adaptive Sensing [10]	0.077334	0.645161	0.840725
AdaM [14]	0.088728	0.75	0.75

Subset-16 of NO_2 pollutant sensed at the site-2 is considered. The comparison of the proposed framework is done with random sampling [8] and CS-based recovery [24] (RS-CS), adaptive sensing [10], and AdaM scheme [14]. For the sake of fair comparison of the proposed sensing framework with the AdaM's scheme, the SBL-based estimation is used in the AdaM scheme. For the RS-CS, M_w is set as per relation $M_w \geq S\log\left(\frac{K}{S}\right)$ [32], where S represents the number of non-sparse components which herein comprises 95% of the total signal energy. The parameters of the RS-CS, adaptive sensing, and AdaM are respectively set as $M_w = 3$, $\{[\alpha, \beta], \epsilon_{th}, \delta_{th}\} = \{\left[1 \times 10^{-5}, 0.6\right], 50, 0.5842\}$, and $\{\alpha, \beta, \gamma, \lambda\} = \{0.55, 1, 0.5, 1\}$. Note that these parameters of the competitive schemes are set such that either the achieved $\overline{\text{SEU}}$ or $\overline{\text{nRMSE}}$ are similar to that of the proposed scheme. Further, the parameter K is set as 4 following the approach outlined in Section V-A. It can be observed from rows $1, 5$, and 7 of Table I that the proposed framework offers 30.1% and 22.9% gain in sensing quality $\left(\overline{\text{nRMSE}}\right)$ compared to the RS-CS and AdaM schemes respectively while keeping nearly same $\overline{\text{SEU}}$. On the contrary, comparing the proposed framework with the RS-CS, adaptive sensing, and AdaM schemes while maintaining nearly same sensing quality (i.e., $\overline{\text{nRMSE}}$ in the rows $\{3, 5\}$, $\{4, 6\}$, and $\{2, 7\}$), the gain in $\overline{\text{SEU}}$ are 22.4%, 18.3%, and 16.6% respectively. Further, in both the comparative cases (same $\overline{\text{SEU}}$ and same $\overline{\text{nRMSE}}$), it can be observed that the $\overline{\text{AoS}}$ of the proposed scheme is lower than that of the competitive schemes. Thus, the proposed scheme is preferred for time-critical applications over the other schemes.

Remark 3. *In contrast to the existing schemes, the proposed AoS-based framework provides energy-efficient temporal sensing with lesser inter-sample delay without degrading the sensing quality. It offers better sensing performance as well while consuming the same sensing energy.*

C. Effect of Sampling Interval ΔT on the Proposed Framework

This section considers site-1's NO_2 sensing data sampled at $T, 2T,$ and $3T$ sampling interval for a total duration of $480T$

TABLE II: Impact of sampling interval ΔT on sensing NO_2 data of site-1

ΔT	K	$\overline{\text{nRMSE}}$	$\overline{\text{SEU}}$
T	12	0.076168	0.42378
$2T$	6	0.090548	0.730769
$3T$	4	0.110148	0.730769

time units. For the period T, the sensing window length K is set as 12 such that the total number of sensing windows evaluates to $W = 40$. Likewise, for the sampling intervals $2T$ and $3T$, the parameters $\{K, W\}$ are set as $\{6, 40\}$ and $\{4, 40\}$ respectively. The proposed framework is run for the above three different cases keeping the same BCRB limit $\gamma = 0.7$. It can be observed from Table II that the data with prospective sampling instants T time units apart provides better sensing quality and energy efficiency of the SN due to lesser $\overline{\text{SEU}}$ compared to the other higher sampling intervals.

Remark 4. *Increase in sampling interval (or decrease in sensing frequency) results in reduction of correlation among consecutive data samples, which deteriorates the sensing quality of the estimated signal.*

D. Sensing Window Length Adaptation

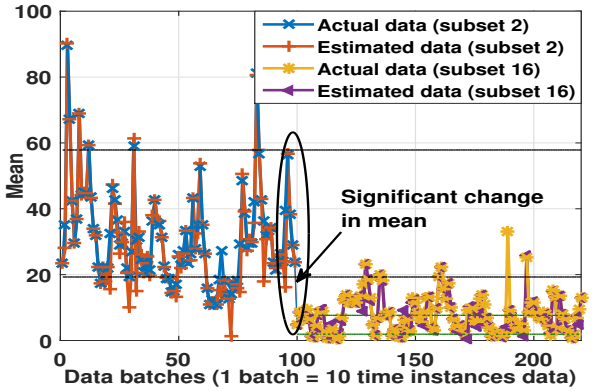

Fig. 7: Mean change detection of non-stationary real NO_2 sensing data of site-2.

For this part, NO_2 data of the site-2 from January 1, 2020 to July 2, 2020 ($\sim$ 16000 samples) is divided into subsets of length 1000 each. Non-stationarity in the data can be observed from Table III in which the subsets 1-15 have high correlation while the subset 16 has low correlation. Data corresponding to the subsets 2 and 16 are considered for simulation. The sensing window length K is initially set as 10 for the subset 2. Thereafter, on detecting change in mean statistics of the estimated data for ~ 5 consecutive windows (using Alg. 2) as observed around 100^{th} batch (i.e., last window of the subset 2) in Fig. 7, the framework is retrained to adjust K to 4 keeping same BCRB constraint (with $\gamma = 0.1$) such that the sensing quality remains nearly same.

From Table IV, it can be observed that the proposed framework uses a longer (shorter) sensing window and consumes lesser (higher) sensing energy for the portion of data with high (low) correlation, while providing better sensing quality

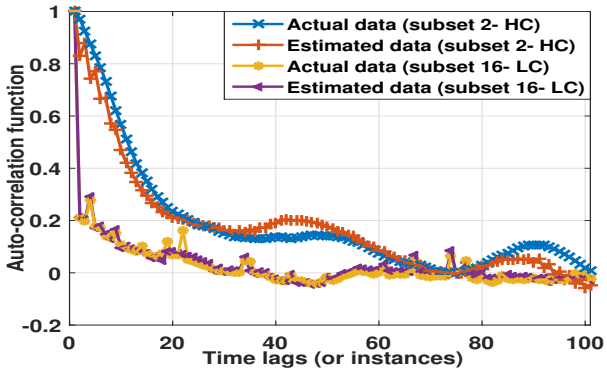

Fig. 8: Autocorrelation of respectively true and estimated data of the two subsets of site-2's NO_2 data.

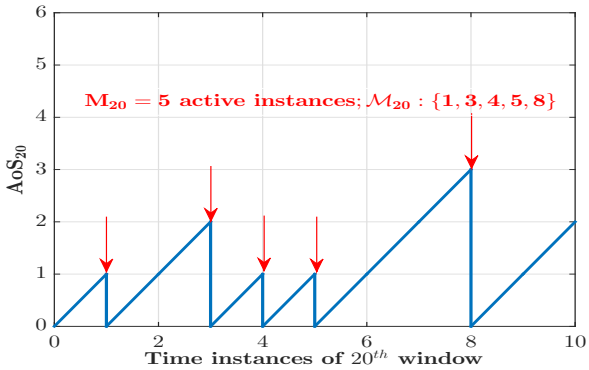

Fig. 9: AoS pattern of 20^{th} sensing window of NO_2 data subset-2.

($\overline{\text{nRMSE}}$). This can be further verified from the similar trend of autocorrelation of actual and estimated process shown in Fig. 8. The sample AoS pattern obtained for $w = 20$ corresponding to the subset 2 is shown in Fig. 9.

Remark 5. *The proposed framework allows selective temporal sensing of highly correlated data over a longer duration with more energy efficiency without deteriorating the sensing performance and increasing the inter-sample delay.*

E. Intelligent Sensing versus Intelligent Pruning

This section compares the resource utilization performance of the proposed intelligent sensing framework with that of the intelligent pruning framework in [33]. The resource utilization is calculated as sum of $\overline{\text{SEU}}$ and average bandwidth utilization $(\overline{\text{BWU}})$. The metric $\overline{\text{BWU}}$ is computed as $\frac{1}{W}\sum_{w=1}^{W}\frac{M_w^{Tx}}{K}$ with M_w^{Tx} being the number of units of sensed information sent to the FC. It is assumed that a SN senses one unit of information in one time instance. In the proposed intelligent sensing, M_w units of information sensed in a sensing window (length K) are transmitted to the FC. Thus, both $\overline{\text{SEU}}$ and $\overline{\text{BWU}}$ are equal to $\frac{1}{W}\sum_{w=1}^{W}\frac{M_w}{K}$. While in the intelligent pruning, K units of information sensed in a sensing window undergoes pruning and M_w^{prun} $(\leq K)$ units are transmitted to the FC, i.e., $M_w^{Tx} = M_w^{prun}$. Thus, $\overline{\text{SEU}} = \frac{1}{W}\sum_{w=1}^{W}\frac{K}{K} = 1$ and $\overline{\text{BWU}} = \frac{1}{W}\sum_{w=1}^{W}\frac{M_w^{prun}}{K}$ in this case. The parameter of intelligent pruning scheme is set such that the achieved

TABLE III: Correlation statistics of non-stationary real NO_2 sensing data of site-2

Time	**Data subsets**													
lags	**1**	**2**	**3**	**4**	**5**	**6**	$\cdots$	**10**	**11**	**12**	**13**	**14**	**15**	**16**
1	1	1	1	1	1	1	$\cdots$	1	1	1	1	1	1	1
2	0.9681	0.9712	0.9621	0.9773	0.9729	0.9569	$\cdots$	0.9774	0.9666	0.9581	0.9806	0.9729	0.9331	0.2127
3	0.9199	0.9233	0.9054	0.9429	0.9325	0.9008	$\cdots$	0.9426	0.9265	0.8983	0.9454	0.9198	0.8616	0.1993
4	0.8705	0.8763	0.8485	0.9051	0.8893	0.8417	$\cdots$	0.9087	0.8880	0.8334	0.9086	0.8605	0.8049	0.2774
5	0.8234	0.8302	0.7903	0.8636	0.8433	0.7836	$\cdots$	0.8770	0.8499	0.7713	0.8741	0.7986	0.7533	0.1751
6	0.7787	0.7840	0.7357	0.8207	0.7951	0.7273	$\cdots$	0.8445	0.8118	0.7147	0.8422	0.7367	0.7066	0.1733
7	0.7356	0.7312	0.6860	0.7769	0.7463	0.6719	$\cdots$	0.8088	0.7735	0.6640	0.8104	0.6822	0.6649	0.1396
8	0.6936	0.6739	0.6382	0.7297	0.6991	0.6223	$\cdots$	0.7732	0.7331	0.6170	0.7777	0.6348	0.6277	0.1344
9	0.6507	0.6174	0.5923	0.6829	0.6543	0.5740	$\cdots$	0.7370	0.6897	0.5732	0.7443	0.5903	0.5936	0.1529
10	0.6040	0.5628	0.5502	0.6364	0.6106	0.5326	$\cdots$	0.7009	0.6454	0.5321	0.7111	0.5483	0.5591	0.1085
11	0.5533	0.5096	0.5125	0.5884	0.5704	0.4893	$\cdots$	0.6650	0.6009	0.4910	0.6781	0.5086	0.5252	0.0964
12	0.5023	0.4599	0.4742	0.5379	0.5333	0.4451	$\cdots$	0.6315	0.5526	0.4557	0.6451	0.4711	0.4963	0.0881
13	0.4538	0.4155	0.4346	0.4891	0.4979	0.4026	$\cdots$	0.6007	0.5032	0.4194	0.6117	0.4345	0.4705	0.0814
14	0.4097	0.3775	0.3954	0.4410	0.4623	0.3652	$\cdots$	0.5732	0.4543	0.3847	0.5783	0.3987	0.4504	0.1027
15	0.3707	0.3457	0.3541	0.3953	0.4283	0.3295	$\cdots$	0.5479	0.4118	0.3521	0.5436	0.3641	0.4325	0.0737
16	0.3342	0.3165	0.3171	0.3521	0.3956	0.2928	$\cdots$	0.5209	0.3737	0.3235	0.5092	0.3297	0.4122	0.0632

TABLE IV: Sensing window adaptation for non-stationary NO_2 data of site-2

Data subset	**Sensing window length K**	$\overline{\textbf{nRMSE}}$	$\overline{\textbf{SEU}}$
2 (HC)	10	0.040844	0.683838
16 (LC)	4	0.068345	0.7449799

$\overline{\text{nRMSE}}$ is nearly same to that in the intelligent sensing case. $M_w^{prun} = 1.7s\log\left(\frac{K}{s}\right)$ with s being the sparsity in the signal [33]. For the intelligent sensing and pruning cases, the resource utilization respectively evaluates to 1.367676 $(= 0.683838 + 0.683838)$ and 1.505050 $(= 1 + 0.505050)$ with $\overline{\text{nRMSE}} \approx 0.04$. Also, $\overline{\text{AoS}}$ is 0.930303 in the former case and 0.5 (minimum value) in the latter case.

Remark 6. *Intelligent sensing saves both sensing and transmission energy; while intelligent pruning saves only transmission energy. Thus, intelligent sensing is preferred for SNs that consume more energy in sensing than in transmission, e.g. PM sensors, gas sensors, etc.*

*F. Tightness of the Convex Relaxation Used in the MOP **P2***

To investigate the tightness of convex relaxation used in the MOP **P2** (10) (where the constraint **(C11)** is relaxed to the constraint **(C21)**), performance of the MOP **P2** (10) with the exact constraint (**C11**) (solved using exhaustive search based algorithm) is compared with the performance using the relaxed constraint (**C21**) (solved using $\epsilon-$constraint method). In the MOP performance with **(C21)**, the objective function $\text{E}_{cons} = \frac{1}{K}\sum_k^K a_w^{(k)} = \frac{M_w}{K}$ is set as constraint using the value of M_w obtained from the MOP solution with (**C11**). This ensures a fair analysis of the tightness of the convex relaxation. Also note that, similar to [26], the result of the relaxed case $(a_w^{(k)}, \forall k, \forall w)$ is rounded to obtain a solution feasible to the exact case.

Three scenarios with different values of γ $(\gamma = 0.75, 0.2, 0.07)$ are considered for simulation and the tightness of the relaxed case is captured by considering the difference between the number of active instants of the actual case $(M_w|_{Exhaustive})$ and relaxed case $(M_w|_{Relaxed})$ across all sensing windows, i.e. $(M_w|_{Exhaustive} - M_w|_{Relaxed})$. Thereafter, the degree of tightness between the two cases' results across all the sensing windows is computed as $\frac{1}{W} \times |\{(M_w|_{Exhaustive} - M_w|_{Relaxed}) = 0, 1 \leq w \leq W\}|$. The degree of tightness obtained in the three scenarios $\gamma = 0.75, 0.2, 0.07$ is respectively $0.53, 0.69, 0.88$. It can be observed that, as the parameter γ decreases, the degree of tightness of the relaxed case increases.

Remark 7. *The convex relaxation becomes tighter with the decreasing value of maximum allowable BCRB.*

VI. Conclusion and Future Directions

This paper has proposed a novel energy-efficient temporal sensing framework for a SN. The framework jointly optimizes inter-sample delay and energy consumption of the SN to select a few temporal sensing instances of a sensing window for activation such that a required sensing quality is maintained. To capture the inter-sample delay, a novel AoS function has been developed. Further, a weighted AoS function and its upper bound have been formulated to incorporate process variations in the proposed selective sensing framework. Exploiting the temporal correlation inherent in the monitored process, the signal across entire sensing window has been estimated using the few sensed measurements. A sensing window length adaptation logic has been designed which detects the time at which the statistics of non-stationary process changes and adapts the sensing window length accordingly. Simulation analysis using real data-sets of two gaseous pollutants have demonstrated that, using the proposed AoS-based temporal sensing framework, the sensing quality and energy saving gain are consistently achieved with respect to the state-of-the-art competitive approaches.

While in this study two limited available air pollutant data from two sites have been considered, a more extensive study is needed on other pollutants and at very different weather conditions, for air as well as water pollution monitoring. Further, AoS-based temporal monolithic and multi-sensing aspects are required to be investigated at the network-level for real-time monitoring applications. Since the current work has considered linear increase in time between two sampling instants, there is a scope of exploring the applicability of non-linear inter-sample time and the changes required in the structure of the AoS-based temporal sensing framework. The possibility of modeling the AoS function to jointly incorporate the effect of sensing delay and transmission delay can also be analyzed for delay-sensitive IoT applications.

APPENDIX A
BAYESIAN CRAMÉR-RAO BOUND (BCRB)

For the system model (2), the BCRB is computed as $\text{BCRB} = \text{Tr}\left\{\mathbf{J}_B^{-1}\right\}$, where $\mathbf{J}_B \in \mathbb{R}^{K\times K}$ represents the Bayesian Fisher information matrix (FIM) [34] given by,

$$\mathbf{J}_B = \underbrace{-\text{E}_{(\mathbf{y}_w,\mathbf{x}_w)}\left\{\frac{\partial^2 \mathcal{L}(\mathbf{y}_w|\mathbf{x}_w;\mathbf{\Gamma}_w)}{\partial \mathbf{x}_w \left(\partial \mathbf{x}_w\right)^T}\right\}}_{\mathbf{J}_D} \underbrace{-\text{E}_{(\mathbf{x}_w)}\left\{\frac{\partial^2 \mathcal{L}(\mathbf{x}_w;\mathbf{\Gamma}_w)}{\partial \mathbf{x}_w \left(\partial \mathbf{x}_w\right)^T}\right\}}_{\mathbf{J}_P}, \quad (10)$$

where the terms $\mathcal{L}\left(\mathbf{y}_w|\mathbf{x}_w;\mathbf{\Gamma}_w\right), \mathcal{L}\left(\mathbf{x}_w;\mathbf{\Gamma}_w\right)$ and $\mathbf{J}_D, \mathbf{J}_P \in \mathbb{R}^{K\times K}$ in (11) respectively represent the log-likelihood functions of the vectors $\mathbf{y}_w$, $\mathbf{x}_w$ parameterized by $\mathbf{\Gamma}_w$ and FIMs with respect to $\mathbf{y}_w, \mathbf{x}_w$. The component $\mathbf{\Gamma}_w = \text{diag}\left(\boldsymbol{\alpha}_w\right) \in \mathbb{R}^{K\times K}$ is a diagonal matrix of the hyperparameters of prior distribution of the vector $\mathbf{x}_w$ considered in the SBL scheme. This distribution is given by $p\left(\mathbf{x}_w;\boldsymbol{\alpha}_w\right) = \prod_{k=1}^{K}\left(2\pi\alpha_w\left(k\right)\right)^{-1/2} e^{-\frac{\left(x_w(k)\right)^2}{2\alpha_w(k)}}$ [25], with the component $\alpha_w\left(k\right)$ of the hyperparameter vector $\boldsymbol{\alpha}_w = \left[\alpha_w\left(1\right),\cdots,\alpha_w\left(K\right)\right]^T \in \mathbb{R}^{K\times 1}$ corresponds to the variance of component $x_w\left(\cdot\right)$. Using this distribution, the log-likelihood function is evaluated as: $\mathcal{L}\left(\mathbf{x}_w;\mathbf{\Gamma}_w\right) = \left(\widetilde{c} - \frac{1}{2}\left(\mathbf{x}_w\right)^T\left(\mathbf{\Gamma}_w\right)^{-1}\mathbf{x}_w\right)$, with $\widetilde{c}$ being a parametric constant. Its second order derivative with respect to $\mathbf{x}_w$ gives $\frac{\partial^2 \mathcal{L}(\mathbf{x}_w)}{\partial \mathbf{x}_w \left(\partial \mathbf{x}_w\right)^T} = \left(\mathbf{\Gamma}_w\right)^{-1}$. Thus, the resulting FIM is given by $\mathbf{J}_P = \left(\mathbf{\Gamma}_w\right)^{-1}$. Likewise, the log-likelihood function $\mathcal{L}\left(\mathbf{y}_w;\mathbf{x}_w\right)$, after ignoring the constant terms is obtained as: $\mathcal{L}\left(\mathbf{y}_w|\mathbf{x}_w;\mathbf{\Gamma}_w\right) = \frac{1}{2\sigma^2}\left\|\mathbf{y}_w - \mathbf{A}_w\mathbf{B}_w\mathbf{x}_w\right\|^2$. Thus, $\mathbf{J}_D = \frac{1}{\sigma^2}\left(\mathbf{B}\right)^T\left(\mathbf{A}_w\right)^T\mathbf{A}_w\mathbf{B}$. The BCRB is expressed as:

$$\text{BCRB} = \text{Tr}\left\{\left(\frac{1}{\sigma^2}\mathbf{B}^T\mathbf{A}_w^T\mathbf{A}_w\mathbf{B} + \mathbf{\Gamma}_w^{-1}\right)^{-1}\right\}. \quad (11)$$

REFERENCES

[1] D. J. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "Codeblue: An ad hoc sensor network infrastructure for emergency medical care," in *Proc. Mobisys. Workshop Appl. Mobile Embedded Syst. (WAMES)*. Boston, MA, USA, 2004, pp. 12–14.

[2] Y.-C. Wang and G.-W. Chen, "Efficient data gathering and estimation for metropolitan air quality monitoring by using vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7234–7248, 2017.

[3] G. Zhao *et al.*, "Wireless sensor networks for industrial process monitoring and control: A survey." *Netw. Protoc. Algorithms*, vol. 3, no. 1, pp. 46–63, 2011.

[4] R. C. Carrano, D. Passos, L. C. Magalhaes, and C. V. Albuquerque, "Survey and taxonomy of duty cycling mechanisms in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 181–194, 2013.

[5] K. Stone and M. Colagrosso, "Efficient duty cycling through prediction and sampling in wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 7, no. 9, pp. 1087–1102, 2007.

[6] R. Masiero, G. Quer, D. Munaretto, M. Rossi, J. Widmer, and M. Zorzi, "Data acquisition through joint compressive sensing and principal component analysis," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*. Honolulu, HI, USA, 2009, pp. 1–6.

[7] G. Quer, R. Masiero, G. Pillonetto, M. Rossi, and M. Zorzi, "Sensing, compression, and recovery for WSNs: Sparse signal modeling and monitoring framework," *IEEE Trans. Wireless Commun.*, vol. 11, no. 10, pp. 3447–3461, 2012.

[8] M. Hooshmand, M. Rossi, D. Zordan, and M. Zorzi, "Covariogram-based compressive sensing for environmental wireless sensor networks," *IEEE Sensors J.*, vol. 16, no. 6, pp. 1716–1729, 2015.

[9] M. Leinonen, M. Codreanu, and M. Juntti, "Sequential compressed sensing with progressive signal reconstruction in wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1622–1635, 2014.

[10] V. Gupta and S. De, "SBL-based adaptive sensing framework for WSN-assisted IoT applications," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4598–4612, 2018.

[11] R. Prabha, M. V. Ramesh, V. P. Rangan, P. Ushakumari, and T. Hemalatha, "Energy efficient data acquisition techniques using context aware sensing for landslide monitoring systems," *IEEE Sensors J.*, vol. 17, no. 18, pp. 6006–6018, 2017.

[12] V. Gupta and S. De, "Collaborative multi-sensing in energy harvesting wireless sensor networks," *IEEE Trans. Signal Inform. Process. over Netw.*, vol. 6, pp. 426–441, 2020.

[13] M. Andreolini, M. Colajanni, M. Pietri, and S. Tosi, "Adaptive, scalable and reliable monitoring of big data on clouds," *J. Parallel Distrib. Comput.*, vol. 79, pp. 67–79, 2015.

[14] D. Trihinas, G. Pallis, and M. Dikaiakos, "Low-cost adaptive monitoring techniques for the internet of things," *IEEE Trans. Services Comput.*, 2018.

[15] Y. Inoue and T. Takine, "AoI perspective on the accuracy of monitoring systems for continuous-time markovian sources," in *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*. IEEE, 2019, pp. 183–188.

[16] R. Wang, Y. Gu, H. Chen, Y. Li, and B. Vucetic, "On the age of information of short-packet communications with packet management," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[17] Y. Sun, Y. Polyanskiy, and E. Uysal-Biyikoglu, "Remote estimation of the Wiener process over a channel with random delay," in *Proc. IEEE Intl. Symp. Inf. Theory (ISIT)*. IEEE, 2017, pp. 321–325.

[18] Y. Sun and B. Cyr, "Information aging through queues: A mutual information perspective," in *IEEE Intl. Wkshp. Signal Process. Adv. Wireless Commun. (SPAWC)*. IEEE, 2018, pp. 1–5.

[19] C. Kam, S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, "Towards an effective age of information: Remote estimation of a Markov source," in *Proc. IEEE Conf. Comput. Commun. Wkshps. (INFOCOM WKSHPS)*. IEEE, 2018, pp. 367–372.

[20] W. Chen and I. J. Wassell, "Optimized node selection for compressive sleeping wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 827–836, 2015.

[21] A. Ramakrishnan and S. Saha, "Ecg coding by wavelet-based linear prediction," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 12, pp. 1253–1261, 1997.

[22] Q. Abbas, S. Zeb, S. A. Hassan, R. Mumtaz, and S. A. R. Zaidi, "Joint optimization of Age of Information and Energy Efficiency in IoT Networks," in *Proc. IEEE Veh. Technol. Conf. (VTC2020-Spring)*. IEEE, 2020, pp. 1–5.

[23] N. Jain, A. Gupta, and V. A. Bohara, "PCI-MDR: Missing data recovery in wireless sensor networks using partial canonical identity matrix," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 673–676, 2018.

[24] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[25] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

[26] S. P. Chepuri and G. Leus, "Sparsity-promoting sensor selection for non-linear measurement models," *IEEE Trans. Signal Process.*, vol. 63, no. 3, pp. 684–698, 2014.

[27] Z. Fei, B. Li, S. Yang, C. Xing, H. Chen, and L. Hanzo, "A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 550–586, 2016.
[28] I. CVX Research, "CVX: Matlab software for disciplined convex programming, version 2.0," http://cvxr.com/cvx, Aug. 2012.
[29] S. Silvestri, R. Urgaonkar, M. Zafer, and B. J. Ko, "A framework for the inference of sensing measurements based on correlation," *ACM Trans. Sensor Netw.*, vol. 15, no. 1, p. 4, 2018.
[30] *Data-sets*. [Online]. Available: https://1drv.ms/u/s!AoEH3xyPnnfYfV3xmzgNaX3xgOk?e=P3LKFX
[31] J. Qin, Q. Zhao, H. Yin, Y. Jin, and C. Liu, "Numerical simulation and experiment on optical packet header recognition utilizing reservoir computing based on optoelectronic feedback," *IEEE Photon. J.*, vol. 9, no. 1, pp. 1–11, 2017.
[32] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.
[33] S. Tripathi and S. De, "An efficient data characterization and reduction scheme for smart metering infrastructure," *IEEE Trans. Ind. Inform.*, vol. 14, no. 10, pp. 4300–4308, 2018.
[34] H. L. Van Trees, *Detection, estimation, and modulation theory*. John Wiley & Sons, 2004.