Resource Optimization for Machine-Type Communications with Lossy Links Based on Compressive Sensing

Hung-Hsien Chen, Cheng-Yeh Chen, Ching-Wei Yang, and Hung-Yun Hsieh

***Abstract*—Compressive sensing (CS) has been applied as an effective technique to fully reconstruct collective data from a subset of sensors in a wireless sensor network (WSN). Such technique is able to reduce the required bandwidth resource for data collection and reconstruction over a massive number of sensors. Although many algorithms have been proposed to exploit the structure of spatial correlation among sensors and the channel condition for each sensor, reconstruction quality is still severely affected by lossy links. In this work, we propose to incorporate the retransmission mechanism in CS to effectively mitigate the impact of packet loss. Unlike previous work that does not allow retransmission opportunity or determines the number of retransmissions based purely on channel condition, this paper is the first work to jointly consider channel uncertainty and spatial correlation among sensors to optimize allocation of transmission and retransmission resources. To proceed, we first apply CS based on Bayesian estimation to derive the analytical lower bound for data reconstruction error and incorporate the effect of retransmission into the problem formulation. Based on the rank-one adjustment for matrix inversion, we efficiently compute the matrix inversion of the packet-delivery-ration (PDR) matrix required to minimize reconstruction error. We then propose a greedy yet effective algorithm to determine the resource allocation and a meta-heuristic algorithm to further optimize the solution. Our proposed algorithms are evaluated under simulated as well as real-world environments over various resource constraints and spatial correlations to validate the improvement of reconstruction quality brought by the retransmission mechanism. Our algorithms outperform existing algorithm in terms of reconstruction error and computation efficiency.**

***Index Terms*—Lossy wireless sensor network, compressive sensing, retransmission, simulated annealing.**

I. Introduction

Machine-to-machine (M2M) communications, also known as machine-type communications (MTC), that involves a large number of machines (sensors) for data collection without or with very little human intervention have emerged as a new communication paradigm for supporting Internet of Things (IoT) applications. Various applications have already flourished in several sectors including industrial automation [1], autonomous vehicles [2], and environmental monitoring [3]. Despite the promising applications and benefits, MTC face the problem that when a massive number of wireless sensors need to access the data aggregator through the base station (or access point) simultaneously, the radio access network could easily become overloaded, making it impractical to allocate sufficient resource to every sensor for data collection.

Since sensors are typically deployed to perform a particular task or to measure specific quantity collectively in some physical fields, data originated from various sensors is not totally independent but rather correlated in nature, especially for those with close proximity. For most of applications, the performance of MTC is determined by the aggregated quality of collective data, instead of the quality of data measured from individual sensors. The data aggregator only requires data from a subset of sensors in the network due to the redundancy incurred by correlation between each sensor. In [4]–[6], the authors provide a *data-centric* viewpoint focusing on the quality of overall data instead of optimizing *machine-centric* metrics like the quality of service (QoS) on individual links.

In the same vein, compressive sensing (CS) [7] can be considered as a technique suitable for data-centric applications that prioritizes data integrity over individual sensors. CS promises to deliver a full recovery of the collective data with high probability based on samples far fewer than those required by the Nyquist–Shannon sampling theorem as long as the data is sparse over some basis. By addressing the spatial correlation between data measured from neighboring sensors, CS has been utilized to formulate the sensing, compression, and recovery process in wireless sensor networks (WSNs). In [8], CS is proposed in combination with Principle Component Analysis (PCA) to establish the underlying sparse basis and reconstruct the collective data sampled from any subset of sensors. Multivariate Bayesian estimation is formulated with CS [9] to accommodate the scenario with nonuniform variance of Gaussian noise among different sensors. In [10], covariogram estimation technique is embedded into the construction of sparse basis for CS to further incorporate spatial and temporal correlation of signal. [11] applies matrix completion technique accompanied by clustering structure in WSNs to reconstruct compressed data.

One popular topic investigated in CS-based WSNs is to optimize the *node selection problem* for data transmission and reconstruction to reduce the required bandwidth consumption and prolong the network life time. The spatial correlation among sensor nodes should be utilized to intelligently determine a subset of nodes which are the most representative of all the nodes in the network and the most informative for data reconstruction. The authors in [8] first proposed the

H.-H. Chen, C.-Y. Chen, and C.-W. Yang are with the Graduate Institute of Communication Engineering, National Taiwan University, Taipei 106, Taiwan. E-mail: r05942101, r08942083, r09942160@ntu.edu.tw.

H.-Y. Hsieh is with the Department of Electrical Engineering and Graduate Institute of Communication Engineering, National Taiwan University, Taipei 106, Taiwan. E-mail: hungyun@ntu.edu.tw.

random node selection (RNS) for its low complexity and low communication overhead among sensor nodes. The authors in [9] proposed centralized and decentralized node selection algorithm to greedily minimize estimation error while the authors in [10] proposed the deterministic node selection (DNS) algorithm based on spatial correlation among sensors to successively decide important nodes that bring the largest improvement in terms of reconstruction quality. In [12], the authors designed a node selection algorithm for multi-hop WSNs based on [10], which not only took advantage of the correlation structure but also took the paths cost, measured by the number of hops, into consideration.

Although CS-based WSNs have been intensively discussed and developed in the literature, these endeavors focused on the mathematical structure and formulation of CS-based data reconstruction. *Practical communications issues such as the impact of underlying channel conditions are often neglected or over-simplified.* However, packet loss is inevitable especially when a large number of sensors are distributed in a vast physical field to collect information, which could substantially impact the quality of CS-based data reconstruction as mentioned in [13], [14]. When transmission loss occurs, the remaining data received at the aggregator may be insufficient to represent the collective data for all the nodes in a WSN. How to incorporate channel uncertainty into the node selection and resource allocation mechanism to mitigate the impact on data collection is crucial to pave the road to practical applications of CS in lossy WSNs. In [13], sparsest random scheduling (SRS) was proposed to mitigate the impact of unreliable links in tree-based WSNs by restricting one measurement for one sample value to avoid single point of failure for transmission of a group of nodes packaged in one multi-hop delivery. Similar to [13], sparsest random sampling strategy was proposed in [14] with block diagonal matrix to take inter-cluster and intra-cluster correlation into consideration. The authors in [15] selects the most informative yet reliable nodes by combining the packet loss matrix into the objective function to minimize expected reconstruction error.

In addition to taking channel statistics into consideration during node selection, another countermeasure against packet loss is the retransmission mechanism. If some informative nodes in a WSN happen to suffer from the most unreliable links, retransmitting data from these informative but lossy nodes may be the most effective way to gather information for data reconstruction. Although [13] has conducted experiments on CS-based WSNs with tree-based routing and revealed that retransmission could hardly improve reconstruction quality despite the increased resource consumption due to a single point of failure, their retransmission mechanism does not consider the spatial correlation and packet loss rate. For MTC derived from one-hop transmissions in WSNs, our previous work [16] mitigated the impact of lossy links by allocating the number of time slots based on expected transmission count (ETX) for each link. A recent work [17] proposed dynamic retransmission to guarantee the quality of data reconstruction by the minimum expected transmission success rate per link. *However, these endeavors determined the number of retransmission simply by a given metric without addressing the effect of spatial correlation among nodes.*

In this work, we focus on a typical one-hop WSN comprising a massive number of sensors distributed in a physical field to collect information of interest for applications such as climate monitoring or industrial monitoring, where radio resource in the network is relatively insufficient and channel is lossy. Unlike most previous work, we generalize the node selection problem into a resource allocation problem allowing multiple retransmission opportunities for each sensor. The data aggregator (co-located with the base station or access point) needs to decide the set of sensors with the most informative data as well as the number of time slots to allocate for each sensor to conquer packet loss. We first apply CS with Bayesian estimation [9], [15] to derive analytical lower bound for reconstruction error and incorporate the effect of retransmission into the problem formulation. We then utilize the rank-one adjustment for matrix inversion [18] to efficiently compute the inversion of the packet-delivery-ratio (PDR) matrix required to minimize reconstruction error. Such a technique allows us to develop the proposed greedy algorithm to solve the resource optimization problem with retransmissions. We further propose a meta-heuristic, stochastic algorithm to further optimize the solution. Again, by applying rank-one adjustment for matrix inversion, we could efficiently obtain the reconstruction error by reusing the computation in the previous iteration.

Although the assumption and problem formulation of this work is derived from [9] and [15], both of these research endeavors allocate the same amount of radio resource only for transmission. *That is, they focus on node selection instead of resource allocation.* Introducing the retransmission mechanism would substantially complicate the searching space and hence efficient algorithm is required to obtain the solution. Furthermore, unlike [16], [17], which determine the amount of retransmission slots for each sensors based merely on specific metrics like "expected transmission count" or "successful transmission rate," we allocate transmission opportunity (including retransmission) jointly based on the packet loss rate per link and the overall structure of spatial correlation in the WSN. The proposed algorithms are evaluated based on a real-world deployment of sensors against related work.

In conclusion, our contribution could be summarized into the following three aspects:

- We incorporate the packet-delivery-ratio (PDR) matrix with retransmission based on Bayesian CS to derive the lower bound of mean square error (MSE) between the reconstructed data. We formulate a resource optimization problem able to solve node selection and resource allocation among lossy WSNs.
- We apply the theorem of rank-one adjustment for matrix inversion to efficiently compute the inversion of error covariance matrix required for MSE computation and elegantly developed a greedy algorithm able to iteratively allocate radio resource to the most informative nodes for data reconstruction considering channel uncertainty and spatial correlation.
- A stochastic optimization algorithm based on simulated annealing (SA) is developed to further minimize the reconstruction error. We propose several neighboring gen-

eration strategies for SA by applying the efficient computation of matrix inversion to improve the convergence and the convergence rate.

II. System model

A. Network scenario

In this paper, we consider a typical data-gathering application to collect data from a physical field with N sensors and a base station (BS) that doubles up as a data aggregator in Fig. 1. Each sensor periodically measures the data of interest from the data field and waits for transmission opportunity to report the data back to the BS through one-hop uplink communication based on the wide-area cellular technology. It is assumed that the sensed data exhibits spatial correlation among various sensors. Hence, it is possible to reconstruct the data from all sensors using only the data sensed by a subset of sensors. Due to limited radio resource, the BS needs to determine a subset of sensors for uplink transmission and coordinates channel access among these active sensors via the downlink in control plane. Unlike most recent works that assume ideal data transmission, we consider lossy links in this work. To address loss in a given data-gathering period, the BS could pre-allocate more than one transmission opportunity (i.e., time slot) to some sensors to increase the delivery rate of informative data.

B. Data transmission model

Let $\mathcal{S} = \{1, 2, \cdots, N\}$ denote the set of all sensors randomly deployed in a service area. Each data-gathering period consists of T time slots. One time slot represents one transmission opportunity, which is pre-allocated to a subset of $\mathcal{S}$ before each data-gathering period starts. We capture the uncertainty of each link by the expected packet delivery ratio (PDR) as q_i from sensor i to the BS, which is the probability of successful one-hop transmission. The BS may allocate more than one time slot to a given sensor to secure the data. To avoid the need to feedback transmission status in each time slot within one period, sensors would use up all allocated time slots for transmissions, irrespective of the transmission status of preceding time slots in a period. Under such transmission scheme, the data is only lost when all of the transmission opportunities fail. The overall PDR Q_i within each data-gathering period for sensor i could be written as

$$Q_i = 1 - (1 - q_i)^{\theta_i}, \tag{1}$$

where θ_i denotes the number of time slots allocated to sensor i in the data-gathering period.

C. Data aggregation model

Let $\mathbf{x} = [x_1, \cdots, x_N]^T$ represent the vector of data gathered by N sensors within a time period. Since these measurements exhibit spatial correlation, there exists an $N \times N$ matrix $\mathbf{\Psi}$ (sparsifying matrix) and an $N \times 1$ vector $\mathbf{u}$ (sparse vector) with only K non-zero elements (K-sparse) such that

$$\mathbf{x} = \mathbf{\Psi u}. \tag{2}$$

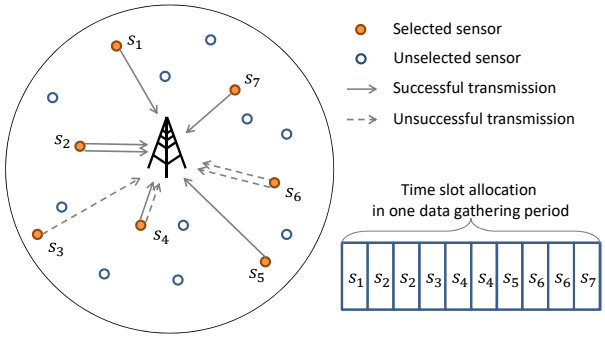

Fig. 1. Scenario of MTC involving a single BS and lossy, one-hop uplink communication for transmission and retransmission. A corresponding example of time slot allocation in one data gathering period is demonstrated.

The sparsifying matrix $\mathbf{\Psi}$ could be determined by principal component analysis on historical data [8], which is known by the BS in our assumption. Since $\mathbf{x}$ is K-sparse, the BS could select only a subset of M sensors with $M \geq K$ in each data gathering period to save radio resource while successfully reconstructing $\mathbf{x}$.

1) Selection matrix: Supposed M sensors are selected in the network. We define $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$ $(M \leq N)$ as the selection matrix to record the state of node selection. The structure of selection matrix satisfies the following constraints:

$$\begin{cases} \sum_{i=1}^{M} \Phi_{i,j} \leq 1, & j = 1, \cdots, N \\ \sum_{j=1}^{N} \Phi_{i,j} = 1, & i = 1, \cdots, M \\ \Phi_{i,j} \in \{0, 1\}, & i = 1, \cdots, M,\ j = 1, \cdots, N. \end{cases} \tag{3}$$

The entries of $\mathbf{\Phi}$ are all zeros except M unity entries, one for each row. The column index of a unity entry corresponds to the sensor's state, i.e., selected or not selected. If sensor i is selected, there is a unity element in the i-th column of $\mathbf{\Phi}$. Otherwise, all the elements of the i-th column are zeros.

2) Packet-loss matrix: Assuming M sensors are selected to transmit data to the BS, we define $\mathbf{H} \in \mathbb{R}^{L \times M}$ $(L \leq M)$ as the packet-loss matrix to capture the stochastic packet loss in each data-gathering period, where the data from L out of the M selected sensors are successfully received by the data aggregator. The structure of $\mathbf{H}$ is similar to $\mathbf{\Phi}$. The entries of $\mathbf{H}$ are all zeros except L unity entries, one for each row. A unity element in a given column of $\mathbf{H}$ indicates the packet of the corresponding sensor selected in $\mathbf{\Phi}$ is successfully received; otherwise, it is lost in transmission. If a sensor is allocated more than one time slot for transmission, any successful transmission in a time slot results in a unity element in the corresponding row and column.

Given the selection matrix $\mathbf{\Phi}$ and the packet-loss matrix $\mathbf{H}$, the measurement vector $\mathbf{y}$ received at the BS in a lossy network can be expressed as

$$\mathbf{y} = \mathbf{H\Phi x} + \mathbf{z}, \tag{4}$$

where $\mathbf{z} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2 \mathbf{I}\right)$ is independently identically distributed (i.i.d) additive white Gaussian noise with mean of 0 and variance of σ^2. Combining (2) and (4), one could obtain

$$\mathbf{y} = \mathbf{H\Phi\Psi u} + \mathbf{z} = \mathbf{Au} + \mathbf{z}, \tag{5}$$

where $\mathbf{A} = \mathbf{H}\mathbf{\Phi}\mathbf{\Psi}$ denotes the equivalent sensing matrix.

III. Problem formulation

In this section, we formulate the optimization problem into the minimization of mean square error (MSE) between the reconstructed data and the original data from the compressive measurement with radio resource constraint and retransmisstion mechanism. We analyze the difficulty to solve such an optimization problem due to stochastic characteristic of channel condition and reformulate the objective function by Jensen's inequality to capture the expected behavior of lossy links.

A. Minimization of MSE for CS scheme

Using the sparse property of $\mathbf{x}$, the BS can reconstruct the desired signal $\mathbf{x}$ from observation $\mathbf{y}$. Since $M < N$, we have an underdetermined system of linear equations, which in general has infinitely many solutions. However, adding the constraint that the initial signal is sparse enables one to solve such an undetermined system. The approach taken for CS decoding process is to find the sparsest solution by solving the following optimization problem:

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}} \|\mathbf{u}\|_0 \ \text{ subject to } \mathbf{y} = \mathbf{A}\mathbf{u} + \mathbf{z}. \tag{6}$$

One could first reconstruct an estimated sparse vector $\hat{\mathbf{u}}$ by (6) to obtain the estimated signal $\hat{\mathbf{x}}$ through (2),

$$\hat{\mathbf{x}} = \mathbf{\Psi}\hat{\mathbf{u}}. \tag{7}$$

Our goal is to find the allocation set $\mathbf{\Theta} = \{\theta_1, \cdots, \theta_N\}$ where θ_i denotes the number of transmission allocated to sensor i subject to the limited resource constraints to minimize MSE between $\hat{\mathbf{x}}$ and $\mathbf{x}$. Let T denote the total number of time slots in one data-gathering period. The minimization of MSE is given as follows:

$$\begin{aligned} \min_{\mathbf{\Theta}} \quad & \mathbb{E}_{\mathbf{z},\mathbf{H}}\left(\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2\right) \\ \text{subject to} \quad & \sum_{i\in\mathcal{S}} \theta_i \le T, \\ & \theta_i \in \{0, 1, \cdots, \theta_{\max}\},\ \forall\, i \in \mathcal{S}, \end{aligned} \tag{8}$$

where $\theta_{\max}$ is the maximum number allowed for a single sensor. Notice that $\theta_i = 0$ indicates the i-th sensor is not selected while $\theta_i > 0$ represents the i-th sensor is selected for data gathering and it is allocated θ_i slots for transmission.

To analyze the interplay between reconstruction performance and different settings of selected sensor nodes, we utilize the Bayesian estimation to convert the MSE into analytical form. Following the derivation in [9], we could rewrite the objective function of (8) into

$$\begin{aligned} & \mathbb{E}_{\mathbf{z},\mathbf{H}}\left(\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2\right) \\ =& \mathbb{E}_{\mathbf{H}}\left[\mathrm{tr}\left((\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{A}^T\mathbf{A})^{-1}\right)\right] \\ =& \mathbb{E}_{\mathbf{H}}\left[\mathrm{tr}\left((\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{\Phi}^T\mathbf{H}^T\mathbf{H}\mathbf{\Phi}\mathbf{\Psi})^{-1}\right)\right] \end{aligned} \tag{9}$$

by the well-known fact that the mean squared error is the trace of the error covariance matrix, $\left(\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{A}^T\mathbf{A}\right)^{-1}$, obtained in [9]. The unknown hyperparameters $\mathbf{\Lambda}$ and σ^2 can be estimated through an expectation-maximization algorithm [19]. In this work, we assume that perfect information about the hyperparameters and the sparsifying matrix $\mathbf{\Psi}$ are provided to the BS in advance.

B. Simplification by Jensen's inequality

One could observe that reconstruction performance in (9) is affected by the packet-loss matrix $\mathbf{H}$, whose statistics is determined by the allocation set $\mathbf{\Theta}$ since the number of transmission affects the overall PDR and packet loss. However, it is difficult to obtain a closed-form for the expectation in (9) due to the matrix inversion. Fortunately, such difficulty also appeared in [15]. Instead of directly estimating (9), [15] obtained a lower bound by Jensen's inequality. Therefore, we derive the closed-form of the lower bound by Jensen's inequality:

$$\begin{aligned} & \mathbb{E}_{\mathbf{H}}\left(\mathrm{tr}\left(\left(\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{\Phi}^T\mathbf{H}^T\mathbf{H}\mathbf{\Phi}\mathbf{\Psi}\right)^{-1}\right)\right) \\ \ge\ & \mathrm{tr}\left(\left(\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{\Phi}^T\mathbb{E}_{\mathbf{H}}\left(\mathbf{H}^T\mathbf{H}\right)\mathbf{\Phi}\mathbf{\Psi}\right)^{-1}\right) \\ =\ & \mathrm{tr}\left(\left(\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)^{-1}\right), \end{aligned} \tag{10}$$

where $\mathbf{Q} = \mathbf{\Phi}^T\mathbb{E}_{\mathbf{H}}\left(\mathbf{H}^T\mathbf{H}\right)\mathbf{\Phi}$ is defined as the expected packet-delivery-ratio (PDR) matrix, which is a diagonal matrix with value:

$$\mathbf{Q}_{i,i} = 1 - (1 - q_i)^{\theta_i}, \quad \forall\, i \in \mathcal{S}, \tag{11}$$

as introduced in (1). After the simplification by Jensen's inequality, the objective function could be explicitly written as a function of $\mathbf{\Theta}$ by the utilization of $\mathbf{Q}$:

$$\begin{aligned} \min_{\mathbf{\Theta}} \quad & \mathrm{tr}\left(\left(\mathbf{\Lambda} + \frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)^{-1}\right) \\ \text{subject to} \quad & \sum_{i\in\mathcal{S}} \theta_i \le T, \\ & \theta_i \in \{0, 1, \cdots, \theta_{\max}\},\ \forall\, i \in \mathcal{S}. \end{aligned} \tag{12}$$

C. Complexity analysis

1) Complexity for brute-force search: We first analyze the complexity of the formulated problem based on brute-force search. The search complexity of this problem is similar to the problem of calculating the number of ways to write T as a sum of N non-negative integers, which is

$$\theta_1 + \theta_2 + \cdots + \theta_N = T \tag{13}$$

The number of ways could be solved by permuting $N-1$ addition signs in a row of T ones, which is $\binom{T+N-1}{N-1}$. In the resource allocation problem, there is an upper bound $\theta_{\max}$ for resource allocation of each sensor. To further calculate the search complexity, we must subtract the cases in which at least

a sensor is allocated more than $\theta_{\max}$ time slots. For the case that k sensors are allocated more than $\theta_{\max}$ slots, the number of combination is

$$\binom{N}{k}\binom{T+N-1-k\left(\theta_{\max}+1\right)}{N-1}. \tag{14}$$

We can observe that at most $k_{\max}$ machines could be allocated more than $\theta_{\max}$ time slots, where

$$k_{\max}=\left\lfloor\frac{T}{\theta_{\max}+1}\right\rfloor. \tag{15}$$

By the inclusion-exclusion principle, the number of ways to allocate T slots among N sensors with at most $\theta_{\max}$ slots for each sensor is

$$\sum_{k=0}^{k_{\max}}(-1)^k\binom{N}{k}\binom{T+N-1-k\left(\theta_{\max}+1\right)}{N-1}, \tag{16}$$

whose complexity is combinatorial.

2) Comparison to the knapsack problem: As a matter of fact, the resource allocation problem in (12) is similar to a classical NP-hard problem, the bounded knapsack problem [20], [21]. Given N items indexed from 1 to N, a classical bounded knapsack problem could be formulated as follows:

$$\begin{aligned}&\max_{x_i}\quad\sum_{i=1}^{N}f_i(x_i),\\ &\text{subject to}\quad\sum_{i=1}^{N}g_i(x_i)\leq W,\ x_i\in\{0,1,2,...,x_{\max}\},\end{aligned} \tag{17}$$

where $f_i(x_i)$ and $g_i(x_i)$ represent the value and weight function of item i while x_i represents the number of copies of item i to be chosen to maximize the total value. Such a knapsack problem is similar to the resource allocation problem in this paper with the difference that the objective function in the bounded knapsack problem is separable (the value for each item i could be separably represented by f_i). However, the objective function in (12) is non-separable due to the matrix inversion in the error covariance matrix derived from (9). As a result, although many algorithms have been proposed to solve the linear [20] or non-linear [21] bounded knapsack problems, they are not applicable to our resource allocation problem. To solve the non-separable and non-linear knapsack problem formulated in this paper, we propose customized algorithms that are effective and efficient as follows.

IV. Proposed greedy algorithm

To solve such an NP-hard combinatorial optimization problem, we first propose a greedy algorithm based on the matrix inversion theorem for rank-one adjustment to efficiently compute matrix inversion and find local optimal selection to minimize the reconstruction error.

TABLE I
Notation

Notation	Definition
$\mathcal{S}$	Set of all sensors.
N	Total number of sensors in the network.
M	Number of selected sensors.
T	Total number of time slots for allocation.
q_i	Packet delivery ratio for sensor i to the BS.
$\mathbf{x}$	Column vector of sensed data from all sensors, $\mathbf{x}\in\mathbb{R}^{N\times 1}$.
$\mathbf{y}$	Data vector received at BS with $\mathbf{y}\in\mathbb{R}^{L\times 1}$.
$\mathbf{\Psi}$	Sparsifying matrix $\mathbf{\Psi}\in\mathbb{R}^{N\times N}$ known by the BS in advance.
$\mathbf{\Phi}$	Selection matrix $\mathbf{\Phi}\in\mathbb{R}^{M\times N}$ denoting the state of node selection.
$\mathbf{H}$	Packet-loss matrix $\mathbf{H}\in\mathbb{R}^{L\times M}$ corresponding to transmission state of selected sensors.
$\mathbf{\Theta}$	Set for time slot allocation. Each element indicates the number of assigned time slots for each sensor.
$\mathbf{z}$	Noise vector $\mathbf{z}\in\mathbb{R}^{L\times 1}$, i.i.d additive white Gaussian noise with mean $\mathbf{0}$ and variance $\sigma^2\mathbf{I}$.
$\mathbf{A}$	Equivalent sensing matrix $\mathbf{A}\in\mathbb{R}^{L\times N}$, where $\mathbf{A}=\mathbf{H}\mathbf{\Phi}\mathbf{\Psi}$.
$\mathbf{\Lambda}$	Sparseness prior of $\mathbf{u}$ in Bayesian estimation.

A. Conversion to rank-one matrix

First of all, we rewrite the objective function (12) into

$$\begin{aligned}&\operatorname{tr}\left[\left(\mathbf{\Lambda}+\frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)^{-1}\right]\\ =&\operatorname{tr}\left[\mathbf{\Psi}\left(\mathbf{\Lambda}+\frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)^{-1}\mathbf{\Psi}^T\right]\\ =&\operatorname{tr}\left[\left(\mathbf{\Psi}\left(\mathbf{\Lambda}+\frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)\mathbf{\Psi}^T\right)^{-1}\right]\\ =&\operatorname{tr}\left[\left(\mathbf{\Psi}\mathbf{\Lambda}\mathbf{\Psi}^T+\frac{1}{\sigma^2}\mathbf{Q}\right)^{-1}\right]\end{aligned} \tag{18}$$

by the property of similarity invariance of trace function and $\mathbf{\Psi}^T\mathbf{\Psi}=\mathbf{I}$. One could decompose $\mathbf{Q}$ as $\mathbf{Q}_1+\mathbf{Q}_2+\cdots+\mathbf{Q}_N$, where $\mathbf{Q}_i$ is an $N\times N$ zero matrix except for the entry at the i-th row and the i-th column equal to $\mathbf{Q}_{i,i}$. As defined in (11), $\mathbf{Q}_{i,i}=1-(1-q_i)^{\theta_i}$ is the overall PDR after θ_i times of transmission from sensor i to the BS. Each $\mathbf{Q}_i$ could be further decomposed as $\mathbf{Q}_i^{(1)}+\cdots+\mathbf{Q}_i^{(\theta_i)}$, where $\mathbf{Q}_i^{(j)}$ is an $N\times N$ zero matrix except for the entry at the i-th row and the i-th column $\left[\mathbf{Q}_i^{(j)}\right]_{i,i}$ defined as

$$\begin{aligned}\left[\mathbf{Q}_i^{(j)}\right]_{i,i}&=\left[1-(1-q_i)^j\right]-\left[1-(1-q_i)^{j-1}\right]\\ &=q_i(1-q_i)^{j-1},\end{aligned} \tag{19}$$

$\forall\ j\in\mathbb{N},\ j\leq\theta_i,\ i\in\mathcal{S}$. $\mathbf{Q}_i^{(j)}$ is treated as the improvement on PDR that the j-th transmission could contribute on the lossy link from sensor i to the BS.

According to the above definition, we could further decompose the term $\left(\mathbf{\Psi\Lambda\Psi}^T + \frac{1}{\sigma^2}\mathbf{Q}\right)$ in (18) into

$$\begin{aligned}
&\mathbf{\Psi\Lambda\Psi}^T + \frac{1}{\sigma^2}\mathbf{Q} \\
=&\mathbf{\Psi\Lambda\Psi}^T + \frac{1}{\sigma^2}\left(\mathbf{Q}_1 + \cdots + \mathbf{Q}_N\right) \\
=&\mathbf{\Psi\Lambda\Psi}^T + \frac{1}{\sigma^2}\left(\mathbf{Q}_1^{(1)} + \cdots + \mathbf{Q}_1^{(\theta_1)} + \mathbf{Q}_2^{(1)} + \cdots + \mathbf{Q}_2^{(\theta_2)}\right. \\
&\left. + \cdots + \mathbf{Q}_N^{(1)} + \cdots + \mathbf{Q}_N^{(\theta_N)}\right).
\end{aligned} \tag{20}$$

One could observe that $\mathbf{Q}_i^{(j)}$ are all rank-one matrices, and hence, we could apply the inversion theorem for rank-one adjustment [18] to compute our objective function.

Theorem 1. *Let* $\mathbf{G}$ *and* $\mathbf{G}+\mathbf{H}$ *be non-singular matrices and let* $\mathbf{H}$ *have positive rank* r*. Let* $\mathbf{H} = \mathbf{E}_1 + \mathbf{E}_2 + \cdots + \mathbf{E}_r$ *where each* $\mathbf{E}_k$ *has rank one and* $\mathbf{C}_{k+1} = \mathbf{G} + \mathbf{E}_1 + \cdots + \mathbf{E}_k$ *is non-singular for* $k = 1, \cdots, r$*. Then if* $\mathbf{C}_1 = \mathbf{G}$*, we have*

$$\mathbf{C}_{k+1}^{-1} = \mathbf{C}_k^{-1} - v_k \mathbf{C}_k^{-1}\mathbf{E}_k\mathbf{C}_k^{-1}, \quad k = 1, \cdots, r \tag{21}$$

where

$$v_k = \frac{1}{1 + \operatorname{tr}\left(\mathbf{C}_k^{-1}\mathbf{E}_k\right)}. \tag{22}$$

Proof. The complete proof process is detailed in [18]. □

From Theorem 1, we could rewrite the objective function into T times of rank-one adjustment, and each time we consider only the improvement induced by one time slot. Initially, no time slots have be allocated, we have $\mathbf{C}_1 = \mathbf{\Psi\Lambda\Psi^T}$ and $\mathbf{C}_1^{-1} = \mathbf{\Psi\Lambda^{-1}\Psi^T}$. If we allocate one time slot to sensor $i_1 \in \mathcal{S}$, we have

$$\mathbf{C}_2 = \mathbf{C}_1 + \frac{1}{\sigma^2}\mathbf{Q}_{i_1}^{(1)} \tag{23}$$

and

$$\mathbf{C}_2^{-1} = \mathbf{C}_1^{-1} - v_1\mathbf{C}_1^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_1}^{(1)}\mathbf{C}_1^{-1}, \tag{24}$$

where

$$v_1 = \frac{1}{1 + \operatorname{tr}\left(\mathbf{C}_1^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_1}^{(1)}\right)}. \tag{25}$$

Similarly, if we allocate the k-th time slot out of T time slots to sensor i_k, which is sensor i_k's j-th allocation, we have

$$\mathbf{C}_{k+1} = \mathbf{C}_k + \frac{1}{\sigma^2}\mathbf{Q}_{i_k}^{(j)} \tag{26}$$

and

$$\mathbf{C}_{k+1}^{-1} = \mathbf{C}_k^{-1} - v_k\mathbf{C}_k^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_k}^{(j)}\mathbf{C}_k^{-1}, \tag{27}$$

where

$$v_k = \frac{1}{1 + \operatorname{tr}\left(\mathbf{C}_k^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_k}^{(j)}\right)}. \tag{28}$$

Note that retransmission is allowed and hence, a sensor could be selected multiple times, resulting in $j > 1$.

We simplify the notation using the property of trace function. Let c_{i_k,i_k} denote $\left[\mathbf{C}_k^{-1}\right]_{i_k,i_k}$ and $q_{i_k}^{(j)}$ denote $\left[\mathbf{Q}_{i_k}^{(j)}\right]_{i_k,i_k}$. The coefficient v_k could be expressed into

$$\begin{aligned}
v_k &= \frac{1}{1 + \operatorname{tr}\left(\mathbf{C}_k^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_k}^{(j)}\right)} \\
&= \frac{1}{1 + \frac{1}{\sigma^2}c_{i_k,i_k}q_{i_k}^{(j)}}.
\end{aligned} \tag{29}$$

To further compute the objective function in (18), we use $\mathbf{c}_{i_k}$ and $\bar{\mathbf{c}}_{i_k}$ to denote the i_k-th column vector of matrix $\mathbf{C}_k^{-1}$ and the i_k-th column vector of matrix $\left(\mathbf{C}_k^{-1}\right)^T$, respectively. We have

$$\begin{aligned}
\operatorname{tr}\left(\mathbf{C}_{k+1}^{-1}\right) =& \operatorname{tr}\left[\mathbf{C}_k^{-1} - v_k\mathbf{C}_k^{-1}\frac{1}{\sigma^2}\mathbf{Q}_{i_k}^{(j)}\mathbf{C}_k^{-1}\right] \\
=& \operatorname{tr}\left(\mathbf{C}_k^{-1}\right) - \frac{\frac{1}{\sigma^2}q_{i_k}^{(j)}}{1 + \frac{1}{\sigma^2}c_{i_k,i_k}q_{i_k}^{(j)}}\operatorname{tr}\left(\mathbf{c}_{i_k}\bar{\mathbf{c}}_{i_k}^T\right) \\
=& \operatorname{tr}\left(\mathbf{C}_k^{-1}\right) - \frac{q_{i_k}^{(j)}}{\sigma^2 + c_{i_k,i_k}q_{i_k}^{(j)}}\bar{\mathbf{c}}_{i_k}^T\mathbf{c}_{i_k}.
\end{aligned} \tag{30}$$

B. Greedy selection procedure

Our goal is to sequentially repeat the selection for T times to fill up the time slot allocation. We present a greedy algorithm by selecting a sensor i_k^* that could result in the smallest $\operatorname{tr}\left(\mathbf{C}_{k+1}^{-1}\right)$ from $k = 1$ to $k = T$:

$$\begin{aligned}
i_k^* =& \arg\min_{i_k \in \mathcal{S}} \operatorname{tr}\left(\mathbf{C}_{k+1}^{-1}\right) \\
=& \arg\min_{i_k \in \mathcal{S}} \left(\operatorname{tr}\left(\mathbf{C}_k^{-1}\right) - \frac{q_{i_k}^{(j)}}{\sigma^2 + c_{i_k,i_k}q_{i_k}^{(j)}}\bar{\mathbf{c}}_{i_k}^T\mathbf{c}_{i_k}\right) \\
=& \arg\max_{i_k \in \mathcal{S}} \left(\frac{q_{i_k}^{(j)}}{\sigma^2 + c_{i_k,i_k}q_{i_k}^{(j)}}\bar{\mathbf{c}}_{i_k}^T\mathbf{c}_{i_k}\right),
\end{aligned} \tag{31}$$

where the final equality follows as the value of $\operatorname{tr}\left(\mathbf{C}_k^{-1}\right)$ does not depend on i_k. Eq. (31) could be treated as finding the most informative sensor i_k^* considering channel uncertainty given that $k-1$ time slots have been allocated for data reconstruction.

The detailed procedure is summarized in Algorithm 1. The algorithm first initializes $\mathbf{C}_1^{-1}$ and $\mathbf{\Theta}$ in line 1 and starts the iteration to greedily allocate T time slots from line 2 to line 8. In the k-th iteration, the most informative sensor i_k^* is selected for data reconstruction and corresponding $\mathbf{C}_k^{-1}$ is updated for the next iteration.

C. Complexity analysis on the greedy algorithm

In Algorithm 1, the complexity to compute the initial $\mathbf{C}_1^{-1}$ is $\mathcal{O}(N)$ since $\mathbf{\Lambda}$ is a diagonal matrix. For each iteration, selecting i_k^* in (31) involves the computation of the trace of $\mathbf{C}_k^{-1}$ for all the N nodes in $\mathcal{S}$ to find the minimum value and the corresponding i_k^*. By the simplification of (30), the complexity to obtain $\operatorname{tr}\left(\mathbf{C}_k^{-1}\right)$ is $\mathcal{O}(N)$. As a result, the overall complexity to find i_k^* is $\mathcal{O}\left(N^2\right)$. The greedy algorithm needs to run T times to find the solution, indicating the overall complexity of the greedy algorithm is $\mathcal{O}\left(TN^2\right)$.

Algorithm 1 Proposed greedy algorithm

1: $k=1,\ \mathbf{C}_k^{-1}=\mathbf{\Psi}\mathbf{\Lambda}\mathbf{\Psi}^T,\ \mathbf{\Theta}=\{\theta_1,\cdots,\theta_N\}\ s.t.\ \theta_i=0,\ \forall i\in\mathcal{S},$ and T is the total time slots to be allocated.
2: **repeat**
3: Select the sensor i_k^* specified in (31).
4: $\theta_{i_k^*}\leftarrow\theta_{i_k^*}+1$
5: $T\leftarrow T-1$
6: $k\leftarrow k+1$
7: **update** matrix $\mathbf{C}_k^{-1}$ with (27).
8: **until** ($T=0$ is reached)
9: **return** $\mathbf{\Theta}$, $\mathrm{tr}\left(\mathbf{C}_k^{-1}\right)$

V. Proposed simulated annealing algorithm

Although our greedy algorithm is efficient, its performance with respect to reconstruction error could be further improved by a more general optimization method. For solving an optimization problem in a large search space, simulated annealing (SA) is a general meta-heuristic method to approximate the global optimum of any given function especially when the search space is discrete. Therefore, we propose an SA method to solve the MSE minimization problem.

SA could be modeled as a random walk on a search graph, whose vertices are all feasible states of $\mathbf{\Theta}=\{\theta_1,\cdots,\theta_N\}$ such that $\sum_{i=1}^N\theta_i=T$ and $\theta_i\in\{0,1,\cdots,\theta_{\max}\},\forall i\in\mathcal{S}$. Given $\mathbf{\Theta}$ as the current allocation state, we heuristically generate a neighboring state $\mathbf{\Theta}'$ with a slight modification from the current state $\mathbf{\Theta}$ by switching one time slot allocation from sensor i and sensor j:

$$\mathbf{\Theta}'=\{\cdots,\theta_i-1,\cdots,\theta_j+1,\cdots\}. \tag{32}$$

The searching process for SA on the state space is called a stochastic descent procedure. In order to guide the procedure in a more efficient way, we refer to a guided searching method that considers the payoff such that the neighbor with lower payoff would have higher probability to be generated. Let $\mathbb{P}(\mathbf{\Theta})$ denote the payoff for state $\mathbf{\Theta}$, which is directly defined as the objective function in (12):

$$\mathbb{P}(\mathbf{\Theta})=\mathrm{tr}\left(\left(\mathbf{\Lambda}+\frac{1}{\sigma^2}\mathbf{\Psi}^T\mathbf{Q}\mathbf{\Psi}\right)^{-1}\right). \tag{33}$$

By modeling the searching process as a random walk on a graph of state space, in each iteration, SA should determine which neighboring states $\mathbf{\Theta}'$ to explore and probabilistically traverse a neighboring state according to $\mathbb{P}(\mathbf{\Theta})$.

A. Strategies for neighbor generation

To proceed, we propose three strategies to generate neighboring states by choosing 1) random neighbors, 2) all neighbors, or 3) informative neighbors, and propose the corresponding probability for state transition derived from $\mathbb{P}(\mathbf{\Theta})$.

1) Random neighbors: For the strategy of random neighbors, we randomly select two nodes i and j and switch one time slot allocation from one node to the other. If the number of allocated time slots for these two nodes θ_i and θ_j are neither 0 nor $\theta_{\max}$, there are two feasible neighboring states:

$$\begin{cases}\mathbf{\Theta}_1=\{\cdots,\theta_i+1,\cdots,\theta_j-1,\cdots\},\\ \mathbf{\Theta}_2=\{\cdots,\theta_i-1,\cdots,\theta_j+1,\cdots\}.\end{cases} \tag{34}$$

During each SA iteration, these two neighboring states $\mathbf{\Theta}_1$ and $\mathbf{\Theta}_2$ are generated and $\mathbb{P}(\mathbf{\Theta}_1)$ and $\mathbb{P}(\mathbf{\Theta}_2)$ are calculated. The probability to traverse a neighboring state $\mathbf{\Theta}'\in\{\mathbf{\Theta}_1,\mathbf{\Theta}_2\}$ is determined by:

$$\frac{1/\mathbb{P}(\mathbf{\Theta}')}{\sum_{i=1}^2 1/\mathbb{P}(\mathbf{\Theta}_i)}. \tag{35}$$

By the design of (35), the state with lower payoff is stochastically preferred. On the other hand, if one of θ_i and θ_j equals 0 or $\theta_{\max}$, there is only one feasible neighboring state to traverse. If both of θ_i and θ_j equals 0 or $\theta_{\max}$, we skip this iteration because there is no neighboring states.

2) All neighbors: The strategy of random neighbors only considers two neighboring states at most in each iteration, and thus, few computation overhead is involved to calculate the payoff in each iteration. However, two neighboring states are relatively small compared to the number of all the feasible neighboring states. The improvement of payoff or the possibility to reach the global optimum in each iteration is negligible, which may require higher number of iterations before convergence or even converge to poor local optimum. As a result, we propose another strategy to take all the neighboring states for $\mathbf{\Theta}$ into consideration. All the neighboring states consist of the permutation produced by swapping one time slot for arbitrary pair of nodes. Since there are N sensors in the field, at most $2\cdot C_2^N$ neighboring states exist for a given state. Similarly, at each SA iteration, we calculate the payoff of all the feasible neighboring states, and the probability of moving to each neighboring state $\mathbf{\Theta}'$ can be written as:

$$\frac{1/\mathbb{P}(\mathbf{\Theta}')}{\sum_{\mathbf{\Theta}''\in\mathbf{\Theta}_{all}}1/\mathbb{P}(\mathbf{\Theta}'')}, \tag{36}$$

where $\mathbf{\Theta}_{all}$ denotes the set containing all the neighboring states for $\mathbf{\Theta}$.

3) Informative neighbors: A complete search of all feasible neighboring states with the calculation of corresponding payoff is time-consuming. We aim to offer an eclectic strategy able to wisely explore a considerable number of neighboring states without consuming too much time within each iteration. We use a heuristic method that removes one time slot from the most "uninformative" sensor and adds that time slot to the most "informative" sensor by computing the change of MSE. Given current state $\mathbf{\Theta}$, if we remove a time slot from sensor i, the change of MSE can be expressed as

$$\Delta\mathrm{MSE}_i=\frac{q_i^{(\theta_i)}}{\sigma^2+c_{i,i}q_i^{(\theta_i)}}\bar{\mathbf{c}}_i^T\mathbf{c}_i \tag{37}$$

by applying the result obtained in (30). We define the most "uninformative" sensor i^* as

$$i^*=\arg\min_{i\in\mathcal{S},\,\theta_i>0}\left(\frac{q_i^{(\theta_i)}}{\sigma^2+c_{i,i}q_i^{(\theta_i)}}\bar{\mathbf{c}}_i^T\mathbf{c}_i\right), \tag{38}$$

which is to find the sensor with minimum contribution to reduce MSE. On the other hand, if we add a time slot to sensor j, the change of MSE can be expressed as

$$\Delta\text{MSE}_j = -\frac{q_j^{(\theta_j+1)}}{\sigma^2 + c_{j,j}q_j^{(\theta_j+1)}}\bar{\mathbf{c}}_j^T\mathbf{c}_j \tag{39}$$

Symmetrically, we define the most "informative" sensor j^* as

$$\begin{aligned} j^* &= \arg\min_{j\in\mathcal{S},\,\theta_j<T}\left(-\frac{q_j^{(\theta_j+1)}}{\sigma^2 + c_{j,j}q_j^{(\theta_j+1)}}\bar{\mathbf{c}}_j^T\mathbf{c}_j\right) \\ &= \arg\max_{j\in\mathcal{S},\,\theta_j<T}\left(\frac{q_j^{(\theta_j+1)}}{\sigma^2 + c_{j,j}q_j^{(\theta_j+1)}}\bar{\mathbf{c}}_j^T\mathbf{c}_j\right), \end{aligned} \tag{40}$$

which is to find the sensor with maximum contribution to reduce MSE.

Finally, we pair the k most "uninformative" sensors and the k most "informative" sensors to generate k^2 neighboring states denoted by the set $\mathbf{\Theta}_k$. The probability to move to state $\mathbf{\Theta}' \in \mathbf{\Theta}_k$ can be expressed as

$$\frac{1/\mathbb{P}\left(\mathbf{\Theta}'\right)}{\sum_{\mathbf{\Theta}''\in\mathbf{\Theta}_k} 1/\mathbb{P}\left(\mathbf{\Theta}''\right)}. \tag{41}$$

By varying k, we could control the completeness of exploration and the computation overhead in each SA iteration.

Algorithm 2 Proposed Simulated Annealing (SA) Algorithm

1: Initialize $t_{\max}$, γ_0, γ_{final}, $p_{\min}$, $l = 0$, and $\mathbb{P}_{best} = \infty$
2: Obtain initial state $\mathbf{\Theta}$ from Algorithm 1.
3: **for** $t = 1 \to t_{\max}$ **do**
4: Generate a candidate neighbor $\mathbf{\Theta}'$ by one of the proposed strategies for neighbor generation.
5: Accept the state transition with the probability of $p\left(\mathbb{P}\left(\mathbf{\Theta}\right), \mathbb{P}\left(\mathbf{\Theta}'\right)\right)$.
6: **if** the state transition is accepted **then**
7: $\mathbf{\Theta} \leftarrow \mathbf{\Theta}'$
8: $\gamma_t \leftarrow \alpha\gamma_{t-1}$
9: $l \leftarrow 0$
10: **else**
11: $t \leftarrow t - 1$
12: $l \leftarrow l + 1$
13: **if** $l = L_{\max}$ and $p\left(\mathbb{P}\left(\mathbf{\Theta}\right), \mathbb{P}\left(\mathbf{\Theta}'\right)\right) < p_{\min}$ **then**
14: **return** $\mathbf{\Theta}^*$
15: **end if**
16: **end if**
17: **if** $\mathbb{P}\left(\mathbf{\Theta}\right) < \mathbb{P}_{best}$ **then**
18: $\mathbf{\Theta}^* \leftarrow \mathbf{\Theta}$
19: **end if**
20: **end for**
21: **return** $\mathbf{\Theta}^*$

B. Overall algorithm

The SA algorithm is shown in Algorithm 2. In the following context, we describe four steps of SA in details.

TABLE II
PARAMETERS FOR THE SYNTHETIC SIMULATION

Parameter	Value
Node Location Distribution	Random Distribution
Position of Base Station	(0,0)
Number of sensor (N)	40
Side length of square area (R)	60 m
Transmission Power	0 dBm
Frequency Band	2.4 GHz
Correlation Factor (κ)	2000 m^{-2}

1) Initial solution state: The determination of the start point in the feasible search space is influential since a good start point may be closer to the optimal solution with high probability. We use the solution obtained in our greedy algorithm to serve as the initial solution state in line 2.

2) Acceptance probability: When a new state $\mathbf{\Theta}'$ is generated by any of the aforementioned strategies, we have to decide whether to traverse or not. The probability of making the transition from the current state $\mathbf{\Theta}$ to a candidate new state $\mathbf{\Theta}'$ is specified by an acceptance probability function depending on the payoff $\mathbb{P}\left(\mathbf{\Theta}\right)$ and $\mathbb{P}\left(\mathbf{\Theta}'\right)$ and the global time-varying temperature γ_t. If $\mathbb{P}\left(\mathbf{\Theta}'\right)$ is smaller than $\mathbb{P}\left(\mathbf{\Theta}\right)$, the state transition is accepted. However, if $\mathbb{P}\left(\mathbf{\Theta}'\right)$ is greater than $\mathbb{P}\left(\mathbf{\Theta}\right)$, there is still a probability of $e^{\left(\mathbb{P}\left(\mathbf{\Theta}\right)-\mathbb{P}\left(\mathbf{\Theta}'\right)\right)/\gamma_t}$ that the state transition will be accepted. Specifically, the acceptance function p could be given by

$$p\left(\mathbb{P}\left(\mathbf{\Theta}\right), \mathbb{P}\left(\mathbf{\Theta}'\right)\right) = \min\left(1, e^{\left(\mathbb{P}\left(\mathbf{\Theta}\right)-\mathbb{P}\left(\mathbf{\Theta}'\right)\right)/\gamma_t}\right), \tag{42}$$

where γ_t is the temperature in the t-th iteration defined in the following step. Probability of the transition from a state with lower payoff to a state with higher payoff prevents the method to be stuck at local minimum. The implementation for transition acceptance is shown in line 5.

3) Cooling schedule: The cooling schedule depends on three parameters, the initial temperature γ_0, the final temperature γ_{final}, and the iteration length $t_{\max}$. We use a common cooling function $\gamma_t = \alpha\gamma_{t-1}$ in line 8 to control the temperature at each iteration by a constant multiplication factor α:

$$\alpha = \exp\left(\frac{\log\left(\frac{\gamma_{final}}{\gamma_0}\right)}{t_{\max}}\right). \tag{43}$$

Note that iteration counts only when the state transition is accepted. Hence, if the state transition is not accepted, we minus t by one to maintain the number of iteration in line 11.

4) Stopping criteria: The default stopping criteria is when the number of iteration t reaches $t_{\max}$ (or equivalently, when the temperature γ_t reaches γ_{final}). However, if the current state does not move to a new state before reaching $t_{\max}$ for $L_{\max}$ consecutive iterations and the acceptance ratio falls below a given small value $p_{\min}$, we halt the algorithm as well from line 13 to 15.

VI. EVALUATION

In this section, we evaluate the performance of several existing algorithms and our proposed algorithms by the simulation

over synthetic environment and real-world environment. We compare the reconstruction MSE and the computation time among different methods to demonstrate how retransmission mechanism could enhance the reconstruction MSE with competitive time complexity under our proposed algorithms. We also analyze the convergence of SA under different strategies for neighbor generation to confirm that selecting informative neighbors could achieve the lowest MSE given similar computation time.

For the synthetic environment, $N = 40$ sensors are distributed uniformly among a square area with side length of $R = 60$ meters. The BS is placed at the center of the square. The collected data from sensor i is assumed to follow Gaussian distribution with mean μ_i and variance σ_i. The covariance $\sigma_{i,j}$ between the data collected by sensor i and j is determined by the distance $d_{i,j}$ between these two sensors and a spatial correlation factor κ as $\sigma_{i,j} = \sigma_i \sigma_i e^{-d_{i,j}^2/\kappa}$, a common spatial correlation used in literature [22]. The default transmission power of each sensor is set to 0 dBm. We follow the path loss model and frequency band specified in [23]. To model link loss, the received signal strength indicator (RSSI) values are converted to the packet delivery ratio (PDR) values by a conversion table based on real-world deployment specified in [23]. The default setting of simulation parameters is summarized in Table II.

A. Baseline algorithms

To demonstrate how the proposed Greedy and SA algorithm could benefit the system performance in a lossy WSN, we consider the following approaches proposed in related work addressing the node selection problem in lossless or lossy WSN.

1) Enhanced Correlation Based Deterministic Node Selection (ECB-DNS): ECB-DNS, proposed in [10], successively decides the node that brings the largest improvement in terms of reconstruction quality with respect to spatial correlation among sensors. The selected nodes are all equally allocated one transmission opportunity regardless of the link condition.

2) Genie-Aided Least-Square estimation (GALS): Notice that ECB-DNS is designed for lossless WSNs. To better compare the performance in lossy WSNs, we include GALS algorithm proposed in [15] considering packet loss in the problem formulation similar to our assumption with the difference that GALS only allocates one transmission opportunity to each selected sensor.

3) Greedy Node Selection with Expected Transmission Count (GNS-ETX): GNS-ETX [16] proposed a greedy node selection method based on the concept of enhanced correlation structure formulated in ECB-DNS to further incorporate the impact of lossy links. The transmission opportunity for each selected sensor is decided by the expected transmission count (ETX), which is the expected number of transmissions for a packet to be received at its destination. This method determines the number of retransmission solely based on link condition, unlike Greedy and SA proposed in this paper, which also include correlation structure among the network to determine the number of retransmission for each selected sensor.

B. Performance of the proposed algorithms

1) Reconstruction MSE under various available time slots: We first compare all the algorithms under various number of available time slots to evaluate how resource scarcity could influence the performance of each algorithm. Fig. 2 shows the performance in a lossless WSN. Although our algorithms (Greedy and SA) are designed for lossy WSN, they outperform existing algorithms like ECB-DNS and GNS-ETX[1] with respect to the reconstruction MSE in lossless WSN presented in Fig. 2, which means our algorithms could select more informative sensors compared to ECB-DNS and GNS-ETX even when the network is lossless. Our algorithms achieve similar performance as GALS achieves. For lossy WSN in Fig. 3, our algorithms outperform all of the existing algorithms. One could find that even for $T = N = 40$, when the number of available time slots is the same as the number of sensors, our algorithms achieve lower reconstruction MSE since we do not allocate time slot to each node. We allocate more than one time slots to informative but lossy sensors and therefore achieve even lower reconstruction MSE.

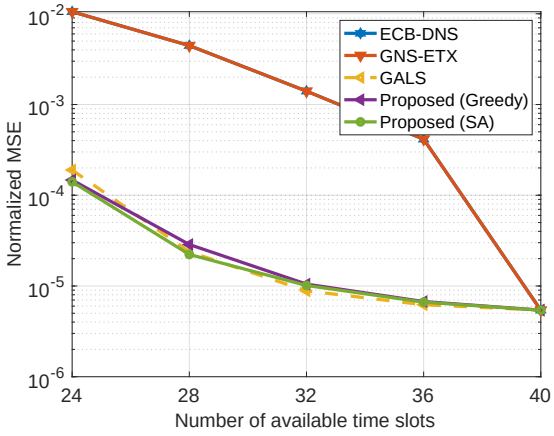

Fig. 2. Reconstruction MSE versus available time slots with transmission power 0 dBm in a lossless WSN with 40 sensors.

2) Reconstruction MSE under various spatial correlations: We further compare the performance under networks with different spatial correlations between sensors. We modify the spatial correlation factor κ to make the sensed data $\mathbf{x}$ become K-sparse and define the sparsity ratio as K/N. We plot the reconstruction MSE over various sparsity ratio under the constraint of $T = K + 1$ in Fig. 4 to make the sensed data reconstructable. One could observe that when both the sparsity ratio and the available resource become smaller (from $K/N = 0.8$ to $K/N = 0.5$), the reconstruction MSE for existing algorithms is not effectively reduced. However, the reconstruction MSE for our proposed algorithms goes down steadily with decreasing sparsity ratio. Such result indicates our algorithms could better reconstruct signal under constrained resource, especially for low sparsity ratio ($K/N < 0.7$). That is, our algorithms performs much better when the data from various sensors is highly correlated. From the

[1]Note that GNS-ETX is equivalent to ECB-DNS when the link is lossless.

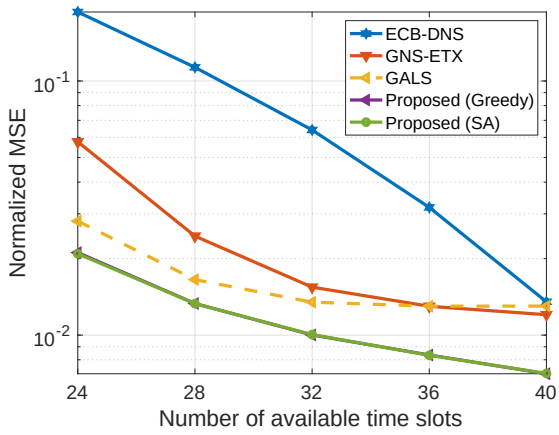

Fig. 3. Reconstruction MSE versus available time slots with transmission power 0 dBm in a lossy WSN with 40 sensors.

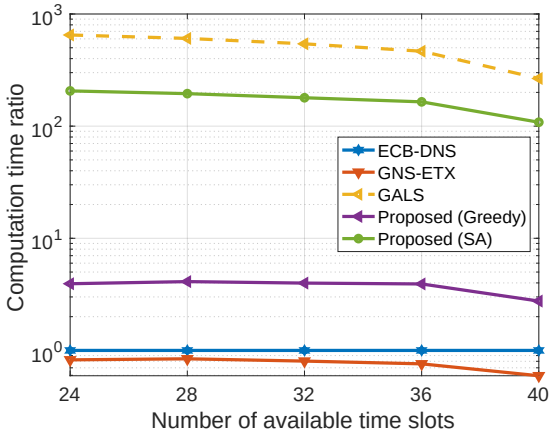

Fig. 5. Computation time ratio versus available time slots with respect to ECB-DNS.

microscopic observation in Section VI-C, we attribute such improvement to the ability of our algorithms to identify the most stable link among neighboring sensors and concentrate sufficient transmission opportunities to that link to guarantee successful data collection.

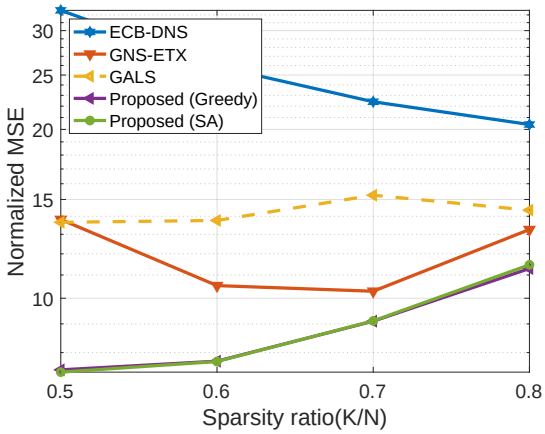

Fig. 4. Reconstruction MSE versus sparsity ratio with transmission power 0 dBm in a lossy WSN with available time slots $T = K + 1$.

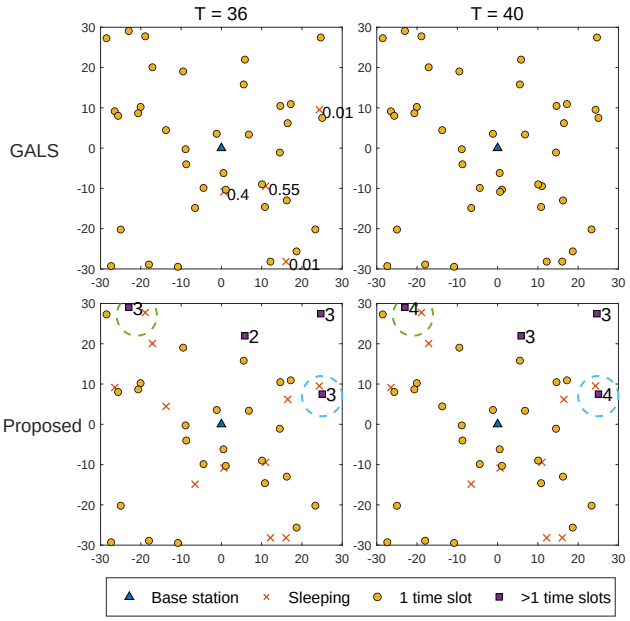

Fig. 6. Layout of resource allocation for GALS and our proposed algorithm ("Proposed") for $T = 36, 40$. The decimal number in GALS for $T = 36$ is the PDR for the nearby sleeping sensors. The green and blue dashed circles in "Proposed" highlight the retransmission of resource allocation.

3) Required computation time: Finally, Fig. 5 demonstrates the ratio of computation time for all the algorithms with respect to ECB-DNS, the baseline algorithm designed for lossless WSNs. Although ECB-DNS and GNS-ETX are the most efficient algorithms, their decision for node selection and resource allocation in lossless and lossy WSNs is not comparable with the remaining algorithms. On the other hand, our proposed Greedy and SA outperform GALS both in performance and efficiency.

C. Microscopic observation of resource allocation

Most of recent endeavors focused on node selection of informative sensors. The optimization on resource allocation, or the number of retransmission, among the selected sensors is not jointly handled. In this subsection, we elaborate the benefit of optimization on the number of retransmission via microscopic observation. Fig. 6 displays the layout of all the sensors and the number of allocated transmission for GALS and our Greedy under two different resource constraints, $T = 36$ and $T = 40$. We choose GALS for comparison since GALS is designed for lossy WSNs without retransmission, a suitable baseline solution to demonstrate the benefit of retransmission proposed by our solution. In GALS, it is obvious that the four extra selected sensors from $T = 36$ to $T = 40$ are all sensors with poor channel condition (low PDR of 0.01, 0.01, 0.4, and 0.55). Also, these four sensors share high spatial correlation with their selected neighbors in similar region.

In other words, the extra selected sensors are lossy and not informative. Although we merely display the case for $T = 36$ to $T = 40$, it is a common phenomenon that algorithms without retransmission have no choice but to choose lossy and non-informative sensors even when T is lower than 36. In contrast, our proposed algorithm could avoid selecting lossy and non-informative sensors by reallocating resource to the selected sensors. Fig. 6 reveals that our algorithm does not allocate any resource to the four lossy but non-informative sensors mentioned previously. Instead, time slots are allocated to informative but lossy sensors. For example, in the green and blue dashed circles in Fig. 6, we allocate 3 to 4 time slots to the selected sensors but 0 time slots to their adjacent neighbors. These adjacent sensors in the circles are equally informative (representing similar spatial information in that region), but only one sensor in each dashed circle has higher PDR than the others. Therefore, it is wiser to concentrate resource on the one with the highest PDR to better conquer the impact of channel uncertainty. The allocation result shows our algorithm is able to tackle the situation and allocate resource wisely.

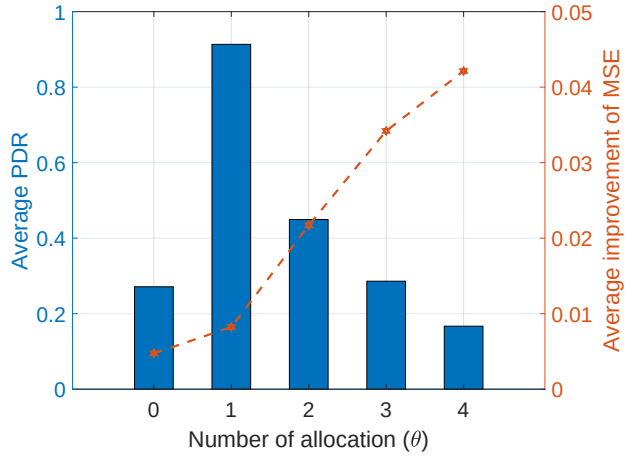

Fig. 7. The average PDR and average improvement of MSE over sensors with different number of time slots allocation.

We conclude our observation in Fig. 7, where the average values of PDR (q_i) and the improvement of reconstruction MSE (ΔMSE_i) of sensor i over 100 experiments are displayed. In these experiments, the number of time slot allocation for each sensor ranges from 0 (not selected) to 4 (selected with 3 times of retransmission opportunity). We group all the sensors with the same number of time slot allocation into the same set to plot the corresponding bars for PDR and the curve for MSE improvement. We find that the sensors receiving more than 2 time slots exhibit higher improvement of MSE (more informative) with lower PDR (more lossy) while the sensors receiving 1 time slot tend to possess higher PDR (less lossy) since retransmission is not that required. Finally, the sensors with 0 time slots exhibit lower improvement of MSE and lower PDR. The ability to jointly consider reconstruction information and channel condition is the reason that our algorithm outperforms previous work.

D. Convergence analysis of the proposed SA algorithm

We compare different strategies for neighbor generation as we have described in Section V. In Fig. 8, we show the objective payoff value obtained by different strategies for neighbor generation. One can observe that "Informative" neighbor generation reaches the lowest payoff, followed by "All" and "Random" neighbor generation. "Informative" neighbor generation requires approximately 200 iterations to converge to an optimal payoff value. In comparison, much more iterations are required for the other two schemes to converge. Although "Informative" neighbor generation has a faster convergence rate than the other two strategies for neighbor generation, it requires longer computation time for each SA iteration with respect to "Random" neighbor strategy. For "Informative" neighbor generation to converge, approximately 200 iterations is required, leading to 0.6875 second in total. In contrast, it takes approximately 1200 iterations for "Random" neighbor generation to converge, requiring 0.6732 second on average. For "All" neighbor generation, much higher accumulated time is observed. In conclusion, "Informative" neighbor generation achieves the lowest payoff with similar accumulated time for convergence compared to "Random" neighbor generation.

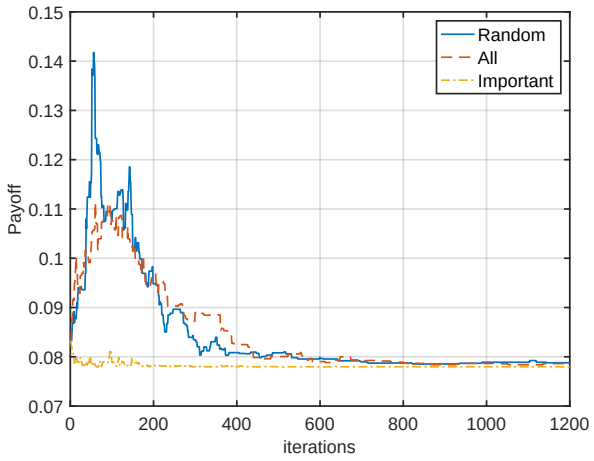

Fig. 8. Payoff versus number of iterations with different neighbor generation strategies

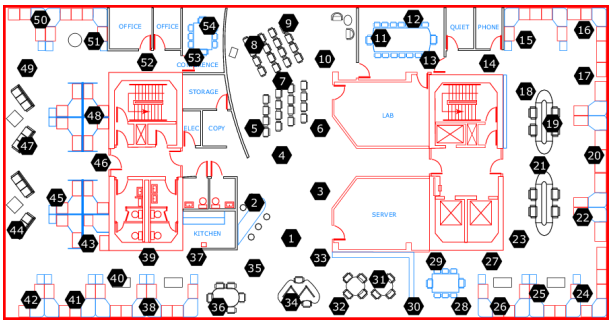

Fig. 9. Layout of the WSN of the Intel Berkeley Research Lab

E. Evaluation on real data

To further validate our proposed algorithms, we use the temperature data of Intel lab [24], which is a dataset collected from 54 sensors deployed in the Intel Berkeley Research lab between February 28th and April 5th, 2004. The sensors are arranged in the lab with size 40×30 (m^2) as shown in Figure 9 [24]. Each sensor collects timestamped temperature information once every 31 seconds and report data back to the data aggregator.

In the Intel lab data, each sensor has measured the environmental temperature for more than one month and the datalog file contains about 2.3 million readings collected from these sensors. The epoch number is a monotonically increasing sequence number for each sensor. Multiple readings from the same epoch number are produced from different sensors at the same time. There are some missing epochs in this dataset. We observe that the temperature data has very high temporal correlation in the Intel lab data. Therefore, we use interpolation to deal with missing data and use the repaired data as our experimental dataset. The location of the aggregator is not revealed. Therefore, we assume that the data aggregator is placed at the center of the floor and each sensor transmits the sensed data back through one-hop communication. For each sensor, we calculate the distance d between it and the aggregator to calculate the received RSSI value based on propagation model in 6TiSCH simulator [25].

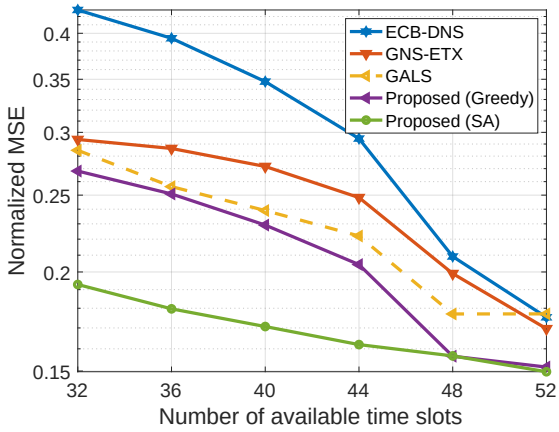

Fig. 10. Reconstruction MSE versus available time slots with transmission power -10 dBm for the Intel lab data

In each trial, we randomly select a start epoch t and use 100 consecutive epochs of data measured from all the sensors as dataset $\mathcal{X} = \{\mathbf{x}_t, \mathbf{x}_{t+1}, \cdots, \mathbf{x}_{t+100}\}$. The sparsifying matrix $\mathbf{\Psi}$ is constructed from PCA [8]. We evaluate the reconstruction MSE performance over 100 trials. The sparsifying matrix $\mathbf{\Psi}$ is constructed from PCA, a statistical procedure utilizing the sample covariance matrix $\mathbf{\Sigma}$ measured from historical data from all the sensors in the network to convert the correlated signal into a sparse signal. The first step of PCA is to center the values of all of the input variables [8]. Accordingly, we refer to the temporal mean of $\mathcal{X}$ as $\bar{\mathbf{x}}$. Hence, we have

$$\mathbf{x}_t - \bar{\mathbf{x}} = \mathbf{\Psi}\mathbf{u}. \tag{44}$$

In our experiments, to focus on the MSE performance of different resource allocation strategies, we assume the temporal mean $\bar{\mathbf{x}}$ and sample covariance matrix $\mathbf{\Sigma}$ of data $\mathcal{X}$ is computed in prior. Once we obtain the $\mathbf{\Psi}$ and $\mathbf{\Sigma}$, we are able to derive the hyperparameters $\mathbf{\Lambda}$ of sparse vector $\mathcal{U} = \{\mathbf{u}_t, \mathbf{u}_{t+1}, \cdots, \mathbf{u}_{t+100}\}$, that is,

$$\mathbf{\Lambda} = \mathbf{\Psi}^T \mathbf{\Sigma} \mathbf{\Psi}. \tag{45}$$

In each trial, we will random select a start epoch t and use 100 consecutive epochs of data from 52 sensors as dataset $\mathcal{X} = \{\mathbf{x}_t, \mathbf{x}_{t+1}, \cdots, \mathbf{x}_{t+100}\}$, and we will evaluate the reconstruction MSE performance over 100 trials.

In Fig. 10, we show the reconstruction MSE versus available time slots under different algorithms for the real Intel lab data. One could find that even if the proposed Greedy suffers from higher MSE for lower number of available time slots compared to the proposed SA, both of our proposed algorithms achieve better reconstruction quality for all numbers of available time slots with respect to existing algorithms.

VII. Conclusion

In this paper, we investigate the resource allocation problem with retransmission mechanism for CS-based lossy WSN. We determine the number of transmission opportunities considering both the packet deliver ratio of each sensor and the overall spatial correlation among the network. By simulation on synthetic and real-world environments, we find that the proposed Greedy and SA algorithm could efficiently allocate transmission opportunities to the most informative yet lossy sensors, paving a robust and efficient way to CS-based lossy WSNs.

References

[1] Q. Wang and J. Jiang, "Comparative Examination on Architecture and Protocol of Industrial Wireless Sensor Network Standards," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 2197–2219, 2016.

[2] A. Ali, Y. Ming, S. Chakraborty, and S. Iram, "A comprehensive survey on real-time applications of WSN," *Future internet*, vol. 9, no. 4, p. 77, 2017.

[3] M. T. Lazarescu, "Design of a WSN Platform for Long-Term Environmental Monitoring for IoT Applications," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 3, no. 1, pp. 45–54, 2013.

[4] H.-Y. Hsieh, C.-H. Chang, and W.-C. Liao, "Not Every Bit Counts: Data-Centric Resource Allocation for Correlated Data Gathering in Machine-to-Machine Wireless Networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 11, no. 38, Feb. 2015.

[5] H.-Y. Hsieh, T.-C. Juan, Y.-D. Tsai, and H.-C. Huang, "Minimizing Radio Resource Usage for Machine-to-Machine Communications through Data-Centric Clustering," *IEEE Transactions on Mobile Computing*, vol. 15, pp. 3072–3076, Dec. 2016.

[6] H.-Y. Hsieh and C.-Y. Su, "Data-Centric Scheduling for Minimizing Queue Length in Wireless Machine-to-Machine Networks," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 2019, pp. 1–6.

[7] D. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 1289–1306, pp. 471–480, Apr. 2006.

[8] G. Quer, R. Masiero, G. Pillonetto, M. Ross, and M. Zorzi, "Sensing, compression, and recovery for WSNs: Sparse signal modeling and monitoring framework," *IEEE Transactions on Wireless Communications*, vol. 11, no. 10, p. 3447–3461, Oct. 2012.

[9] S. Hwang, R. Ran, J. Yang, and D. K. Kim, "Multivariated Bayesian Compressive Sensing in Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 16, pp. 2196 – 2206, Dec. 2015.

[10] M. Hooshmand, M. Rossi, D. Zordan, and M. Zorzi, "Covariogram-Based Compressive Sensing for Environmental Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 16, no. 6, pp. 1716–1729, 2015.

[11] M. Kortas, O. Habachi, A. Bouallegue, V. Meghdadi, T. Ezzedine, and J.-P. Cances, "Robust Data Recovery in Wireless Sensor Network: A Learning-Based Matrix Completion Framework," *Sensors*, vol. 21, no. 3, 2021.

[12] M. Kortas, V. Meghdadi, A. Bouallegue, T. Ezzeddine, O. Habachi, and J.-P. Cances, "Routing aware space-time compressive sensing for Wireless Sensor Networks," *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct. 2017.

[13] X. Wu, P. Yang, T. Jung, Y. Xiong, and X. Zheng, "Compressive Sensing Meets Unreliable Link: Sparsest Random Scheduling for Compressive Data Gathering in Lossy WSNs," in *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 13–22.

[14] C. Zhang, O. Li, Y. Yang, G. Liu, and X. Tong, "Energy-efficient data gathering algorithm relying on compressive sensing in lossy WSNs," *Measurement*, vol. 147, p. 106875, 2019.

[15] W. Chen and I. J. Wassell, "Optimized Node Selection for Compressive Sleeping Wireless Sensor Networks," *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY*, vol. 65, Feb. 2016.

[16] H.-H. Chen and H.-Y. Hsieh, "Data-Centric Node Selection for Machine-Type Communications with Lossy Links," in *2020 European Conference on Networks and Communications (EuCNC)*, 2020, pp. 358–363.

[17] B. Jiang, G. Huang, F. Li, and S. Zhang, "Compressed Sensing With Dynamic Retransmission Algorithm in Lossy Wireless IoT," *IEEE Access*, vol. 8, pp. 133 827–133 842, 2020.

[18] K. S. Miller, "On the Inverse of the Sum of Matrices," *Mathematics Magazine*, vol. 54, pp. 67–72, Mar. 1981.

[19] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *The Journal of Machine Learning Research*, vol. 1, pp. 211–244, Jun. 2001.

[20] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*. USA: John Wiley amp; Sons, Inc., 1990.

[21] K. M. Bretthauer and B. Shetty, "The nonlinear knapsack problem–algorithms and applications," *European Journal of Operational Research*, vol. 138, no. 3, pp. 459–472, 2002.

[22] B. S. James, O. Berger, and V. de Oliveira, "Objective bayesian analysis of spatially correlated data," *Journal of the American Statistical Association*, vol. 96, no. 456, p. 1361–1374, Dec. 2001.

[23] E. Municio, G. Daneels, M. Vučinić, and S. Latré, "Simulating 6TiSCH networks," *Transactions on Emerging Telecommunications Technologies*, Sep. 2018.

[24] P. Bodik, W. Hong, C. Guestrin, S. Madden, M. Paskin, and R. Thibaux, "Intel Lab Data," Available online: http://db.csail.mit.edu/labdata/labdata.html.

[25] M. Esteban, D. Glenn, M. Vucinic, S. Latre, J. Famaey, Y. Tanaka, K. Brun-Laguna, X. Vilajosana, K. Muraoka, and T. Watteyne, "The 6TiSCH Simulator," Available online https://bitbucket.org/6tisch/simulator/src/master/.