

## Research Article

# A DCT Regularized Matrix Completion Algorithm for Energy Efficient Data Gathering in Wireless Sensor Networks

Kefu Yi,<sup>1</sup> Jiangwen Wan,<sup>1</sup> Tianyue Bao,<sup>1,2</sup> and Lei Yao<sup>1</sup>

<sup>1</sup>*School of Instrumentation Science and Opto-Electronics Engineering, Beihang University, Beijing 100191, China*

<sup>2</sup>*Northeast Petroleum University, Daqing 163318, China*

Correspondence should be addressed to Jiangwen Wan; [jwwan@buaa.edu.cn](mailto:jwwan@buaa.edu.cn)

Received 16 April 2015; Accepted 25 June 2015

Academic Editor: Qilian Liang

Copyright © 2015 Kefu Yi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a novel matrix completion algorithm to enable energy efficient data gathering in wireless sensor networks. The algorithm takes full advantage of both the low-rankness and the DCT compactness features in the sensory data to improve the recovery accuracy. The time complexity of the algorithm is analyzed, which indicates it has a low computational cost. Moreover, the recovery error is carefully analyzed and a theoretical upper bound is derived. The error bound is then validated by experimental results. Extensive experiments are conducted on three datasets collected from two testbeds. Experimental results show that the proposed algorithm outperforms state-of-the-art methods for low sampling rate and achieves a good recovery accuracy even if the sampling rate is very low.

## 1. Introduction

Wireless sensor networks (WSNs) have been widely used in a variety of applications such as precision agriculture [1], personal health monitoring [2], and environment surveillance [3], in which sensor nodes with limited battery energy are deployed and periodically report their sensor readings to the base station or sink node. Therefore, a key issue in wireless sensor networks is how to efficiently gather these data from sensors provided with only limited energy resources. In the classical data gathering approach [4], each sensor node simply forwards its sensor readings to the sink, resulting in a large amount of traffic and energy consumption.

Recently, Compressive Sensing (CS) [5, 6] has emerged as a new approach to tackle the efficient data gathering problem in WSNs. Taking advantage of the sparsity in sensor readings, CS based methods [7–9] require fewer data packets than the classical approach. However, there are many practical problems when applying CS to data gathering in WSNs. Firstly, CS based methods require a prior dictionary to sparsify the sensor readings. Secondly, the measurement matrix in CS is composed of independent and identically distributed random Gaussian entries, which is dense with very few zero elements. Therefore, sensor nodes need to sample all sensor readings

and perform a considerable number of measurement operations, resulting in a large amount of energy waste. Thirdly, CS requires the number of measurements to exceed a certain threshold (depending on the sparsity level of sensor readings) to achieve exact recovery. However, the realistic sensor signal is not always exactly sparse as it should be. Therefore, low sampling rate may lead to insufficient measurements and result in a bad recovery accuracy.

As an extension of CS, matrix completion [10] has shown its potential for enabling efficient data gathering in WSNs. Because the sensor readings are highly temporal-spatial correlated, the data matrix structured by the sensor readings will approximate to a low-rank matrix. Therefore, the sink node can gather only a few of the total sensor readings and adopt the matrix completion algorithm to reconstruct the missing data. However, unlike CS based methods, matrix completion based methods do not require the prior dictionary to sparsify the original signal. Furthermore, the sampling matrix (or measurement matrix) in matrix completion is much sparser than CS, which makes it more suitable for wireless sensor networks.

Utilizing the low-rankness feature in sensory data, there are many pioneering works [11–13] on applying matrix completion to WSN, which adopt the alternating least squares

technique to estimate the low-rank matrix. The recovery accuracy is further improved by utilizing the spatiotemporal structure in the WSN data. However, the improvement is limited because the spatiotemporal structure directly implies the low-rank feature, and in some sense, these two features are equivalent. Moreover, the alternating least squares algorithm does not scale to large rank.

Besides the low-rankness feature, we observe that the sensor readings in WSNs also exhibit Discrete Cosine Transform (DCT) compactness feature. In other words, the sensor readings can be approximated by only a small number of DCT coefficients. Therefore, by taking full advantage of the DCT compactness feature of the WSNs data, in this paper, we propose a DCT Regularized Matrix Completion (DRMC) algorithm. We analyze the time complexity of DRMC, which indicates that the proposed algorithm has a low computational cost. Moreover, we analyze the recovery error of DRMC and derive a theoretical upper bound. The error bound is then validated by experimental results. Extensive experiments are carried out on three datasets that are collected from two realistic WSN testbeds. We compare the performance of DRMC with state-of-the-art methods. Experimental results show that DRMC outperforms state-of-the-art methods for low sampling rate and achieves a good recovery accuracy even if the sampling rate is very low.

The main contributions of this paper are summarized in the following:

- (i) We examine the sensor data collected from real-world WSNs, which reveal two data features: (1) *low-rankness*, (2) *DCT compactness*.
- (ii) Inspired by these observations, we design a novel *DCT Regularized Matrix Completion* (DRMC) algorithm to estimate the missing sensory data. Experimental results indicate that DRMC outperforms state-of-the-art methods when the sampling rate is low.
- (iii) We analyze the time complexity of the DRMC algorithm, which indicates that DRMC has a low computational cost.
- (iv) We analyze the recovery error of the DRMC algorithm and derive a theoretical upper bound, which is then validated by experimental results.

The rest of this paper is organized as follows. Section 2 reviews the state-of-the-art methods. Section 3 models the problem. Section 4 examines the data features in WSNs. Section 5 proposes the DRMC algorithm. Section 6 analyzes the time complexity of the algorithm. Section 7 analyzes the recovery error of the algorithm. Section 8 evaluates the effectiveness of the proposed algorithm. Section 9 concludes this paper.

## 2. Related Works

In this section, we make a brief review of previous works related to the data gathering problem in wireless sensor networks.

**2.1. Compressive Sensing.** Compressive Sensing (CS) theory suggests that sparse signals can be accurately reconstructed from only a small number of measurements [5, 6]. It is a new paradigm for signal processing of networked data [7] and there are many CS based methods for data gathering in wireless sensor networks. Luo et al. proposed a data gathering scheme that applies Compressive Sensing theory to reduce communication cost [14]. Quer et al. presented a framework for data gathering and signals recovery in actual WSN deployments with the integration of CS [9]. Ebrahimi and Assi recently proposed a decentralized method to apply the Compressive Sensing to data gathering in wireless sensor networks [15].

**2.2. Matrix Completion.** Recently, there are many applications that apply matrix completion technique to wireless sensor networks. Utilizing the low-rankness and spatiotemporal correlation, Zhang et al. proposed a method to recover the lost data in internet traffic matrices [13]. Kong et al. designed an algorithm using the low-rank structure, time stability, space similarity, and multiattribute correlation to estimate the missing data in highly incomplete data matrix [12]. Cheng et al. presented a Spatiotemporal Compressive Data Collection (STCDG) algorithm that utilizes the low-rankness and short-term stability features to reduce data traffic in WSNs [11].

In our earlier work [16], we have studied the data recovery problem in wireless sensor networks when historical data are available and proposed a DCT-Regularized Partial Matrix Completion (DCT-RPMC) algorithm. However, the new algorithm proposed in this paper does not depend on any historical data, which greatly widens its applicability to more general scenarios.

## 3. Problem Formulation

In this section, we formally formulate the data gathering and recovery problem in wireless sensor networks and state the goal of this paper. The main notations that will be used in the rest of this paper are listed in Summary of Notations.

Assume that the network is composed of  $n$  sensor nodes. During a certain sampling period, the  $i$ th sensor node acquires  $m$  samples, which are modeled as an  $m$ -dimensional sensor vector  $\vec{x}_i$ ,

$$\vec{x}_i = [x(i, 0), x(i, \Delta_T), \dots, x(i, (m-1)\Delta_T)]^T, \quad (1)$$

where  $\Delta_T$  is the sampling period. Therefore, the entire samples in the network can be organized as an environment matrix  $X \in \mathbb{R}^{m \times n}$ ,

$$X = [\vec{x}_1, \vec{x}_2, \dots, \vec{x}_n]. \quad (2)$$

In order to reduce energy consumption, only a fraction of the entries of  $X$  will be transmitted to the sink node. We then define a matrix  $M \in \mathbb{R}^{m \times n}$  as the sampling matrix to indicate which parts of  $X$  are transmitted to the sink:

$$M(i, j) = \begin{cases} 1, & \text{if } X(i, j) \text{ is transmitted,} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

TABLE I: Experimental datasets.

Data name	Matrix size	Time interval
Intel Temperature	54 nodes $\times$ 96 intervals	10 minutes
Intel Humidity	54 nodes $\times$ 96 intervals	10 minutes
PARED Temperature	50 nodes $\times$ 96 intervals	10 minutes

And the sampling rate  $\tau$  is defined in the following:

$$\tau = \frac{\sum_{i,j} M(i, j)}{mn}. \quad (4)$$

Let  $Y \in \mathbb{R}^{m \times n}$  denote the data transmitted to the sink.  $Y$  is an incomplete version of  $X$ , with missing entries replaced by zeros. Therefore, we have

$$Y = M \circ X. \quad (5)$$

The symbol  $\circ$  denotes the element-wise matrix production operator.

After obtaining  $Y$ , the sink can reconstruct the original environment matrix  $X$  with the proposed algorithm in Section 5. Our goal is to generate a reconstructed matrix  $\hat{X}$  that approximates to the original environment matrix  $X$  as closely as possible. We measure the recovery performance by the Normalized Mean Absolute Error (NMAE):

$$\text{NMAE} = \frac{\sum_{i,j:M(i,j)=0} |X(i, j) - \hat{X}(i, j)|}{\sum_{i,j:M(i,j)=0} |X(i, j)|}. \quad (6)$$

## 4. Exploring the Data Features

In this section, we examine the data features in real-world wireless sensor networks.

**4.1. Experimental Datasets.** We use three datasets, which are collected from two WSN testbeds, to serve as the ground truth. The summary of the datasets is shown in Table I.

The first category of datasets is collected by 54 Mica2Dot nodes deployed in the Intel Berkeley Research Lab [17] between February 28 and April 5, 2004. The Mica2Dot node reports collected sensor data including humidity and temperature once every 30 seconds. However, we find that the raw dataset has considerable missing data. Therefore, we have rearranged the raw data (by changing the reporting interval from 30 seconds to 10 minutes) to avoid the missing data.

The second category of dataset consists of temperature readings, which are collected with a 10-minute interval by our own testbed, namely, PARED. PARED consists of 50 sensor nodes. More details about PARED can be found in [18].

**4.2. Low-Rank Structure.** We first examine the low-rank structure in WSN datasets using the Singular Value Decomposition (SVD). The environment matrix  $X$  can be decomposed into three matrices by SVD:

$$X = U \Sigma V^T, \quad (7)$$

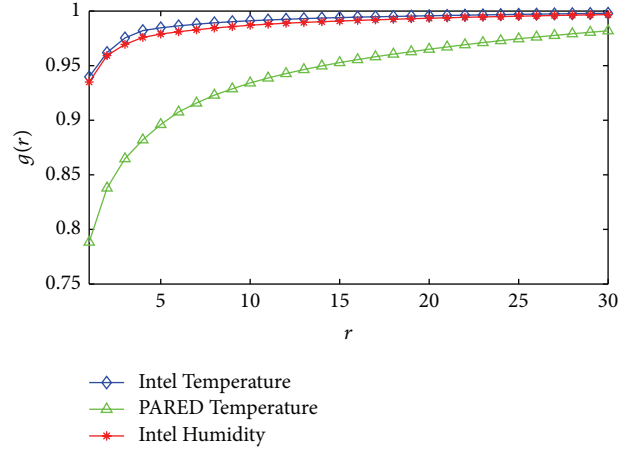


FIGURE 1: Low-rankness feature.

where  $U$  is an  $m \times m$  orthonormal matrix,  $V$  is an  $n \times n$  orthonormal matrix, and  $\Sigma$  is an  $m \times n$  diagonal matrix with singular values  $\sigma_1, \sigma_2, \dots, \sigma_p$  sorted in a descending order ( $p = \min(m, n)$ ).

Because sensor readings in a WSN are spatiotemporal correlated, the environment matrix  $X$  would exhibit low-rank feature. More exactly,  $X$  should approximate to a low-rank matrix of rank  $r$ . So, the first  $r$  singular values will occupy the most energy of  $X$ . We use the following as the metric to examine the quality of the low-rank approximation:

$$g(r) = \frac{\sum_{i=1}^r \sigma_i}{\|X\|_*} = \frac{\sum_{i=1}^r \sigma_i}{\sum_{i=1}^p \sigma_i}, \quad (8)$$

where  $\|\cdot\|_*$  is the nuclear norm and  $\|X\|_* = \sum_{i=1}^p \sigma_i$ .

Figure 1 shows the low-rank approximation quality of the three datasets. We found that the largest 10 singular values occupy the 93%–99% of the total energy, which suggests that the WSN datasets exhibit a good low-rank feature.

**4.3. DCT Compactness.** We also observed that the sensor readings in WSN exhibit DCT compactness feature. In other words, the first  $k$  DCT coefficients of the sensor vector  $\vec{x}_i$  concentrate the most energy of  $\vec{x}_i$ .

We first define  $D$  as the  $m \times m$  Discrete Cosine Transform Matrix:

$$D(i, j) = \sqrt{\frac{2}{m}} \cos \left[ \frac{\pi}{m} \left( i - \frac{1}{2} \right) \left( j - \frac{1}{2} \right) \right]. \quad (9)$$

Then, the  $m \times m$  orthonormal matrix  $D$  can be divided into two submatrices:

$$D = \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}, \quad (10)$$

where  $D_1$  consists of the first  $k$  rows of  $D$  and  $D_2$  consists of the last  $m - k$  rows of  $D$ .

Therefore, if the first  $k$  DCT coefficients occupy the most energy of  $\vec{x}_i$ , we will have  $\|D_1 \vec{x}_i\|_2 / \|\vec{x}_i\|_2 \approx 1$  and  $\|D_2 \vec{x}_i\|_2 / \|\vec{x}_i\|_2 \approx 0$ . Similarly, for the matrix form, we will have

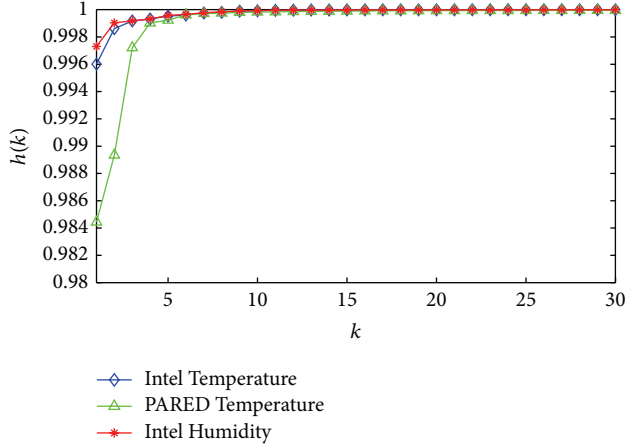


FIGURE 2: DCT compactness feature.

$\|D_1 X\|_F / \|X\|_F \approx 1$  and  $\|D_2 X\|_F / \|X\|_F \approx 0$ , where  $\|*\|_F$  is the Frobenius norm with  $\|X\|_F = \sqrt{\sum_{i,j} X(i,j)^2}$ . So, we use the following function to examine the DCT compactness feature:

$$h(k) = \frac{\|D_1 X\|_F}{\|X\|_F}. \quad (11)$$

From Figure 2, we can see that the first 10 DCT coefficients concentrate 99% of the total energy, which suggests that these WSN datasets exhibit a good DCT compactness feature.

## 5. Algorithm

In this section, we proposed a novel matrix completion algorithm, namely, DCT Regularized Matrix Completion (DRMC), to solve the data recovery problem in WSN data gathering. DRMC takes full advantage of the low-rankness and DCT compactness features to improve the recovery accuracy.

**5.1. Utilization of Low-Rankness.** As mentioned before in Section 3, the goal of the recovery problem is to estimate  $X$  from only a fraction of known entries. According to [10], we can recover  $X$  by solving the following rank optimization problem if  $X$  is a low-rank matrix:

$$\begin{aligned} & \text{minimize} \quad \text{rank}(X) \\ & \text{subject to} \quad M \circ X = Y. \end{aligned} \quad (12)$$

However, the rank minimization problem (12) is NP-hard and is not solvable in polynomial time. Since the nuclear norm is the optimal convex approximation of the rank function, a reasonable solution is to solve a convex relaxation problem with the rank function replaced by the nuclear norm:

$$\begin{aligned} & \text{minimize} \quad \|X\|_* \\ & \text{subject to} \quad M \circ X = Y. \end{aligned} \quad (13)$$

However, in a more realistic occasion, the environment matrix  $X$  is not an exactly low-rank matrix. There may not

### Input:

$Y \in \mathbb{R}^{m \times n}$ : collected data matrix  
 $M \in \mathbb{R}^{m \times n}$ : sampling matrix  
 $\bar{\lambda}$ : nuclear norm regularization parameter  
 $\mu$ : DCT regularization parameter

### Output:

$\hat{X} \in \mathbb{R}^{m \times n}$ : reconstructed environment matrix

### Main procedure:

- (1)  $L \leftarrow 1 + 2\mu$ ;
- (2)  $X_{\text{old}} \leftarrow \mathbf{0}$ ;  $X_{\text{new}} \leftarrow \mathbf{0}$ ;
- (3) Select  $\lambda_1 > \lambda_2 > \dots > \lambda_K = \bar{\lambda}$
- (4) **for**  $\lambda = \lambda_1, \lambda_2, \dots, \lambda_K$  **do**
- (5)  $t_{\text{old}} \leftarrow 1$ ;  $t_{\text{new}} \leftarrow 1$ ;
- (6) **repeat**
- (7)  $Z \leftarrow X_{\text{new}} + \frac{t_{\text{old}} - 1}{t_{\text{new}}} (X_{\text{new}} - X_{\text{old}})$ ;
- (8)  $X_{\text{old}} \leftarrow X_{\text{new}}$ ;
- (9)  $G \leftarrow Z - L^{-1}(M \circ Z - Y + 2\mu D_2^T D_2 Z)$ ;
- (10)  $X_{\text{new}} \leftarrow S_{\lambda L^{-1}}(G)$ ;
- (11)  $t_{\text{old}} \leftarrow t_{\text{new}}$ ;
- (12)  $t_{\text{new}} \leftarrow \frac{1 + \sqrt{1 + 4t_{\text{old}}^2}}{2}$ ;
- (13) **until**  $\|X_{\text{new}} - X_{\text{old}}\|_F / \|X_{\text{old}}\|_F < \epsilon$
- (14) **end for**
- (15)  $\hat{X} \leftarrow X_{\text{new}}$ ;
- (16) **return**  $\hat{X}$

ALGORITHM 1: DRMC algorithm.

exist low-rank matrices that exactly satisfy the constraints in problem (13). So, we converted the constrained optimization problem (13) into the following nuclear norm regularized optimization problem:

$$\text{minimize} \quad \frac{1}{2} \|M \circ X - Y\|_F^2 + \lambda \|X\|_*, \quad (14)$$

where  $\lambda > 0$  is the nuclear regularization parameter.

**5.2. Utilization of DCT Compactness.** Though we can estimate  $X$  by solving the optimization problem (14), it will overfit to the known entries of  $X$  when the sampling rate is low, which will lead to large recovery error in the estimation of the missing entries.

Therefore, to reduce the overfitting in (14), we exploit the DCT compactness feature of the sensor data. As mentioned in Section 4.3,  $\|D_2 X\|_F / \|X\|_F \approx 0$ . So, we added  $\|D_2 X\|_F$  as the DCT regularization term to (14), and finally, we obtain the following optimization problem:

$$\text{minimize} \quad \frac{1}{2} \|M \circ X - Y\|_F^2 + \lambda \|X\|_* + \mu \|D_2 X\|_F^2, \quad (15)$$

where  $\mu$  is the DCT regularization parameter.

**5.3. The DRMC Algorithm.** We present the DRMC algorithm by solving the optimization problem in (15). The pseudocode is shown in Algorithm 1. Next, we will describe the design of DRMC in details.

The object function in (15) can be rewritten into the following form:

$$F(X) := f(X) + P(X), \quad (16)$$

with

$$f(X) = \frac{1}{2} \|M \circ X - Y\|_F^2 + \mu \|D_2 X\|_F^2, \quad (17)$$

$$P(X) = \lambda \|X\|_*. \quad (18)$$

Note that  $P(X)$  is a proper, convex, lower semicontinuous (lsc) [19] function but it is nonsmooth, while  $f(X)$  is a convex smooth function and is continuously differentiable, with

$$\nabla f(X) = (M \circ X - Y) + 2\mu D_2^T D_2 X. \quad (19)$$

Furthermore,  $\nabla f(X)$  is Lipschitz continuous with a positive constant  $L$ :

$$\|\nabla f(X_1) - \nabla f(X_2)\|_F \leq L \|X_1 - X_2\|_F, \quad (20)$$

$$\forall X_1, X_2 \in \text{dom } P,$$

where  $\text{dom } P = \{X \mid P(X) < \infty\}$ . Proposition 2 indicates that the Lipschitz constant of  $f(X)$  is  $L = 1 + 2\mu$ .

Since  $P(X)$  is nonsmooth, it is difficult to directly minimize the objective function  $F(X)$ . Instead, we choose to iteratively minimize a sequence of quadratic approximations of  $F(X)$ , which is an effective way to minimize the unconstrained nonsmooth convex function [20–22]. The quadratic approximation of  $F(\cdot)$  at point  $Z$  is defined as the following:

$$Q_L(X, Z) := f(Z) + \langle \nabla f(Z), X - Z \rangle + \frac{L}{2} \|X - Z\|_F^2 + P(X). \quad (21)$$

And the objective variable  $X_{\text{new}}$  is repeatedly updated to the minimizer of  $Q_L(X, Z)$ , until  $\|X_{\text{new}} - X_{\text{old}}\|_F / \|X_{\text{old}}\|_F < \epsilon$ . The convergence of such iterative process is well studied in [21].

We then introduce an auxiliary variable  $G$  to minimize  $Q_L(X, Z)$ . As suggested by Proposition 5, we can minimize  $Q_L(X, Z)$  using the singular value shrinkage operator defined in (26). Thus, we have  $X_{\text{new}} = S_{\lambda L^{-1}}(G)$ .

What is more, we consider a warm-start technique for the nuclear regularization parameter  $\lambda$ . Rather than remaining unchanged,  $\lambda$  is monotonically decreasing in the iterative process. The nuclear regularization parameter  $\lambda$  starts with an initial value  $\lambda_1$  and gradually declines to  $\bar{\lambda}$ , forming a sequence of  $\lambda_1 > \lambda_2 > \dots > \lambda_K$ , ( $\lambda_K = \bar{\lambda}$ ).

**Lemma 1.**  $D_2$  is defined as in (10). Then, one has

$$\|D_2^T D_2 X\|_F \leq \|X\|_F, \quad \forall X \in \mathbb{R}^{m \times n}. \quad (22)$$

*Proof.* Since  $D$  is orthonormal,  $D_2 D_2^T = I$ ,  $\|DX\|_F = \|X\|_F$ .

And note that  $\|X\|_F = \sqrt{\text{Tr}(X^T X)}$ , then we can obtain

$$\begin{aligned} \|D_2^T D_2 X\|_F &= \sqrt{\text{Tr}(X^T D_2^T D_2 D_2^T D_2 X)} \\ &= \sqrt{\text{Tr}(X^T D_2^T D_2 X)} = \|D_2 X\|_F \\ &\leq \|DX\|_F = \|X\|_F. \end{aligned} \quad (23)$$

□

**Proposition 2.** Assume that  $f(X)$  is defined as (17); then,  $f(X)$  is Lipschitz continuous with

$$L = 1 + 2\mu. \quad (24)$$

*Proof.* Note that  $\|M \circ X\|_F \leq \|X\|_F$ , and by using the Lemma 1, we obtain that

$$\begin{aligned} \|\nabla f(X_1) - \nabla f(X_2)\|_F &= \|M \circ (X_1 - X_2) + 2\mu D_2^T D_2 (X_1 - X_2)\|_F \\ &\leq \|M \circ (X_1 - X_2)\|_F + 2\mu \|D_2^T D_2 (X_1 - X_2)\|_F \\ &\leq \|X_1 - X_2\|_F + 2\mu \|X_1 - X_2\|_F \\ &= (1 + 2\mu) \|X_1 - X_2\|_F. \end{aligned} \quad (25)$$

□

**Definition 3.** Decompose the matrix  $X \in \mathbb{R}^{m \times n}$  of rank  $r$  by SVD:  $X = U \Sigma V^T$ , where  $U \in \mathbb{R}^{m \times r}$  and  $V \in \mathbb{R}^{n \times r}$  are orthonormal matrices and  $\Sigma = \text{diag}(\{\sigma_i\}_{1 \leq i \leq r})$ . Define the singular value shrinkage operator [23]  $S_\lambda$  as follows:

$$\begin{aligned} S_\lambda(X) &= U S_\lambda(\Sigma) V^T, \\ S_\lambda(\Sigma) &= \text{diag}(\{(\sigma_i - \lambda)_+\}_{1 \leq i \leq r}), \end{aligned} \quad (26)$$

where  $t_+$  is the positive part of  $t$ ,  $t_+ = \max(0, t)$ .

**Lemma 4.** Let  $G \in \mathbb{R}^{m \times n}$ . Then,

$$S_\lambda(G) \equiv \arg \min_{X \in \mathbb{R}^{m \times n}} \left\{ \frac{1}{2} \|X - G\|_F^2 + \lambda \|X\|_* \right\}, \quad (27)$$

where  $S_\lambda(G)$  is the singular value shrinkage operator of  $G$ .

*Proof.* The proof of Lemma 4 can be found in [23]. □

**Proposition 5.** Let  $Z \in \mathbb{R}^{m \times n}$  and  $P(X) = \lambda \|X\|_*$ . Assume that  $f(X)$  is Lipschitz continuous and define  $G \in \mathbb{R}^{m \times n}$  with

$$G = Z - L^{-1} \nabla f(Z). \quad (28)$$

Then, one has

$$\arg \min_{X \in \mathbb{R}^{m \times n}} Q_L(X, Z) = S_{\lambda L^{-1}}(G). \quad (29)$$

*Proof.* Consider

$$\begin{aligned} Q_L(X, Z) &= f(Z) + \langle \nabla f(Z), X - Z \rangle + \frac{L}{2} \|X - Z\|_F^2 \\ &\quad + P(X) \\ &= f(Z) + \langle \nabla f(Z), X - Z \rangle + \frac{L}{2} \|X - Z\|_F^2 \\ &\quad + \lambda \|X\|_* \end{aligned}$$



$$\begin{aligned}
&= \frac{L}{2} \left\| X - Z - \frac{1}{L} \nabla f(Z) \right\|_F^2 + \lambda \|X\|_* \\
&\quad + f(Z) - \frac{1}{2L} \|\nabla f(Z)\|_F^2 \\
&= L \left\{ \frac{1}{2} \|X - G\|_F^2 + \frac{\lambda}{L} \|X\|_* \right\} + f(Z) \\
&\quad - \frac{1}{2L} \|\nabla f(Z)\|_F^2.
\end{aligned} \tag{30}$$

Thus, combined with Lemma 4, we can obtain that

$$\begin{aligned}
&\arg \min_{X \in \mathbb{R}^{m \times n}} Q_L(X, Z) \\
&= \arg \min_{X \in \mathbb{R}^{m \times n}} \left\{ L \left( \frac{1}{2} \|X - G\|_F^2 + \frac{\lambda}{L} \|X\|_* \right) + f(Z) \right. \\
&\quad \left. - \frac{1}{2L} \|\nabla f(Z)\|_F^2 \right\} = \arg \min_{X \in \mathbb{R}^{m \times n}} \left\{ \frac{1}{2} \|X - G\|_F^2 \right. \\
&\quad \left. + \frac{\lambda}{L} \|X\|_* \right\} = S_{\lambda L^{-1}}(G).
\end{aligned} \tag{31}$$

□

## 6. Complexity Analysis

In this section, we discuss the time complexity of the proposed algorithm.

After analyzing the steps in Algorithm 1, we find that the most computationally intensive step is the singular value shrinkage operation that performs SVD on  $G$ , which dominates the computational complexity of this algorithm. The time complexity of SVD for an  $m \times n$  matrix is  $O(mn^2)$  [24]. And for any  $\epsilon > 0$ , the iterative process in Algorithm 1 will terminate in  $O(\sqrt{L}/\epsilon)$  iterations with an  $\epsilon$ -optimal solution [21, 25]. Consequently, the time complexity of our algorithm is  $O(K\sqrt{L}/\epsilon mn^2)$ .

We can further decrease the time complexity of the DRMC algorithm by applying Partial Reorthogonalization Package (PROPACK) [26] in the singular value shrinkage operation. PROPACK uses the Lanczos method [24] to compute only a partial SVD of  $G$ . However, it cannot a priori compute singular values that are greater than  $\lambda/L$ . Hence, we need to predetermine the number of singular values to be computed (denoted as  $sv_i$ ) at the beginning of the  $i$ th iteration, and PROPACK can then compute the  $sv_i$  largest singular values and corresponding singular vectors. We adopt the prediction rule proposed in [27]:

$$sv_{i+1} = \begin{cases} sv_i + 1, & \text{if } sv_i p_i < sv_i. \\ \min\{sv_i + 10, m, n\}, & \text{if } sv_i p_i = sv_i, \end{cases} \tag{32}$$

where  $sv_i$  is the predicted number of singular values,  $sv_i p_i$  is the actual number of singular values that are larger than  $\lambda/L$ , and  $sv_0 = 10$ .

The time complexity of the Lanczos method is  $O(rmn)$  for  $m \times n$  matrix with rank of  $r$  [24]. Therefore, the time complexity of DRMC algorithm is  $O(K\sqrt{L}/\epsilon rmn)$  if PROPACK is used. For fixed number of iterations, the complexity of DRMC can be simplified as  $O(rmn)$ , while the state-of-the-art matrix completion based methods [11–13] require a complexity of  $O(r^2 mn)$ . So, DRMC is more computationally efficient.

## 7. Error Analysis

In this section, we analyze the recovery error of the DRMC algorithm and present a theoretical upper bound.

Before starting the analysis, we first introduce some assumptions and lemmas.

**Assumption 6.** The original data matrix  $X$  can be approximated by the first  $k$  DCT coefficients, with approximation error  $\xi$ :

$$\xi = \|X - D_1^T D_1 X\|_F = \|D_2^T D_2 X\|_F. \tag{33}$$

**Assumption 7.** Let  $\mathcal{A} : \mathbb{R}^{m \times n} \rightarrow \mathbb{R}^{m \times n}$  be the linear operator defined as

$$\mathcal{A}(X) = 2\mu L^{-1} D_2^T D_2 X + L^{-1} M \circ X \tag{34}$$

and  $\mathcal{I}$  be the identity operator.

Then, there exists a constant  $0 \leq \eta < 1$ , such that

$$\sup_{\mathcal{A}(X) \neq 0, \|X\|_F = 1} \|(\mathcal{I} - \mathcal{A})(X)\|_F \leq \eta. \tag{35}$$

**Lemma 8.** Suppose that  $S_\lambda$  is the singular value shrinkage operator; then,

$$\begin{aligned}
&\|S_\lambda(X_1) - S_\lambda(X_2)\|_F \leq \|X_1 - X_2\|_F, \\
&\forall X_1, X_2 \in \mathbb{R}^{m \times n}.
\end{aligned} \tag{36}$$

*Proof.* The detailed proof of Lemma 8 can be found in [28]. □

**Lemma 9.** Suppose that  $X \in \mathbb{R}^{m \times n}$  with rank of  $r$ ; then,

$$\|S_\lambda(X) - X\|_F \leq r\lambda. \tag{37}$$

*Proof.* Consider

$$\begin{aligned}
\|S_\lambda(X) - X\|_F &= \|U(S_\lambda(\Sigma) - \Sigma)V^T\|_F \\
&= \|U(\Sigma - S_\lambda(\Sigma))V^T\|_F \\
&= \|U(\text{diag}(\{\sigma_i - (\sigma_i - \lambda)_+\}_{1 \leq i \leq r}))V^T\|_F \\
&= \|U \text{diag}(\{d_i\}_{1 \leq i \leq r})V^T\|_F = \sqrt{\sum_{i=1}^r d_i^2},
\end{aligned} \tag{38}$$

where

$$d_i = \begin{cases} \lambda, & \text{if } \sigma_i \geq \lambda; \\ \sigma_i, & \text{if } \sigma_i < \lambda. \end{cases} \tag{39}$$

Therefore, we have

$$\|S_{\lambda}(X) - X\|_F \leq r\lambda. \quad (40)$$

□

The upper bound of the recovery error is then given in the following theorem.

**Theorem 10.** Suppose that  $\widehat{X}$  is the estimate of  $X$  obtained by the DRMC algorithm; then, one has

$$\|\widehat{X} - X\|_F \leq \frac{r\bar{\lambda} + 2\xi\mu}{(1-\eta)(1+2\mu)}. \quad (41)$$

*Proof.* When the iterative process in Algorithm 1 has converged,  $X_{\text{new}}$  will be equal to  $X_{\text{old}}$ . Hence,  $\widehat{X}$  will be the fixed point of the following:

$$\begin{aligned} \widehat{X} \\ = S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) \widehat{X} - L^{-1} M \circ (\widehat{X} - X) \right]. \end{aligned} \quad (42)$$

Therefore, we have

$$\begin{aligned} \|\widehat{X} - X\|_F &= \|S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) \widehat{X} - L^{-1} M \right. \\ &\quad \left. \circ (\widehat{X} - X) \right] - X\|_F = \|S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) \widehat{X} \right. \\ &\quad \left. - L^{-1} M \circ (\widehat{X} - X) \right] \\ &\quad - S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) X \right] \\ &\quad + S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) X \right] - X\|_F \\ &\leq \|S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) \widehat{X} - L^{-1} M \right. \\ &\quad \left. \circ (\widehat{X} - X) \right] - S_{\bar{\lambda}L^{-1}} \left[ (I - 2\mu L^{-1} D_2^T D_2) X \right]\|_F \\ &\quad + \|S_{\bar{\lambda}L^{-1}}(X) - X\|_F + 2\mu L^{-1} \|D_2^T D_2 X\|_F. \end{aligned} \quad (43)$$

Combining (43) and Lemma 8, we have

$$\begin{aligned} \|\widehat{X} - X\|_F \\ \leq \|(I - 2\mu L^{-1} D_2^T D_2) (\widehat{X} - X) - L^{-1} M \circ (\widehat{X} - X)\|_F \\ + \|S_{\bar{\lambda}L^{-1}}(X) - X\|_F + 2\mu L^{-1} \|D_2^T D_2 X\|_F \\ = \|(\mathcal{I} - \mathcal{A})(\widehat{X} - X)\|_F + \|S_{\bar{\lambda}L^{-1}}(X) - X\|_F \\ + 2\mu L^{-1} \|D_2^T D_2 X\|_F, \end{aligned} \quad (44)$$

where  $\mathcal{I}$  is the identity operator and  $\mathcal{A}$  is the operator defined in (34).

Then, applying Assumption 6 and Lemma 9 to (44), we have

$$\|\widehat{X} - X\|_F \leq \|(\mathcal{I} - \mathcal{A})(\widehat{X} - X)\|_F + \bar{\lambda}L^{-1}r + 2\mu L^{-1}\xi. \quad (45)$$

TABLE 2: Parameter settings for DRMC.

Parameter name	$\bar{\lambda}$	$\mu$	$k$
Default value	0.001	1	10

Combining Assumption 7, (24), and (45), we finally obtain the following error bound:

$$\|\widehat{X} - X\|_F \leq \frac{r\bar{\lambda} + 2\xi\mu}{(1-\eta)L} = \frac{r\bar{\lambda} + 2\xi\mu}{(1-\eta)(1+2\mu)}. \quad (46)$$

□

Let  $E$  represent the upper bound of the recovery error. Then, according to Theorem 10,  $E = (r\bar{\lambda} + 2\xi\mu)/(1-\eta)(1+2\mu)$ . Note that  $E(\bar{\lambda})$  is an increasing function of  $\bar{\lambda}$ . So, we expect the actual recovery error of the DRMC algorithm to increase with  $\bar{\lambda}$ , which is confirmed later by simulation results in Section 8.3.

## 8. Evaluation

We designed a data gathering scheme based on the proposed DRMC algorithm. The data gathering procedure is similar to [11]. Firstly, sink node broadcasts a sampling rate to all sensor nodes. Secondly, each sensor node randomly and independently decides whether to forward its readings to the sink according to the sampling rate. Finally, the sink node collects the incomplete data matrix and uses DRMC to retrieve the missing data. After implementing this data gathering scheme by Matlab, we carried out extensive experiments on three real-world datasets (as shown in Table 1) to evaluate the effectiveness of DRMC.

**8.1. Baseline Methods.** We select two state-of-the-art methods to compare with DRMC. The first method is Compressive Sensing (CS). We choose the DCT matrix defined in (9) to serve as the orthonormal basis in CS. The second method is Spatiotemporal Compressive Data Collection (STCDG). The parameters of STCDG are set to  $\lambda = 0.5$ ,  $r = 10$ . Note that since our earlier work, namely, DCT-RPMC, depends on historical data while the proposed algorithm does not, we do not select DCT-RPMC as the baseline method.

**8.2. Recovery Accuracy.** Firstly, we compared the recovery accuracy of the proposed algorithm with two baseline methods described above. The parameters of DRMC are listed in Table 2.

Simulation experiments are carried out on three real-world datasets. Each simulation is conducted for 100 independent trials. The recovery errors are computed according to (6) and are averaged over the 100 trials.

Comparison results are shown in Figures 3–5. For experiments on Intel Temperature Trace, all methods achieve nearly the same recovery accuracy when the sampling rate is high. When the sampling rate is below a certain value ( $\tau < 0.1$ ), recovery performance of baseline methods deteriorates

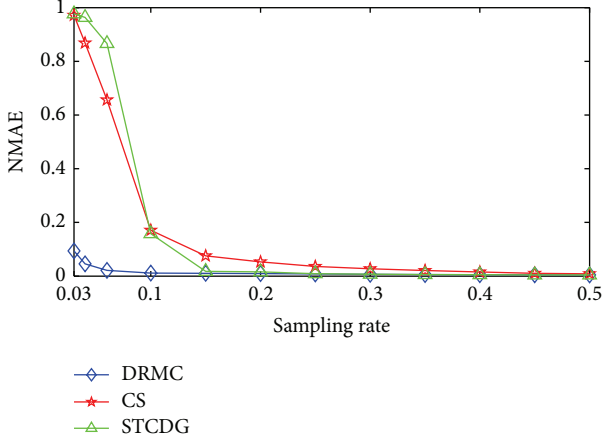


FIGURE 3: Recovery accuracy on Intel Temperature Trace.

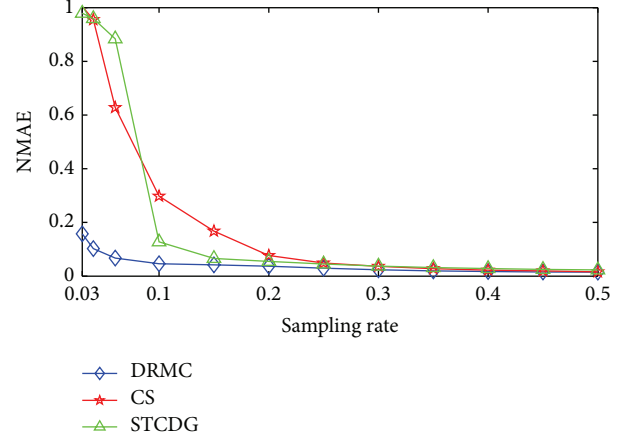


FIGURE 5: Recovery accuracy on PARED Temperature Trace.

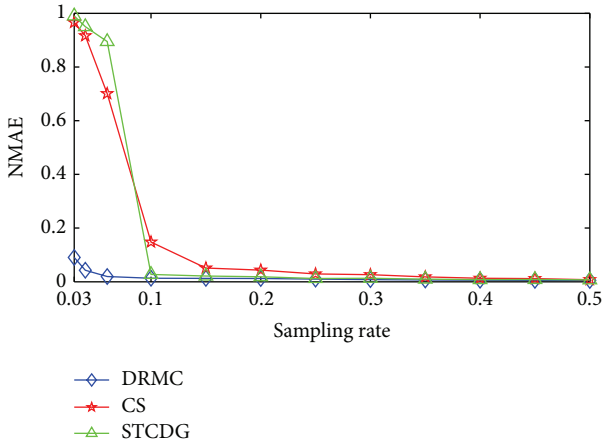
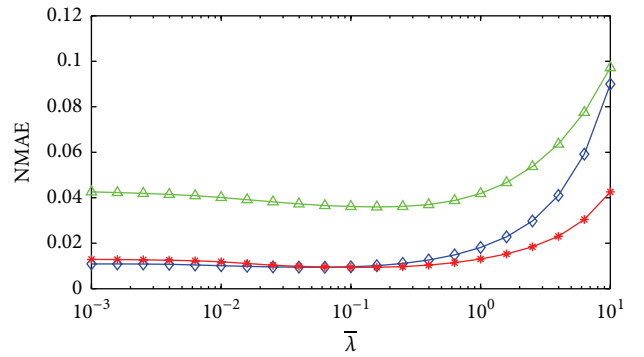


FIGURE 4: Recovery accuracy on Intel Humidity Trace.

FIGURE 6: Effect of parameter  $\bar{\lambda}$ .

quickly, while DRMC still achieves a good recovery accuracy. When the sampling rate is as low as 0.03, which means 97% of data loss, DRMC can reconstruct the lost data with recovery error less than 10%, while recovery error of CS and STCDG is close to 100%.

Comparison results on Intel Humidity Trace are very similar to that on Intel Temperature Trace. Recovery error of DRMC is about 9% when the sampling rate is 0.03, which is noticeably better than that of baseline methods.

For experiments on PARED Temperature Trace, DRMC still outperforms baseline methods for low sampling rate. The recovery error of DRMC is about 16% when the sampling rate is 0.03, which is slightly worse than that on other two traces. This is because the low-rankness and DCT compactness features of PARED Temperature are not as good as that of other two traces, as shown in Figures 1 and 2.

**8.3. Parameter Settings.** The DRMC algorithm depends on several input parameters. Clearly, the choice of these parameters will affect the recovery performance of DRMC. In this subsection, we discuss how to choose the parameters for DRMC.

The nuclear norm regularization parameter  $\lambda$  is an important parameter to ensure the low-rank feature of the reconstructed data matrix. In DRMC, we adopted a warm-start strategy for  $\lambda$ , in which  $\lambda$  is linearly reduced from  $\lambda_1$  to  $\lambda_K$ . In the implementation,  $\lambda_1 = 10\bar{\lambda} + 1$ ,  $\lambda_K = \bar{\lambda}$ , and  $K = 20$ . We tested DRMC on a range of values of  $\bar{\lambda}$  to investigate how  $\bar{\lambda}$  effects the performance of DRMC. Figure 6 shows the experimental results. The recovery errors of DRMC increase with  $\bar{\lambda}$ , just as what we predicted in Section 7 by the theoretical error bound (41). Note that when  $\bar{\lambda} < 0.1$ , the recovery performance of DRMC is not sensitive to  $\bar{\lambda}$ . So, in practice, we just set it to a small enough value,  $\bar{\lambda} = 0.001$ .

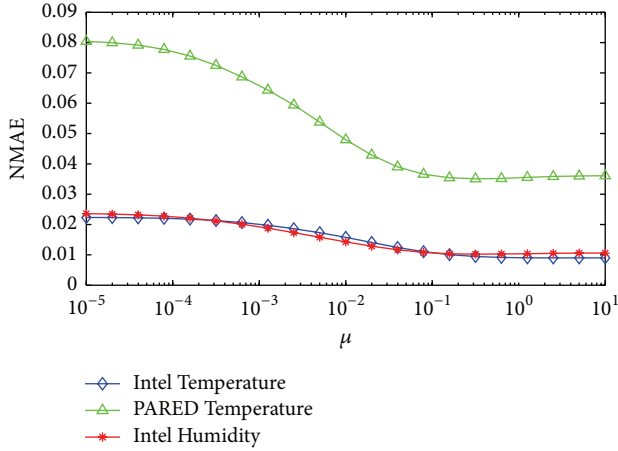
Parameter  $\mu$  is a regularization parameter that guarantees the DCT compactness feature of the recovered signal. Figure 7 shows the effects of  $\mu$  on the recovery performance of DRMC. The recovery errors decline with  $\mu$  and are stable when  $\mu > 0.1$ . So, we choose to use  $\mu = 1$  for our experiments.

Recall that  $k$  represents the number of concentrated DCT coefficients. As discussed before in Section 4.3, the first 10 DCT coefficients concentrate 99% of the total energy. As expected, Figure 8 shows that the recovery errors are



TABLE 3: Simulation configuration for network lifespan.

Parameter name	Value
Number of nodes	1000
Initial energy	1J
Sampling period	10 seconds
Data size	16 bits
$E_{Tx}$	100 nJ/bit
$E_{Rx}$	120 nJ/bit
$E_{Amp}$	0.01 nJ/(bit·m <sup>2</sup> )

FIGURE 7: Effect of parameter  $\mu$ .

dropping fast with  $k$  and are stable when  $k > 10$ . So, we choose to use  $k = 10$  for our experiments. Note that the recovery errors are slightly increasing with the growth of  $k$  when  $k > 10$ . We explain this by considering the extreme case when  $k = m$ . If  $k = m$ ,  $D_1$  is equal to  $D$ . As a result, the DCT regularization term  $\|D_2 X\|_F^2$  in (15) is automatically equal to zero. Therefore, the DCT compactness property of the sensory data is not utilized and the recovery errors will increase, just as the case in Figure 7 when  $\mu = 0$ .

**8.4. Energy Consumption and Network Lifespan.** DRMC-based data gathering protocol is more energy efficient because it transmits less packets than the classic one (receiving and forwarding). As a result, DRMC-based protocol can save more energy and prolong the lifespan of wireless sensor networks. To verify this, simulation experiments are conducted and the simulation configuration is shown in Table 3.

In the simulation, sensor nodes are randomly deployed in a 500 m × 500 m area and the sink node is deployed in the center. Each sensor node is equipped with 1J energy. To evaluate the energy consumption, we adopt the following energy model [29]:

$$E_T(k, d) = \begin{cases} (E_{Tx} + d^2 \times E_{Amp}) \times k, & \text{if } d < d_{Thres}, \\ (E_{Tx} + d^4 \times E_{Amp}) \times k, & \text{if } d \geq d_{Thres}, \end{cases} \quad (47)$$

$$E_R(k) = kE_{Rx}.$$

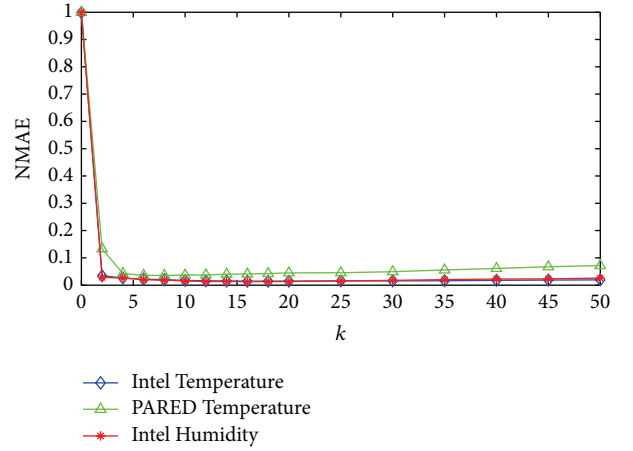
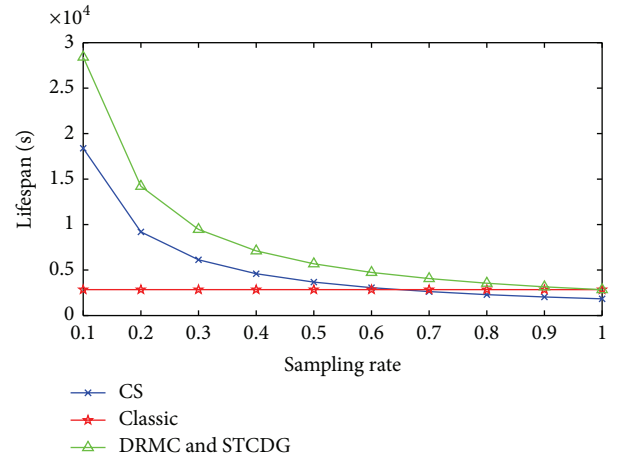
FIGURE 8: Effect of parameter  $k$ .

FIGURE 9: Relation of network lifespan to sampling rate.

$E_T(k, d)$  represents the energy consumption of transmitting  $k$  bits data with distance  $d$ .  $E_R(k)$  denotes the energy consumption of receiving  $k$  bits data.  $E_{Tx}$  is the energy consumed by the transmitting circuit to process 1 bit data.  $E_{Rx}$  is the energy consumed by the receiving circuit to process 1 bit data.  $E_{Amp}$  is the energy consumed by power amplifying circuit.

Figure 9 demonstrates the network lifespan of DRMC-based protocol and other baseline protocols under different sampling rate. Note that the network lifespan is defined as the time when the first energy exhausted node appears. Apparently, the sampling rate does not play a role in the classic data gathering, since it directly transmits all data without compression. Therefore, the lifespan curve of the classic protocol in Figure 9 is a straight line. For the CS method, the smaller the sampling rate is, the less measurements are taken. And as shown in Figure 9, the lifespan of CS is decreasing with the sampling rate. However, when the sampling rate is above a certain value, the lifespan of CS is even worse than the classic one. The reason why CS performs badly for large sampling rate is well analyzed in [30].

Figure 9 shows that DRMC-based protocol achieves the best lifespan. Similarly, the lifespan of DRMC is decreasing with the sampling rate. When the sampling rate decreases to 1, DRMC-based protocol is equivalent to the classic protocol and the lifespan of DRMC is equal to the classic one. Note that the lifespan of STCDG is exactly the same as DRMC, because both of the two methods are based on matrix completion. The lifespan of DRMC is longer than CS because the sampling matrix in DRMC is much sparser than that in CS.

## 9. Conclusion

In this paper, we studied the data gathering and reconstruction problem in WSNs. We modeled the problem as matrix completion problem and investigated the data features in real WSN datasets. Then, by taking advantage of the low-rankness and DCT compactness features in WSNs, we proposed a DCT Regularized Matrix Completion (DRMC) algorithm to reconstruct the missing data. The recovery error of DRMC is carefully analyzed and a theoretical error upper bound is presented. Experimental results show that DRMC outperforms state-of-the-art methods for low sampling rate and achieves a good recovery accuracy even if the sampling rate is very low.

## Summary of Notations

$n$ :	Number of sensor nodes
$m$ :	Number of samples in a sampling period
$k$ :	Number of dominant DCT coefficients
$\lambda$ :	Nuclear norm regularization parameter
$\bar{\lambda}$ :	Final nuclear norm regularization parameter
$\mu$ :	DCT regularization parameter
$\tau$ :	Sampling rate
$\xi$ :	Approximation error of dominant DCT coefficients
$X$ :	Environment matrix
$\hat{X}$ :	Estimated environment matrix
$M$ :	Sampling matrix
$Y$ :	Data matrix collected by the sink
$D$ :	DCT matrix
$D_1$ :	Dominant DCT matrix
$D_2$ :	Subdominant DCT matrix.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgment

This work is funded by the National Nature Science Foundation of China under Grant no. 61371135.

## References

- [1] A. Baggio, "Wireless sensor networks in precision agriculture," in *Proceedings of the ACM Workshop on Real-World Wireless Sensor Networks (REALWSN '05)*, Stockholm, Sweden, 2005.
- [2] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: issues and an implementation," *Computer Communications*, vol. 29, no. 13-14, pp. 2521–2533, 2006.
- [3] L. Mo, Y. He, Y. Liu et al., "Canopy closure estimates with greenorbs: sustainable sensing in the forest," in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pp. 99–112, ACM, November 2009.
- [4] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TinyDB: an acquisitional query processing system for sensor networks," *ACM Transactions on Database Systems*, vol. 30, no. 1, pp. 122–173, 2005.
- [5] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies?" *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.
- [6] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.
- [7] J. Haupt, W. U. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 92–101, 2008.
- [8] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Efficient measurement generation and pervasive sparsity for compressive data gathering," *IEEE Transactions on Wireless Communications*, vol. 9, no. 12, pp. 3728–3738, 2010.
- [9] G. Quer, R. Masiero, G. Pillonetto, M. Rossi, and M. Zorzi, "Sensing, compression, and recovery for WSNs: sparse signal modeling and monitoring framework," *IEEE Transactions on Wireless Communications*, vol. 11, no. 10, pp. 3447–3461, 2012.
- [10] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational Mathematics*, vol. 9, no. 6, pp. 717–772, 2009.
- [11] J. Cheng, Q. Ye, H. Jiang, D. Wang, and C. Wang, "STCDG: an efficient data gathering algorithm based on matrix completion for wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 2, pp. 850–861, 2013.
- [12] L. Kong, M. Xia, X.-Y. Liu et al., "Data loss and reconstruction in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 11, pp. 2818–2828, 2014.
- [13] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 267–278, 2009.
- [14] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, pp. 145–156, ACM, September 2009.
- [15] D. Ebrahimi and C. Assi, "A distributed method for compressive data gathering in wireless sensor networks," *IEEE Communications Letters*, vol. 18, no. 4, pp. 624–627, 2014.
- [16] K. Yi, J. Wan, L. Yao, and T. Bao, "Partial matrix completion algorithm for efficient data gathering in wireless sensor networks," *IEEE Communications Letters*, vol. 19, no. 1, pp. 54–57, 2015.
- [17] Intel Lab Data, <http://db.lcs.mit.edu/labdata/labdata.html>.
- [18] K. Yi, R. Feng, N. Yu, and P. Chen, "PARED: a testbed with parallel reprogramming and multi-channel debugging for WSNs," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '13)*, pp. 4630–4635, IEEE, Shanghai, China, April 2013.
- [19] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2009.

- [20] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.
- [21] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pacific Journal of Optimization*, vol. 6, no. 3, pp. 615–640, 2010.
- [22] S. Ji and J. Ye, "An accelerated gradient method for trace norm minimization," in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML '09)*, pp. 457–464, ACM, June 2009.
- [23] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [24] G. H. Golub and C. F. van Loan, *Matrix Computations*, vol. 3, JHU Press, 2012.
- [25] P. Tseng, "On accelerated proximal gradient methods for convex-concave optimization," submitted to *The SIAM Journal on Optimization*.
- [26] R. M. Larsen, "Propack-software for large and sparse SVD calculations," 2004, <http://sun.stanford.edu/~rmunk/PROPACK/>.
- [27] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," <http://arxiv.org/abs/1009.5055>.
- [28] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Mathematical Programming*, vol. 128, no. 1-2, pp. 321–353, 2011.
- [29] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on Wireless Communications*, vol. 1, no. 4, pp. 660–670, 2002.
- [30] J. Luo, L. Xiang, and C. Rosenberg, "Does compressed sensing improve the throughput of wireless sensor networks?" in *Proceedings of the IEEE International Conference on Communications (ICC '10)*, pp. 1–6, IEEE, May 2010.

