On Balancing Energy Efficiency and Estimation Error in Compressed Sensing

Donglin Hu†, Shiwen Mao†, and Nedret Billor‡
†Department of Electrical and Computer Engineering, Auburn University, Auburn, AL, USA
‡Department of Mathematics and Statistics, Auburn University, Auburn, AL, USA
Email: dzh0003@tigermail.auburn.edu, smao@ieee.org, billone@auburn.edu

***Abstract*—Compressed sensing (CS) refers to the process of reconstructing a signal that is supposed to be sparse or compressible. CS has wide applications, such as in cognitive radio networks. In this paper, we investigate effective CS schemes for balancing energy efficiency and estimation error. We propose an enhancement to a Bayesian estimation approach and an enhancement to the isotonic regression approach that is based on nearly isotonic regression. We also show how to compute the routing matrix for selecting active sensor nodes. The proposed enhancements are evaluated with trace-driven simulations. Considerable gaps are observed between the original approaches and the proposed enhancements in the simulation results. The near isotonic regression method achieves the best performance among all the CS schemes examined in this paper.**

I. Introduction

Compressed sensing (CS) refers to the process of reconstructing a signal that is supposed to be sparse or compressible [1]. It has found wide applications in communications and networking. For example, in cognitive radio networks, spectrum sensing is a critical component in dynamic spectrum access and enforcement of spectrum usage and sharing [2]. Given the wide range of activities in space, time, and frequency, it would be extremely challenging and costly to have a full range and dense sampling. In such situations, compressed sensing becomes a powerful tool for efficient spectrum sensing. Generally, a spatially distributed wireless sensor network (WSN) is used to monitor physical or environmental conditions. The sparse signals are obtained by collecting readings from sensor nodes by a server through wireless transmissions. The server will then use CS to process the sparse data to achieve certain design goals.

CS has received considerable interest from the wireless community recently. For example, several CS papers have focused on network optimization and scheduling, where two important factors, *network performance* and *power consumption*, are considered in the design of CS. The objectives of the schemes presented in these papers are to prolong the life time of wireless sensor nodes, while meeting certain network performance requirements. Energy can be conserved by turning off some sensor nodes. Network performance can be measured in terms of surveillance quality such as the number of active nodes [3], [4], network coverage [5], minimum degree of connection [6], [7], and surveillance delay [8], [9]. The proposed schemes usually find a trade-off between network performance and power consumption.

Other CS papers have laid emphasis on signal compression and reconstruction. These papers seek a trade-off between *compression efficiency* and *reconstruction quality*. For instance, the conventional CS approach is based on orthonormal basis. An important example of this approach is wavelet transform. The original sensing data is compressed and delivered, therefore less network resource will be needed for transmitting the compressed data [10]–[12].

In this paper, we investigate effective CS schemes for balancing *energy efficiency* and *estimation error*. As discussed, a compressed signal can be obtained by collecting readings from a subset of the deployed sensor nodes, so that the rest of the nodes can be turned off to save energy. At the server, the intact signal is reconstructed via signal estimation. Therefore, there is a fundamental trade-off between how many active nodes to choose (thus how much energy to spend) and the corresponding accuracy of prediction. Recently, only a few papers have investigated the problem of joint *network optimization* and *signal processing*. In [13], sensor nodes are divided into subgroups and all the readings can be recovered from one of the subgroups using isotonic regression. The objective is to maximize the number of subgroups while keeping the estimation error below a tolerable threshold. In [14], nodes are grouped into pairs. In each pair, the node with lower battery capacity goes to sleep, while the node with higher battery capacity estimates the measurement of the sleeping node using linear regression. In [15], [16], the authors investigate the problem of reconstructing a distributed signal through the collection of a small number of sensor readings, where CS is jointly applied with principle component analysis. The data of the entire network is recovered with Bayesian estimation.

In particular, we aim to develop effective CS schemes that balance the fundamental trade-off between energy saving and recovery accuracy, which are in terms of the number of active sensor nodes and the corresponding estimation error. We examine three classes of CS approaches in related work discussed above. We first implement the Bayesian estimation approach presented in [15], and propose an enhancement to this approach by scaling the variance of the sensor readings. We next revisit the isotonic regression approach presented in [13], and propose an improved method based on nearly isotonic regression. Finally, we briefly introduce two polynomial regression approaches, i.e., linear regression and quadratic regression, which are used as benchmarks in the performance evaluation. Bayesian estimation selects active nodes randomly, while isotonic regression chooses active nodes based on estimation errors, which results in better reconstruction quality. Unlike LR and QR, isotonic regression considers the distribution of readings for more accurate estimation. The proposed

978-1-4673-0921-9/12/$31.00 ©2012 IEEE

enhancements are evaluated with trace-driven simulations. We find considerable gaps between the original approaches and the enhancements in the simulation results. We also find that the near isotonic regression method achieves the best performance among all the CS schemes examined in this paper.

The remainder of this paper is organized as follows. In Section II, we describe the system model. We introduce Masiero's method and its enhancement in Section III. In Section IV-A, we discuss Koushanfar's method and its enhancement. In Section V, we present two polynomial regression schemes. Performance evaluation is presented in Section VI. Section VII concludes this paper.

II. System Model and Assumptions

The network model considered in this paper consists of autonomous sensor nodes that are deployed in an area to monitor physical conditions (e.g., spectrum availability) and a data processing center (or, a server). We assume that the sensor nodes collect information data following a synchronized slot structure. Once a node obtains its data in a time slot, it transmits the data to the data processing center through its wireless interface in the same time slot.

We assume the processes of sensing and transmission consume certain amount of energy, denoted as E, which is a constant. If there are N sensor nodes deployed, the total amount of energy consumed in each time slot is $N \times E$. It is easy to see that a convenient way to conserve energy is to reduce the number of active nodes. On the other hand, the data from the idle nodes has to be recovered through CS. The estimation errors of the recovered data should meet the minimum precision requirement of the target sensing application. Therefore, *there is a fundamental trade-off between the number of active nodes and estimation error*.

In the following sections, we introduce three classes of CS approaches and propose enhanced algorithms. The first class is *Bayesian Estimation*, which focuses on energy saving by managing the number of active sensor nodes. Specifically, L active sensor nodes will be chosen from the set of N sensor nodes in each time slot. The second class is *Isotonic Regression* and the third class is *Polynomial Regression*, which guarantee the quality of recovered data by controlling the estimation error. With these techniques, we choose as few active sensor nodes as possible, while keeping the estimation error below a prescribed threshold.

III. Bayesian Estimation Approaches

In Bayesian Estimation (BE), the parameter set θ is obtained by maximizing the posterior probability density function (pdf)

$$p(\theta|\mathcal{D},\mathcal{M}) = \frac{p(\mathcal{D}|\theta,\mathcal{M})p(\theta|\mathcal{M})}{p(\mathcal{D}|\mathcal{M})}, \tag{1}$$

where $\mathcal{D}$ is the data set and $\mathcal{M}$ the plausible model. Since $p(\mathcal{D}|\mathcal{M})$ does not include θ, maximizing the posterior probability $p(\theta|\mathcal{D},\mathcal{M})$ is equivalent to maximizing the product of $p(\mathcal{D}|\theta,\mathcal{M})$ and $p(\theta|\mathcal{M})$.

In [16], Masiero et al. propose an approach in which *Principal Component Analysis* (PCA) and CS are jointly considered. In the following, we first introduce this method in Section III-A for the sake of completeness. We then propose an enhanced joint PCA and CS method in Section III-B.

A. Joint PCA and CS

In the network, the readings of sensor nodes are collected with a fixed sampling rate at discrete times $k = 1, 2, \cdots, K$. Let $\mathbf{x}_k \in \mathbb{R}^N$ be an $N \times 1$ vector that contains the measurements collected at time k. Then $\mathbf{x}_k$ can be considered as an N-dimensional stochastic process.

With PCA, the centered $\mathbf{x}_k$ can be projected to a new $N \times 1$ variable $\mathbf{s}_k$, which is written as

$$\mathbf{s}_k = \mathbf{U}^T \cdot (\mathbf{x}_k - \overline{\mathbf{x}}), \tag{2}$$

where $\overline{\mathbf{x}}$ is the mean of $\mathbf{x}$ and $\mathbf{U}$ is an orthonormal matrix whose column vectors are the eigenvectors of the covariance matrix of $\mathbf{x}_k$. The eigenvectors are placed in the decreasing order of the corresponding eigenvalues (λ_i). The mean vector and covariance matrix can be replaced by the sample mean vector and the sample covariance matrix as

$$\overline{\mathbf{x}} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{x}_k \ \text{ and } \ \Sigma = \frac{1}{K}\sum_{k=1}^{K}(\mathbf{x}_k - \overline{\mathbf{x}})(\mathbf{x}_k - \overline{\mathbf{x}})^T. \tag{3}$$

According to the properties of PCA, it can be shown that s_k^i, the i-th element of $\mathbf{s}_k$, is uncorrelated with s_k^j, i.e.,

$$\mathbb{E}[s_k^i s_k^j] = \mathbb{E}[s_k^i]\mathbb{E}[s_k^j] = 0, \ \ \forall i \neq j. \tag{4}$$

If the elements of $\mathbf{s}_k$ follow Gaussian distribution, they are independent of each other. The joint pdf of $\mathbf{s}_k$ can be decomposed into the product of the individual pdfs of its elements.

In CS, not all the elements of $\mathbf{x}_k$ can be obtained at time k because a portion of sensor nodes are turned off to reduce power consumption. Similarly, we only have partial elements of $\mathbf{s}_k$ from the projection of the partial elements of the centered $\mathbf{x}_k$. Assume there are L active nodes. Let $\mathbf{x}'_k$ be an $L \times 1$ vector that collects measurements from the L active nodes. The L active nodes are selected randomly from the N sensor nodes with an $L \times N$ *routing matrix* $\mathbf{\Phi}$, which is a binary matrix comprised of two values 0 and 1. Each row of $\mathbf{\Phi}$ has only a single 1 element whose position corresponds to the index of the selected sensor node.

The relationship between $\mathbf{x}'_k$ and $\mathbf{x}_k$ can be written as

$$\mathbf{x}'_k = \mathbf{\Phi} \cdot \mathbf{x}_k. \tag{5}$$

Let $\mathbf{z}_k = \mathbf{x}'_k - \mathbf{\Phi}\overline{\mathbf{x}}$. Recall that $\mathbf{x}'_k$ is a vector containing the measurements from active nodes and $\overline{\mathbf{x}}$ is the sample mean vector. Both of them are available at time k. Thus, $\mathbf{z}_k$ is a known data set at time k. By combining Equations (2) and (5), $\mathbf{z}_k$ can also be obtained from $\mathbf{s}_k$, as

$$\mathbf{z}_k = \mathbf{\Phi}(\mathbf{x}_k - \overline{\mathbf{x}}) = \mathbf{\Phi} \cdot \mathbf{U}\mathbf{s}_k. \tag{6}$$

The objective of this method is to estimate the parameter set $\mathbf{s}_k$ from data set $\mathbf{z}_k$. The estimate of the parameter set $\mathbf{s}_k$ is denoted as $\hat{\mathbf{s}}_k$. Once $\hat{\mathbf{s}}_k$ is obtained, the intact signal $\mathbf{x}_k$ can be recovered from $\hat{\mathbf{s}}_k$ according to (2), as

$$\hat{\mathbf{x}}_k = \overline{\mathbf{x}} + \mathbf{U} \cdot \hat{\mathbf{s}}_k, \tag{7}$$

where $\hat{\mathbf{x}}_k$ is the estimate of $\mathbf{x}_k$. Two types of distribution assumptions of $\mathbf{s}_k$ were made in [15], i.e., Gaussian Distribution and Laplace Distribution. They are good models to represent the principal components of typical sensing data. Interested readers are referred to [15] for details.

B. Improved PCA-CS

The major disadvantage of the above approach is that the difference in the variances of the principal components is not considered. It is assumed that the elements of $\mathbf{s}_k$ follow the same distribution with identical variance. However, the following proposition shows that the above assumption is generally not true.

Proposition 1. *For any given pair of eigenvalues $\lambda_i > \lambda_j$, the corresponding variance of s_k^i is greater than that of s_k^j, i.e., $\mathbb{E}[(s_k^i)^2] > \mathbb{E}[(s_k^j)^2]$ if $\lambda_i > \lambda_j$.*

Proof: From (2), it follows that

$$\begin{aligned}\mathbb{E}[\mathbf{s}_k\mathbf{s}_k^T] &= \mathbf{U}^T\mathbb{E}[(\mathbf{x}_k-\overline{\mathbf{x}})(\mathbf{x}_k-\overline{\mathbf{x}})^T]\mathbf{U} \\ &= \mathbf{U}^T\Sigma\mathbf{U} = \text{diag}(\lambda_1,\cdots,\lambda_N), \end{aligned} \quad (8)$$

where $\text{diag}(\lambda_1,\cdots,\lambda_N)$ is a diagonal matrix with diagonal elements $(\lambda_1,\cdots,\lambda_N)$. Then we have $\mathbb{E}[(s_k^i)^2] = \lambda_k$ and the statement follows. ∎

To remove this unrealistic assumption, the main idea of our improved method is to scale s_i^k according to its corresponding variance λ_i. We substitute $\mathbf{U}$ with another matrix $\mathbf{V} = \mathbf{U}\Lambda$ where $\Lambda = \text{diag}(1/\sqrt{\lambda_1}, 1/\sqrt{\lambda_2},\cdots,1/\sqrt{\lambda_N})$. Obviously, by definition, $\mathbf{V}$ is also an orthonormal matrix. Letting $\mathbf{d}_k \stackrel{def}{=} \mathbf{V}^T(\mathbf{x}_k-\overline{\mathbf{x}})$, it follows that

$$\begin{aligned}\mathbb{E}[\mathbf{d}_k\mathbf{d}_k^T] &= \mathbf{V}^T\mathbb{E}[(\mathbf{x}_k-\overline{\mathbf{x}})(\mathbf{x}_k-\overline{\mathbf{x}})^T]\mathbf{V} \\ &= \Lambda^T\mathbf{U}^T\Sigma\mathbf{U}\Lambda = \Lambda^T\text{diag}(\lambda_1,\cdots,\lambda_N)\Lambda = \mathbf{I}, \end{aligned} \quad (9)$$

where $\mathbf{I}$ is an $N \times N$ identity matrix. In our proposed algorithm, we replace matrix $\mathbf{U}$ with the new matrix $\mathbf{V}$. Then the elements of the new projected vector $\mathbf{d}_k$ will have identical variances.

The two distribution assumptions used in [15] can also be adopted for the enhanced method. In both models, the elements are assumed to follow the same distribution. Since the elements of $\mathbf{d}_k$ now have identical variance, it fits the model assumption much better than $\mathbf{s}_k$ and consequently, the accuracy of recovered data will be improved.

IV. Isotonic Regression Approaches

With Bayesian Estimation, we are able to manage L, the number of active nodes, and the percentage of power saved is computed as $(N-L)/N$. However, there is no consideration of the distance between the true measurements $\mathbf{x}_k$ and the recovered measurements $\hat{\mathbf{x}}_k$. In many applications, the quality of reconstruction is no less important than energy saving. We introduce *Isotonic Regression* (IR) in this section, which is able to address this problem.

Isotonic regression aims to find a weighted least-square fit $\boldsymbol{\beta} \in \mathbb{R}^n$ to a vector $\mathbf{y} \in \mathbb{R}^n$ with weight vector $\mathbf{g} \in \mathbb{R}^n$, subject to a set of monotonicity constraints. The IR problem is defined as follows.

$$\begin{aligned}\text{minimize:} \quad & \sum_{i=1}^{n} g_i\left(y_i-\beta_i\right)^2 \\ \text{subject to:} \quad & \beta_1 \le \cdots \le \beta_n, \end{aligned} \quad (10)$$

where y_i, g_i and β_i are the i-th element of $\mathbf{y}$, $\mathbf{g}$ and $\boldsymbol{\beta}$, respectively. The optimal solution $\hat{\boldsymbol{\beta}}$ is a function of $\mathbf{y}$.

This is motivated by the positive correlation among the readings of neighboring nodes. Suppose sensors s_X and s_Y are exposed to the same stimuli source. Intuitively, increasing the stimuli value may result in higher readings at both sensors. It is thus possible to predict the reading of s_Y from the reading of s_X, and vice versa.

A. Combinatorial Isotonic Regression

In [13], the authors propose a method, named *Combinatorial Isotonic Regression* (CIR), for solving the isotonic regression problem. A four-step method is presented. The first two steps are used to convert isotonic regression into a combinatorial problem. Dynamic programming is then used to solve the combinatorial problem.

Denote the time series of reading from two sensors as $\mathbf{X} = \{x_k\}$ and $\mathbf{Y} = \{y_k\}$, $k = 1, 2, \cdots, K$. CIR sorts the values of $\mathbf{X}$ and $\mathbf{Y}$ in strictly increasing order and eliminates redundancies by grouping the identical values. We then obtain $x_{(1)} < \cdots < x_{(i)} < \cdots < x_{(n)}$ and $y_{(1)} < \cdots < y_{(j)} < \cdots < y_{(m)}$. The objective is to find a mapping $\mathbf{X} \to \hat{\mathbf{Y}}$ by solving

$$\begin{aligned}\text{minimize:} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} g_{i,j}\left(y_{(j)}-\beta_i\right)^2 \\ \text{subject to:} \quad & \beta_1 \le \cdots \le \beta_n, \end{aligned} \quad (11)$$

where $g_{i,j}$ is the number of $\{x_{(i)}, y_{(j)}\}$ pairs of readings from sensor s_X and s_Y. Note that the L-1 norm is used in [13] and their method also works in the case of L-2 norm. To be consistent with the other methods, we adopt the L-2 norm in the remainder of this paper.

CIR consists of four steps, which are executed for each pair of sensor nodes. The algorithm is given below.

Step 1 Build the relative importance matrix $\mathbf{R}$, which captures the number of the pairs $\{x_{(i)}, y_{(j)}\}$.

Step 2 Build the error matrix $\mathbf{E}$ whose element $\{e_{i,j}\}$ describes the error that would be made if the prediction of $\hat{\mathbf{Y}}$ was chosen $y_{(j)}$ while the reading $x_{(i)}$ was obtained from s_X. The error element $e_{i,j}$ is defined as follows.

$$e_{i,j} = \sum_{l=1}^{m} r_{i,j} \times \left(y_{(l)} - y_{(j)}\right)^2 \quad \text{for } i = 1,\cdots,n; j = 1,\cdots,m, \quad (12)$$

where $r_{i,j}$ is the element of matrix $\mathbf{R}$ in the i-th row and j-th column. Recall that m and n are the number of grouped x_k and the number of grouped y_k, respectively.

Step 3 Build a cumulative error matrix $\mathbf{C}$. Each element $c_{i,j}$ of the matrix $\mathbf{C}$ describes the accumulated errors from left column to right. CIR computes $c_{i,j}$ as follows.

$$c_{i,j} = \begin{cases} e_{i,j}, & \text{if } j = 1 \\ e_{i,j} + \min_{1\le i'\le i}\{c_{i',j-1}\}, & \text{otherwise.} \end{cases} \quad (13)$$

For each value in $\mathbf{C}$, keep the index $d_{i,j} = \text{argmin}_{1\le i'\le i}\{c_{i',j-1}\}$ that tracks the minimum value

Algorithm 1: Algorithm for Combinatorial Isotonic Regression

```
for i = 1 to m do
    c_{i,1} = e_{i,1};
end
for j = 1 to n do
    for i = 1 to m do
        c_{i,j} = e_{i,j} + min_{1≤i'≤i} {c_{i',j-1}};
        d_{i,j} = argmin_{1≤i'≤i} {c_{i',j-1}};
    end
end
l = argmin_{1≤i≤m} {c_{i,n}};
ŷ_(n) = y_(l);
for j = m − 1 down to 1 do
    l = d_{l,j};
    ŷ_(j) = y_(l);
end
```

in its previous column, which leads to the current estimate of the cumulative error.

Step 4 Once the cumulative matrix $\mathbf{C}$ is obtained, find the minimum value in the last column of $\mathbf{C}$ and the path leading to this value by back tracking the indices.

Steps 3 and 4 are presented in Algorithm 1.

B. Nearly Isotonic Regression

CIR is based on dynamic programming [13]. However, in Step 2, it can be seen that the estimate of the reading from sensor s_Y can only be selected from the set of $y_{(j)}$'s. To improve the CIR method, we propose a Nearly Isotonic Regression (NIR) based method in this section.

We adopt a method originally proposed for solving NIR problems in [18], to estimate the reading of s_Y from that of s_X. Again, we group the same value of $\mathbf{X}$ and average the values of readings at s_Y when $\mathbf{X}$ takes value $x_{(i)}$. Denote $\overline{y}_i$ as the average reading at s_Y when $\mathbf{X}$ takes $x_{(i)}$. The problem can be formulated as:

$$\begin{aligned} \text{minimize:} \quad & \sum_{i=1}^{n} (\overline{y}_i - \beta_i)^2 \\ \text{subject to:} \quad & \beta_1 \leq \cdots \leq \beta_n. \end{aligned} \tag{14}$$

The optimal solution $\hat{\beta}_i$ of this problem is an estimate of y_k when reading x_k is $x_{(i)}$ at sensor s_X. After obtaining $\widehat{\beta}_i$, the reading from sensor s_X is used to estimate that of s_Y. If the reading of s_X is $x_{(i)}$, the estimate of the reading of s_Y is β_i.

In [18], Tibshirani et al. seek a nearly-monotone approximation and considers the following problem

$$\hat{\beta}_\lambda = \underset{\beta_i}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{n} (\overline{y}_i - \beta_i)^2 + \lambda \sum_{i=1}^{n-1} (\beta_i - \beta_{i+1})_+ \right\}, \tag{15}$$

where x_+ indicates the positive part of x, i.e., $x_+ = x \cdot 1_{[x>0]}$. This is a convex problem for each fixed $\lambda \geq 0$. We have Proposition 2 from [18] to solve this problem.

Proposition 2. *Suppose that, for some λ, two adjacent coordinates of the solution satisfy $\hat{\beta}_{\lambda,i} = \hat{\beta}_{\lambda,i+1}$. Then we have $\hat{\beta}_{\lambda_0,i} = \hat{\beta}_{\lambda_0,i+1}$, for all $\lambda_0 \geq \lambda$ [18].*

Proposition 2 states that the pieces in the solution can only be joined together, not split apart, as λ increases. Consider a parameter value λ with K_λ joined pieces, which are represented by groups of coordinates $A_1, \cdots, A_{K_\lambda}$. Problem (15) can be rewritten as

$$\frac{1}{2} \sum_{i=1}^{K_\lambda} \sum_{j \in A_i} (\overline{y}_j - \beta_{\lambda,A_i})^2 + \lambda \sum_{i=1}^{K_\lambda - 1} (\beta_{\lambda,A_i} - \beta_{\lambda,A_{i+1}})_+. \tag{16}$$

To find the optimal values $\hat{\beta}_{\lambda,A_i}$, we examine the subgradient of (16), as

$$-\sum_{j \in A_i} \overline{y}_j + |A_i| \hat{\beta}_{\lambda,A_i} + \lambda(w_i - w_{i-1}) = 0 \quad \text{for } i = 1, \cdots, K_\lambda, \tag{17}$$

where $w_i = 1_{[\hat{\beta}_{\lambda,A_i} - \hat{\beta}_{\lambda,A_{i+1}} > 0]}$ and $w_0 = w_{K_\lambda} = 0$. We then differentiate (17) with respect to λ to get the slope m_i as

$$m_i = \frac{d\hat{\beta}_{\lambda,A_i}}{d\lambda} = \frac{w_{i-1} - w_i}{|A_i|}. \tag{18}$$

The slope m_i is a constant, indicating that each $\hat{\beta}_{\lambda,A_i}$ is a linear function of λ.

Using this slope, it can be shown that groups A_i and A_{i+1} will merge at

$$t_{i,i+1} = \frac{b_{i+1} - b_i}{m_i - m_{i+1}}, \quad \text{for each } i = 1, \cdots, K_\lambda - 1, \tag{19}$$

where $b_i = \frac{\sum_{j \in A_i} \overline{y}_j}{|A_i|}$. We can move to the next value of λ

$$\lambda^* = \min_{i: t_{i,i+1} > \lambda} \{t_{i,i+1}\}, \tag{20}$$

and merge groups A_{i^*} and A_{i^*+1}, where $i^* = \operatorname{argmin}_{i: t_{i,i+1} > \lambda} \{t_{i,i+1}\}$. Note that the minimization problem (20) is only taken over the values of $t_{i,i+1}$ that are larger than λ. If none of the $t_{i,i+1}$ are larger than λ, then none of the existing groups will merge and the algorithm is terminated. The algorithm is presented in Algorithm 2. The worst case complexity of this algorithm is $O(n)$.

C. Compute Routing Matrix

Recall that the routing matrix $\mathbf{\Phi}$ in Bayesian Estimation is a binary matrix that randomly picks L active nodes from the set of N sensors. In Isotonic Regression, the routing matrix is not a random matrix; it chooses active nodes based on the estimation errors of each pair of sensor nodes.

Define $\epsilon_{X,Y}$ as the estimation error at sensor s_Y due to reconstruction of the readings at sensor s_X, as

$$\epsilon_{X,Y} = \frac{1}{K} \sum_{k=1}^{K} (y_k - f(x_k))^2 \quad \text{for } X \neq Y \text{ and } X, Y \in \{1, \cdots, N\}, \tag{21}$$

Algorithm 2: Algorithm for Nearly Isotonic Regression

```
Initialize λ = 0, K_λ = n, A_i = {i} and β̂_{λ,i} = ȳ_i for each i;
while TRUE do
    Compute the slope m_i according to (18);
    Compute the collision time t_{i,i+1} according to (19);
    if all t_{i,i+1} ≤ λ then
        The algorithm is terminated;
    else
        Compute the critical value λ* as in (20) and update
        β̂_{λ,i} as: β̂_{λ*,i} = β̂_{λ,i} + m_i(λ* − λ);
        Merge groups A_{i*} and A_{i*+1};
    end
end
```

Algorithm 3: Algorithm for Routing Matrix

```
1  Initialize the estimation error matrix Ψ;
2  Initialize Ω = {1, ··· , N} and Ω_1 = ∅;
3  while Ω is not empty do
4      Find {i*, j*} = argmin_{i≠j; i∈Ω∪Ω_1; j∈Ω} {ε_{i,j}};
5      if ε_{i*,j*} ≤ ξ then
6          Remove i* from Ω;
7          if j* ∈ Ω then
8              Move j* to Ω_1;
9          end
10     else
11         The algorithm is terminated;
12     end
13 end
```

where y_k is a reading from s_Y and $f(x_k)$ is the estimate of y_k based on the reading of s_X. We assume $\epsilon_{X,X} = 0$ for all nodes. For the N sensor nodes, we can construct an $N \times N$ matrix $\mathbf{\Psi}$ whose element is the estimation error of each pair of sensors. We compare the elements of matrix $\mathbf{\Psi}$ with a preset threshold ξ and choose the pair whose corresponding estimation error is not greater than ξ.

The algorithm for finding the routing matrix $\mathbf{\Phi}$ is presented in Table 3. Define Ω as the set of all sensor nodes and Ω_1 as the set of active sensor nodes. In the beginning, we let $\Omega = \{1, \cdots, N\}$ and $\Omega_1 = \emptyset$. In Lines $4-12$, we find the minimum estimation error ϵ_{i^*,j^*} and compare it with the threshold ξ. If it is not greater than ξ, we remove i^* from Ω and move j^* from Ω to Ω_1. These lines are executed until all nodes in Ω are removed. Once the algorithm is terminated, the set Ω_1 is the set of nodes we would like to turn on. The nodes not in Ω_1 is scheduled to be switched off.

V. Polynomial Regression Approaches

If we treat the readings from s_X as predictor variables and those from s_Y as response variables, it is natural to use Polynomial Regression to estimate the readings of s_Y from those of s_X. In this section, we show how to use *Linear Regression* (LR) and *Quadratic Regression* (QR) in CS.

A. Linear Regression

As in [14], we also adopt the simple LR method in CS. The reading of sensor node s_Y can be estimated from the reading of sensor node s_X with an LR model as follows.

$$\hat{y}_k = \alpha_0 + \alpha_1 \cdot x_k, \text{ for } k = 1, \cdots, K, \tag{22}$$

where α_0 and α_1 are the intercept and slope of the linear regression model.

Once $\hat{y}_k$ is obtained, the estimation error $\epsilon_{X,Y}$ and estimation error matrix $\mathbf{\Psi}$ can be computed as in Section IV-C. Similarly, the routing matrix for LR is obtained as in Table 3.

B. Quadratic Regression

To compare with the LR method, we also choose a QR model as follows.

$$\hat{y}_k = \alpha_0 + \alpha_1 \cdot x_k + \alpha_2 \cdot (x_k)^2, \text{ for } k = 1, \cdots, K, \tag{23}$$

where α_2 is the coefficient of the second order term of x_k. The estimation error and routing matrix are obtained as in the case of LR. The complexity of LR and QR is $O(1)$ because the coefficients can be computed with explicit expressions.

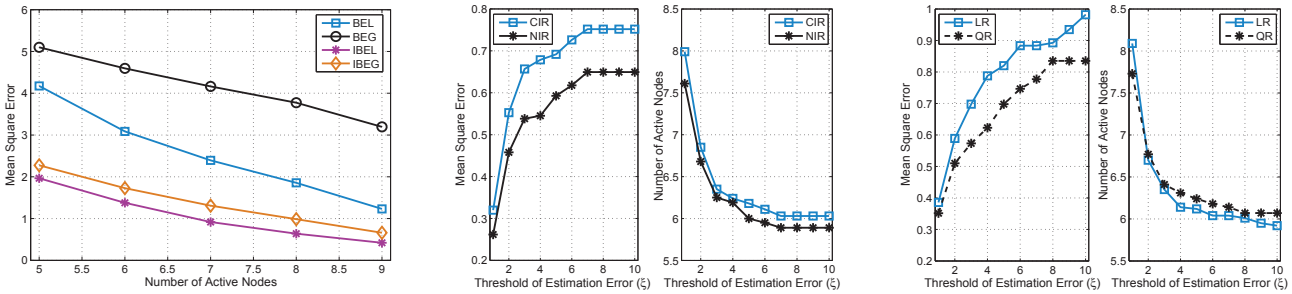
VI. Performance Evaluation

In this section, we evaluate the performance of all the methods with trace-driven simulations. We use the traces from the sensors deployed in an outdoor environment [19]. There are $N = 16$ sensors that measure the illumination in a field. The range of the measurements is from 0 to 255 and they are collected from time $k = 1$ to 150. We define Mean Square Error (MSE) as: $\text{MSE} = \frac{1}{K \times N} \sum_{k=1}^{K} \sum_{i=1}^{N} (\hat{x}_k^i - x_k^i)^2$, where x_k^i is the reading at the i-th sensor at time k and $\hat{x}_k^i$ the estimate of x_k^i. Note that $\hat{x}_k^i = x_k^i$ if the i-th sensor is active. MSE is computed for $k = 51$ to 150. At each time k, the previous fifty data vectors are used for parameter estimation.

We first compare the performance of the BE approaches. We name the BE approaches with Gaussian and Laplace assumptions as BEG and BEL, respectively. Our proposed improvements are named IBEG and IBEL. In the simulation, we increase the number of active nodes from 5 to 9. Four curves of the BE approaches are plotted in Fig. 1(a). As expected, the more active nodes, the more accurate the prediction and the smaller the estimation error. We find our proposed enhancements yield lower MSE than those of the original schemes. IBEL has the best performance among the four BE-based methods, indicating that Laplace assumption is preferred in our proposed method.

Next, we investigate the impact of estimation error threshold ξ on CIR and NIR. In Fig. 1(b), we increase ξ from 1 to 10 and plot MSE and the number of active nodes. Intuitively, a larger ξ means a larger tolerance on estimation error, allowing more nodes to be turned off to save more energy. It can be seen from the figure that the MSE of NIR is up to 0.13 lower than that of CIR. Furthermore, we find NIR requires less active nodes than CIR. Therefore, NIR is superior to CIR with respect to both estimation error and energy savings. We thus choose it to compare with other methods in the following simulations.

In Fig. 1(c), we examine the impact of estimation error threshold ξ on LR and QR. We increase ξ from 1 to 10 and plot MSE and the number of active nodes in the figure. We find that although the MSE of QR is lower than that of LR, QR requires more active nodes and thus consumes more power



(a) The BE approaches. (b) The IR approaches. (c) The PR approaches.

Fig. 1. Comparison of the schemes in each CS class.

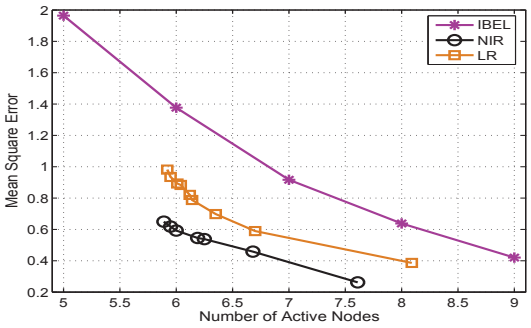


Fig. 2. Comparison of three approaches

for computation and communication. Compared with QR, LR may be a sufficiently good choice for CS.

Finally, we choose one method from each of the three CS classes discussed above and compare their performance. Specifically, we choose IBEL, NIR and LR in this simulation. For each threshold of Estimation Error in Figs. 1(b) and 1(c), the Y-axis value is the corresponding MSE and the X-axis value is the corresponding number of active nodes. Three curves are plotted in Fig. 2 for the three selected schemes, respectively. Although IBEL requires as few as 5 active nodes, its MSE is very high. Among these three methods, it is obvious that NIR achieves the best performance because it has smallest MSE and requires the minimum number of active nodes.

VII. Conclusion

In this paper, we investigated three classes of CS approaches. We first introduced Bayesian Estimation based approaches and propose an enhanced scheme by scaling the variance of projected random variables. We then adopted NIR to improve the performance of CIR and examined LR and QR approaches. We compared the three classes of CS approaches with trace-driven simulations. Our simulation results showed that NIR is the best choice among all the CS approaches considered in this paper.

Acknowledgment

This work is supported in part by the US National Science Foundation (NSF) under Grants CNS-0953513, and through the NSF Wireless Internet Center for Advanced Technology at Auburn University.

References

[1] E. J. Candés and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[2] Y. Zhao, S. Mao, J. O. Neel, and J. H. Reed, "Performance evaluation of cognitive radios: Metrics, utility functions, and methodology," *Proc. IEEE*, vol. 97, no. 4, pp. 642–659, 2009.

[3] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang, "PEAS: A robust energy conserving protocol for long-lived sensor networks," in *Proc. IEEE ICDCS'03*, Providence, RI, May 2003, pp. 28–37.

[4] H. Ling and T. Znati, "Energy efficient adaptive sensing for dynamic coverage in wireless sensor networks," in *Proc. IEEE WCNC'09*, Budapest, Hungary, Apr. 2009, pp. 1–6.

[5] C.-F. Hsin and M. Liu, "Network coverage using low duty-cycled sensors: Random & coordinated sleep algorithms," in *Proc. ACM IPSN'04*, Berkeley, CA, Apr. 2004, pp. 433–442.

[6] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *Proc. ACM SenSys'03*, Los Angeles, CA, Nov. 2003, pp. 28–39.

[7] T. Yan, T. He, and J. A. Stankovic, "Differentiated surveillance for sensor networks," in *Proc. ACM SenSys'03*, Los Angeles, CA, Nov. 2003, pp. 51–62.

[8] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-efficient surveillance system using wireless sensor networks," in *Proc. ACM MobiSys'04*, Boston, MA, June 2004, pp. 270–283.

[9] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic, "Towards optimal sleep scheduling in sensor networks for rare-event detection," in *Proc. ACM IPSN'05*, Los Angeles, CA, Apr. 2005, pp. 20–27.

[10] D. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 4036–4048, Apr. 2006.

[11] E. Candes and T. Tao, "Near optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inform. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[12] J. R. E. Candes and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inform. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.

[13] F. Koushanfar, N. Taft, and M. Potkonjak, "Sleeping coordination for comprehensive sensing using isotonic regression and domatic partitions," in *Proc. IEEE INFOCOM'06*, Barcelona, Spain, Apr. 2006, pp. 1–13.

[14] G. Ollos and R. Vida, "Adaptive regression algorithm for distributed dynamic clustering in wireless sensor networks," in *Proc. 2nd IFIP Wireless Days (WD)*, Paris, France, Dec. 2009, pp. 1–5.

[15] R. Masiero, G. Quer, D. Munaretto, M. Rossi, J. Widmer, and M. Zorzi, "Data acquisition through joint compressive sensing and principal component analysis," in *Proc. IEEE Globecom 2009*, Honolulu, HI, Nov.-Dec. 2009, pp. 1–6.

[16] R. Masiero, G. Quer, M. Rossi, and M. Zorzi, "A Bayesian analysis of compressive sensing data recovery in wireless sensor networks," in *Proc. 2009 Int. Conf. Ultra Modern Telecommun. Workshops*, St.-Petersburg, Russia, Oct. 2009, pp. 1–6.

[17] M. Grant, "CVX: Matlab software for disciplined convex programming, version 1.21," [online] Available: http://cvxr.com/cvx, Apr. 2011.

[18] R. Tibshirani, H. Hoefling, and R. Tibshirani, "Nearly-isotonic regression," *Technometrics*, vol. 53, no. 1, pp. 54–61, Feb. 2011.

[19] Wireless Sensor Network Illumination Data Trace. [online] Available: http://lightning.med.monash.edu/Illumination.zip.