Energy-Aware Sensor Selection in Field Reconstruction

Sijia Liu, *Student Member, IEEE*, Aditya Vempaty, *Student Member, IEEE*, Makan Fardad, *Member, IEEE*, Engin Masazade, *Member, IEEE*, and Pramod K. Varshney, *Fellow, IEEE*

***Abstract*—In this letter, a new sparsity-promoting penalty function is introduced for sensor selection problems in field reconstruction, which has the property of avoiding scenarios where the same sensors are successively selected. Using a reweighted ℓ_1 relaxation of the ℓ_0 norm, the sensor selection problem is reformulated as a convex quadratic program. In order to handle large-scale problems, we also present two fast algorithms: accelerated proximal gradient method and alternating direction method of multipliers. Numerical results are provided to demonstrate the effectiveness of our approaches.**

***Index Terms*—Alternating direction method of multipliers, convex relaxation, field reconstruction, proximal gradient method, reweighted ℓ_1, sensor selection, sparsity.**

I. Introduction

In many field reconstruction problems, multiple sensors observe the field intensity (e.g., pressure, temperature, or pollution concentration) at their own locations, and transmit their measurements to a fusion center (FC) that determines the global estimate of field intensity at an unobserved location and time instant. Due to resource constraints such as communication bandwidth and sensor battery life, it is important to select a small number of sensors for field reconstruction at each time. In this letter, we aim to schedule sensors over a finite time horizon in an energy-efficient way.

In [1], the sensor selection problem is interpreted as an optimization problem of selecting the best subset of sensors in wireless sensor networks with a given estimation performance. Recently, a sparsity-promoting technique has been used for sensor selection [2]–[5], where the desire to reduce the number of selected sensors is characterized by a sparsity-promoting penalty term in the objective function. This provides a tractable optimization framework to obtain the trade-off between estimation performance and the number of selected sensors. Also, the unconstrained formulation for sensor selection in the sparsity-promoting framework can be efficiently handled by several approaches, e.g., alternating direction method of multipliers (ADMM) in [2]–[4] and the projected subgradient method in [5].

The current sparsity-promoting techniques [2]–[5] may lead to scenarios in which a fixed set of sensors, which we hereafter refer to as the most 'informative sensors', are successively selected, e.g., due to their larger mutual information with the target [6] or stronger correlation with the field point of interest [7]. This behavior would result in faster energy depletion of the most informative sensors. From the perspective of network lifetime [8], the death of a sensor (or a percentage of sensors) can make the network nonfunctional. Therefore, it is desired to have a balanced use of sensors while discouraging the excessive use of any group of sensors in the network.

In this letter, we propose a new sparsity-promoting penalty function, which penalizes successive selection of the same sensors. This framework generates sparse sensor schedules which achieve a trade-off between activating the most informative sensors and balancing the energy consumption in the network. We summarize our contributions in this letter as follows.

- A new sparsity-promoting penalty function is introduced for sensor selection problems to avoid scenarios in which the same sensors are successively selected.
- We formulate a new sensor selection problem which is combinatorial in nature. By relaxation, we convert it into a standard quadratic program (QP).
- We develop two efficient optimization approaches based on accelerated proximal gradient method (APGM) and ADMM, to address large-scale sensor selection problems.

II. Problem Formulation

A. System Model

We consider a generic system where multiple sensors are deployed in a sensor network to monitor a spatio-temporally correlated physical field. Measurements are taken from multiple sensors at multiple time instants to construct the field value at an unobserved location over multiple time steps. We use the following notations throughout the letter: M is the number of sensors, K is the number of measurements taken by each sensor, N is the number of time instants where the field intensity is estimated, and let $L := KMN$. The field intensity at time t and

Manuscript received May 24, 2014; revised July 16, 2014; accepted July 20, 2014. Date of current version July 31, 2014. The work of S. Liu, A. Vempaty, and P. K. Varshney was supported by the U.S. Air Force Office of Scientific Research (AFOSR) under Grants FA9550-10-1-0263 and FA9550-10-1-0458. The work of M. Fardad was supported by the National Science Foundation under Awards CNS-1329885 and CMMI-0927509. The work of E. Masazade was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 113E220. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Qian He.

S. Liu, A. Vempaty, M. Fardad, and P. K. Varshney are with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13244 USA (e-mail: sliu17@syr.edu; avempaty@syr.edu; makan@syr.edu; varshney@syr.edu).

E. Masazade is with the Department of Electrical and Electronics Engineering, Yeditepe University, Istanbul 34755, Turkey (e-mail: engin.masazade@yeditepe.edu.tr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2014.2342198

location $\mathbf{s}$ is denoted by $f(t,\mathbf{s})$. Sensor locations are represented as $\mathbf{s}_m$ for $m=1,2,\ldots,M$.

The sensor measurements are given by

$$y_{k,m}=f(t_k,\mathbf{s}_m)+v_{k,m}, v_{k,m}\sim\mathcal{N}(0,\sigma_v^2), \tag{1}$$

for $k\in[K]$ and $m\in[M]$, where σ_v^2 is the variance of measurement noise, and for notational simplicity, let $[K]:=\{1,2,\ldots,K\}$ and $[M]:=\{1,2,\ldots,M\}$. We assume that $v_{k,m}$ is uncorrelated with $f(t_k,\mathbf{s}_m)$. Let $\mathbf{y}=[y_{1,1},\ldots,y_{K,1},\ldots,y_{1,M},\ldots,y_{K,M}]^T$, $\mathbf{f}=[f(t_1,\mathbf{s}_1),\ldots,f(t_K,\mathbf{s}_1),\ldots,f(t_1,\mathbf{s}_M),\ldots,f(t_K,\mathbf{s}_M)]^T$ and $\mathbf{v}=[v_{1,1},\ldots,v_{K,1},\ldots,v_{1,M},\ldots,v_{K,M}]^T$. Therefore, the observation model can be written in a compact form as

$$\mathbf{y}=\mathbf{f}+\mathbf{v}. \tag{2}$$

We assume that the second order statistics of the physical field is known in advance. Without loss of generality, let $\mathrm{E}[f(t,\mathbf{s})]=0$.

Our goal is to reconstruct the field intensity at an unobserved location $\boldsymbol{\varsigma}$ over N time instants $\tau_1,\tau_2,\cdots,\tau_N$. Denote the vector of field intensities to be estimated by $\boldsymbol{\varphi}:=[f(\tau_1,\boldsymbol{\varsigma}),f(\tau_2,\boldsymbol{\varsigma}),\ldots,f(\tau_N,\boldsymbol{\varsigma})]^T$. To perform the estimation task, we consider an unbiased linear estimator

$$\hat{\boldsymbol{\varphi}}=\mathbf{W}\mathbf{y}+\mathbf{c}, \mathbf{W}\in\mathbb{R}^{N\times KM}, \tag{3}$$

where $\mathbf{W}$ is an unknown coefficient matrix determined by the minimum mean square error criterion, and $\mathbf{c}=\mathbf{0}$ due to the unbiased condition.

According to (3), the trace of mean square error $\mathrm{tr}\{\mathbb{E}[(\boldsymbol{\varphi}-\hat{\boldsymbol{\varphi}})(\boldsymbol{\varphi}-\hat{\boldsymbol{\varphi}})^T]\}$ can be written as

$$\begin{aligned}J(\mathbf{W}):&=\sum_{n=1}^{N}\mathbb{E}[\varphi_n-(\mathbf{w}^n)^T\mathbf{y}]^2\\&=\sum_{n=1}^{N}(\mathbf{w}^n)^T\boldsymbol{P}\mathbf{w}^n-2\sum_{n=1}^{N}\boldsymbol{q}_n^T\mathbf{w}^n+N\sigma_f^2,\end{aligned} \tag{4}$$

where the nth *row* vector of $\mathbf{W}$ is denoted by $\mathbf{w}^n\in\mathbb{R}^{KM}$, $\mathbf{P}=\mathbb{E}[\mathbf{y}\mathbf{y}^T]=\mathrm{Cov}(\mathbf{f},\mathbf{f})+\sigma_v^2\mathbf{I}$, $\mathbf{q}_n=\mathbb{E}[\varphi_n\mathbf{y}]=\mathrm{Cov}(\mathbf{f},f(\tau_n,\boldsymbol{\varsigma}))$, and σ_f^2 is the variance of the field intensity. Note that the performance measure $J(\mathbf{W})$ is a quadratic function with respect to rows of $\mathbf{W}$. Therefore, we also write $J(\mathbf{W})$ as $J(\mathbf{w}_{\mathrm{row}})$, where $\mathbf{w}_{\mathrm{row}}=[(\mathbf{w}^1)^T,(\mathbf{w}^2)^T,\ldots,(\mathbf{w}^N)^T]^T\in\mathbb{R}^L$ is the row-wise vector of $\mathbf{W}$.

To address the problem of sensor selection in the context of field estimation, it was shown in [3] that the *non-zero columns* of coefficient matrix $\mathbf{W}$ characterize the selected sensor measurements. Suppose, for example, that only the mth sensor reports its kth measurement to the FC. In this case, it follows from (3) that $\mathbf{W}\mathbf{y}=\mathbf{w}_{k,m}y_{k,m}$, where $\mathbf{w}_{k,m}\in\mathbb{R}^N$ is the $(k+mK-K)$th column of $\mathbf{W}$. For ease of notation, we denote the column-wise vector of $\mathbf{W}$ by $\mathbf{w}_{\mathrm{col}}=[\mathbf{w}_1^T,\mathbf{w}_2^T,\ldots,\mathbf{w}_M^T]^T$, where $\mathbf{w}_{\mathrm{col}}\in\mathbb{R}^L$, and $\mathbf{w}_m=[\mathbf{w}_{1,m}^T,\mathbf{w}_{2,m}^T,\ldots,\mathbf{w}_{K,m}^T]^T\in\mathbb{R}^{KN}$ is a vector formed by stacking K column vectors of $\mathbf{W}$. Now, the problem of sensor selection can be interpreted as an optimization problem by promoting the column sparsity of $\mathbf{W}$

$$\underset{\mathbf{W}}{\text{minimize}}\quad\frac{1}{2}J(\mathbf{w}_{\mathrm{row}})+\gamma h(\mathbf{w}_{\mathrm{col}}), \tag{5}$$

where we interchangeably use the notations $\mathbf{W}$, $\mathbf{w}_{\mathrm{row}}$, and $\mathbf{w}_{\mathrm{col}}$ to denote the optimization variables, γ is a positive scalar, and

$$h(\mathbf{w}_{\mathrm{col}})=\sum_{m=1}^{M}\sum_{k=1}^{K}\|\|\mathbf{w}_{k,m}\|_p\|_0, \tag{6}$$

which gives the number of nonzero columns of $\mathbf{W}$, and is referred to as the sparsity-promoting penalty function; see more details in [3, Sec. III]. In (6), $\|\cdot\|_0$ is the ℓ_0 norm, and $\|\cdot\|_p$ denotes an arbitrary ℓ_p norm ($p=1$ or 2 typically). Here, we assume $p=1$ for ease of optimization.

It is clear from (5) that the rows and columns of $\mathbf{W}$ determine the estimation distortion and sensor activations, respectively. It is helpful to derive the relationship between $\mathbf{w}_{\mathrm{row}}$ and $\mathbf{w}_{\mathrm{col}}$: $\mathbf{w}_{\mathrm{col}}=\mathbf{D}\mathbf{w}_{\mathrm{row}}$, where $\mathbf{D}=[\mathbf{e}_{j_1},\mathbf{e}_{j_2},\cdots,\mathbf{e}_{j_{KMN}}]^T\in\mathbb{R}^{L\times L}$, and $\mathbf{e}_{j_l}\in\mathbb{R}^L$ ($l=1,2,\ldots,L$) denotes the basis vector with a 1 in the j_lth coordinate and 0s elsewhere. The value j_l is given by $j_l=(n_l-1)KM+(m_l-1)K+k_l$, where[1]

$$m_l=\left\lfloor\frac{l-1}{KN}\right\rfloor+1, k_l=\left\lfloor\frac{l-(m_l-1)KN-1}{N}\right\rfloor+1, \text{ and}$$
$$n_l=l-(m_l-1)KN-(k_l-1)N.$$

We emphasize that $\mathbf{D}$ is invertible since $\{\mathbf{e}_{j_l}\}_{l=1,2,\ldots,L}$ are linearly independent. Therefore, the estimation performance $J(\mathbf{w}_{\mathrm{row}})$ can be written as $J(\mathbf{D}^{-1}\mathbf{w}_{\mathrm{col}})$.

According to (5), sparser sensor schedules can be achieved by making γ larger. However, it may lead to scenarios in which a set of informative sensors is always selected. For instance, sensors closest to the field point of our interest would be successively selected due to their larger correlation with the field intensity to be estimated. As a consequence, these sensors may have early energy depletion, which reduces the network lifetime [8]. Therefore, we propose a new sparsity penalty function given by (7), which characterizes the 'cost' of each sensor being repeatedly selected:

$$g(\mathbf{w}_{\mathrm{col}}):=\sum_{m=1}^{M}\left(\sum_{k=1}^{K}\|\|\mathbf{w}_{k,m}\|_1\|_0\right)^2, \tag{7}$$

where the quantity $\kappa_m:=\sum_{k=1}^{K}\|\|\mathbf{w}_{k,m}\|_1\|_0$ is the number of times the mth sensor is selected over K time steps. The rationale behind using a quadratic function κ_m^2 is its relatively fast growth as a function of κ_m. Consider a system with $M=2$ sensors. It is clear from (7) that the penalty value of using the first sensor 4 times and the second sensor 0 times ($4^2+0^2=16$ units) is greater than the penalty of using each sensor 2 times ($2^2+2^2=8$ units). This motivates us to pose the following optimization problem

$$\underset{\mathbf{w}_{\mathrm{col}}}{\text{minimize}}\quad\frac{1}{2}J(\mathbf{D}^{-1}\mathbf{w}_{\mathrm{col}})+\gamma h(\mathbf{w}_{\mathrm{col}})+\eta g(\mathbf{w}_{\mathrm{col}}), \tag{8}$$

where we remark that the parameters γ and η characterize our relative emphasis on the conventional penalty of sensor activations and a new penalty term for successively selecting the same set of sensors.

[1] $\lfloor\cdot\rfloor$ maps a real number to the largest integer smaller than itself.

III. Reweighted ℓ_1 Based Quadratic Program

Due to the presence of the ℓ_0 norm, the proposed sensor selection problem is combinatorial in nature. For tractability, we relax problem (8) by using the reweighted ℓ_1 method [9, Sec. 2.2]. Specifically, we replace the ℓ_0 norm $\|\cdot\|_0$ with a weighted ℓ_1 norm $\alpha_{k,m}^p\|\cdot\|_1$, where $\alpha_{k,m}^p$ is an appropriately chosen positive weight at iteration $p = 0, 1, \cdots$, and $\alpha_{k,m}^0 = 1$ for $k \in [K]$ and $m \in [M]$. During each iteration, we solve an ℓ_1 norm optimization problem

$$\begin{aligned}\underset{\mathbf{w}_{\rm col}}{\text{minimize}} \quad & \frac{1}{2}J(\mathbf{D}^{-1}\mathbf{w}_{\rm col}) + \gamma\sum_{m=1}^{M}\sum_{k=1}^{K}\alpha_{k,m}^p\|\mathbf{w}_{k,m}\|_1 \\ & + \eta\sum_{m=1}^{M}\left(\sum_{k=1}^{K}\alpha_{k,m}^p\|\mathbf{w}_{k,m}\|_1\right)^2, \end{aligned} \tag{9}$$

where we denote its solution by $\{\mathbf{w}_{k,m}^*\}$ for $k \in [K]$ and $m \in [M]$. Based on $\{\mathbf{w}_{k,m}^*\}$, the weights with respect to optimization variables $\mathbf{w}_{k,m}$ are updated as $\alpha_{k,m}^{p+1} = 1/(\|\mathbf{w}_{k,m}^*\|_1 + \iota)$ for the next iteration, where the parameter $\iota > 0$ is a small number to ensure that the denominator does not become zero.

Proposition 1 shows that problem (9) can be solved by transformation to a QP.

Proposition 1: The optimal solution $\mathbf{w}_{\rm col}$ (or its matrix form $\mathbf{W}$) to (9) can be obtained by solving the relaxed problem

$$\begin{aligned}\underset{\mathbf{w}_{\rm col},\{\mathbf{u}_{k,m}\}}{\text{minimize}} \quad & \frac{1}{2}J(\mathbf{D}^{-1}\mathbf{w}_{\rm col}) + \gamma\sum_{m=1}^{M}\sum_{k=1}^{K}\alpha_{k,m}^p\mathbf{1}^T\mathbf{u}_{k,m} \\ & + \eta\sum_{m=1}^{M}\left(\sum_{k=1}^{K}\alpha_{k,m}^p\mathbf{1}^T\mathbf{u}_{k,m}\right)^2, \\ \text{subject to} \quad & |\mathbf{w}_{k,m}| \le \mathbf{u}_{k,m}, k \in [K], m \in [M], \end{aligned} \tag{10}$$

where $\mathbf{u}_{k,m} \in \mathbb{R}^N$ is a vector of introduced auxiliary variables, and the absolute operator $|\cdot|$ is defined in element-wise fashion.

Proof: Note that problem (10) is equivalent to (9) when $|\mathbf{w}_{k,m}| = \mathbf{u}_{k,m}$ for $k \in [K]$ and $m \in [M]$. Therefore, the main idea of the proof is to show that a strict inequality leads to a contradiction. ∎

For ease of notation, we define $\mathbf{u}_{\rm col} := [\mathbf{u}_1^T, \mathbf{u}_2^T, \ldots, \mathbf{u}_M^T]^T$ and $\mathbf{u}_m := [\mathbf{u}_{1,m}^T, \mathbf{u}_{2,m}^T, \ldots, \mathbf{u}_{K,m}^T]^T$. According to (4), $\mathbf{u}_{\rm col}$, and the relationship between $\mathbf{w}_{\rm row}$ and $\mathbf{w}_{\rm col}$, problem (10) can be written as

$$\begin{aligned}\underset{\mathbf{w}_{\rm col},\mathbf{u}_{\rm col}}{\text{minimize}} \quad & \frac{1}{2}\mathbf{w}_{\rm col}^T(\mathbf{D}^{-1})^T\tilde{\mathbf{P}}\mathbf{D}^{-1}\mathbf{w}_{\rm col} - \mathbf{w}_{\rm col}^T(\mathbf{D}^{-1})^T\tilde{\mathbf{q}} \\ & + \gamma\mathbf{u}_{\rm col}^T\tilde{\mathbf{r}} + \eta\mathbf{u}_{\rm col}^T\tilde{\mathbf{R}}\mathbf{u}_{\rm col} \\ \text{subject to} \quad & -\mathbf{u}_{\rm col} \le \mathbf{w}_{\rm col} \le \mathbf{u}_{\rm col}, \end{aligned} \tag{11}$$

where $\tilde{\mathbf{P}} := \mathbf{I}_{N\times N} \otimes \mathbf{P}$, $\otimes$ denotes the Kronecker product, $\tilde{\mathbf{q}} := [\boldsymbol{q}_1^T, \boldsymbol{q}_2^T, \ldots, \boldsymbol{q}_N^T]^T$, $\tilde{\mathbf{r}} := [\boldsymbol{\alpha}_1^T, \boldsymbol{\alpha}_2^T, \ldots, \boldsymbol{\alpha}_M^T]^T$, $\boldsymbol{\alpha}_m := [\alpha_{1,m}^p\mathbf{1}_N^T, \alpha_{2,m}^p\mathbf{1}_N^T, \ldots, \alpha_{K,m}^p\mathbf{1}_N^T]^T$ for $m \in [M]$, and $\tilde{\mathbf{R}}$ is a block diagonal matrix formed by $\{\boldsymbol{\alpha}_m\boldsymbol{\alpha}_m^T\}_{m\in[M]}$.

Further, let $\mathbf{x} := \mathbf{C}[\mathbf{w}_{\rm col}^T, \mathbf{u}_{\rm col}^T]^T$, where $\mathbf{C} = \begin{bmatrix} \mathbf{I} & -\mathbf{I} \\ -\mathbf{I} & -\mathbf{I} \end{bmatrix}$. Then, problem (11) can be simplified as

$$\begin{aligned}\underset{\mathbf{x}\in\mathbb{R}^{2L}}{\text{minimize}} \quad & \frac{1}{2}\mathbf{x}^T\tilde{\mathbf{H}}x - \mathbf{x}^T\tilde{\mathbf{h}} \\ \text{subject to} \quad & \mathbf{x} \le 0, \end{aligned} \tag{12}$$

where $\mathbf{A} := [\mathbf{I}_{L\times L}|\mathbf{0}_{L\times L}]$, $\mathbf{B} := [\mathbf{0}_{L\times L}|\mathbf{I}_{L\times L}]$, $\tilde{\mathbf{H}} := (\mathbf{D}^{-1}\mathbf{A}\mathbf{C}^{-1})^T\tilde{\mathbf{P}}\mathbf{D}^{-1}\mathbf{A}\mathbf{C}^{-1} + 2\eta(\mathbf{B}\mathbf{C}^{-1})^T\mathbf{R}\tilde{\mathbf{B}}\mathbf{C}^{-1}$, and $\tilde{\mathbf{h}} := (\mathbf{D}^{-1}\mathbf{A}\mathbf{C}^{-1})^T\tilde{\mathbf{q}} - \gamma(\mathbf{B}\mathbf{C}^{-1})^T\tilde{\mathbf{r}}$.

We remark that problem (12) is a *convex QP*. Convexity holds since $\tilde{\mathbf{P}}$ and $\tilde{\mathbf{R}}$ are positive semidefinite. To solve such a QP, the complexity of using standard solvers such as CVX [10] is roughly $O(L^3)$ per Newton iteration [11], where L is proportional to the dimension of the optimization variable $\mathbf{x}$.

IV. Fast Algorithms for Optimization

To solve (12), instead of using a QP solver, we present two fast algorithms which yield lower computational complexity. Note that problem (12) can be expressed in the following form

$$\underset{\mathbf{x}}{\text{minimize}} \quad \phi(\mathbf{x}) + \psi(\mathbf{x}), \tag{13}$$

where $\phi(\mathbf{x}) := \frac{1}{2}\mathbf{x}^T\tilde{\mathbf{H}}x - \mathbf{x}^T\tilde{\mathbf{h}}$ and $\psi(\mathbf{x})$ is defined by an indicator function $\psi(\mathbf{x}) := \begin{cases} 0 & \text{if } \mathbf{x} \le 0 \\ +\infty & \text{otherwise} \end{cases}$.

References [12], [13] show that problems in the form of (13) (whose objective consists of a differentiable convex function ϕ and a nonsmooth convex function ψ) can be efficiently solved by using the accelerated proximal gradient method (APGM) and the alternating direction method of multipliers (ADMM). The advantages of APGM and ADMM are that we can split the objective function into two terms ϕ and ψ, which facilitates the optimization procedure.

A. Accelerated Proximal Gradient Method

We employ the accelerated proximal gradient method [12, Sec. 4.3] to solve problem (13) and summarize APGM in Algorithm 1. We refer to [12] for more details.

Algorithm 1 APGM-based algorithm for solving (13)

Require: Given $\mathbf{x}^{-1} = \mathbf{x}^0$, λ^0, $\varrho = 0.5$ and $\epsilon_{\rm prox} > 0$.
1: **for** $i = 1, 2, \ldots$ **do**
2: Let $\lambda := \lambda^{i-1}$ and $\mathbf{s}^i := \mathbf{x}^i + \frac{i}{i+3}(\mathbf{x}^i - \mathbf{x}^{i-1})$.
3: **repeat**
4: $\boldsymbol{\chi} := \mathbf{prox}_{\lambda\psi}(\mathbf{s}^i - \lambda\nabla\phi(\mathbf{s}^i)) = (\mathbf{s}^i - \lambda\nabla\phi(\mathbf{s}^i))_-$.
5: **until** if $\phi(\boldsymbol{\chi}) \le \hat{\phi}_{\lambda,\mathbf{x}^i}(\boldsymbol{\chi})$ where $\hat{\phi}_{\lambda,\mathbf{x}^i}(\boldsymbol{\chi})$ is given by (15); else $\lambda = \varrho\lambda$, go to step 3.
6: Let $\lambda^i := \lambda$ and $\mathbf{x}^{i+1} := \boldsymbol{\chi}$.
7: **until** $\|\phi(\mathbf{x}^{i+1}) - \phi(\mathbf{x}^i)\| < \epsilon_{\rm prox}$.
8: **end for**

In Step 4, the proximal operator $\mathbf{prox}_{\lambda\psi}(\cdot)$ of ψ with parameter λ is given by

$$\mathbf{prox}_{\lambda\psi}(\cdot) = \arg\min_{\mathbf{x}}\left(\psi(\mathbf{x}) + \frac{1}{2\lambda}\|\mathbf{x} - \cdot\|_2^2\right). \tag{14}$$

By recalling the definition of $\psi(\mathbf{x})$, the proximal operator (14) precisely defines a Euclidean projection onto the halfspace $\{\mathbf{x}|\mathbf{x} \le 0\}$. Namely, $\mathbf{prox}_{\lambda\psi}(\cdot) = (\cdot)_-$, where $(\mathbf{v})_-$ takes non-positive elements of a vector $\mathbf{v}$ and sets 0 for its positive elements. In Step 5, the function $\hat{\phi}_{\lambda,\mathbf{x}^i}(\boldsymbol{\chi})$ is defined by

$$\hat{\phi}_{\lambda,\mathbf{x}^i}(\boldsymbol{\chi}) \underset{=}{:} \phi(\mathbf{x}^i) + \nabla\phi(\mathbf{x}^i)^T(\boldsymbol{\chi} - \mathbf{x}^i) + \frac{1}{2\lambda}\|\boldsymbol{\chi} - \mathbf{x}^i\|_2, \tag{15}$$

where $\nabla\phi(\mathbf{x}^i) = \tilde{\mathbf{H}}\mathbf{x}^i - \tilde{\mathbf{h}}$. In (15), $\hat{\phi}_{\lambda,\mathbf{x}^i}(\boldsymbol{\chi})$ can be interpreted as a quadratic approximation of $\phi(\boldsymbol{\chi})$ [14]. Steps 3-5 constitute a procedure of backtracking line search [15], which is used to determine an appropriate step size λ at iteration i.

At each iteration of Algorithm 1, the computation cost is dominated by matrix multiplication, which has $O(L^2)$ complexity [16]. The total number of iterations required for APGM is approximated by $O(1/\sqrt{\epsilon})$ [12]. In our implementation, it takes around 100 iterations as $\epsilon_{\text{prox}}= 10^{-4}$.

B. Alternating Direction Method of Multipliers

Another efficient optimization method for solving problem (13) is ADMM, which is summarized in Algorithm 2. We refer to [13, Sec. 3] for more details.

Algorithm 2 ADMM-based algorithm for solving (13)

Require: Given $\mathbf{x}^0$, $\mathbf{z}^0$, $\boldsymbol{\nu}^0$, ρ and $\epsilon_{\text{ad}} > 0$.
1: **for** $i = 1, 2, \ldots$ **do**
2: $\mathbf{x}^{i+1} := \arg\min_{\mathbf{x}}\{\phi(\mathbf{x}) + \frac{\rho}{2}\|\mathbf{x} - \mathbf{a}^i\|_2^2\}$, $\mathbf{a}^i := \mathbf{z}^i - \frac{1}{\rho}\boldsymbol{\nu}^i$.
3: $\mathbf{z}^{i+1} := \arg\min_{\mathbf{z}}\{\psi(\mathbf{z}) + \frac{\rho}{2}\|\mathbf{z} - \mathbf{b}^i\|_2^2\}$, $\mathbf{b}^i := \mathbf{x}^{i+1} + \frac{1}{\rho}\boldsymbol{\nu}^i$.
4: $\boldsymbol{\nu}^{i+1} := \boldsymbol{\nu}^i + \rho(\mathbf{x}^{i+1} - \mathbf{z}^{i+1})$,
5: **until** $\|\phi(\mathbf{x}^{i+1}) - \phi(\mathbf{x}^i)\| < \epsilon_{\text{ad}}$.
6: **end for**

In Step 2, by recalling the definition of $\phi(\mathbf{x})$ in (13), it is not difficult to obtain $\mathbf{x}^{i+1} = (\tilde{\mathbf{H}} + \rho\mathbf{I})^{-1}(\tilde{\mathbf{h}} + \rho\mathbf{a}^i)$, where we remark that the matrix inversion $(\tilde{\mathbf{H}} + \rho\mathbf{I})^{-1}$ is *independent* of iteration index i and, therefore, we can compute this matrix inversion before employing ADMM. Further, similar to (14), the minimizer in Step 3 is given by $\mathbf{z}^{i+1} = (\mathbf{b}^i)_-$. This implies that ADMM also requires the computation of the proximal operator of a nonsmooth function ψ. Note that ADMM yields similar computational complexity per iteration as APGM (by getting rid of the line search step). Typically, ADMM takes a few tens of iterations to converge with modest accuracy [2]–[4], [13]. In our implementation, it takes around 100 iterations as $\epsilon_{\text{ad}}= 10^{-4}$.

V. Numerical Results

In this section, numerical examples consider the task of estimating the field intensities with $\boldsymbol{\varsigma} = (2, 3)$, $\tau_n = 0.2n - 0.1$ for $n = 1, 2, \ldots, 5$ ($N = 5$). Each of $M = 5$ sensors takes $K = 5$ measurements. Specifically, the sampling time for sensor measurements is given by $t_k = 0.2k$ for $k = 1, 2, \ldots, 5$. The correlation model is given by $\text{Cov}(f(t_1, \mathbf{s}_1), f(t_2, \mathbf{s}_2)) = \sigma_f^2 \exp\{-c_s\|\mathbf{s}_1 - \mathbf{s}_2\|_2 - c_t(t_1 - t_2)^2\}$, where $\sigma_f^2 = 1$, $c_s = 0.1$, and $c_t = 0.1$. For APGM and ADMM, we choose $\mathbf{x}^0 = \mathbf{1}$, $\mathbf{z}^0 = \mathbf{0}$, $\lambda^0 = 1$, $\rho = 1$ and $\epsilon_{\text{prox}} = \epsilon_{\text{ad}}= 10^{-4}$.

For results presented in Table I, we employ a reweighted ℓ_1 based APGM to solve problem (8) and obtain different sensor selection schemes by varying the sparsity-promoting parameters γ and η. As we can see in rows 1, 4 and 7, only the 1st and 4th sensors are selected as $\gamma > \eta$, since a larger γ places more emphasis on minimizing the conventional sparsity-promoting function, which encourages fewer but more informative sensors to be selected (the 1st and 4th sensors have the strongest correlation with the field point of our interest). As η increases (for a fixed nnz), we observe that sensors are selected as uniformly as possible. This is to be expected, since the new sparsity-promoting penalty enforces sensor energy to be consumed in a balanced way. Moreover, for a fixed value of γ or η (see examples of $\gamma= 1.6^{-2}$ and $\eta \in \{10^{-3}, 4.3 \times 10^{-3}\}$, or $\eta= 10^{-3}$ and $\gamma \in \{1.6 \times 10^{-2}, 5.5 \times 10^{-2}\}$), the total number of selected sensors decreases as η or γ increase, respectively.

TABLE I
Sensor Selection Schemes. The nnz Column Gives the Total Number of Selected Sensors, i.e., $\sum_{m=1}^{M}\sum_{k=1}^{K}\|\|\mathbf{w}_{k,m}\|_1\|_0$. Sensor Locations: $\text{S}_1 : (1, 3)$, $\text{S}_2 : (2, 0)$, $\text{S}_3 : (2, 1)$, $\text{S}_4 : (3, 2)$, $\text{S}_5 : (4, 0)$

γ	η	nnz	S_1	S_2	S_3	S_4	S_5
1.6×10^{-2}	10^{-3}	10	5	0	0	5	0
1.3×10^{-4}	4.8×10^{-2}	10	3	2	2	2	1
1.8×10^{-6}	10^{-1}	10	2	2	2	2	2
5.5×10^{-2}	10^{-3}	7	5	0	0	2	0
2.6×10^{-3}	7×10^{-3}	7	3	0	2	2	0
7×10^{-5}	8×10^{-2}	7	2	1	1	2	1
1.6×10^{-2}	4.3×10^{-3}	6	4	0	0	2	0
2.6×10^{-3}	1.8×10^{-2}	6	3	0	1	2	0
1.2×10^{-4}	8×10^{-2}	6	2	1	1	1	1

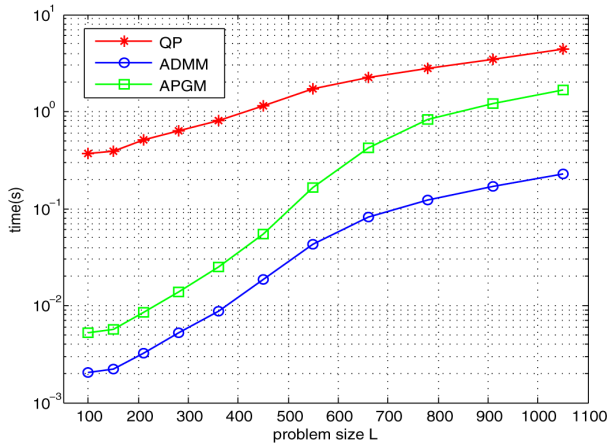

Fig. 1. Computation time against problem size.

In Fig. 1, we show the computation time as a function of the problem size ($L = KMN$) for different optimization methods: a QP solver in CVX, APGM, and ADMM. We first remark that in our extensive simulations, the three algorithms yielded similar objective function values; see similar results in [12, Sec. 7]. Thus, we compare their computation efficiency. We observe that APGM and ADMM perform much faster than solving a QP by CVX. Furthermore, ADMM requires less computation time than APGM. This is because at each iteration of APGM, line search consumes more time.

VI. Conclusion

In this letter, a new sparsity-promoting penalty function was proposed to penalize successive selections (i.e., overuse) of any group of sensors. We relaxed the problem using reweighted ℓ_1 minimization and converted the resulting problem to a QP. We also introduced two algorithms that scale more gracefully in problem size compared to QP. Numerical results show that the proposed framework ensures a sparse sensor schedule that strikes a balance between activating the most informative sensors and uniformly allocating sensor energy over the entire network. In the future, it would be worthwhile to seek guidelines for choosing the sparsity-promoting parameters for a desired sparsity level. Also, we will explore the relationship between the sparsity of sensor schedules and network lifetime in our framework.

REFERENCES

[1] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, Feb. 2009.

[2] E. Masazade, M. Fardad, and P. K. Varshney, "Sparsity-promoting extended Kalman filtering for target tracking in wireless sensor networks," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 845–848, Dec. 2012.

[3] S. Liu, M. Fardad, E. Masazade, and P. K. Varshney, "Optimal periodic sensor scheduling in networks of dynamical systems," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3055–3068, Jun. 2014.

[4] S. Liu, M. Fardad, E. Masazade, and P. K. Varshney, "Sparsity-aware field estimation via ordinary Kriging," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 3976–3980.

[5] S. P. Chepuri and G. Leus, *Sparsity-promoting sensor selection for non-linear measurement models*, 2013 [Online]. Available: http://arxiv.org/abs/1310.5251

[6] E. Ertin, J. W. Fisher, and L. C. Potter, "Maximum mutual information principle for dynamic sensor query problems," in *Proc. 2nd Int. Conf. Information Processing in Sensor Networks*, 2003, pp. 405–416.

[7] S. Liu, E. Masazade, and P. K. Varshney, "Temporally staggered sensing for field estimation with quantized data in wireless sensor networks," in *IEEE Statistical Signal Processing Workshop (SSP)*, Aug. 2012, pp. 512–515.

[8] Y. Chen and Q. Zhao, "On the lifetime of wireless sensor networks," *IEEE Commun. Lett.*, vol. 9, no. 11, pp. 976–978, Nov 2005.

[9] E. Candes, M. Wakin, and S. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Applicat.*, vol. 14, pp. 877–905, 2008.

[10] Inc. CVX Research, "CVX: Matlab software for disciplined convex programming, version 2.0," [Online]. Available: http://cvxr.com/cvx Aug. 2012

[11] A. Nemirovski, "Interior point polynomial time methods in convex programming," 2012 [Online]. Available: http://www2.isye.gatech.edu/nemirovs/Lect_IPM.pdf

[12] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 123–231, 2013.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Optim.*, vol. 3, no. 1, pp. 1–122, 2011.

[14] A. Beck and M. Teboulle, "Gradient-based algorithms with applications to signal recovery problems," in *Convex Optimization in Signal Processing and Communications*, D. Palomar and Y. Eldar, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2010, pp. 1–49.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[16] A. J. Stothers, "On the Complexity of Matrix Multiplication," Ph.D. thesis, Unive. Edinburgh, Edinburgh, U.K., 2010.