Nested performance bounds and approximate solutions for the sensor placement problem

Muhammad Sharif Uddin, Anthony Kuh, Aleksandar Kavcic and Toshihisa Tanaka

ORIGINAL PAPER

Nested performance bounds and approximate solutions for the sensor placement problem

MUHAMMAD SHARIF UDDIN[1], ANTHONY KUH[1], ALEKSANDAR KAVCIC[1] AND TOSHIHISA TANAKA[2]

This paper considers the placement of m sensors at n > m possible locations. Given noisy observations, knowledge of the state correlation matrix, and a mean-square error criterion (equivalently maximizing an efficacy cost criterion), the problem is formulated as an integer programming problem. Computing the solution for large m and n is infeasible, requiring us to look at approximate algorithms and bounding optimal performance. Approximate algorithms include greedy algorithms and variations based on examining the efficacy cost function and projection-based methods that all run in polynomial time of m and n. A sequence of nested bounds are found that upper bound the optimal performance (with analysis based on using matrix pencils and generalized eigenvectors). Finally, we show through simulations that the approximate algorithms perform well and provide tight implementable lower bounds to optimal performance and the nested bounds provide upper bounds to optimal performance with tighter bounds achieved with increasing complexity. The sensor placement problem has many energy applications where we are often confronted with limited resources. Some examples include where to place environmental sensors for an area in which there are many distributed solar photovoltaic generators and where to place grid monitors on an electrical distribution microgrid.

I. INTRODUCTION

This paperdevelops a number of tools and algorithms to analyze the sensor placement problem. The problem is formulated as an optimization problem where a limited number of sensors is placed to maximize an efficacy cost criterion. The solution involves solving an integer programming problem that becomes infeasible when the number of sensors and locations becomes large. The paper makes two key contributions; development of set of approximation algorithms (greedy algorithms and variations using the efficacy cost and also a projection-based method) that run in polynomial time in the number of parameters and derives a set of analytical nested performance upper bounds to the optimal solution based on the structure of the data correlation matrix. Simulations are conducted on a number of experiments from random correlation matrices, to a simulated sensor network, to an IEEE 57- bus test system [1] that show how tight the nested performance upper bounds are

[1]Department of Electrical Engineering, University of Hawaii at Manoa, Honolulu, HI 96822, USA
[2]Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, Tokyo 184-8588, Japan

Corresponding author:
M. Sharif Uddin
Email: uddin@hawaii.edu

and that the approximation algorithms give lower bounds to the optimal solution that are close to the optimal solution.

Data gathering, data mining, or data analytics have become increasingly more important in a wide range of areas from energy to health care to social networking to business to the environment. The National Science Foundation recently had a call for proposals on "Big Data" to study these problems. An important part is how data are gathered which in many cases require using distributed sensor networks and monitors. In applications ranging from the electrical power grid to the natural environment to biomedical monitoring there is a need to deploy sensors to monitor and estimate the behavior of different complex systems. Often the number of sensors that can be deployed is limited by cost and other constraints (e.g. communication and energy constraints of the sensors).

There has been much previous work on the optimal placement of sensors. In Dhillon et al. [2, 3] the optimal placement of sensors is considered where the probability of sensor detection depends upon distance with sensors placed on a two- or three-dimensional grid. Other research has considered sensor placement for the power grid by considering placement of phasor measurement units (PMUs) [4, 5]. In their research the problem is formulated as a state estimation problem with PMU placement depending on a key condition to make the system observable. In [6, 7], the optimization criterion is to maximize the

measurement redundancy while minimizing the required number of PMUs. To solve this problem, Dua et al. [6] considered the phasing of PMU deployment in an integer linear programming (ILP) framework, while Chakrabarti et al. [7] took a binary particle swarm optimization (BPSO)-based approach. In work by Krause et al. [8] they model spatial phenomena as a Gaussian Process and consider placement of sensors again using optimal experimental design. The goal is to maximize mutual information and the problem again becomes a combinatorial optimization problem that is NP-complete. This paper also discusses a greedy algorithm and shows that mutual information is submodular and is monotonically increasing for small number of sensors. Performance bounds are obtained for the greedy algorithm. Simpler reduced computation algorithms and robust algorithms are also considered. More recently in [9], PMU placement is considered in a different context where observability is assumed and the goal is to optimize the experimental design using a different criterion. The solution involves solving an integer programming problem, which is NP-complete. However, an approximate greedy solution is found that gives good results and runs in polynomial time. Then the greedy algorithm is tied to submodular and monotonic functions where bounds can be obtained to the greedy algorithm in relationship to the optimal algorithm. An estimation-theoretic approach to the PMU placement problem is proposed in [10, 11]; after posing system state estimation as a linear regression problem, a convex relaxation is developed to suboptimally solve the PMU placement problem. In [12], the authors consider the sensor selection problem in a wireless sensor network for event detection, under two hypotheses – event occurring and event not occurring. They propose the maximization of the Kullback–Liebler and Chernoff distances between the probability distributions of the selected sensor measurements, under these hypotheses, as the optimization criteria. After proving that this problem is NP-hard, the authors propose a greedy algorithm as a general approach to solve this problem suboptimally.

We consider a static discrete optimization problem where we have n discrete node locations where we can deploy sensors and we have m sensors to place. The objective is to minimize the sum of the mean-squared error at all node locations. This has some close ties to [8, 9] and has applications to energy problems. Examples include placement of meters such as advanced metering infrastructure (AMI) on the distribution grid or to deploy environmental resource sensors where distributed photovoltaic (PV) solar panels are located.

The optimization criteria we consider is also known as the *A*- optimality design of experiments maximizing the trace of the inverse of the information matrix also considered in [8–11]. A key difference between this paper and the others is that here we consider a variety of computationally efficient approximation algorithms for the sensor placement problem and come up with analytical nested lower and upper bounds (depending on the structure of the correlation matrix) for the cost function of the optimal

sensor placement. In [9], the authors present a bound on the optimal PMU placement problem under the assumption that the reward function is submodular. Under the submodularity assumption, the optimal solution is upper bounded by the greedy solution factored by $e/(e - 1)$. This bound is usually not very tight, because it does not take the covariance matrix structure (i.e. eigenvalues, eigenvectors, etc.) into account. In contrast, we do not assume any submodularity condition and the upper bounds are obtained analytically in terms of the eigenvalues of the covariance matrix. This paper significantly extends work by the authors in [13, 14] by considering a broader array of approximation algorithms including projection-based algorithms, presenting nested upper performance bounds with a complete analysis of these bounds (based on matrix pencils, generalized eigenvectors, and matrix manipulations), and a more complete set of simulation studies.

Section II gives a formulation of the problem. Since the optimal solution is computationally difficult to obtain for large n and m, we present some approximation solutions and lower bounds to the optimal solution in Section III. These include the expedient solution, greedy algorithm, and an algorithm based on dynamic programming heuristics. Next, in Section IV we find a family of upper bounds to the optimal solution. In Section V, we present some more lower bounds based on the projections of the solution subspace that achieved the upper bounds in Section IV. In Section VI, we present simulation results for randomly generated data, a 5×5 grid model, and data generated from the IEEE 57-bus test system [1]. Finally, Section VII summarizes results of this paper and suggests directions for further work. In this paper, we do not present any iterative solutions, however, any of the approximate solutions presented can be used as a starting point of an iterative algorithm [15].

Notation:

Upper and lower case letters denote random variables and their realizations, respectively; underlined letters stand for vectors; boldface upper case letters denote matrices, and $\mathbf{I}$ denotes the identity matrix; $\langle \cdot, \cdot \rangle$ denotes a matrix pencil; $(\cdot)^T$ and $\mathbf{E}(\cdot)$ stand for transposition and expectation, respectively.

II. PROBLEM STATEMENT

We assume that the state vector is $\underline{X} \in \mathbb{R}^n$ and the observation vector is $\underline{Y} \in \mathbb{R}^m$ where $m \leq n$. Sensors are all identical and give noisy readings of the state, where the variance of each sensor reading is σ^2. The model is described by

$$\underline{Y} = \mathbf{C}(\underline{X} + \sigma \underline{N}), \qquad (1)$$

where $\underline{N}$ is a random vector with mean $\mathbf{0}$ and covariance matrix $\Sigma_{\underline{N}} = \mathbf{I}_n$ with $\mathbf{I}_n$ being an identity matrix of dimension n and $\underline{X}$ are random vectors with mean $\mathbf{0}$ and covariance matrix $\Sigma_{\underline{X}}$. $\underline{X}$ and $\underline{N}$ are jointly independent. $\mathbf{C}$ is a binary matrix with orthonormal rows, where each row has one '1'. In other words, $\mathbf{C}$ is composed of m rows of the $n \times n$

identity matrix $\mathbf{I}_n$. The positions of ones in the matrix $\mathbf{C}$ denote the position of the sensors.

The linear minimum mean-squared error estimator is given by [16]

$$\hat{\underline{X}}(\underline{Y}) = \mathbf{E}\left(\underline{X}\,\underline{Y}^T\right)\mathbf{E}\left(\underline{Y}\,\underline{Y}^T\right)^{-1}\underline{Y}. \qquad (2)$$

The error is defined by $\underline{\mathcal{E}} = \underline{X} - \hat{\underline{X}}(\underline{Y})$ and the error covariance matrix is given by [16]

$$\mathbf{E}\left(\underline{\mathcal{E}}\,\underline{\mathcal{E}}^T\right) = \Sigma_{\underline{X}} - \mathbf{E}\left(\underline{X}\,\underline{Y}^T\right)\mathbf{E}\left(\underline{Y}\,\underline{Y}^T\right)^{-1}\mathbf{E}\left(\underline{Y}\,\underline{X}^T\right), \quad (3)$$

where

$$\mathbf{E}\left(\underline{X}\,\underline{Y}^T\right) = \Sigma_{\underline{X}}\mathbf{C}^T, \qquad (4)$$

and

$$\mathbf{E}\left(\underline{Y}\,\underline{Y}^T\right) = \mathbf{C}\Sigma_{\underline{X}}\mathbf{C}^T + \sigma^2\mathbf{I}_m. \qquad (5)$$

Our task is to find the matrix $\mathbf{C}$ that minimizes the total error $\text{tr}\,\mathbf{E}\left(\underline{\mathcal{E}}\,\underline{\mathcal{E}}^T\right)$.

Definition. *Let $\mathcal{C}^{[m \times n]}$ denote the **set** of all $m \times n$ matrices composed of m rows of the identity matrix $\mathbf{I}_n$.*

The optimization problem is then given by

$$\mathbf{C}^* = \underset{\mathbf{C}\in\mathcal{C}^{[m\times n]}}{\arg\min}\,\text{tr}\,\mathbf{E}\left(\underline{\mathcal{E}}\,\underline{\mathcal{E}}^T\right) = \underset{\mathbf{C}\in\mathcal{C}^{[m\times n]}}{\arg\min}\,\mathbf{E}\left(\underline{\mathcal{E}}^T\underline{\mathcal{E}}\right). \quad (6)$$

Since the first term in (3) (i.e. $\Sigma_{\underline{X}}$) does not depend on the choice of matrix $\mathbf{C}$, we can restate the optimization problem as an equivalent maximization problem using the following definition.

Definition. *Let the **efficacy** of matrix $\mathbf{C}$ be defined as*

$$J(\mathbf{C}) \triangleq \text{tr}\left\{\mathbf{E}\left(\underline{X}\,\underline{Y}^T\right)\mathbf{E}\left(\underline{Y}\,\underline{Y}^T\right)^{-1}\mathbf{E}\left(\underline{Y}\,\underline{X}^T\right)\right\} \qquad (7)$$

$$= \text{tr}\left\{\left[\mathbf{C}(\Sigma_{\underline{X}} + \sigma^2\mathbf{I})\mathbf{C}^T\right]^{-1}\mathbf{C}\Sigma_{\underline{X}}^2\mathbf{C}^T\right\}. \qquad (8)$$

where the final equality follows from the properties of the trace operator. (Note that (8) has the form of the generalized Rayleigh quotient.)

The optimization problem (6) is then equivalent to

$$\mathbf{C}^* = \underset{\mathbf{C}\in\mathcal{C}^{[m\times n]}}{\arg\max}\,J(\mathbf{C}), \qquad (9)$$

which is an integer programming problem of choosing m rows of the identity matrix $\mathbf{I}_n$ that maximize the efficacy. The optimum solutions to (9) requires an exhaustive search by testing all $\binom{n}{m}$ possible choices of m rows. Even for a moderately sized n and m this becomes computationally infeasible. In fact, the sensor placement problem is NP-complete [17].

We can further rewrite the efficacy to take advantage of the eigenstructure of the underlying matrices.

Definition. *Let $\bar{\mathbf{C}}$ denote **a complement** of $\mathbf{C}$, with constraints $\bar{\mathbf{C}} \in \mathcal{C}^{[(n-m)\times n]}$ and $\bar{\mathbf{C}}\mathbf{C}^T = \mathbf{0}$.*

We perform the eigendecomposition of $\mathbf{C}\Sigma_{\underline{X}}\mathbf{C}^T = \mathbf{U}_{(\mathbf{C})}\mathbf{D}_{(\mathbf{C})}\mathbf{U}_{(\mathbf{C})}^T$ where the columns of $\mathbf{U}_{(\mathbf{C})}$ are the eigenvectors of $\mathbf{C}\Sigma_{\underline{X}}\mathbf{C}^T$ and $\mathbf{D}_{(\mathbf{C})}$ is a diagonal matrix whose diagonal entries are the eigenvalues of $\mathbf{C}\Sigma_{\underline{X}}\mathbf{C}^T$. Let $\lambda_{(\mathbf{C}),1},\cdots,\lambda_{(\mathbf{C}),m}$ be the eigenvalues of $\mathbf{C}\Sigma_{\underline{X}}\mathbf{C}^T$. Let the permutation matrix $\mathbf{P} = [\mathbf{C}^T, \bar{\mathbf{C}}^T]$ and note that

$$\mathbf{C}\Sigma_{\underline{X}}^2\mathbf{C}^T = \mathbf{C}\Sigma_{\underline{X}}\mathbf{P}\mathbf{P}^T\Sigma_{\underline{X}}\mathbf{C}^T = \mathbf{U}_{(\mathbf{C})}\mathbf{D}_{(\mathbf{C})}^2\mathbf{U}_{(\mathbf{C})}^T$$
$$+ \mathbf{C}\Sigma_{\underline{X}}\bar{\mathbf{C}}^T\bar{\mathbf{C}}\Sigma_{\underline{X}}\mathbf{C}^T \qquad (10)$$

Then combining this with (8) we have that

$$J(\mathbf{C}) = \text{tr}\left\{\mathbf{U}_{(\mathbf{C})}\left[\mathbf{D}_{(\mathbf{C})} + \sigma^2\mathbf{I}\right]^{-1}\mathbf{D}_{(\mathbf{C})}^2\mathbf{U}_{(\mathbf{C})}^T\right\}$$
$$+ \text{tr}\left\{\bar{\mathbf{C}}\Sigma_{\underline{X}}\mathbf{C}^T\mathbf{U}_{(\mathbf{C})}\left[\mathbf{D}_{(\mathbf{C})} + \sigma^2\mathbf{I}\right]^{-1}\mathbf{U}_{(\mathbf{C})}^T\mathbf{C}\Sigma_{\underline{X}}\bar{\mathbf{C}}^T\right\}. \qquad (11)$$

The first term in (11) is the trace of a diagonal matrix and it contributes to the efficacy by summing the diagonal terms $\lambda_{(\mathbf{C}),i}^2/(\lambda_{(\mathbf{C}),i} + \sigma^2)$. The second term accounts for the state correlations. It is the second term that is most difficult to deal with when attempting to solve (9). Intuitively, we want to pick the matrix $\mathbf{C}$ such that the second term contributes considerably to the efficacy, i.e., we want to place sensors in locations that are highly correlated to the remaining states.

We can restate (11) using inner products (i.e., correlations).

Definition. *Let $\underline{g}_i$ be the **vector of inner products** between the ith eigenvector in $\mathbf{U}_{(\mathbf{C})}$ and the columns of $\mathbf{C}\Sigma_{\underline{X}}\bar{\mathbf{C}}^T$, i.e.,*

$$\underline{g}_i = \left(\bar{\mathbf{C}}\Sigma_{\underline{X}}\mathbf{C}^T\right)\left(\mathbf{U}_{(\mathbf{C})}\underline{e}_i^T\right), \qquad (12)$$

where $\underline{e}_i$ is the ith unit row vector and $\left(\mathbf{U}_{(\mathbf{C})}\underline{e}_i^T\right)$ is the ith eigenvector in $\mathbf{U}_{(\mathbf{C})}$.

The efficacy in (11) now takes the form

$$J(\mathbf{C}) = \sum_{i=1}^{m}\frac{\lambda_{(\mathbf{C}),i}^2}{\lambda_{(\mathbf{C}),i} + \sigma^2} + \sum_{i=1}^{m}\frac{\underline{g}_i^T\underline{g}_i}{\lambda_{(\mathbf{C}),i} + \sigma^2}. \qquad (13)$$

We can readily interpret the second term in (13) as the contribution of the *energy* in the correlations (between sensor readings and the remaining states) to the efficacy. Clearly, we would like to find a matrix $\mathbf{C}$, so that the eigenvalues are large and the measurements are maximally correlated to the remaining states.

Problem (9) is an integer-programming problem of choosing m rows of the identity matrix $\mathbf{I}_n$ that maximize the efficacy in (13). This can be solved by an exhaustive search requiring testing all $\binom{n}{m}$ possible choices of m rows. Even for a moderately sized n and m this becomes computationally infeasible. The sensor placement problem is in fact NP-complete [17]. Section III gives computationally feasible approximation algorithms to solve (9) and Section IV gives upper bounds to maximum efficacy in (9).

III. EFFICACY-BASED APPROXIMATE SOLUTIONS

Since the optimization in (9) is difficult to perform, we resort to approximate solutions. Each approximate solution is in fact an *ad hoc* solution because the exact solution requires an exhaustive search. If $\mathbf{C}$ is an *ad hoc* solution to (9), then it provides a lower bound on the optimal efficacy $J(\mathbf{C}^*)$, i.e. $J(\mathbf{C}) \leq J(\mathbf{C}^*)$. Therefore, the search for good (suboptimal) solutions to (9) is equivalent to constructing tight lower bounds on $J(\mathbf{C}^*)$. Here we consider approximate solutions to the optimization problem requiring a much lower search complexity than $\mathcal{O}\left(\binom{n}{m}\right)$.

A) Expedient solution

This is a trivial approximate solution to consider. Let $J(\underline{e}_k)$ be the efficacy of the kth unit row vector, i.e., the efficacy of the sensor placed at the location of the kth state variable when $m = 1$. Then using (8) we have

$$J(\underline{e}_k) = \sum_{i=1}^{n} \frac{\left(\underline{e}_k \, \Sigma_{\underline{X}} \, \underline{e}_i^T\right)^2}{\underline{e}_k \, \Sigma_{\underline{X}} \, \underline{e}_k^T + \sigma^2}. \tag{14}$$

We rank the vectors $\underline{e}_k$ in descending order of their efficacies $J(\underline{e}_k)$. For any arbitrary m, we pick the m highest ranked vectors $\underline{e}_k$ and stack them to be the rows of the approximate solution $\mathbf{C}_E$. Clearly we have $J(\mathbf{C}_E) \leq J(\mathbf{C}^*)$.

Since this algorithm requires sorting and picking m highest ranked vectors $\underline{e}_k$, it has search complexity at most $\mathcal{O}(n \log n)$. Thus, this algorithm finds an approximate solution very fast. Hence, the solution obtained by this algorithm can be used as a good starting point of an iterative algorithm [15].

B) Greedy solution

A greedy algorithm obtains an approximate solution to (9) by making a sequence of choices [18]. At each step t, it assumes that t sensor locations are fixed, and makes a greedy choice where to place the $(t+1)$-st sensor. Let $\mathbf{C}_G$ denote the solution provided by the greedy algorithm. The algorithm can be described by the following.

Greedy Algorithm [18]

1. Initialization: Set iteration $t = 1$ and choose $\mathbf{C}_t = \underline{e}^*$ such that $\underline{e}^* = \arg\max_{\underline{e} \in \mathcal{C}^{[1 \times n]}} J(\underline{e})$.

2. Find $\underline{e}^* = \arg\max_{\substack{\underline{e} \in \mathcal{C}^{[1 \times n]} \\ : \mathbf{C}_t \underline{e}^T = 0}} J\left(\begin{bmatrix} \mathbf{C}_t \\ \underline{e} \end{bmatrix}\right)$.

3. Set $\mathbf{C}_{t+1} = \begin{bmatrix} \mathbf{C}_t \\ \underline{e}^* \end{bmatrix}$.

4. Increment: $t \leftarrow t + 1$.

5. If $t = m$ set $\mathbf{C}_G = \mathbf{C}_t$ and stop, else go to 2. $\square$

Note that the greedy solution may not be optimal even for $m = 2$, but it has search complexity $\mathcal{O}(mn)$ which is much smaller than $\mathcal{O}\left(\binom{n}{m}\right)$ required to find the optimal solution $\mathbf{C}^*$.

C) *n*-path greedy solution

We propose the n-path greedy method to compute n candidate solutions where each candidate solution is attained by starting the greedy algorithm using each of the unit row vectors $\underline{e}_k$, where $k = 1, \ldots, n$. Let $\mathbf{C}_m^{(k)}$ denote the candidate solution when the greedy algorithm is initiated with vector $\underline{e}_k$. Finally, we choose the n-path greedy solution $\mathbf{C}_{nG}$ to be the best of the n-different candidate solutions.

$$\mathbf{C}_{nG} = \arg\max_{\mathbf{C} \in \left\{ \mathbf{C}_m^{(1)}, \mathbf{C}_m^{(2)}, \cdots, \mathbf{C}_m^{(n)} \right\}} J(\mathbf{C}). \tag{15}$$

The n-path greedy algorithm runs in polynomial time. It has search complexity $\mathcal{O}(mn^2)$, which is larger than the $\mathcal{O}(mn)$ search complexity of the plain greedy algorithm in Section III.B, but the n-path greedy algorithm performs better than the plain greedy algorithm, thus giving a tighter lower bound $J(\mathbf{C}_{nG})$ on the optimal efficacy $J(\mathbf{C}^*)$, i.e. $J(\mathbf{C}_G) \leq J(\mathbf{C}_{nG}) \leq J(\mathbf{C}^*)$.[1]

D) Backtraced *n*-path solution

We now propose a *backtraced* version of the n-path greedy algorithm to solve the optimization problem (9). This algorithm solves optimization problem (9) by dividing the problem into smaller subproblems, which is similar to the heuristics of the dynamic programming algorithm [18, 19]. However, the sensor placement problem does not have the optimal substructure property. Thus, the backtraced n-path solution is not optimal in general. In this algorithm, we use a *bottom-up* approach to rank the subproblems in terms of their problem sizes, smallest first. We save the intermediate solutions of the subproblems in a table and later use them to solve larger subproblems. For our optimization problem, we define a subproblem of size (number of sensors) t as finding the best candidate solution of size $t - 1$ for a newly added sensor in a fixed location. Therefore, for any arbitrary number of sensors t, we have n subproblems of size t. Let $\mathbf{C}_t^{(j)}$ be the solution to a subproblem of size t, where $t \in \{1, 2, \ldots, m\}$ is the number of sensors and $j \in \{1, 2, \ldots, n\}$ is the index of the subproblem. In short, the goal of the backtraced algorithm is to append the best existing solution $\mathbf{C}_{t-1}^{(j)}$ to a fixed $\underline{e}_k$ and thus construct a solution for a subproblem of size t.

Let $\mathbf{C}_{BT}$ denote the approximate solution to (9) computed by the backtraced n-path algorithm, in terms of the solutions of the subproblems as

$$\mathbf{C}_{BT} = \arg\max_{\mathbf{C} \in \left\{ \mathbf{C}_m^{(1)}, \mathbf{C}_m^{(2)}, \cdots, \mathbf{C}_m^{(n)} \right\}} J(\mathbf{C}), \tag{16}$$

---

[1]Further improvement on the performance of the n-path greedy algorithm may be possible by considering a larger initial search space, such as $\mathcal{O}(n^k)$, so that the locations of the first k sensors are guaranteed to be optimal. However, this comes at the cost of increased search complexity of $\mathcal{O}(mn^{k+1})$. Usually, there is no or only marginal performance gain since the n-path greedy solution often performs close or identical to the optimal solution.

where $\left\{ \mathbf{C}_m^{(1)}, \mathbf{C}_m^{(2)}, \ldots, \mathbf{C}_m^{(n)} \right\}$ is the set of the solutions to the subproblems of size m. The following procedure implements the backtraced n-path algorithm.

Backtraced n-path Algorithm

1. Initialization: Set iteration $t = 1$ and initial matrices $\mathbf{C}_1^{(1)} = \underline{e}_1, \mathbf{C}_1^{(2)} = \underline{e}_2, \ldots, \mathbf{C}_1^{(n)} = \underline{e}_n$.
2. Set $k \leftarrow 1$.
3. Find $j^* = \arg\max_{j: \mathbf{C}_t^{(j)} \underline{e}_k^T = \mathbf{0}} J\left(\begin{bmatrix} \mathbf{C}_t^{(j)} \\ \underline{e}_k \end{bmatrix} \right)$.
4. $\mathbf{C}_{t+1}^{(k)} = \begin{bmatrix} \mathbf{C}_t^{(j^*)} \\ \underline{e}_k \end{bmatrix}$.
5. $k \leftarrow k + 1$.
6. If $k = n$, continue to step 7, else go to step 3.
7. $t \leftarrow t + 1$.
8. If $t = m$, set $\mathbf{C}_{DP} = \arg\max_{\mathbf{C} \in \left\{ \mathbf{C}_t^{(1)}, \cdots, \mathbf{C}_t^{(n)} \right\}} J(\mathbf{C})$ and stop, else go to 2.

The backtraced version has the same search complexity $\mathcal{O}(mn^2)$ as the n-path greedy algorithm, and performs better than the plain greedy approximation, i.e., $J(\mathbf{C}_G) \leq J(\mathbf{C}_{BT}) \leq J(\mathbf{C}^*)$. However, we cannot provide an *a priori* comparison between $J(\mathbf{C}_{nG})$ and $J(\mathbf{C}_{BT})$ without explicitly computing both values.

IV. UPPER BOUNDS ON THE OPTIMAL EFFICACY

It is clear from the previous section that there exist numerous ways of obtaining a lower bound on the optimal efficacy $J(\mathbf{C}^*)$. However, to evaluate the performance of these lower bounds we want to obtain a numerically computable upper bound for the difference $J(\mathbf{C}^*) - J(\mathbf{C})$. One way to achieve this goal is to find a numerically computable upper bound, say $\bar{J}$, on the optimal $J(\mathbf{C}^*)$ such that

$$J(\mathbf{C}^*) - J(\mathbf{C}) \leq \bar{J} - J(\mathbf{C}). \qquad (17)$$

Hence, we devote this section to finding a family of upper bounds $\bar{J}_k$ on the optimal efficacy $J(\mathbf{C}^*)$ by relaxing conditions on $\mathbf{C}$. In Section IV.A we present some definitions, which describe the relaxation of the optimization constraints for problem (9). Next, Section IV.B gives the canonic theorems used to calculate the family of upper bounds. Specifically, in Lemma B, we devise a method of calculating a family of upper bounds if the optimal solution is available for some $k \leq m$. Finally, in Section IV.C, we show that these bounds are nested and can be calculated in terms of the generalized eigenvalues of matrix pencils, using Theorem 1 and Lemma B.

A) Definitions

To develop a family of upper bounds on the optimal efficacy, we generalize the reward function (efficacy), and generalize the optimization problem and its constraints. Instead of considering two matrices $\Sigma_{\underline{X}}^2$ and $\Sigma_{\underline{X}} + \sigma^2 \mathbf{I}$, in this section we consider a general matrix pencil $\langle \mathbf{A}, \mathbf{B} \rangle$, where $\mathbf{A}$ and

$\mathbf{B}$ do not necessarily equal $\Sigma_{\underline{X}}^2$ and $\Sigma_{\underline{X}} + \sigma^2 \mathbf{I}$, respectively. Next, instead of considering a matrix $\mathbf{C}$ whose entries take values in the set $\{0, 1\}$, in this section we consider a generalized matrix $\mathbf{F}$ whose entries take values in $\mathbb{R}$. Finally, we introduce a modified optimization problem (different from the one in Section II) that leads to the upper bounds. The following definitions set the stage.

Definition. *For two $n \times n$ matrices $\mathbf{A}$ and $\mathbf{B}$, we define the* **efficacy** *of a matrix $\mathbf{C}$, with respect to the matrix pencil $\langle \mathbf{A}, \mathbf{B} \rangle$, as*

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{C}) \triangleq \mathrm{tr}\left\{ \left(\mathbf{C}\mathbf{B}\mathbf{C}^T \right)^{-1} \mathbf{C}\mathbf{A}\mathbf{C}^T \right\}, \qquad (18)$$

[should the inverse $(\mathbf{C}\mathbf{B}\mathbf{C}^T)^{-1}$ exist].

Definition. *For $m \leq n$, let $\mathcal{F}^{[m \times n]}$ be the* **set** *of all $m \times n$ matrices with rank m.*

Definition. *We define $\mathbf{F}_{\langle \mathbf{A}, \mathbf{B} \rangle}^*$ to be the argument that solves the following optimization problem:*

$$\mathbf{F}_{\langle \mathbf{A}, \mathbf{B} \rangle}^* \triangleq \arg\max_{\mathbf{F} \in \mathcal{F}^{[m \times n]}} J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{F}) \qquad (19)$$

$$= \arg\max_{\mathbf{F} \in \mathcal{F}^{[m \times n]}} \mathrm{tr}\left\{ \left(\mathbf{F}\mathbf{B}\mathbf{F}^T \right)^{-1} \mathbf{F}\mathbf{A}\mathbf{F}^T \right\}. \qquad (20)$$

Definition. *We define $J_{\langle \mathbf{A}, \mathbf{B} \rangle}^*$ as the solution to the optimization problem in (19).*

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}^* \triangleq \max_{\mathbf{F} \in \mathcal{F}^{[m \times n]}} J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{F}) = J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\mathbf{F}_{\langle \mathbf{A}, \mathbf{B} \rangle}^* \right). \qquad (21)$$

B) Canonic theorem

If $\mathbf{A}$ and $\mathbf{B}$ are $n \times n$ symmetric matrices, there exist n generalized eigenvectors $\underline{v}_1, \underline{v}_2, \ldots, \underline{v}_n$, with corresponding generalized eigenvalues $\delta_1, \delta_2, \ldots, \delta_n$ such that $\mathbf{A}\underline{v}_j = \delta_j \mathbf{B}\underline{v}_j$. [Note: $\delta_1, \delta_2, \ldots, \delta_n$ need not be distinct.] We arrange the generalized eigenvalues as the diagonal elements of a diagonal matrix $\mathbf{D}$,

$$\mathbf{D} \triangleq \begin{bmatrix} \delta_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \delta_n \end{bmatrix}, \qquad (22)$$

and we arrange the generalized eigenvectors as the columns of a matrix $\mathbf{V}$,

$$\mathbf{V} \triangleq \begin{bmatrix} \underline{v}_1 & \cdots & \underline{v}_n \end{bmatrix}. \qquad (23)$$

Lemma A. *If $\mathbf{A}$ and $\mathbf{B}$ are symmetric $n \times n$ matrices and $\mathbf{B}$ is positive definite, then*

$$\mathbf{V}^T \mathbf{B} \mathbf{V} = \mathbf{I}, \qquad (24)$$

and

$$\mathbf{V}^T \mathbf{A} \mathbf{V} = \mathbf{D}. \qquad (25)$$

Proof: See in [20]. □

Theorem 1. *Let* **A** *and* **B** *be symmetric and* **B** *be positive definite, and let* **D** *and* **V** *denote the generalized eigenvalue matrix and generalized eigenvector matrix as in* (22) *and* (23), *respectively. If the eigenvalues are ordered as* $\delta_1 \geq \delta_2 \geq \cdots \geq \delta_n \geq 0$, *then*

$$J^*_{\langle \mathbf{A}, \mathbf{B} \rangle} = \mathrm{tr}\left\{ \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix} \mathbf{D} \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix}^T \right\} = \sum_{j=1}^{m} \delta_j, \quad (26)$$

and

$$\mathbf{F}^*_{\langle \mathbf{A}, \mathbf{B} \rangle} = \begin{bmatrix} \mathbf{I}_m & \mathbf{0} \end{bmatrix} \mathbf{V}^T = \begin{bmatrix} \underline{v}_1 & \cdots & \underline{v}_m \end{bmatrix}^T. \quad (27)$$

[Note: The solution $\mathbf{F}^*_{\langle \mathbf{A}, \mathbf{B} \rangle}$ *in* (27) *is not unique.]*

Proof: [21] provides a proof for this theorem. We provide an alternative proof using Lemma A in Appendix A. □

Remark 1. *If* $\mathbf{A} = \Sigma_{\underline{X}}^2$ *and* $\mathbf{B} = \Sigma_{\underline{X}} + \sigma^2 \mathbf{I}$, *and* $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ *are the eigenvalues of* $\Sigma_{\underline{X}}$, *then the generalized eigenvalues of the pencil* $\left\langle \Sigma_{\underline{X}}^2, \ \Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right\rangle$ *are*

$$\delta_j = \frac{\lambda_j^2}{\lambda_j + \sigma^2}. \quad (28)$$

Thus, using Theorem 1 we can write

$$J^*_{\left\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right\rangle} = \sum_{j=1}^{m} \frac{\lambda_j^2}{\lambda_j + \sigma^2}. \quad (29)$$

Theorem 1 provides an upper bound for the optimal efficacy $J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{C}^*)$ in terms of the generalized eigenvalues of the pencil $\langle \mathbf{A}, \mathbf{B} \rangle$. We now devise a method to calculate a family of upper bounds for the optimal efficacy when the optimal solution is available for some $k \leq m$. These upper bounds get tighter as k increases. To develop a family of upper bounds, we find it useful to solve a series of modified efficacy maximization problems for all $k \leq m$. The next definition addresses the modified efficacy maximization problem.

Definition. *For any* $k \leq m$, *we define* $\mathbf{F}^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle}$ *and* $J^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle}$ *as the solution pair of the following modified efficacy maximization:*

$$\mathbf{F}^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} \triangleq \underset{\mathbf{F} \in \mathcal{F}^{[(m-k) \times (n-k)]}}{\arg\max} J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix} \right), \quad (30)$$

and

$$J^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} \triangleq \underset{\mathbf{F} \in \mathcal{F}^{[(m-k) \times (n-k)]}}{\max} J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix} \right) \quad (31)$$

$$= J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F}^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} \end{bmatrix} \right). \quad (32)$$

In order to solve the modified efficacy maximization problem in (30)–(31), it is convenient to split the efficacy

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix} \right)$$

into two terms such that

(i) the first term does not depend on **F**, and
(ii) the second term equals the efficacy of **F** with respect to a modified pencil of lower dimensions.

We formulate the split in the following lemma.

Lemma B. *In the optimization problem* (31), *the efficacy can be expressed as*

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix} \right) = t_k + J_{\langle \mathbf{A}_k, \mathbf{B}_k \rangle}(\mathbf{F}), \quad (33)$$

where the additive term t_k *and the modified pencil* $\langle \mathbf{A}_k, \mathbf{B}_k \rangle$ *satisfy*

$$t_k = \mathrm{tr}\left\{ \mathbf{A} \begin{bmatrix} \mathbf{P}_k^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right\}, \quad (34)$$

$$\mathbf{A}_k = \begin{bmatrix} \mathbf{P}_k^{-1}\mathbf{Q}_k \\ -\mathbf{I}_{n-k} \end{bmatrix}^T \mathbf{A} \begin{bmatrix} \mathbf{P}_k^{-1}\mathbf{Q}_k \\ -\mathbf{I}_{n-k} \end{bmatrix}, \quad (35)$$

$$\mathbf{B}_k = \mathbf{R}_k - \mathbf{Q}_k^T \mathbf{P}_k^{-1} \mathbf{Q}_k, \quad (36)$$

$$\mathbf{P}_k = \begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix}^T \mathbf{B} \begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix}, \quad (37)$$

$$\mathbf{Q}_k = \begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix}^T \mathbf{B} \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{n-k} \end{bmatrix}, \quad (38)$$

$$\mathbf{R}_k = \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{n-k} \end{bmatrix}^T \mathbf{B} \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{n-k} \end{bmatrix}. \quad (39)$$

Proof: See Appendix B. □

Lemma B now lets us express the solution of the modified optimization problem (30)–(31) equivalently as the solution of a regular efficacy maximization (i.e. using Theorem 1), but for a modified matrix pencil. Hence, we have the following corollary of Theorem 1:

Corollary 1.1.

$$\mathbf{F}^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} = \mathbf{F}^*_{\langle \mathbf{A}_k, \mathbf{B}_k \rangle}, \quad (40)$$

and

$$J^{(k)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} = t_k + J^*_{\langle \mathbf{A}_k, \mathbf{B}_k \rangle}. \quad (41)$$

Proof: In (33), t_k does not depend on **F**. Therefore, (40) and (41) hold. □

Remark 2.

$$J^{(0)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} = J^*_{\langle \mathbf{A}, \mathbf{B} \rangle}. \quad (42)$$

Remark 3.

$$J^{(m)*}_{\langle \mathbf{A}, \mathbf{B} \rangle} = t_m. \quad (43)$$

From Corollary 1.1, we observe that the nested bounds are calculated in terms of the generalized eigenvalues of a matrix pencil with smaller dimensions.

C) Nested bounds

We dedicate this section to applying the upper bounds, obtained for the modified optimization problem in Section IV.B, to the optimization problem defined in Section II. We obtain these nested upper bounds assuming that the optimal solution to (9) is calculable for any $k \leq m$. We define the nested upper bounds $\bar{J}_k$ as follows.

Definition.

$$\bar{J}_k \triangleq \max_{\mathbf{C} \in \mathcal{C}^{[k \times n]}} \left\{ \max_{\substack{\mathbf{F} \in \mathcal{F}^{[(m-k) \times n]} \\ \mathbf{F}\mathbf{C}^T = \mathbf{0}}} J_{\left\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right\rangle} \left(\begin{bmatrix} \mathbf{C} \\ \mathbf{F} \end{bmatrix} \right) \right\}. \quad (44)$$

From Remark 2, we clearly see that $\bar{J}_0 \geq J(\mathbf{C}^*)$. We next show that $\bar{J}_k \geq J(\mathbf{C}^*)$ for any $k \leq m$.

Theorem 2. *For any $k \leq m$,*

$$\bar{J}_k \geq J(\mathbf{C}^*). \quad (45)$$

Proof:

$$J(\mathbf{C}^*) = \max_{\mathbf{C} \in \mathcal{C}^{[m \times n]}} J(\mathbf{C}) \quad (46)$$

$$= \max_{\mathbf{C}_1 \in \mathcal{C}^{[k \times n]}} \max_{\substack{\mathbf{C}_2 \in \mathcal{C}^{[(m-k) \times n]} \\ \mathbf{C}_2 \mathbf{C}_1^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{bmatrix} \right) \quad (47)$$

$$\leq \max_{\mathbf{C}_1 \in \mathcal{C}^{[k \times n]}} \max_{\substack{\mathbf{F} \in \mathcal{F}^{[(m-k) \times n]} \\ \mathbf{F}\mathbf{C}_1^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C}_1 \\ \mathbf{F} \end{bmatrix} \right) \quad (48)$$

$$= \bar{J}_k, \quad (49)$$

where the inequality follows from the set relationship $\mathcal{C}^{[(m-k) \times n]} \subset \mathcal{F}^{[(m-k) \times n]}$. □

Remark 4. *Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of $\Sigma_{\underline{X}}$. Then, for $k = 0$,*

$$\bar{J}_0 = J^*_{\left\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right\rangle} = \sum_{j=1}^m \frac{\lambda_j^2}{\lambda_j + \sigma^2}. \quad (50)$$

Remark 5. *When $k = m$,*

$$\bar{J}_m = J(\mathbf{C}^*) \quad (51)$$

We now show that the upper bounds are nested.

Theorem 3. *For any $k \leq m - 1$,*

$$\bar{J}_k \geq \bar{J}_{k+1}. \quad (52)$$

Proof:

$$\bar{J}_{k+1} = \max_{\mathbf{C} \in \mathcal{C}^{[(k+1) \times n]}} \max_{\substack{\mathbf{F} \in \mathcal{F}^{[(m-k-1) \times n]} \\ \mathbf{F}\mathbf{C}^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C} \\ \mathbf{F} \end{bmatrix} \right) \quad (53)$$

$$= \max_{\mathbf{C}_1 \in \mathcal{C}^{[k \times n]}} \max_{\substack{\underline{e} \in \mathcal{C}^{[1 \times n]} \\ \mathbf{C}_1 \underline{e}^T = \mathbf{0}}} \max_{\substack{\mathbf{F} \in \mathcal{F}^{[(m-k-1) \times n]} \\ \mathbf{F}\mathbf{C}_1^T = \mathbf{0} \\ \mathbf{F}\underline{e}^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C}_1 \\ \underline{e} \\ \mathbf{F} \end{bmatrix} \right) \quad (54)$$

$$\leq \max_{\mathbf{C}_1 \in \mathcal{C}^{[k \times n]}} \max_{\substack{\underline{f} \in \mathcal{F}^{[1 \times n]} \\ \mathbf{C}_1 \underline{f}^T = \mathbf{0}}} \max_{\substack{\mathbf{F} \in \mathcal{F}^{[(m-k-1) \times n]} \\ \mathbf{F}\mathbf{C}_1^T = \mathbf{0} \\ \mathbf{F}\underline{f}^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C}_1 \\ \underline{f} \\ \mathbf{F} \end{bmatrix} \right) \quad (55)$$

$$= \max_{\mathbf{C}_1 \in \mathcal{C}^{[k \times n]}} \max_{\substack{\mathbf{F}_1 \in \mathcal{F}^{[(m-k) \times n]} \\ \mathbf{F}_1 \mathbf{C}_1^T = \mathbf{0}}} J \left(\begin{bmatrix} \mathbf{C}_1 \\ \mathbf{F}_1 \end{bmatrix} \right) \quad (56)$$

$$= \bar{J}_k, \quad (57)$$

where the inequality is the consequence of the relationship $\mathcal{C}^{[1 \times n]} \subset \mathcal{F}^{[1 \times n]}$. □

Corollary 3.1.

$$\sum_{j=1}^m \frac{\lambda_j^2}{\lambda_j + \sigma^2} = \bar{J}_0 \geq \bar{J}_1 \geq \cdots \geq \bar{J}_m = J(\mathbf{C}^*). \quad (58)$$

Proof: Combine (50)–(52). □

We next want to utilize Theorem 1 (more specifically, Corollary 1.1) to efficiently compute the upper bounds $\bar{J}_k$. To that end, we define the matrix pencil $\left\langle \mathbf{A}_{(\mathbf{C})}, \mathbf{B}_{(\mathbf{C})} \right\rangle$ as a permutation of the matrix pencil $\left\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right\rangle$.

Definition. *Let $\bar{\mathbf{C}}$ denote **a complement** of $\mathbf{C}$, with constraints $\bar{\mathbf{C}} \in \mathcal{C}^{[(n-m) \times n]}$ and $\bar{\mathbf{C}} \mathbf{C}^T = \mathbf{0}$.*

Definition. *We define*

$$\mathbf{A}_{(\mathbf{C})} = \begin{bmatrix} \mathbf{C} \\ \bar{\mathbf{C}} \end{bmatrix} \Sigma_{\underline{X}}^2 \begin{bmatrix} \mathbf{C} \\ \bar{\mathbf{C}} \end{bmatrix}^T, \quad (59)$$

and

$$\mathbf{B}_{(\mathbf{C})} = \begin{bmatrix} \mathbf{C} \\ \bar{\mathbf{C}} \end{bmatrix} \left(\Sigma_{\underline{X}} + \sigma^2 \mathbf{I} \right) \begin{bmatrix} \mathbf{C} \\ \bar{\mathbf{C}} \end{bmatrix}^T. \quad (60)$$

Combining Theorem 1 and Lemma B, we now reformulate the upper bounds $\bar{J}_k$, so that the bounds can be calculated in terms of the generalized eigenvalues of a matrix pencil of smaller dimensions.

Corollary 3.2.

$$\bar{J}_k = \max_{\mathbf{C} \in \mathcal{C}^{[k \times n]}} J^{(k)*}_{\left\langle \mathbf{A}_{(\mathbf{C})}, \mathbf{B}_{(\mathbf{C})} \right\rangle}. \quad (61)$$

Proof: Let $\mathbf{F} \in \mathcal{F}^{[(m-k)\times(n-k)]}$ and let $\mathbf{F}_1 = \mathbf{F}\bar{\mathbf{C}}$, for any $\mathbf{C} \in \mathcal{C}^{[k\times n]}$. Then, from (18), (59), and (60), it follows that

$$J_{\langle \mathbf{A}_{(\mathbf{C})}, \mathbf{B}_{(\mathbf{C})}\rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix}\right) = J_{\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}}+\sigma^2\mathbf{I}\rangle}\left(\begin{bmatrix} \mathbf{C} \\ \mathbf{F}_1 \end{bmatrix}\right). \quad (62)$$

Since $\mathrm{rank}(\mathbf{F}_1) = \mathrm{rank}(\mathbf{F}) = m-k$, and $\mathbf{F}_1\mathbf{C}^T = \mathbf{0}$, using (31), (44), and (62) we can write

$$\bar{J}_k = \max_{\mathbf{C}\in\mathcal{C}^{[k\times n]}} \max_{\substack{\mathbf{F}_1\in\mathcal{F}^{[(m-k)\times n]} \\ \mathbf{F}_1\mathbf{C}^T=\mathbf{0}}} J_{\langle \Sigma_{\underline{X}}^2, \Sigma_{\underline{X}}+\sigma^2\mathbf{I}\rangle}\left(\begin{bmatrix} \mathbf{C} \\ \mathbf{F}_1 \end{bmatrix}\right) \quad (63)$$

$$= \max_{\mathbf{C}\in\mathcal{C}^{[k\times n]}} \max_{\mathbf{F}\in\mathcal{F}^{[(m-k)\times(n-k)]}} J_{\langle \mathbf{A}_{(\mathbf{C})}, \mathbf{B}_{(\mathbf{C})}\rangle}\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{F} \end{bmatrix}\right) \quad (64)$$

$$= \max_{\mathbf{C}\in\mathcal{C}^{[k\times n]}} J_{\langle \mathbf{A}_{(\mathbf{C})}, \mathbf{B}_{(\mathbf{C})}\rangle}^{(k)*}. \quad (65)$$

$\square$

For any $k \leq m$, the computation of the upper bound $\bar{J}_k$ requires searching over all $\binom{n}{k}$ matrices $\mathbf{C} \in \mathcal{C}^{[k\times n]}$. Therefore, computation of $\bar{J}_k$ has search complexity $\mathcal{O}\left(\binom{n}{k}\right)$.

V. PROJECTION-BASED APPROXIMATE SOLUTIONS

In the previous section, we obtained the subspace that maximizes the modified optimization problem (20) under the constraint $\mathbf{F} \in \mathcal{F}^{[m\times n]}$. The rows of the solution $\mathbf{F}^*$, i.e. $\underline{v}_1^T, \ldots, \underline{v}_m^T$ form a (not necessarily orthogonal) basis of the maximizing subspace. However, the solution subspace of the original optimization problem (18) has unit row vectors as its basis. Hence, we may adopt the approximate solution strategy that finds the best subspace spanned by unit vectors using a projection approach. That is, we project n unit row vectors[2] onto the subspace $\mathbf{F}^*$ to find good choices for m rows of $\mathbf{C}$ (sensor locations).

So, instead of utilizing the efficacy as our reward function, here we utilize the norm of the projection as the reward function. Thereby, similar to the approaches taken in Section III, we may now develop several projection-based approximate solutions (the expedient, greedy, and n-path greedy) that have varying search complexities.

A) Expedient projection-based solution

This is a single shot solution. Let $z(\underline{e}_k)$ be the norm of the projection of the vector $\underline{e}_k$ onto the subspace spanned by $\mathbf{F}^*$

$$z(\underline{e}_k) \triangleq \left\| \mathbf{F}^{*T}\left(\mathbf{F}^*\mathbf{F}^{*T}\right)^{-1}\mathbf{F}^*\underline{e}_k^T \right\|. \quad (66)$$

We rank the vectors $\underline{e}_k$ in descending order of $z(\underline{e}_k)$. For any $m \leq n$, the approximate solution $\mathbf{C}_{E\text{-}proj}$ is constructed by picking the m highest ranked vectors $\underline{e}_k$ as rows of $\mathbf{C}_{E\text{-}proj}$. The projection based expedient solution has the

[2]The projection of a unit row vector $\underline{e}_k$ on the subspace $\mathbf{F}$ is given by $\mathbf{F}^T(\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F}\underline{e}_k^T$.

same search complexity as the efficacy-based expedient solution, i.e. $\mathcal{O}(n\log n)$. Owing to their low search complexity, both are extremely good candidates for a solution when a quick first-order approximation is needed, e.g., such when starting an iterative refinement algorithm [15]. A direct comparison of the efficacy-based expedient solution $\mathbf{C}_E$ and the projection-based expedient solution $\mathbf{C}_{E\text{-}proj}$ cannot be made *a priori*, but must be done on a case by case basis.

B) Greedy projection-based solution

From Corollary 3.2 and Lemma B, we know that for any lower order $k < m$, an optimal low-order matrix $\mathbf{F}$ (achieving the upper bound $\bar{J}_k$) consists of the generalized eigenvectors of a lower-dimension pencil $\langle \mathbf{A}_k, \mathbf{B}_k \rangle$ given by (35) and (36). Assuming that the generalized eigenvalues of $\langle \mathbf{A}_k, \mathbf{B}_k \rangle$ are sorted in descending order, the matrix that upper bounds the efficacy is given by (27) as $\mathbf{F}_{\langle \mathbf{A}_k, \mathbf{B}_k\rangle}^* = \begin{bmatrix} \mathbf{I}_{m-k+1} & \mathbf{0} \end{bmatrix}\mathbf{V}_{\langle \mathbf{A}_k, \mathbf{B}_k\rangle}^T$, where $\mathbf{V}_{\langle \mathbf{A}_k, \mathbf{B}_k\rangle}^T$ is the generalized eigenvector matrix of $\langle \mathbf{A}_k, \mathbf{B}_k \rangle$. The greedy algorithm takes advantage of this property. In each iteration $k < m$, we assume that the solution is known for k sensors, and we pick the $(k+1)$-st sensor location as the unit row vector $\underline{e}$ that has the highest norm when projected onto the subspace spanned by $\mathbf{F}_{\langle \mathbf{A}_k, \mathbf{B}_k\rangle}^*$.

Greedy Algorithm [18]

1. Initialization: Set iteration $k = 1$, $\mathbf{C}_{k-1} = []$, and calculate $\mathbf{A}_{(\mathbf{C}_{k-1})}$ and $\mathbf{B}_{(\mathbf{C}_{k-1})}$ using (59) and (60), respectively.
2. Find $\mathbf{A}_k$, $\mathbf{B}_k$ using (35) and (36).
3. Set $\mathbf{F} = \begin{bmatrix} \mathbf{I}_{m-k+1} & \mathbf{0} \end{bmatrix}\mathbf{V}_{\langle \mathbf{A}_k, \mathbf{B}_k\rangle}^T$.
4. Find $\underline{e}^* = \arg\max_{\underline{e}\in\mathcal{C}^{[1\times(n-k+1)]}} \left\| \mathbf{F}^T\left(\mathbf{F}\mathbf{F}^T\right)^{-1}\mathbf{F}\underline{e}^T \right\|$.
5. Set $\mathbf{C}_k = \begin{bmatrix} \mathbf{C}_{k-1} \\ \underline{e}^*\bar{\mathbf{C}}^{k-1} \end{bmatrix}$.
6. If $k = m$, set $\mathbf{C}_{G\text{-}proj} = \mathbf{C}_k$ and stop, else set $k \leftarrow k+1$ and go to step 2.

The projection-based greedy algorithm has search complexity $\mathcal{O}(mn)$.

C) n-path greedy projection-based solution

In the projection-based n-path greedy algorithm, n candidate solutions are constructed by initializing the projection-based greedy algorithm with each of n unit row vectors $\underline{e}_k$. The best of the n candidate solutions is picked as the approximate solution. The projection-based n-path greedy algorithm runs in polynomial time with search complexity $\mathcal{O}(mn^2)$.

VI. SIMULATION RESULTS

We considered different test scenarios to evaluate the performances of the approximate solutions and the nested family of upper bounds $\bar{J}_k$ (for $k = 0$–5). In Section VI.A we considered realizations of the covariance matrix $\Sigma_{\underline{X}}$ generated at random for different system sizes $n = 20$, and 50.
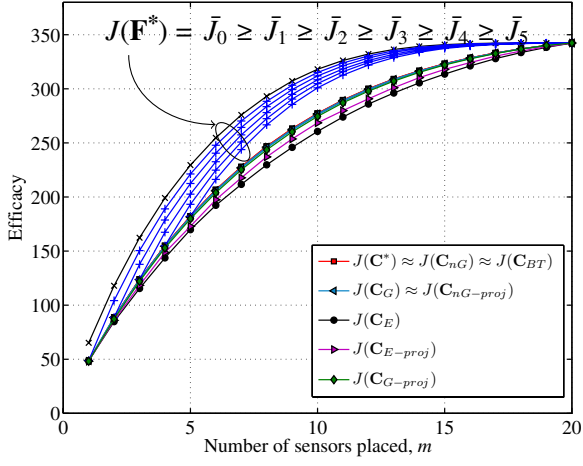
Fig. 1. $n = 20$: average efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$ compared to the average optimal efficacy $J(\mathbf{C}^*)$.
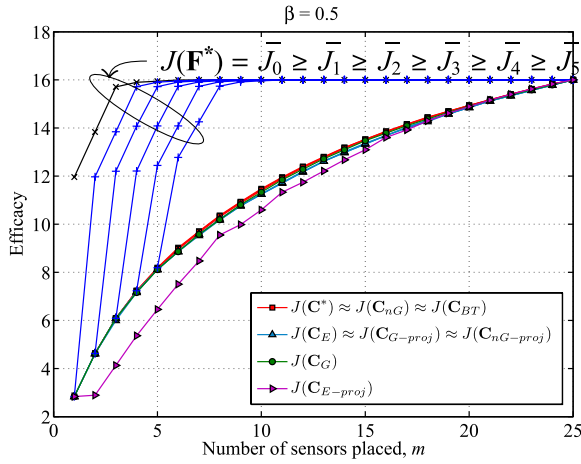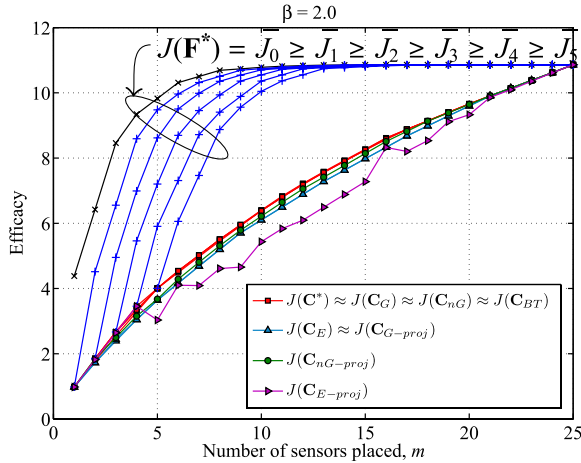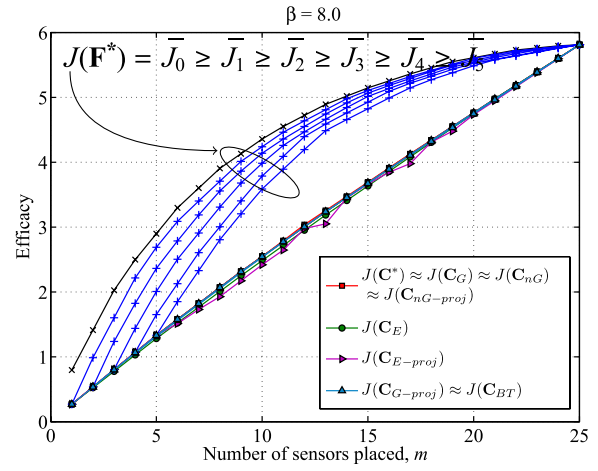

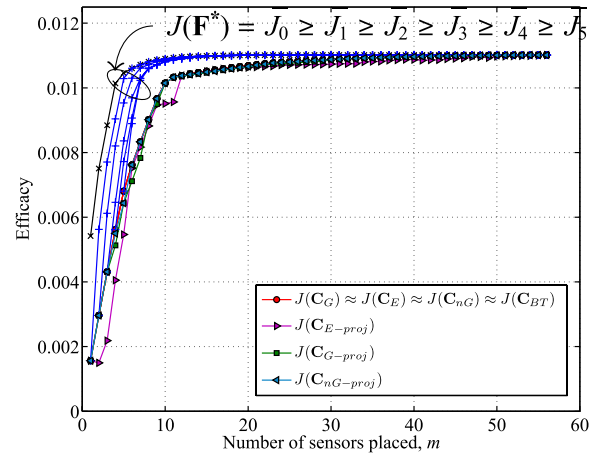**Fig. 2.** $n = 50$: average efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$.

Next in Section VI.B we created a 5×5 grid with unit distance between neighboring points, i.e. each point is located at unit distance from its horizontal and vertical neighbors. We generated a covariance matrix using a Gaussian distribution, where the variance between the points, considered as vectors $\underline{x}_1$ and $\underline{x}_2$ is given by [8]:

$$\sum \odot (\underline{x}_1, \underline{x}_2) = \exp\left(-\beta \left\| \underline{x}_1 - \underline{x}_2 \right\|_2^2 \Big/ (2\pi)\right), \quad (67)$$

where $\beta > 0$. Finally, we used the standard IEEE 57 bus test system [1] in Section VI.C to show the performances of our proposed algorithms.

A) Simulations with data generated at random

First, we consider the case where $n = 20$. We generated 100 realizations of the correlation matrix Σ_X at random; σ^2 was kept constant for all realizations of Σ_X. We compared the efficacies of the expedient, optimal, greedy, n-path greedy, and backtraced n-path solutions and the upper bounds for each of the 100 realizations.

Figure 1 shows the average efficacies and the upper bounds, averaged over 100 realizations. From Fig. 1 we note that both efficacy-based expedient solution and projection-based expedient solution get reasonably close to the optimal solution for each value of m. In this scenario, the projection-based expedient solution performs better than the efficacy-based one. However, the efficacy-based greedy solution performs better than the projection-based greedy solution. The projection-based n-path greedy does better than the efficacy-based greedy, but does worse than the efficacy-based n-path greedy and backtraced n-path solution. Also the upper bound $J(\mathbf{F}^*) = \bar{J}_0$ is not tight, but the nested bounds $\bar{J}_k$ become tighter as k increases.

We then considered the case for $n = 50$. For this case, we could not compute the optimal solution because of the prohibitive complexity when $n = 50$. In Fig. 2, we observe that the projection-based expedient algorithm performs better than the efficacy-based expedient algorithm for some values of m, whereas the latter performs better for other values of
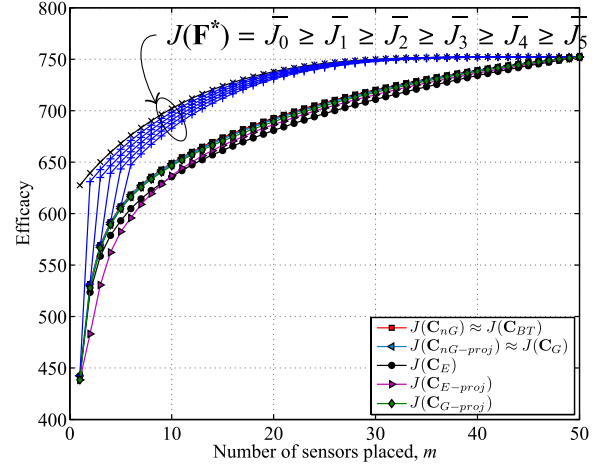
Table 1. Average runtime of proposed algorithms.

Algorithms		Average Runtime (sec)	
		$n = 20$, $m = 10$	$n = 50$, $m = 25$
Efficacy based	Expedient	0.0003	0.0005
	Greedy	0.008	0.15
	n-path greedy	0.22	10.93
	Backtraced n-path	0.25	12.76
Projection based	Expedient	0.0003	0.001
	Greedy	0.01	0.13
	n-path Greedy	0.2	6.21

m. We also observe that the projection-based n-path greedy algorithm does not always perform better than the efficacy-based greedy algorithm. Therefore, we cannot make an *a priori* comparison between these algorithms. However, we note that the efficacy-based n-path greedy solution and the backtraced n-path solution perform better than the rest of the algorithms. The upper bound in this case is not tight. Table 1 shows a comparison of average execution time of different proposed algorithms on an Intel Core i3 @ 2.10 GHz machine with 4 GB RAM. From the table, we observe that the projection-based algorithms have similar average runtime as the efficacy-based algorithms.

B) 5×5 grid model

Next, we apply the proposed algorithms to a 5×5 grid model where the variance between two points is given in (67). We increased the parameter β from 0.5 to 8.0 in three steps to evaluate the solutions. From Figs. 3–5, we observe that the projection-based expedient solution is not always monotonically increasing. Since the dimension of the upper bounding subspace $\mathbf{F}^*$ increases as m is increased, the dimension of the subspace closest to $\mathbf{F}^*$ also increases. As a result the solution subspace for $(m + 1)$ sensors does not always contain the solution subspace for m sensors.

Fig. 3. $\beta = 0.5$: efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$ compared to the optimal efficacy $J(\mathbf{C}^*)$.

Fig. 5. $\beta = 8.0$: Efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$ compared to the optimal efficacy $J(\mathbf{C}^*)$.

Fig. 4. $\beta = 2.0$: efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$ compared to the optimal efficacy $J(\mathbf{C}^*)$.

Fig. 6. IEEE 57-bus test case (voltage magnitudes): efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$.

Thus, the efficacy obtained by projection-based expedient algorithm is not always monotonic with increasing m.

We also note that the projection-based greedy and the n-path greedy algorithm perform better than the efficacy-based expedient solution, but perform worse than the efficacy-based greedy solution in Fig. 3. But the projection-base n-path greedy performs better than efficacy-based greedy solution in Fig. 4, and performs as good as the efficacy-based n-path greedy and backtraced solutions in Fig. 5. The efficacy-based n-path greedy solution and the backtraced n-path solution are almost indistinguishable from the optimal solution in Figs. 3 and 4. In Fig. 5, the back-traced algorithm performs slightly worse than the efficacy-based n-path greedy algorithm. The family of upper bounds from Section IV is not tight in Figs. 3–5. However, the bounds get tighter as β increases.

C) IEEE 57-bus test system

We next apply the proposed sensor placement algorithms to the IEEE 57-bus test system [1]. Traditional state estimators

(SE) in the power grid utilize the redundant measurements taken by supervisory control and data acquisition (SCADA) systems, and the states are estimated using iterative algorithms [22]. With the advent of phasor technology, time synchronized measurements can be obtained using PMUs. The PMUs can directly measure the states at the PMU-installed buses, and the states of all the connected buses (given enough channels are available) [23]. Thus, the measurement model (1) can be used as the PMU measurement model for the state estimation problem in the power grid. The storage and computational costs of the SE approaches may be further reduced by assuming the voltage magnitudes and phase angles are independent [24, 25]. Thus, we consider two independent measurement models for PMU placement, one for voltage magnitudes and the other for phase angles. For simulation purposes, without loss of generality, we assume that the conventional measurements of the test bus system are available to us. We further assume that a sensor is always placed at the swing bus [23] and therefore, the swing bus is not considered in our sensor placement algorithms.
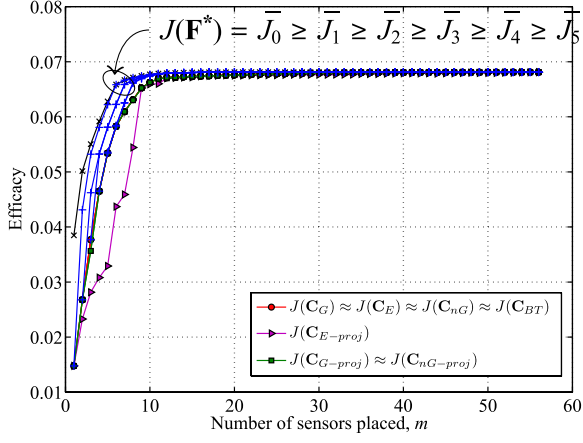
Fig. 7. IEEE 57-bus test case (phase angles): efficacies of approximate solutions and the upper bound $J(\mathbf{F}^*)$.

For each measurement model, we construct the sample covariance matrix Σ_X by running traditional SE algorithms 1000 times. We used the iterative state estimator based on non-linear AC power flow equations in the MATPOWER software package [26] for this purpose. Since the measurement noise in the traditional state estimator is Gaussian, the estimated state vectors are also Gaussian [9]. In the state estimation process, the standard deviations for voltage magnitudes, bus power injections, and line power flows measurements are 0.01, 0.015, and 0.02, respectively, in accordance with the setups in [26]. The simulations results of our proposed algorithms and nested bounds for voltage magnitude measurement model and phase angle measurement model are shown in Figs. 6 and 7, respectively. Again due to the size of the system, the exhaustive search for optimal solution was not possible. But it is observed that, for both measurement models, all of the proposed solutions perform very close to each other.

VII. SUMMARY AND FURTHER DIRECTIONS

This paper considers the optimal placement of m sensors among n locations. The optimal solution involves solving an integer-programming problem and is NP-complete [17]. We came up with a series of approximation algorithms, including a greedy algorithm and a dynamic programming-based algorithm, and their variations that have polynomial complexities. Through simulations we verified that their performances are close to the optimal solution. To further understand the performance of the optimal solution we come up with a series of nested upper bounds (using matrix pencils, generalized eigenvectors, and matrix manipulations) that give tighter upper bounds at increasing complexity. From these nested upper bounds, we found a new set of approximation algorithms that are based on finding projections of unit binary vectors on the subspace generated by the nested upper bounds.

There are many further directions for this research. We would like to come up with tighter upper bounds for the optimal performance. Through simulations we showed that many of the approximation algorithms perform quite well when compared to the optimal algorithm. It would be interesting to come up with tighter theoretical bounds. Algorithms can also be considered when there are additional constraints to the problem (e.g., sensors are not allowed at certain locations), for incremental cases (e.g., want to add some sensors given some sensors have already been placed, which is a generalization of the greedy algorithm), and for dynamic state estimation problems (important for state estimation for the electrical power grid).

In the simulation section, we had an example for an IEEE 57-bus test system. Approximation algorithms perform well and we could apply these algorithms for placement of meters at the distribution level for microgrid state estimation problems. The microgrid has specific local and hierarchical correlations that can be represented by graphs that for the most part are radial. In these cases, perhaps distributed algorithms (such as *belief propagation*) can be used to find good approximate solutions to the sensor placement problem.

APPENDIX A: PROOF OF THEOREM 1

Proof: Using (18), for any $\mathbf{F} \in \mathcal{F}^{[m \times n]}$, we have

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{F}) = \operatorname{tr}\left\{ \left[\mathbf{F}\mathbf{B}\mathbf{F}^T \right]^{-1} \mathbf{F}\mathbf{A}\mathbf{F}^T \right\}. \tag{A.1}$$

Let

$$\mathbf{G} = \mathbf{F}\mathbf{B}\mathbf{F}^T. \tag{A.2}$$

Since $\mathbf{G}$ is symmetric positive definite, there exists an $m \times m$ invertible matrix $\mathbf{G}^{1/2}$ such that

$$\mathbf{G} = \mathbf{G}^{1/2}\mathbf{G}^{T/2}. \tag{A.3}$$

Define $\mathbf{M}$ of size $m \times n$ as

$$\mathbf{M} = \mathbf{G}^{-1/2}\mathbf{F}\mathbf{V}^{-T}. \tag{A.4}$$

Combining Lemma A with (A.3)–(A.4) we have

$$\mathbf{M}\mathbf{M}^T = \mathbf{I}_m. \tag{A.5}$$

Further applying Lemma A and (A.2)–(A.4), the efficacy now reads

$$J_{\langle \mathbf{A}, \mathbf{B} \rangle}(\mathbf{F}) = \operatorname{tr}\left[\mathbf{G}^{-1}\mathbf{F}\mathbf{A}\mathbf{F}^T \right] \tag{A.6}$$

$$= \operatorname{tr}\left[\mathbf{G}^{-1}\mathbf{F}\mathbf{V}^{-T}\mathbf{D}\mathbf{V}^{-1}\mathbf{F}^T \right] \tag{A.7}$$

$$= \operatorname{tr}\left[\mathbf{G}^{-1}\mathbf{G}^{1/2}\mathbf{M}\mathbf{D}\mathbf{M}^T\mathbf{G}^{T/2} \right] \tag{A.8}$$

$$= \operatorname{tr}\left[\mathbf{M}\mathbf{D}\mathbf{M}^T \right]. \tag{A.9}$$

[22] Abur, A.; Expósito, A.G.: Power System State Estimation: Theory and Implementation, *Marcel Dekker*, New York, NY, 2004.

[23] Phadke, A.G.; Thorp, J.S.: Synchronized Phasor Measurements and Their Applications, *Springer*, 2008.

[24] Huang, Y.-F.; Werner, S.; Huang, J.; Kashyap, N.; Gupta, V.: State estimation in electric power grids: meeting new challenges presented by the requirements of the future grid. IEEE Signal Process, Mag., **29**(5) (2012), 33–43.

[25] Monticelli, A.: Electric power system state estimation. Proc. IEEE, **88**(2) (2000), 262–282.

[26] Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J.: MAT-POWER: steady-state operations, planning, and analysis tools for power systems research and education. IEEE Trans. Power Syst., **26**(1) (2011), 12–19.

[27] Abadir, K.M.; Magnus, J.: Matrix Algebra, *Princeton University Press*, Princeton, NJ, 2008.

Muhammad Sharif Uddin received his B.Sc. degree in Electrical and Electronic Engineering from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 2009. He is currently pursuing the Ph.D. degree in Electrical Engineering at the University of Hawaii at Manoa. His research interests include sensor networks, signal processing, machine learning, and smart grid applications.

Anthony Kuh received his B.S. degree in Electrical Engineering and Computer Science at the University of California, Berkeley in 1979, an M.S. degree in Electrical Engineering from Stanford University in 1980, and a Ph.D. degree in Electrical Engineering from Princeton University in 1987. Dr. Kuh previously worked at AT&T Bell Laboratories and has been on the faculty in Electrical Engineering at the University of Hawaii since 1986. He is currently a Professor in the Department and is also currently serving as the director of the Interdisciplinary Renewable Energy and Island Sustainability (REIS) Group. Previously, he served as the Department Chair of Electrical Engineering. Dr. Kuh's research is in the area of neural networks and machine learning, adaptive signal processing, sensor networks, communication networks, and renewable energy and smart grid applications. Dr. Kuh won a National Science Foundation Presidential Young Investigator Award, received a Distinguished Fulbright Scholar's Award working at Imperial College, London, and is an IEEE Fellow. He is currently serving as the IEEE Signal Processing Society Regions 1-6 Director at Large, on the Board of Governors of the Asia Pacific Signal and Information Processing Association, and as a senior editor of the IEEE Journal of Selected Topics in Signal Processing.

Aleksandar Kavcic received the Dipl. Ing. degree in Electrical Engineering from Ruhr-University, Bochum, Germany in 1993, and the Ph.D. degree in Electrical and Computer Engineering from Carnegie Mellon University in 1998. Currently, he is a Professor of Electrical Engineering at the University of Hawaii. Professor Kavcic received the IBM Partnership Award in 1999 and NSF CAREER Award in 2000. He is a co-recipient of the 2005 IEEE Best Paper Award in Signal Processing and Coding for Data Storage. He served as Associate Editor of the IEEE Transactions on Information Theory from 2001 to 2004, as Guest Editor of the IEEE Signal Processing Magazine in 2003–2004, and as Guest Editor of the IEEE Journal on Selected Areas in Communications in 2008–2009. From 2005 until 2007, he was the Chair of the DSTC of the IEEE Communications Society, and in 2014 was the general co-chair of the IEEE ISIT in Honolulu, Hawaii.

Toshihisa Tanaka received the B.E., M.E., and Ph.D. degrees from Tokyo Institute of Technology in 1997, 2000, and 2002, respectively. He joined the Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology, in 2004, where he is currently an Associate Professor. In 2005, he was a Royal Society Visiting Fellow at Imperial College London, UK. In 2011, he was a Visiting Faculty Member at University of Hawaii. He is a coeditor of Signal Processing Techniques for Knowledge Extraction and Information Fusion (Springer 2008). His research interests include image and signal processing, statistical signal processing and machine learning, biomedical signal processing, and adaptive systems.

Dr. Tanaka has served on the editorial board for several major international journals. He is a Past-Chair of the TC on Biomedical Signal Processing, APSIPA. He is a senior member of the IEEE, and a member of the IEICE and APSIPA.