

D-Optimum Scanning Node Activation for Large-Scale Monitoring Networks

Dariusz Uciński

Abstract—A method is developed to solve an optimal node activation problem in sensor networks whose measurements are supposed to be used to estimate unknown parameters of the underlying process model. Given a partition of the observation horizon into a finite number of consecutive intervals, the problem is set up to select gaged sites on each interval so that the determinant of the Fisher information matrix associated with the estimated parameters be maximal. The approach adopted here to circumvent the combinatorial nature of the sensor selection problem consists in operating on the spatial density of sensors, rather than on the sensor locations. The original problem then reduces to maximizing the determinant of the sum of finite convex combinations of some nonnegative definite matrices subject to additional box constraints on the weights of these combinations. Some separability characterizations of optimal solutions are indicated first, and then the block coordinate ascent method combined with simplicial decomposition is applied to obtain numerical solutions. As a result, a simple computational scheme is obtained which can be implemented without resorting to sophisticated numerical software.

I. INTRODUCTION

The design of proper measurement strategies for parameter estimation of Distributed Parameter Systems (DPSs) is extremely important in applications such as, e.g., environmental protection [1]. The impossibility to observe the system states over the entire spatial domain implies the question of where to locate discrete sensors so as to accurately estimate the unknown system parameters. This problem has received much attention so far, but the results communicated by most authors are limited to the selection of stationary sensor positions (for reviews, see [2]–[5]).

A natural generalization is to apply sensors which are capable of tracking points providing most valuable information about the parameters. However, communications in this field are rather limited and primarily focus on observations from sensors placed on platforms having some degree of mobility, e.g., mobile robots or pilotless aircraft. This setup was initiated in [6], where the determinant of the Fisher Information Matrix (FIM) was employed as a measure of the identification accuracy and optimal time-dependent measures were sought, rather than the trajectories themselves. This was then generalized in [3], [7]–[10], where many computational algorithms were put forward to attack the associated state-constrained optimal-control problems. Their adaptation to the sensor-network setting was reported in [11], [12].

In many situations, however, the observation system comprises multiple sensors whose positions are already specified

and it is desired to activate only a subset of them during a given time interval while the other sensors remain dormant. For example, this is the usual framework when sensor networks are employed. Such a scanning strategy of taking measurements can equivalently be interpreted in terms of a group of sensors which switch their positions at discrete time instants.

The problem was first exposed in [13] and then more thoroughly examined in [3, Ch. 4]. The key idea originates in the seminal work [14] (see also [15], [16]) and consists in operating on the density of sensors per unit area instead of the positions of individual sensors. This permits to relax the discrete optimization problem in context and to replace it by a more convenient continuous approximation. The transformation is warranted in the case of a relatively large number of measurement sensors and a predefined set of switching times. Mathematically, the relevant procedure involves looking for a family of ‘optimal’ probability measures defined on subsets of the set of feasible measurement points. In spite of its somewhat abstract assumptions, the resulting algorithms of the exchange type to solve the scanning sensor guidance problem are very easy to implement and extremely fast.

In turn, the scanning measurement problem with free switching times was attacked in [17] by treating it as an optimal discrete-valued control problem which is then transformed into an equivalent continuous-valued optimal-control formulation. In principle, the latter method is only applicable to situations when the number of scanning sensors to be scheduled is rather moderate, as it does not prevent the drawback of the “curse of dimensionality” inherent to the original combinatorial problem. In turn, the work [18] extended the sensor selection scheme based on the branch-and-bound scheme set forth in [4] for stationary sensors. As a result, a simple, yet powerful, computational scheme was developed to deal with problems of small or moderate sizes.

The aim of the present note is to propose a better alternative to the above-mentioned greedy exchange algorithm presented in [3, Ch. 4]. The latter works well in most cases, but it relies on some heuristics and its global convergence has not been proven yet (therefore, its use requires many restarts in the hope of getting better solutions). Our starting point is the approach outlined in [19], where the problem of determining optimal sensor densities was reduced to the search for an optimal probability mass function. It was then efficiently solved via simplicial decomposition. The present study was undertaken to demonstrate that, with some turning up involving the block coordinate ascent method, the same technique can be easily adapted to the case of scanning

Dariusz Uciński is with Institute of Control and Computation Engineering, University of Zielona Góra, ul. Podgórna 50, 65-246 Zielona Góra, Poland d.ucinski@issi.uz.zgora.pl

sensors.

Notation: Throughout the paper, \mathbb{R}_+ and \mathbb{R}_{++} stand for the sets of nonnegative and positive real numbers, respectively. The set of real $m \times n$ matrices is denoted by $\mathbb{R}^{m \times n}$. We use \mathbb{S}^m to denote the set of symmetric $m \times m$ matrices, \mathbb{S}_+^m to denote the set of symmetric nonnegative definite $m \times m$ matrices, and \mathbb{S}_{++}^m to denote the set of symmetric positive definite $m \times m$ matrices. The curled inequality symbol \succeq (resp. \succ) is used to denote generalized inequalities. More precisely, between vectors, it represents a componentwise inequality, and between symmetric matrices, it represents the Löwner ordering: given $\mathbf{A}, \mathbf{B} \in \mathbb{S}^m$, $\mathbf{A} \succeq \mathbf{B}$ means that $\mathbf{A} - \mathbf{B}$ is nonnegative definite (resp. positive definite). The symbols $\mathbf{1}$ and $\mathbf{0}$ denote vectors whose all components are one and zero, respectively. The context makes their lengths clear.

Given a set of points A , $\text{conv}(A)$ stands for its convex hull. The probability (or conical) simplex in \mathbb{R}^n is defined as $S_n = \text{conv}(\{\mathbf{e}_1, \dots, \mathbf{e}_n\}) = \{\mathbf{p} \in \mathbb{R}_+^n \mid \mathbf{1}^\top \mathbf{p} = 1\}$, where \mathbf{e}_j is the usual unit vector along the j -th coordinate of \mathbb{R}^n .

Given two vectors \mathbf{x} and \mathbf{y} of dimension n , $\mathbf{x} \cdot \mathbf{y}$ is an n -vector whose i -th component is $x_i y_i$ (the componentwise multiplication operator).

II. SCANNING SENSOR LOCATION PROBLEM

A. Expected Estimation Accuracy

Let $\Omega \subset \mathbb{R}^d$ be a bounded spatial domain with sufficiently smooth boundary $\partial\Omega$, and let $T = (0, t_f]$ be a bounded time interval. Consider a DPS whose scalar state at a spatial point $\mathbf{x} \in \bar{\Omega} \subset \mathbb{R}^d$ and time instant $t \in \bar{T}$ is denoted by $y(\mathbf{x}, t; \boldsymbol{\theta})$, where $\boldsymbol{\theta} \in \mathbb{R}^m$ is a vector of unknown parameters which must be estimated using observations of the state.

Consider an arbitrary partition of the time interval T by choosing points $t_0 < t_1 < \dots < t_K = t_f$ defining subintervals $T_k = (t_{k-1}, t_k]$, $k = 1, \dots, K$ called *scanning stages*. It is assumed here that, over each time interval T_k , the state y is continuously observed by N pointwise sensors. The sensors will possibly be changing their locations at the beginning of every time subinterval but will be remaining stationary for the duration of each of the subintervals. In other words, the measurement process can be formally represented as

$$z_k^j(t) = y(\mathbf{x}_k^j, t; \boldsymbol{\theta}) + \varepsilon(\mathbf{x}_k^j, t), \quad t \in T_k \quad (1)$$

for $j = 1, \dots, N$ and $k = 1, \dots, K$, where $z_k^j(t)$ is the scalar output, $\mathbf{x}_k^j \in X$ stands for the location of the j -th sensor on the subinterval T_k , X signifies the part of Ω where the measurements can be made (X is assumed to be a compact subset of Ω), and ε denotes the measurement noise. It is customary to assume that the noise is zero-mean, Gaussian, uncorrelated in both time and space [5].

We assume that the parameter estimate $\hat{\boldsymbol{\theta}}$ is defined as the solution to the usual output least-squares formulation of the parameter estimation problem [20]. Determination of sensor positions which would guarantee the best accuracy of the least-squares estimates of $\boldsymbol{\theta}$ amounts to choosing \mathbf{x}_k^j for $k = 1, \dots, K$ and $j = 1, \dots, N$ so as to maximize some scalar

measure of performance Ψ defined on the *Fisher Information Matrix* (FIM) [6]

$$\mathbf{M} = \sum_{k=1}^K \sum_{j=1}^N \int_{T_k} \mathbf{g}(\mathbf{x}_k^j, t) \mathbf{g}^\top(\mathbf{x}_k^j, t) dt, \quad (2)$$

where $\mathbf{g}(\mathbf{x}, t) = \nabla_{\boldsymbol{\theta}} y(\mathbf{x}, t; \boldsymbol{\theta})|_{\boldsymbol{\theta}=\boldsymbol{\theta}^0}$ stands for the so-called *sensitivity vector*, $\boldsymbol{\theta}^0$ being a prior estimate to the unknown parameter vector $\boldsymbol{\theta}$ [21]. Up to a constant multiplier, the inverse of the FIM constitutes an approximation to the covariance matrix of any unbiased estimator of $\boldsymbol{\theta}$ [22], [23].

As for Ψ , various choices exist [22], [23], but the most common option is to maximize the D-optimality criterion

$$\Psi(\mathbf{M}) = \log \det(\mathbf{M}), \quad (3)$$

which yields the minimum volume of the uncertainty ellipsoid for the estimates. In the sequel, we define Ψ to be just (3).

The introduction of an optimality criterion makes it possible to formulate the sensor location problem as an optimization one: Maximize $\Psi(\mathbf{M})$ with respect to \mathbf{x}_k^j , $k = 1, \dots, N$ and $j = 1, \dots, N$ belonging to the admissible set X .

B. Conversion to the problem of finding optimal sensor densities

As the number of sensors N becomes very large, which is more and more common, e.g., in applications involving sensor networks, the optimal sensor location problem gets extremely hard from a computational point of view. A way to overcome this predicament is to operate on the spatial density of sensors (i.e., the number of sensors per unit area), rather than on the sensor locations [3], [14].

For the time subinterval T_k , the density of sensors can be approximately described by a probability measure $\xi_k(d\mathbf{x})$ on the space (X, \mathcal{B}) , where \mathcal{B} is the σ -algebra of all Borel subsets of X . Feasible solutions of this form make it possible to apply convenient and efficient mathematical tools of convex programming theory. As regards the practical interpretation of the so produced results (provided that we are in a position to calculate at least their approximations), partitioning X into subdomains X_i , $i = 1, \dots, n$ of relatively small areas, we allocate to each of them the number

$$N_k^i = \left\lceil N \int_{X_i} \xi_k(d\mathbf{x}) \right\rceil \quad (4)$$

of sensors on the interval T_k (here $\lceil \rho \rceil$ is the smallest integer greater than or equal to ρ).

Accordingly, we define the class of admissible designs $\Xi(X)$ as all ordered K -tuples of the form

$$\xi = (\xi_1, \dots, \xi_K), \quad (5)$$

where the probability measures ξ_k over X are absolutely continuous with respect to the Lebesgue measure. Thus,

$$\int_X \xi_k(d\mathbf{x}) = 1, \quad k = 1, \dots, K. \quad (6)$$

Accordingly, we replace (2) by

$$\mathbf{M}(\xi) = \sum_{k=1}^K \int_X \mathbf{r}_k(\mathbf{x}) \xi_k(d\mathbf{x}), \quad (7)$$

where

$$\mathbf{r}_k(\mathbf{x}) = \int_{T_k} \mathbf{g}(\mathbf{x}, t) \mathbf{g}^\top(\mathbf{x}, t) dt$$

and the integration in (6) and (7) is to be understood in the Lebesgue-Stieltjes sense.

C. Reformulation Using Clusterization-Free Designs

Independent observations are convenient from a theoretical point of view, but they can hardly be justified when in an optimal solution several sensors are to take measurements near one another (this phenomenon is called *sensor clusterization*). Indeed, in the spatial data collection schemes there is usually no possibility of replicated measurements, i.e., different sensors cannot take measurements at one point without influencing one another. Anyway, several sensors situated in the close vicinity of one another usually do not give more information than a single sensor.

In order to avoid such clustered sensor configurations, we impose the crucial restriction that the density of sensor allocation must not exceed some prescribed level. This amounts to the following condition [14], [15]:

$$\xi_k(d\mathbf{x}) \leq \omega(d\mathbf{x}), \quad k = 1, \dots, K, \quad (8)$$

where $\omega(d\mathbf{x})$ signifies the maximal possible ‘number’ of sensors per $d\mathbf{x}$ [14]. Clearly,

$$\int_X \omega(d\mathbf{x}) \geq 1. \quad (9)$$

Consequently, we are faced with the following optimization problem:

Problem 1: Find

$$\xi^* = \arg \max_{\xi \in \Xi(X)} \Psi[\mathbf{M}(\xi)] \quad (10)$$

subject to

$$\xi_k(d\mathbf{x}) \leq \omega(d\mathbf{x}), \quad k = 1, \dots, K. \quad (11)$$

The design ξ^* above is then said to be a (D, ω) -optimal design on the analogy of the definition introduced in [15] in the context of directly constrained design measures.

A proper mathematical formulation calls for the following assumptions:

- (A1) X is compact,
- (A2) $\mathbf{g}(\cdot, \cdot)$ is continuous,
- (A3) $\omega(d\mathbf{x})$ is atomless, i.e., for any $\Delta X \subset X$ there exists a $\Delta X' \subset \Delta X$ such that

$$\int_{\Delta X'} \omega(d\mathbf{x}) < \int_{\Delta X} \omega(d\mathbf{x}). \quad (12)$$

In what follows, we write $\bar{\Xi}(X) \subset \Xi(X)$ for the collection of all the design measures (5) which satisfy the requirement¹

$$\xi_k(\Delta X) = \begin{cases} \omega(\Delta X) & \text{for } \Delta X \subset \text{supp } \xi_k, \\ 0 & \text{for } \Delta X \subset X \setminus \text{supp } \xi_k, \end{cases} \quad (13)$$

$k = 1, \dots, K$. Given a design ξ , we will say that the function $\psi_k(\cdot, \xi)$ defined by

$$\psi_k(\mathbf{x}, \xi) = \int_{T_k} \mathbf{g}^\top(\mathbf{x}, t) \mathbf{M}^{-1}(\xi) \mathbf{g}(\mathbf{x}, t) dt, \quad (14)$$

separates sets X_1 and X_2 with respect to $\omega(d\mathbf{x})$ if for any two sets $\Delta X_1 \subset X_1$ and $\Delta X_2 \subset X_2$ with equal non-zero measures we have

$$\int_{\Delta X_1} \psi_k(\mathbf{x}, \xi) \omega(d\mathbf{x}) \geq \int_{\Delta X_2} \psi_k(\mathbf{x}, \xi) \omega(d\mathbf{x}). \quad (15)$$

We can now formulate the main result which provides a characterization of (D, ω) -optimal designs, cf. [3, Thms. 4.1 and 4.2].

Theorem 1: Let Assumptions (A1)–(A3) hold. Then:

- (i) There exists an optimal design $\xi^* \in \bar{\Xi}(X)$, and
- (ii) A necessary and sufficient condition for $\xi^* \in \bar{\Xi}(X)$ to be (D, ω) -optimal is that all the functions $\psi_k(\cdot, \xi^*)$, $k = 1, \dots, K$ separate $X_k^* = \text{supp } \xi_k^*$ and $X \setminus X_k^*$ with respect to the measure $\omega(d\mathbf{x})$.

From a practical point of view, the above result means that at all the support points of an optimal measure ξ_k^* the mapping $\psi_k(\cdot, \xi^*)$ should be greater than anywhere else, i.e., preferably $\text{supp } \xi_k^*$ should coincide with maximum points of $\psi_k(\cdot, \xi^*)$. In practice, this amounts to allocating observations to the points at which we know least of all about the system response [14].

III. CONVERSION TO WEIGHT OPTIMIZATION

We are now confronted with the question of how to discretize Problem 1 to make it tractable. In what follows, the basic idea is to make use of the partition of X into a union of small disjoint subdomains X_i , $i = 1, \dots, n$, i.e., $X = \bigcup_{i=1}^n X_i$, as discussed in Section II-B. Observe that a measure $\xi \in \Xi(X)$ assigns each subdomain X_i a weight $p_k^i = \xi_k(X_i)$. Owing to (4), the knowledge of the p_k^i 's suffices to determine an optimal distribution of sensor nodes. Assuming that the variations of all $\mathbf{r}_k(\cdot)$ over each X_i are negligible (this can be achieved by constructing a sufficiently

¹The support of a measure ξ is defined as the closed set $\text{supp } \xi = X \setminus \bigcup\{G : \xi(G) = 0, G \text{ - open}\}$, cf. [24, p.80].

fine partition of X), we have

$$\begin{aligned}
M(\xi) &= \sum_{k=1}^K \int_{X_k} \Upsilon_k(\mathbf{x}) \xi_k(d\mathbf{x}) \\
&\approx \sum_{k=1}^K \int_{X_k} \left(\sum_{i=1}^n \Upsilon_k(\mathbf{x}_i) \mathbb{1}_{X_i}(\mathbf{x}) \right) \xi_k(d\mathbf{x}) \\
&= \sum_{k=1}^K \sum_{i=1}^n \Upsilon_k(\mathbf{x}_i) \int_{X_k} \mathbb{1}_{X_i}(\mathbf{x}) \xi_k(d\mathbf{x}) \\
&= \sum_{k=1}^K M_k(\mathbf{p}_k),
\end{aligned} \tag{16}$$

where

$$\mathbf{p} = (\mathbf{p}_1, \dots, \mathbf{p}_K), \tag{17}$$

$$\mathbf{p}_k = (p_k^1, \dots, p_k^n), \quad k = 1, \dots, K, \tag{18}$$

$$M_k(\mathbf{p}_k) = \sum_{i=1}^n p_k^i \Upsilon_k(\mathbf{x}_i) \tag{19}$$

\mathbf{x}^i is an arbitrary point in X_i (e.g., the centre of gravity), $\mathbb{1}_A$ stands for the indicator function of a set A . Thus, setting $\mathbf{b} = (\omega(X_1), \dots, \omega(X_n))$, we arrive at the following approximation to Problem 1:

Problem 2: Find a vector of weights $\mathbf{p} \in \mathbb{R}^{Kn}$ to maximize

$$\Phi(\mathbf{p}) = \log \det \left(\sum_{k=1}^K M_k(\mathbf{p}_k) \right) \tag{20}$$

subject to

$$\mathbf{0} \preceq \mathbf{p}_k \preceq \mathbf{b}, \quad k = 1, \dots, K, \tag{21}$$

$$\mathbf{1}^\top \mathbf{p}_k = 1, \quad k = 1, \dots, K. \tag{22}$$

Without restriction of generality, we shall further assume that $\mathbf{b} \succ \mathbf{0}$. The constraint set P for each \mathbf{p}_k , defined by (21) and (22), constitutes the intersection of the probability simplex S_n and the box $B = \{\mathbf{q} \in \mathbb{R}^n \mid \mathbf{0} \preceq \mathbf{q} \preceq \mathbf{b}\}$, which is a rather nice convex set. This property is going to be used for effective weight optimization.

IV. ALGORITHM FOR WEIGHT OPTIMIZATION

A. Block-Coordinate Ascent

Recall that the log-determinant is concave and strictly concave over the cones \mathbb{S}_+^m and \mathbb{S}_{++}^m , respectively, cf. [25]. What is more, it is easy to see that the sets $\mathcal{M}_k = \{M_k(\mathbf{q}) \mid \mathbf{q} \in \mathbb{R}^n, \mathbf{0} \preceq \mathbf{q} \preceq \mathbf{b}, \mathbf{1}^\top \mathbf{q} = 1\}$ are closed convex subsets of \mathbb{S}^m , cf. [3, Lem. 3.2]. Interpreting Problem 2 as maximization of the log-determinant of a sum of matrices over the set $\mathcal{M}_1 \times \dots \times \mathcal{M}_K$, we check at once that this setting makes it possible to apply the block coordinate ascent method to solve it, cf., e.g., [26, Prop. 2.7.1]. The simplicity of this idea and the associated ease of implementation were the crucial factors behind its adaptation to numerically solve Problem 2. The resulting computational scheme is outlined as Algorithm 1.

Algorithm 1 Algorithm model for block coordinate ascent.

Step 0: (Initialization)

Guess a starting point $\mathbf{p}^{(0)} = (\mathbf{p}_1^{(0)}, \dots, \mathbf{p}_K^{(0)}) \in P^n$. Set $\tau = 0$.

Step 1: (Cyclic coordinate update)

For $k = 1, \dots, K$ consecutively update \mathbf{p}_k :

$$\mathbf{p}_k^{(\tau+1)} = \arg \max_{\mathbf{q} \in P} \Phi(\mathbf{p}_1^{(\tau+1)}, \dots, \mathbf{p}_{k-1}^{(\tau+1)}, \mathbf{q}, \mathbf{p}_{k+1}^{(\tau)}, \dots, \mathbf{p}_K^{(\tau)}) \tag{23}$$

Step 2: (Termination check) If

$$\Phi(\mathbf{p}_1^{(\tau+1)}, \dots, \mathbf{p}_K^{(\tau+1)}) = \Phi(\mathbf{p}_1^{(\tau)}, \dots, \mathbf{p}_K^{(\tau)}), \tag{24}$$

then STOP. Otherwise, increment τ by one and go back to Step 1.

Observe that implementation of Step 1 in Algorithm 1 amounts to solving the following auxiliary problem:

Problem 3: Given $\mathbf{D}_0, \mathbf{D}_1, \dots, \mathbf{D}_n \in \mathbb{S}_+^m$, find $\mathbf{q} = (q_1, \dots, q_n) \in P$ which maximizes

$$\Gamma(\mathbf{q}) = \log \det \left(\mathbf{D}_0 + \sum_{i=1}^n q_i \mathbf{D}_i \right) \tag{25}$$

subject to

$$\mathbf{0} \preceq \mathbf{q} \preceq \mathbf{b} \tag{26}$$

$$\mathbf{1}^\top \mathbf{q} = 1. \tag{27}$$

Its solution will be the main topic of the remainder of this section.

B. Simplicial Decomposition for Problem 3

1) *Algorithm model:* The structure of Problem 3 makes it well suited for solution using Simplicial Decomposition (SD). This technique proved extremely useful for large-scale pseudoconvex programming problems [27]. In its basic form, it proceeds by alternately solving linear and nonlinear programming subproblems, called the *column generation problem* (CGP) and the *restricted master problem* (RMP), respectively. In the RMP, the original problem is relaxed by replacing the original constraint set P with its inner approximation being the convex hull of a finite set of feasible solutions. In the CGP, this inner approximation is improved by incorporating a point in the original constraint set that lies furthest along the gradient direction computed at the solution of the RMP. This basic strategy has been discussed in numerous references, cf. [27], where possible extensions have also been proposed. A marked characteristic of the SD method is that the sequence of solutions to the RMP tends to a solution to the original problem in such a way that the objective function strictly monotonically approaches its optimal value.

The SD algorithm may be viewed as a form of modular nonlinear programming, provided that one has an effective computer code for solving the RMP, as well as access to a code which can take advantage of the linearity of the

CGP. Our purpose here is to show that this is the case within the framework of Problem 3. What is more, since we deal with maximization of the concave function Φ over a bounded polyhedral set P , this will automatically imply the convergence of the resulting SD scheme in a finite number of RMP steps.

Tailoring the SD scheme to our needs, we obtain Algorithm 2, in which

$$D(\mathbf{q}) = D_0 + \sum_{i=1}^n q_i D_i, \quad (28)$$

$$\begin{aligned} \phi(\mathbf{q}) &= \nabla \Gamma(\mathbf{q}) \\ &= \left[\text{tr}\{D^{-1}(\mathbf{q})(D_0 + D_1)\}, \dots, \right. \\ &\quad \left. \text{tr}\{D^{-1}(\mathbf{q})(D_0 + D_n)\} \right]^T \end{aligned} \quad (29)$$

In the sequel, its consecutive steps will be discussed in turn.

Algorithm 2 Algorithm model for solving Problem 3 via simplicial decomposition.

Step 0: (Initialization)

Guess an initial solution $\mathbf{q}^{(0)} \in P$ such that $D(\mathbf{q}^{(0)})$ is nonsingular. Set $I = \{1, \dots, n\}$, $Q^{(0)} = \{\mathbf{q}^{(0)}\}$ and $\mu = 0$.

Step 1: (Termination check)

Set

$$I_{\text{ub}}^{(\mu)} = \{i \in I \mid q_i^{(\mu)} = b_i\}, \quad (30)$$

$$I_{\text{im}}^{(\mu)} = \{i \in I \mid 0 < q_i^{(\mu)} < b_i\}, \quad (31)$$

$$I_{\text{lb}}^{(\mu)} = \{i \in I \mid q_i^{(\mu)} = 0\}. \quad (32)$$

If

$$\phi_i(\mathbf{q}^{(\mu)}) \begin{cases} \geq \lambda & \text{if } i \in I_{\text{ub}}^{(\mu)}, \\ = \lambda & \text{if } i \in I_{\text{im}}^{(\mu)}, \\ \leq \lambda & \text{if } i \in I_{\text{lb}}^{(\mu)} \end{cases} \quad (33)$$

for some $\lambda \in \mathbb{R}_+$, then STOP and $\mathbf{q}^{(\mu)}$ is optimal.

Step 2: (Solution of the column generation subproblem)

Compute

$$\mathbf{r}^{(\mu+1)} = \arg \max_{\mathbf{q} \in P} \phi(\mathbf{q}^{(\mu)})^T \mathbf{q} \quad (34)$$

and set

$$Q^{(\mu+1)} = Q^{(\mu)} \cup \{\mathbf{r}^{(\mu+1)}\}. \quad (35)$$

Step 3: (Solution of the restricted master subproblem)

Find

$$\mathbf{q}^{(\mu+1)} = \arg \max_{\mathbf{q} \in \text{conv}(Q^{(\mu+1)})} \Gamma(\mathbf{q}) \quad (36)$$

and purge $Q^{(\mu+1)}$ of all extreme points with zero weights in the resulting expression of $\mathbf{q}^{(\mu+1)}$ as a convex combination of elements in $Q^{(\mu+1)}$. Increment μ by one and go back to Step 1.

C. Characterization of the optimal design and termination of Algorithm 2

In the original SD setting, the criterion for terminating the iterations is checked only after solving the column generation problem. The computation is then stopped if the current point $\mathbf{q}^{(\mu)}$ satisfies the ‘‘basic’’ optimality condition of nonincrease, to first order, in performance measure value in the whole constraint set, i.e.,

$$\max_{\mathbf{q} \in P} \phi(\mathbf{q}^{(\mu)})^T (\mathbf{q} - \mathbf{q}^{(\mu)}) \leq 0. \quad (37)$$

The condition (33) is less costly in terms of the number of floating-point operations. It results from the following characterization of \mathbf{q}^* which has the property that $\Gamma(\mathbf{q}^*) = \max_{\mathbf{q} \in P} \Gamma(\mathbf{q})$.

Theorem 2: Suppose that the matrix $D(\mathbf{q}^*)$ is nonsingular for some $\mathbf{q}^* \in P$. The vector \mathbf{q}^* constitutes a global maximum of Γ over P if, and only if, there exists a number λ^* such that

$$\phi_i(\mathbf{q}^*) \begin{cases} \geq \lambda^* & \text{if } q_i^* = b_i, \\ = \lambda^* & \text{if } 0 < q_i^* < b_i, \\ \leq \lambda^* & \text{if } q_i^* = 0 \end{cases} \quad (38)$$

for $i = 1, \dots, n$.

This result follows from direct application of Lemma 1 in the Appendix after setting $f(\mathbf{q}) = \Gamma(D(\mathbf{q}))$.

D. Solution of the column generation subproblem

In Step 2 of Algorithm 2 we deal with the linear programming problem

$$\begin{aligned} &\text{maximize} && \mathbf{c}^T \mathbf{q} \\ &\text{subject to} && \mathbf{q} \in P, \end{aligned} \quad (39)$$

where $\mathbf{c} = \phi(\mathbf{q}^{(\mu)})$, in which the feasible region is defined by $2n$ bound constraints (26) and one equality constraint (27).

The algorithm to solve this problem is almost as simple as a closed-form solution. The key idea is to make use of the following assertion being a direct consequence of Lemma 1 from the Appendix.

Theorem 3: A vector $\mathbf{q} \in P$ constitutes a global solution to the problem (39) if, and only if, there exists a scalar ρ such that

$$c_i \begin{cases} \geq \rho & \text{if } q_i = b_i, \\ = \rho & \text{if } 0 < q_i < b_i, \\ \leq \rho & \text{if } q_i = 0 \end{cases} \quad (40)$$

for $i = 1, \dots, n$.

We thus see that, in order to solve (39), it is sufficient to pick the consecutive largest components c_i of \mathbf{c} and set the corresponding weights q_i as their maximal allowable values b_i . The process is repeated until the sum of the assigned weights exceeds one. Then the value of the last weight which was set in this manner should be corrected so as to satisfy the constraint (27) and the remaining (i.e., unassigned) weights are set as zeros. This straightforward scheme is implemented as Algorithm 3. Note that its correctness requires satisfaction of the condition $\mathbf{b} \preceq \mathbf{1}$, which is by no means restrictive.

Algorithm 3 Algorithm model for solving the column generation subproblem.

Step 0: (Initialization)

Set $j = 0$ and $v^{(0)} = 0$.

Step 1: (Sorting)

Sort the elements of c in nonincreasing order, i.e., find a permutation π on the index set $I = \{1, \dots, n\}$ such that

$$c_{\pi(i)} \geq c_{\pi(i+1)}, \quad i = 1, \dots, n-1 \quad (41)$$

Step 2: (Identification of nonzero weights)

Step 2.1: If $v^{(j)} + b_{\pi(j+1)} < 1$ then set

$$v^{(j+1)} = v^{(j)} + b_{\pi(j+1)}. \quad (42)$$

Otherwise, go to Step 3.

Step 2.2: Increment j by one and go to Step 2.1.

Step 3: (Form the ultimate solution)

Set

$$q_{\pi(i)} = \begin{cases} b_{\pi(i)} & \text{for } i = 1, \dots, j, \\ 1 - v^{(j)} & \text{for } i = j+1, \\ 0 & \text{for } i = j+2, \dots, n. \end{cases} \quad (43)$$

E. Solution of the restricted master subproblem

Suppose that in the $(\mu + 1)$ -th iteration of Algorithm 2, we have

$$Q^{(\mu+1)} = \{\mathbf{r}_1, \dots, \mathbf{r}_s\}, \quad (44)$$

possibly with $s < \mu + 1$ owing to the built-in deletion mechanism of points in $Q^{(j)}$, $1 \leq j \leq \mu$, which did not contribute to the convex combinations yielding the corresponding iterates $\mathbf{q}^{(j)}$. Step 3 of Algorithm 2 involves maximization of the design criterion (20) over

$$\text{conv}(Q^{(\mu+1)}) = \left\{ \sum_{j=1}^s w_j \mathbf{r}_j \mid \mathbf{w} \succeq \mathbf{0}, \mathbf{1}^\top \mathbf{w} = 1 \right\}. \quad (45)$$

From the representation of any $\mathbf{q} \in \text{conv}(Q^{(\mu+1)})$ as

$$\mathbf{q} = \sum_{j=1}^s w_j \mathbf{r}_j, \quad (46)$$

or, in component-wise form,

$$q_i = \sum_{j=1}^s w_j r_{j,i}, \quad i = 1, \dots, n, \quad (47)$$

$r_{j,i}$ being the i -th component of \mathbf{r}_j , it follows that

$$\begin{aligned} \mathbf{D}(\mathbf{q}) &= \mathbf{D}_0 + \sum_{i=1}^n q_i \mathbf{D}_i \\ &= \mathbf{D}_0 + \sum_{j=1}^s w_j \left(\sum_{i=1}^n r_{j,i} \mathbf{D}_i \right) = \sum_{j=1}^s w_j \mathbf{D}(\mathbf{r}_j) \end{aligned} \quad (48)$$

From this, we see that the RMP can equivalently be formulated as the following problem:

Problem 4: Find a sequence of weights $\mathbf{w} \in \mathbb{R}^s$ to maximize

$$\Psi(\mathbf{w}) = \log \det(\mathbf{H}(\mathbf{w})) \quad (49)$$

subject to the constraints

$$\mathbf{1}^\top \mathbf{w} = 1, \quad (50)$$

$$\mathbf{w} \succeq \mathbf{0}, \quad (51)$$

where

$$\mathbf{H}(\mathbf{w}) = \sum_{j=1}^s w_j \mathbf{H}_j, \quad \mathbf{H}_j = \mathbf{D}(\mathbf{r}_j), \quad j = 1, \dots, s. \quad (52)$$

A very simple and numerically effective sequential procedure for its solution was described in [3, p. 62], cf. also [4]. Its version adapted to the RMP proceeds as summarized in Algorithm 4, in which

$$\chi_j(\mathbf{w}) = \text{tr}[\mathbf{H}^{-1}(\mathbf{w}) \mathbf{H}_j], \quad j = 1, \dots, s. \quad (53)$$

Algorithm 4 Algorithm model for the restricted master problem.

Step 0: (Initialization)

Select a weight vector $\mathbf{w}^{(0)} \in S_s \cap \mathbb{R}_{++}^s$, e.g., set $\mathbf{w}^{(0)} = (1/s)\mathbf{1}$. Set $\ell = 0$.

Step 1: (Termination check)

If

$$\frac{1}{m} \chi(\mathbf{w}^{(\ell)}) \preceq \mathbf{1} \quad (54)$$

then STOP.

Step 2: (Multiplicative update)

Evaluate

$$\mathbf{w}^{(\ell+1)} = \frac{1}{m} \chi(\mathbf{w}^{(\ell)}) \cdot \mathbf{w}^{(\ell)}. \quad (55)$$

Increment ℓ by one and go to Step 1.

V. ILLUSTRATIVE EXAMPLE

Consider the two-dimensional diffusion equation

$$\frac{\partial y}{\partial t} = \nabla \cdot (\kappa \nabla y) + u \quad \text{in } \Omega \times T \quad (56)$$

where $\Omega \subset \mathbb{R}^2$ is the T-shaped spatial domain with boundary $\partial\Omega$ which is shown in Fig. 1 and $T = (0, 1]$. As regards the forcing term in our model, it has the form

$$u(\mathbf{x}, t) = 20 \exp(-50(x_1 - t)^2) \quad (57)$$

which mimics the action of a line source whose support is constantly oriented along the x_2 -axis and moves with constant speed from the left to the right boundary of Ω . The assumed form of the diffusion coefficient is

$$\kappa(\mathbf{x}) = \theta_1 + \theta_2 x_1 + \theta_3 x_2, \quad (58)$$

where θ_1 , θ_2 and θ_3 are unknown parameters which have to be estimated based on the measurements from a sensor

network. Throughout the design, $\theta_1^0 = 0.1$, $\theta_2^0 = -0.05$ and $\theta_3^0 = 0.2$ are to be used as nominal values of θ_1 , θ_2 and θ_3 , respectively.

The PDE (56) is supplemented with the initial and boundary conditions

$$y(\mathbf{x}, 0) = 5 \quad \text{in } \Omega, \quad (59)$$

$$y(\mathbf{x}, t) = 5(1 - t) \quad \text{on } \partial\Omega \times T. \quad (60)$$

The upper bound in (8) is given by

$$\omega(A) = c \frac{|A|}{|X|} \quad (61)$$

for any Borel subset of \bar{X} , where $c = 4.3 \geq 1$ is fixed, i.e., ω corresponds to a uniform distribution on \bar{X} , and $|A|$ stands for the area (i.e., the Lebesgue measure) of A .

The MATLAB PDE toolbox [28] was used to generate the triangular FEM mesh of 1811 nodes and $n = 3440$ triangles shown in Fig. 1. The triangles were then treated as the subdomains X_i .

The values of the sensitivities $\mathbf{g} = \text{col}[g_1, g_2, g_3]$ are defined as solutions to the following system of PDEs [3]:

$$\begin{cases} \frac{\partial y}{\partial t} = \nabla \cdot (\mu \nabla y), \\ \frac{\partial g_1}{\partial t} = \nabla \cdot \nabla y + \nabla \cdot (\mu \nabla g_1), \\ \frac{\partial g_2}{\partial t} = \nabla \cdot (x_1 \nabla y) + \nabla \cdot (\mu \nabla g_2), \\ \frac{\partial g_3}{\partial t} = \nabla \cdot (x_2 \nabla y) + \nabla \cdot (\mu \nabla g_3), \end{cases} \quad (62)$$

in which the first equation constitutes the original state equation and the other three equations result from its differentiation with respect to θ_1 , θ_2 and θ_3 , respectively. The initial and Dirichlet boundary conditions for the sensitivity equations are homogeneous.

We numerically solved (62) using some routines of the MATLAB PDE toolbox and stored matrices $\Upsilon(\mathbf{x}^i)$ computed based on g_1 , g_2 and g_3 interpolated at the gravity centres of individual triangles, cf. Appendix I in [3] for details.

All computations were performed using a low-cost PC (Intel Core 2 Quad CPU Q9400 2.66GHz \times 4, 8 GB RAM) running Ubuntu 12.04 and Matlab R2011b. In spite of that, any computational experiment took no more than 10 seconds.

Simplicial decomposition implemented in accordance with Algorithm 2 lead to solutions in which the weights p_i associated with the respective triangles approximately satisfy the ‘bang-bang’ principle formulated in Theorem 1: they are either zero, or equal to the corresponding upper bound $b_i = c|X_i|/|X|$. Thus, the support of the optimal measure ξ covers approximately $c^{-1} \cdot 100\% = 23.36\%$ of the area of $|X|$, cf. Fig. 1.

Figure 1(a) shows the optimal solution for the case of stationary sensors (i.e., $K = 1$). The value of the determinant of the corresponding FIM is 190.19. In turn, Figs. 1(b)–(f) show consecutive triangles in which sensors are to be placed for $K = 5$ scanning stages. Observe that the values of the diffusion coefficient $\kappa(\mathbf{x})$ in the upper left of Ω are

greater than those in the lower right. This means that the state changes during the system evolution are quicker when we move up and to the left (on the other hand, the system would have reached the steady state there earlier). This fact explains the form of the configurations obtained—the sensors switch to new positions so as to follow the moving source while measuring the state in the regions where the DPS is the most sensitive with respect to the unknown parameter κ , finally terminating the movement in the lower right part of the domain Ω . This time, the value of the determinant of the FIM is 460.82, i.e., it is 2.42 times higher than that for the case of stationary sensors. This confirms that scanning may contribute much to the achievable accuracy of parameter estimates.

VI. SUMMARY

A key aspect which implies the simplicity of the sensor placement technique presented in the paper and its negligible additional memory requirements is the possibility of applying the block coordinate ascent method. This results from nice concavity properties of the optimality criterion in the presented formulation. Consequently, the resulting computational procedure is only slightly more complicated than its counterpart for stationary sensors which was proposed in [19]. Clearly, there is still room for numerous refinements regarding the algorithm performance, but even the basic scheme presented here performs very well in practice.

REFERENCES

- [1] D. G. Cacuci, I. M. Navon, and M. Ionescu-Bujor, *Computational Methods for Data Evaluation and Assimilation*. Boca Raton, FL: CRC Press, 2014.
- [2] M. Patan, *Optimal Sensor Networks Scheduling in Identification of Distributed Parameter Systems*. Berlin: Springer-Verlag, 2012.
- [3] D. Uciński, *Optimal Measurement Methods for Distributed-Parameter System Identification*. Boca Raton, FL: CRC Press, 2005.
- [4] D. Uciński and M. Patan, “D-optimal design of a monitoring network for parameter estimation of distributed systems,” *Journal of Global Optimization*, vol. 39, pp. 291–322, 2007.
- [5] C. S. Kubrusly and H. Malebranche, “Sensors and controllers location in distributed systems — A survey,” *Automatica*, vol. 21, no. 2, pp. 117–128, 1985.
- [6] E. Rafajłowicz, “Optimum choice of moving sensor trajectories for distributed parameter system identification,” *International Journal of Control*, vol. 43, no. 5, pp. 1441–1451, 1986.
- [7] D. Uciński, “Optimal sensor location for parameter estimation of distributed processes,” *International Journal of Control*, vol. 73, no. 13, pp. 1235–1248, 2000.
- [8] D. Uciński and M. Patan, “Sensor network design for the estimation of spatially distributed processes,” *International Journal of Applied Mathematics and Computer Science*, vol. 20, no. 3, pp. 459–481, 2010.
- [9] M. Patan, “Distributed scheduling of sensor networks for identification of spatio-temporal processes,” *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 2, pp. 299–311, 2012.
- [10] M. Patan and D. Uciński, “Configuring a sensor network for fault detection in distributed parameter systems,” *International Journal of Applied Mathematics and Computer Science*, vol. 18, no. 4, pp. 513–524, 2008.
- [11] C. Tricaud and Y. Chen, *Optimal Mobile Sensing and Actuation Policies in Cyber-physical Systems*. London: Springer-Verlag, 2012.
- [12] Z. Song, Y. Chen, C. R. Sastry, and N. C. Tas, *Optimal Observation for Cyber-physical Systems: A Fisher-information-matrix-based Approach*. London: Springer-Verlag, 2009.
- [13] D. Uciński and M. Patan, “Optimal location of discrete scanning sensors for parameter estimation of distributed systems,” in *Proc. 15th IFAC World Congress*, Barcelona, Spain, 22–26 July 2002, 2002, published on CD-ROM.

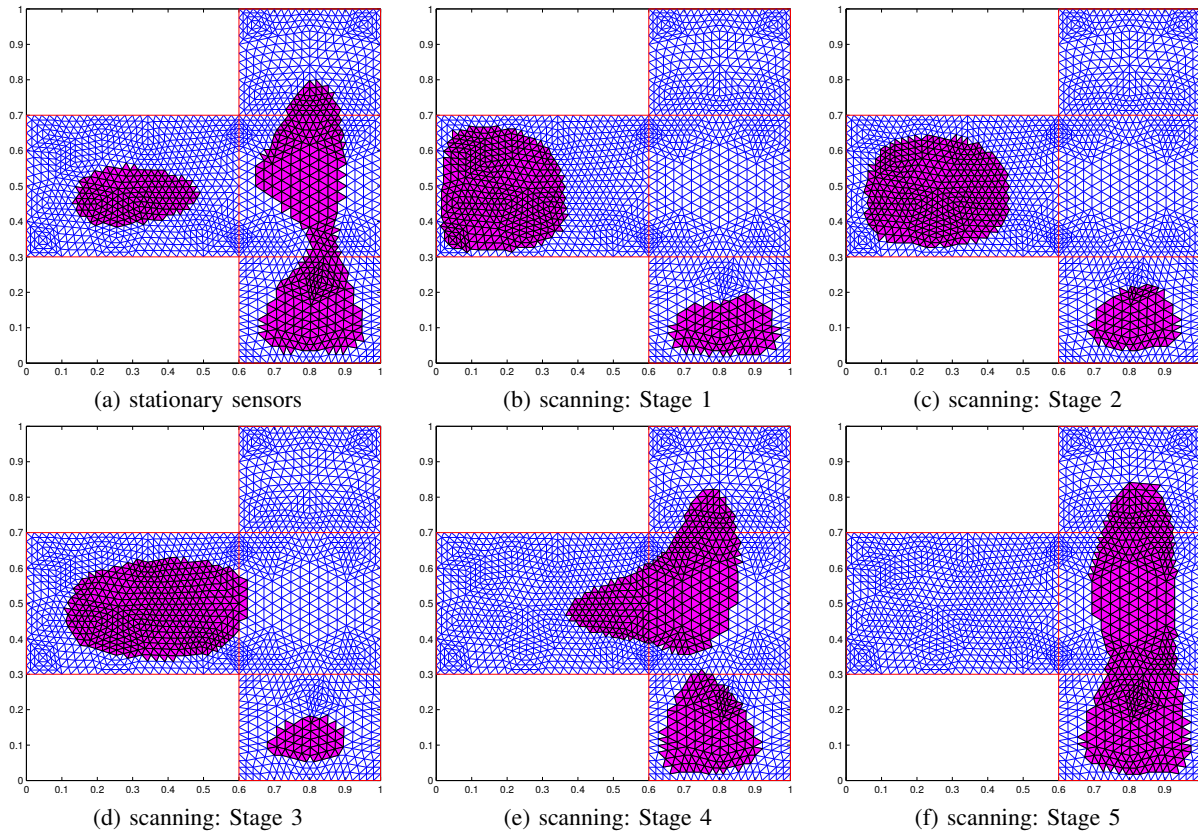


Fig. 1. Supports of the D-optimal design measures for stationary (a) and scanning (b)–(f) sensors.

[14] V. V. Fedorov, “Optimal design with bounded density: Optimization algorithms of the exchange type,” *Journal of Statistical Planning and Inference*, vol. 22, pp. 1–13, 1989.

[15] D. Cook and V. Fedorov, “Constrained optimization of experimental design,” *Statistics*, vol. 26, pp. 129–178, 1995.

[16] V. V. Fedorov and S. L. Leonov, *Optimal Design for Nonlinear Response Models*. Boca Raton, FL: CRC Press, 2014.

[17] M. Patan, “Optimal activation policies for continuous scanning observations in parameter estimation of distributed systems,” *International Journal of Systems Science*, vol. 37, no. 11, pp. 763–775, 2006.

[18] D. Uciński, “D-optimum sensor activity scheduling for distributed parameter systems,” in *Preprints of the 15th IFAC Symposium on System Identification*, Saint-Malo, France, July 6–8, 2009, published on CD-ROM.

[19] D. Uciński, “An algorithm to configure a large-scale monitoring network for parameter estimation of distributed systems,” in *Proceedings of the European Control Conference 2007*, Kos, Greece, July 2–5, 2007, published on CD-ROM.

[20] H. T. Banks and K. Kunisch, *Estimation Techniques for Distributed Parameter Systems*, ser. Systems & Control: Foundations & Applications. Boston: Birkhäuser, 1989.

[21] N.-Z. Sun, *Inverse Problems in Groundwater Modeling*, ser. Theory and Applications of Transport in Porous Media. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1994.

[22] L. Pronzato and A. Pázman, *Design of Experiments in Nonlinear Models. Asymptotic Normality, Optimality Criteria and Small-Sample Properties*. New York: Springer-Verlag, 2013.

[23] A. C. Atkinson, A. N. Donev, and R. D. Tobias, *Optimum Experimental Designs, with SAS*. Oxford: Oxford University Press, 2007.

[24] M. M. Rao, *Measure Theory and Integration*. New York: John Wiley & Sons, 1987.

[25] F. Pukelsheim, *Optimal Design of Experiments*, ser. Probability and Mathematical Statistics. New York: John Wiley & Sons, 1993.

[26] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed., ser. Optimization and Computation Series. Belmont, MA: Athena Scientific, 1999.

[27] M. Patriksson, “Simplicial decomposition algorithms,” in *Encyclopedia of Optimization*, C. A. Floudas and P. M. Pardalos, Eds. Dor-

recht, The Netherlands: Kluwer Academic Publishers, 2001, vol. 5, pp. 205–212.

[28] COMSOL AB, *Partial Differential Equation Toolbox for Use with Matlab. User’s Guide*. Natick, MA: The MathWorks, Inc., 1995.

APPENDIX

Given a vector $\mathbf{b} \in \mathbb{R}_{++}^n$ such that $\mathbf{1}^\top \mathbf{b} \geq 1$, consider Problem A: Maximize $f(\mathbf{p})$ subject to $\mathbf{p} \in P$ where we assume throughout that (a) $P = S_n \cap B$ for S_n being the probability simplex in \mathbb{R}^n and $B = \{\mathbf{p} \in \mathbb{R}^n \mid \mathbf{0} \leq \mathbf{p} \leq \mathbf{b}\}$, (b) the function $f : S_n \rightarrow \mathbb{R}$ is concave and continuously differentiable over P .

Lemma 1: A vector \mathbf{p}^* constitutes a global maximum of Problem A if, and only if, there exists a number λ^* such that

$$\frac{\partial f(\mathbf{p}^*)}{\partial p_i} \begin{cases} \geq \lambda^* & \text{if } p_i^* = b_i, \\ = \lambda^* & \text{if } 0 < p_i^* < b_i, \\ \leq \lambda^* & \text{if } p_i^* = 0 \end{cases} \quad (63)$$

for $i = 1, \dots, n$.

This characterization can be interpreted as separability of the components of \mathbf{p}^* in terms of the gradient $\nabla f(\mathbf{p}^*)$. Specifically, the values of $\partial f(\mathbf{p}^*)/\partial p_i$ for the components p_i^* of \mathbf{p}^* which are interior points of the closed intervals $[0, b_i]$ must equal some constant λ^* , whereas for the components p_i^* coinciding with endpoints 0 or b_i , the values of $\partial f(\mathbf{p}^*)/\partial p_i$ must be no larger and no smaller than λ^* , respectively.

This lemma can be proven using the first-order Karush-Kuhn-Tucker optimality conditions, cf. [4].