Data-Driven Sensor Scheduling for Remote Estimation in Wireless Networks

Marcos M. Vasconcelos and Urbashi Mitra

***Abstract*—Sensor scheduling is a well-studied problem in signal processing and control with numerous applications. Despite its successful history, most of the related literature assumes the knowledge of the underlying probabilistic model of the sensor measurements such as the correlation structure or the entire joint probability density function. Herein, a framework for sensor scheduling for remote estimation is introduced in which the system design and the scheduling decisions are based solely on observed data. Unicast and broadcast networks and corresponding receivers are considered. In both cases, the empirical risk minimization can be posed as a difference-of-convex optimization problem, and locally optimal solutions are obtained efficiently by applying the convex–concave procedure. Our results are independent of the data's probability density function, correlation structure, and the number of sensors.**

***Index Terms*—Decision theory, estimation, networked control systems, optimization, quantization, statistical learning.**

I. Introduction

Sensor scheduling is a classical problem in signal processing and control with a very rich history. The traditional static sensor scheduling problem consists of selecting a subset of k sensors among a group of n sensors such that the expected distortion between the random state-of-the-world and its estimate is minimized [1]. This class of problems has many applications in engineering, especially in sensor networks in which the number of sensors allowed to communicate with a remote fusion center is limited due to bandwidth constraints.

Consider the system described in the block diagram of Fig. 1, where n sensor–estimator pairs share a wireless network, which can operate either in unicast or broadcast modes. Each of the n sensors observes a distinct random variable and reports it to the scheduler. The scheduler selects a single random variable according to a scheduling decision rule and transmits it over the network. If the system is in unicast mode, only the intended estimator receives the sensor's observation, and the remaining estimators observe an erasure symbol. If the system is in broadcast mode, all the sensors receive the same transmitted measurement. Upon seeing the network output, each receiver forms its estimate according to an estimation policy. The system designer's goal is to select scheduling and estimation policies such as to minimize the mean-squared error (MSE) between the observations at the sensors and the estimates at the receivers. This problem lies in the category of team decision problems with a nonclassical information structure, which are, in general, very difficult to solve due to coupling between the scheduling and estimation policies known as signaling [2].

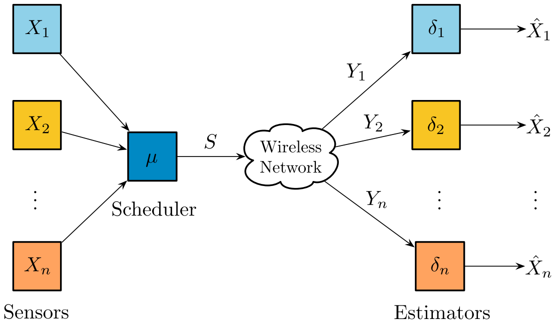

Fig. 1. Schematic diagram for the remote sensing system with n sensor–estimator pairs over a bandwidth constrained wireless network.

In addition to the classical applications of sensor scheduling, the framework proposed here can be used to model real-time communication between the Internet of Things (IoT) devices. Due to the massive number of devices and the very high demand for communication resources, the scheduler selects the pieces of information that are most relevant for a given task and discard the others, keeping the network data flow under control but at the same time achieving excellent system performance. A more specific application of interest is in systems known as wireless body area networks for remote health care monitoring [3]–[5]. In these systems, the sensors collect heterogeneous biometric data and transmit them to a mobile phone, which acts as a scheduler. To preserve battery life and meet bandwidth constraints, the mobile phone selects one of them to transmit it to one or multiple destinations.

Manuscript received July 20, 2020; revised July 30, 2020; accepted December 9, 2020. Date of publication January 8, 2021; date of current version August 24, 2021. This work was supported in part by ONR under Grant N00014-15-1-2550; in part by NSF under Grant CNS-1213128, Grant CCF-1718560, Grant CCF-1410009, and Grant CPS-1446901; and in part by AFOSR under Grant FA9550-12-1-0215. Recommended by Associate Editor S. L. Smith. *(Corresponding author: Marcos M. Vasconcelos.)*

Marcos M. Vasconcelos was with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA. He is now with Commonwealth Cyber Initiative, Virginia Tech, Arlington, VA 22203 USA (e-mail: marcosv@vt.edu).

Urbashi Mitra is with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: ubli@usc.edu).

Digital Object Identifier 10.1109/TCNS.2021.3050136

To the best of our knowledge, most of the literature in sensor scheduling assumes that the joint probability density function (PDF) of the random variables observed at the sensors is known *a priori* to the system designer. However, this is a restrictive assumption because, in most practical applications, this information is typically not available. The main challenge we address in this article is to design such a system in the absence of knowledge of the joint PDF, but in the presence of a dataset of independent and identically distributed (i.i.d.) samples, a standard assumption in statistical learning theory [6]. The results and algorithms presented here combine ideas from quantization theory [7], modern techniques in nonconvex optimization theory [8], [9], and classic results in stochastic programming [10] into a new class of sensor scheduling problems. The findings herein are meant to provide a guide to the art of designing such complex data-driven scheduling for remote estimation systems.

The main contributions of this work are as follows.

1) We provide a systematic data-driven approach for the joint design of scheduling and estimation rules for unicast and broadcast networks.
2) Our algorithm exploits the decompositions of nonconvex objectives as a difference-of-convex (DC) functions. It uses the convex–concave procedure (CCP) to find locally optimal solutions with a fast convergence rate.
3) Our algorithms are universal, working for any joint PDF that generates the dataset, and for any number of sensors.
4) We establish a connection between our algorithms and subgradient methods. The main advantage of our algorithms is that we do not need to select a step size at every iteration in an *ad hoc* manner. Our step sizes are constant and arise naturally from the CCP.

A. Related Literature

With the pervasiveness of data in the design of modern autonomous networked systems, we are experiencing the proliferation of machine learning techniques in control and estimation [11]. These emerging technologies have found applications in problems with unknown stochastic observations and disturbances. Robust control and estimation theory has been successful in designing systems to perform well under model uncertainty [12]. However, the ability to collect and analyze large datasets has allowed us to learn, with high probability, the model parameters of neural networks used to implement optimal policies without knowing the problem's underlying probabilistic model.

In the modern literature of sensor scheduling and control over unknown wireless channels, many articles have tackled the problem of determining an optimal scheduling policy when the statistical model of the measurements is known, but the wireless channel is uncertain [13]. Wu *et al.* [14] considered the sequential scheduling of a single sensor over a channel with unknown packet-drop probability. The strategy consists in estimating the unknown parameter and adapting the scheduling policy to the current estimate of the packet-drop probability. Leong *et al.* [15] considered a similar setup of sensor scheduling in a system with multiple sensors and a single estimator by formulating the problem as a Markov decision process and solving it using the deep Q-network technique. Li *et al.* [16] studied decentralized scheduling in a remote estimation system with multiple sensors and multiple estimators. They formulated the problem as a Markov game and solved it using the concept of Nash Q-learning.

Our problem formulation differs from the existing results in the literature in a fundamental aspect. While the references above consider an *unknown channel* and a *known probabilistic model* for the sensor observations, we consider the reverse situation, in which the *channel is known*, and the *underlying statistics of the observations are unknown*. Another difference between the works mentioned earlier is that they are sequential, and our problem formulation is static. Static sensor scheduling problems also play an important role in the literature, e.g., [1], [17] and references therein, both of which assume complete knowledge of the probability distributions.

Our problem formulation is related to the observation-driven sensor scheduling framework introduced in [18], where the underlying probabilistic model is Gaussian. The subsequent work [19] considered a sequential problem formulation with an energy-harvesting scheduler for sensors making independent observations distributed according to the general class of symmetric and unimodal PDFs. In this work, we study the data-driven version of [18] under minimal assumptions on the probabilistic model, namely, finite first and second moments. Unlike [1], [17]–[19], we do not make any assumptions on correlation, symmetry, and modality of the observations. Our main goal is to design systems suitable for any joint PDF without assumptions on the sensor observations' correlation structure. By relating the remote estimation problem with statistical learning theory [6], we provide a design framework for choosing a scheduler with performance close to the optimal with high probability.

Our problem falls in the broad area of machine learning for regression/estimation. The learning algorithm we use is a particular form of (controlled) piece-wise linear regression. The algorithm used to train the scheduler is the CCP, which we can map into a stochastic subgradient method with a specific constant step-size (or step-matrix, in the broadcast case). Our iterative schemes converge to a local minimum of nonconvex, nonsmooth objective functions. This is the first time the CCP is used in a sensor scheduling problem.

II. Problem Formulation

Consider the system depicted in Fig. 1 with $n \geq 2$ sensor–estimator pairs communicating via a constrained wireless network. We assume that the data observed at the sensors are realizations of the following continuous random vector

$$X \overset{\text{def}}{=} (X_1, X_2, \ldots, X_n) \tag{1}$$

which is distributed according to an arbitrary joint PDF, f_X. We also assume that each X_i, $i \in \{1, \ldots, n\}$ has finite first and second-order moments, which are the only assumptions on the underlying probabilistic model of the problem.

The sensors communicate the measurements to a scheduler. Due to bandwidth constraints, we assume that only one sensor measurement can be transmitted at a time. The scheduler's role is to choose which of the sensor measurements is transmitted over the network to its destination. The scheduling decision, $u \in \{1, \ldots, n\}$, is taken according to a policy $\mu : \mathbb{R}^n \to \{1, \ldots, n\}$ such that

$$u = \mu(x_1, \ldots, x_n). \tag{2}$$

When a sensor is chosen by the scheduler, a communication packet s containing its measurement and identification number is sent over the network, i.e., if $u = j$, then

$$s = (j, x_j). \tag{3}$$

In this work, we will consider *unicast* and *broadcast* networks. In the case of a *unicast network*, only the estimator associated with the chosen sensor receives the transmitted measurement. The remaining estimators receive a special erasure symbol denoted by $\varnothing$. In other words, if $u = j$, then

$$y_i = \begin{cases} (j, x_j), & i = j \\ \varnothing, & i \neq j. \end{cases} \tag{4}$$

When the scheduling policy is properly designed, the erasure symbol also conveys valuable information about x_i to its corresponding estimator. In the case of a *broadcast network*, the packet transmitted by the scheduler is received by all the estimators, i.e., if $u = j$, then

$$y_i = (j, x_j), \quad i \in \{1, \ldots, n\}. \tag{5}$$

Upon receiving y_i, the ith estimator uses a function δ_i to compute an estimate of the ith measurement as follows:

$$\hat{x}_i = \delta_i(y_i), \quad i \in \{1, \ldots, n\}. \tag{6}$$

We denote the collection of estimation functions by

$$\delta \overset{\text{def}}{=} (\delta_1, \ldots, \delta_n). \tag{7}$$

Problem 1 (Observation-Driven Sensor Scheduling): Given the joint PDF of the sensor data f_X and the network operation mode (unicast or broadcast), design the scheduling and estimation policies μ and δ such that the following MSE between observations and estimates is minimized:

$$\mathcal{J}(\mu, \delta) = \mathbb{E}\left[\sum_{i=1}^{n} (X_i - \hat{X}_i)^2\right]. \tag{8}$$

III. Unicast Network

In this setting, the wireless network behaves as independent links between sensors and their corresponding receivers. However, due to bandwidth constraints, only one link may be active at a time. The scheduler then selects which of the n links to be active, and the remaining links are idle. However, the observation of a silent symbol still conveys information about the nontransmitted measurements.

Definition 1 (Estimation Policies for Estimation Over Unicast Networks): An estimation policy for the ith estimator in the unicast network case is a function parameterized by $\theta_i \in \mathbb{R}$ such that

$$\delta_i(y_i) = \begin{cases} x_i & \text{if } y_i = (i, x_i) \\ \theta_i & \text{if } y = \varnothing. \end{cases} \tag{9}$$

Therefore, the collection of estimation policies δ for Problem 1 is completely characterized by a vector $\theta \in \mathbb{R}^n$, where

$$\theta \overset{\text{def}}{=} (\theta_1, \ldots, \theta_n). \tag{10}$$

Theorem 1 (Difference-of-Convex Decomposition—Unicast Case): If the estimators in Problem 1 use policies of the form in Definition 1, the objective function in (8) admits the following decomposition as a difference of two convex functions:

$$\mathcal{J}(\mu_\delta^\star, \delta) = \mathbb{E}\left[\sum_{i=1}^{n} (X_i - \theta_i)^2\right] - \mathbb{E}\left[\max_{j \in \{1,\ldots,n\}} \left\{(X_j - \theta_j)^2\right\}\right] \tag{11}$$

where $\mu_\delta^\star$ is the optimal scheduler for a fixed collection of estimation policies δ, which is parameterized by the vector $\theta \in \mathbb{R}^n$.

Proof: Using the estimators in 1 and the law of total expectation, the cost function in (8) can be expressed in integral form as follows[1]:

$$\mathcal{J}(\mu, \delta) = \sum_{j=1}^{n} \int_{\mathbb{R}^n} \left[\sum_{i \neq j} (x_i - \theta_i)^2\right] \mathbb{I}\left(\mu(x) = j\right) f_X(x) dx \tag{12}$$

For a fixed δ, in other words, for a fixed $\theta \in \mathbb{R}^n$, the optimal scheduler $\mu_\delta^\star$ is determined by the following set of inequalities:

$$\mu_\delta^\star(x) = j \Leftrightarrow |x_j - \theta_j| \geq |x_\ell - \theta_\ell|, \quad \ell \in \{1, \ldots, n\}. \tag{13}$$

This scheduler leads to the following objective function as a function of δ:

$$\mathcal{J}(\mu_\delta^\star, \delta) = \mathbb{E}\left[\min_{j \in \{1,\ldots,n\}} \left\{\sum_{i \neq j} (X_i - \theta_i)^2\right\}\right] \overset{\text{def}}{=} \mathcal{J}(\theta). \tag{14}$$

The $\min\{\cdot\}$ function in the argument of the expectation operator may lead to a nonconvex objective function in (14). Also notice that depending on the continuity of the density, the objective function may also be nonsmooth. However, the identity holds

$$\min_j \left\{\sum_{i \neq j} (x_i - \theta_i)^2\right\} = \sum_{i=1}^{n} (x_i - \theta_i)^2 - \max_j \left\{(x_j - \theta_j)^2\right\}. \tag{15}$$

The result follows from the linearity of the expectation operator. ■

The fact that the optimization problem admits a DC decomposition is attractive because it allows efficient implementation of the branch-and-bound method, which is guaranteed to converge to a globally optimal solution [20]. However, the convergence

[1] The indicator function of a statement $\mathfrak{S}$ is defined as

$$\mathbb{I}(\mathfrak{S}) \overset{\text{def}}{=} \begin{cases} 1 & \text{if } \mathfrak{S} \text{ is true} \\ 0 & \text{otherwise.} \end{cases}$$

of such algorithm is typically very slow for large-dimensional optimization problems, which, in our case, would be prohibitive for a systems with a large number of sensors. On the other hand, the DC decomposition allows the use of a technique known as CCP [8], [21], [22], which is guaranteed to converge to a locally optimal solution [23], and often admits simple implementation and fast convergence.

A. Convex–Concave Procedure

The CCP is an optimization technique used to find local minima of nonconvex cost functions that admit a DC decomposition. The advantage of using CCP over a subgradient method is that the CCP makes use of the structure of the objective function, which in certain cases lead to very efficient algorithms.

Theorem 2: Consider the unconstrained nonconvex optimization problem:

$$\min_{\theta \in \mathbb{R}^n} J(\theta) = F(\theta) - G(\theta) \tag{16}$$

where

$$F(\theta) \overset{\text{def}}{=} \mathbb{E}\left[\sum_{i=1}^{n} (X_i - \theta_i)^2\right] \tag{17}$$

and

$$G(\theta) \overset{\text{def}}{=} \mathbb{E}\left[\max_{j\in\{1,\ldots,n\}} \left\{(X_j - \theta_j)^2\right\}\right]. \tag{18}$$

Let g be any subgradient of the function G. The dynamical system described by the recursion

$$\theta^{(k+1)} = 2^{-1} g(\theta^{(k)}) + \mathbb{E}[X] \tag{19}$$

converges to a local minimum of $J(\theta)$.

Proof: We will apply the CCP to the optimization problem in (16)–(18). The CCP consists of approximating the nonconvex part of J, i.e., G, by its affine approximation at a given point $\theta^{(k)} \in \mathbb{R}^n$:

$$G_{\text{affine}}(\theta; \theta^{(k)}) \overset{\text{def}}{=} G(\theta^{(k)}) + g(\theta^{(k)})^{\mathsf{T}}(\theta - \theta^{(k)}) \tag{20}$$

where $g(\theta^{(k)})$ is any subgradient[2] of the function G at the point $\theta^{(k)}$. The next point in the sequence, $\theta^{(k+1)}$, is found by solving the following convex optimization problem:

$$\theta^{(k+1)} = \arg\min_{\theta\in\mathbb{R}^n} \left\{F(\theta) - G_{\text{affine}}(\theta; \theta^{(k)})\right\}. \tag{21}$$

The unconstrained convex optimization problem in 21 can be solved by using the first-order optimality condition:

$$\nabla(F(\theta) - G_{\text{affine}}(\theta))\Big|_{\theta=\theta^\star} = 0 \tag{22}$$

which, in this case, has a unique solution. Computing the gradient above at $\theta^\star$ yields

$$2(\theta^\star - \mathbb{E}[X]) - g(\theta^{(k)}) = 0. \tag{23}$$

Finally, by solving for $\theta^\star$, we obtain the following dynamical system:

$$\theta^{(k+1)} = 2^{-1} g(\theta^{(k)}) + \mathbb{E}[X]. \tag{24}$$

The sequence of the points generated according to the dynamical system above is guaranteed to converge to one of the local minima of J [23]. ■

B. Relationship With Subgradient Methods

The dynamical system in (19) is related to subgradient methods of the form

$$\theta^{(k+1)} = \theta^{(k)} - \alpha_k j(\theta^{(k)}) \tag{25}$$

where $j(\theta^{(k)})$ is a subgradient of J at $\theta^{(k)}$. Notice that convergence results for such algorithms exist under the condition that J is a convex function and the step sequence satisfies certain summability conditions[3] that typically imply a very slow convergence rate to a global minimum. There are no guarantees in general that a subgradient method like the one in (25) will converge to a local minimum if the objective function is nonconvex.

One remarkable observation is that the dynamical system from the CCP in (19) is equivalent to

$$\hat{x}^{(k+1)} = \hat{x}^{(k)} - 2^{-1} j(x^{(k)}) \tag{26}$$

where

$$j(x^{(k)}) \overset{\text{def}}{=} \nabla F(\hat{x}^{(k)}) - g(\hat{x}^{(k)}). \tag{27}$$

The constant step size $\alpha = 0.5$ is desirable because it yields convergence rate of $\mathcal{O}(1/k)$ to a local minimum despite the fact that the objective function is nonconvex. Furthermore, even for convex objective functions, the constant step size only guarantees convergence to a point within a fixed gap of the optimal solution; and with variable step sizes satisfying the typical summability conditions, the convergence rate is $\mathcal{O}(1/\sqrt{k})$ [24].

C. Computing a Subgradient

The dynamical system in (19) relies on the fact that at every time step k, we are able to evaluate a subgradient g of the function G defined in (18). The fact that only a subgradient is required is important because the function $\max$ inside the expectation G is nonsmooth, which may lead to a nonsmooth G depending on the joint PDF f_X. Next, we will use weak subgradient calculus to compute a subgradient g.

For a fixed vector $x \in \mathbb{R}^n$, define

$$G(\theta; x) \overset{\text{def}}{=} \max_{j\in\{1,\ldots,n\}} \left\{(x_j - \theta_j)^2\right\} \tag{28}$$

and

$$G_j(\theta; x) \overset{\text{def}}{=} (x_j - \theta_j)^2, \;\; j \in \{1, \ldots, n\}. \tag{29}$$

[2] A vector $g \in \mathbb{R}^n$ is a subgradient of $f : \mathbb{R}^n \to \mathbb{R}$ at $x \in \mathbf{dom}\, f$ if for all $z \in \mathbf{dom}\, f$,

$$f(z) \geq f(x) + g^{\mathsf{T}}(z - x).$$

[3] For example, if the step size sequence $\{\alpha_k\}$ satisfies

$$\sum_{k=0}^{\infty} \alpha_k^2 < \infty \text{ and } \sum_{k=0}^{\infty} \alpha_k = \infty.$$

Algorithm 1: Computing a Subgradient of $G(\theta; x)$.

```
procedure subgrad($\theta; x$)
    $G^\star \leftarrow -\infty$
    $j^\star \leftarrow 0$
    for $j \in \{1, \ldots, n\}$ do          ▷ linear search
        $G \leftarrow G_j(\theta; x)$
        if $G \geq G^\star$ then
            $G^\star \leftarrow G$
            $j^\star \leftarrow j$
        end if
    end for
    $g \leftarrow \nabla_\theta G_{j^\star}(\theta; x)$
    return $g$                               ▷ subgradient of $G$
end procedure
```

Therefore,

$$G(\theta; x) = \max_{j \in \{1, \ldots, n\}} G_j(\theta; x). \tag{30}$$

The gradient of each $G_j(\theta; x)$ is given by

$$\nabla_\theta G_j(\theta; x) = -2(x_j - \theta_j)\mathbf{e}_j \tag{31}$$

where $\mathbf{e}_j$ is the jth canonical basis vector in $\mathbb{R}^n$.

The computation of a subgradient for $G(\theta; x)$ is done via an algorithmic procedure, which implements a linear search. For a fixed pair of arguments $(\theta; x)$, the subgradient is computed as follows:

$$g(\theta; x) = \texttt{subgrad}(\theta; x) \tag{32}$$

where `subgrad` is given in the procedure in Algorithm 1.

Finally, weak subgradient calculus states that

$$g(\theta) \overset{\text{def}}{=} \mathbb{E}\left[g(\theta; X)\right] \tag{33}$$

belongs to the subdifferential $\partial G(\theta)$, where the expectation is taken with respect to the random vector X. Thus, (33) is a subgradient of G at θ [25].

Remark 1: The computational procedure derived from the CCP is simple, but still requires the computation of an n-dimensional integral due to the expectation operator in (33). Two things may occur: 1) We know the PDF of the measurement vector X, and the dimension n is small enough to allow for efficient numerical computation of the expectation; 2) we do not have access to the PDF or the dimension n is prohibitively large, but we have access to a (sufficiently large) dataset of i.i.d. samples from f_X. The latter scenario will be explored in Section V.

D. Illustrative Example

In this example, we consider the exact computation of (33) for a system with $n = 2$ with sensors. Each sensor observes a component of a bivariate source $X = (X_1, X_2)$. Let X be distributed according to the following mixture of bivariate Gaussians:

$$X \sim \frac{3}{4}\mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right) + \frac{1}{4}\mathcal{N}\left(\begin{bmatrix}4\\2\end{bmatrix}, \begin{bmatrix}1 & 0.4\\0.4 & 1\end{bmatrix}\right). \tag{34}$$

Assuming that we did not know the number of local minima, we used the algorithm in (19) with 1000 random initial conditions $\theta^{(0)} \in \mathbb{R}^2$, and retain the resulting $\theta^\star$ with the best value. In our case, we obtained

$$\theta^\star = (+0.0045, +1.5900) \tag{35}$$

with an associated value of $J(\theta^\star) = 0.8065$. Therefore, the optimal scheduler is given by

$$\mu^\star(x) = \begin{cases} 1 & \text{if } |x_1 - 0.0045| \geq |x_2 - 1.5900| \\ 2 & \text{otherwise.} \end{cases} \tag{36}$$

To compare the performance of the observation-driven scheduler, consider a "blind" scheduler, μ^{blind}, which does not make use of the observations. Herein, μ^{blind} gives channel access to the sensor with the largest variance. The corresponding blind-estimators δ^{blind} output the expected value of the unobserved random variable, i.e.,

$$\mu^{\text{blind}}(x) = \arg\max_{i \in \{1,2\}} \text{Var}(X_i) \tag{37}$$

and

$$\delta_i^{\text{blind}}(y_i) = \begin{cases} x_i & \text{if } y_i = (i, x_i) \\ \mathbb{E}[X_i] & \text{if } y_i = \varnothing. \end{cases} \tag{38}$$

In this example, the performance of the blind scheduler is

$$J(\mu^{\text{blind}}, \delta^{\text{blind}}) = \min\{4, 1.75\} = 1.75. \tag{39}$$

Notice that the performance of the observation-driven scheduler in this case is approximately 54% better than the blind-scheduler.

Remark 2: Due to the lack of convexity of the optimization problem, we cannot guarantee that the solution obtained via the CCP is globally optimal. In our numerical results, we generate many candidate solutions from uniformly distributed random initial conditions and pick the one with the best value of the objective function. Since the CCP converges quickly due to its constant step size, we can do this efficiently.

IV. Broadcast Network

When the wireless network is of the broadcast type, all the estimators receive the same signal. This signal is then used as side information to estimate the nonreceived observations. Given that $U = j$, the received signals at the estimators are

$$y_i = (j, x_j), \;\; i \in \{1, \ldots, n\}. \tag{40}$$

In this case, X_j serves as side information for the estimates $\hat{X}_i$, $i \neq j$. This must be the case even if the sensors make mutually independent observations.

Proposition 1: Consider Problem 1 over a broadcast network. Let $i, j \in \{1, \ldots, n\}$ such that $i \neq j$. For a fixed scheduling policy μ, the optimal estimator $\delta^\star_{\mu,i}$ is of the following form:

$$\delta^\star_{\mu,i}(x_i) = \begin{cases} x_i & y_i = (i, x_i) \\ \eta_{ij}(x_j) & y_j = (j, x_j) \end{cases} \tag{41}$$

where η_{ij} are functions that depend implicitly on μ.

Proof: For a fixed scheduling policy μ, the MSE objective function implies that the optimal estimator is the conditional mean of the measurement given the channel output, i.e., for $U = j$,

$$\delta^\star_{i,\mu}(j, x_j) = \mathbb{E}\left[X_i \mid \mu(X) = j, X_j = x_j\right]. \tag{42}$$

If $i = j$, then

$$\mathbb{E}\left[X_i \mid \mu(X) = i, X_i = x_i\right] = x_i. \tag{43}$$

If $i \neq j$, then

$$\mathbb{E}\left[X_i \mid \mu(X) = j, X_j = x_j\right] \overset{\text{def}}{=} \eta_{ij}(x_j). \tag{44}$$

■

Remark 3: Without making any assumptions on the probabilistic model or the scheduler, there is nothing we can say about the structure of the optimal representation functions η_{ij}. In fact, even if the observations are jointly Gaussian, the optimal representation functions may be nonlinear [18]. To obtain a tractable finite-dimensional optimization problem over a broadcast network, we will constrain the estimators to the class of piece-wise affine functions.

Definition 2 (Policies for Estimation Over Broadcast Networks): An estimation policy for the ith estimator in the broadcast network case is a function parameterized by weights $w_{ij} \in \mathbb{R}$ and biases $b_{ij} \in \mathbb{R}$, such that

$$\delta_i(y_i) = \begin{cases} x_i & \text{if } y_i = (i, x_i) \\ w_{ij}x_j + b_{ij} & \text{if } y_i = (j, x_j) \text{ and } j \neq i. \end{cases} \tag{45}$$

We are trading off optimality for tractability by constraining the class of estimators to be piece-wise affine, and performing the optimization within that class. The total number of optimization variables in this version of Problem 1 is equal to the number of parameters used to describe all the estimators. In this case, this number is $d = 2n(n-1)$. Therefore, the number of variables scales quadratically with the number of sensors, as opposed to the the linear number of variables in the unicast case. Nevertheless, the number of variables in our algorithm scales polynomially in the number of sensors, and it is still manageable for applications with a large number of sensors. Therefore, the collection of estimation policies δ for Problem 1 is characterized by $\theta \in \mathbb{R}^d$

$$\theta \overset{\text{def}}{=} \operatorname{vec}(\theta_1, \ldots, \theta_n), \text{ where } \theta_j \overset{\text{def}}{=} \operatorname{vec}\left(\left\{\begin{bmatrix} w_{ij} \\ b_{ij} \end{bmatrix}, i \neq j\right\}\right). \tag{46}$$

Theorem 3 (Difference-of-Convex Decomposition—Broadcast Case): If the estimators in Problem 1 use policies of the form in Definition 2, the objective function in (8) admits the following decomposition as a difference of two convex functions:

$$J(\mu^\star_\delta, \delta) = \mathbb{E}\left[\sum_{\ell=1}^{n}\sum_{i\neq\ell}(X_i - (w_{i\ell}X_\ell + b_{i\ell}))^2 - \max_{j\in\{1,\ldots,n\}}\left\{\sum_{\ell\neq j}\sum_{i\neq\ell}\left(X_i - (w_{i\ell}X_\ell + b_{i\ell})\right)^2\right]. \tag{47}$$

Proof: For a fixed collection of estimation policies of the form given in Definition 2, i.e., for a fixed vector $\theta \in \mathbb{R}^d$, and using the law of total expectation, the cost function in (8) can be expressed in integral form as follows:

$$J(\mu, \delta) = \sum_{j=1}^{n}\left[\int_{\mathbb{R}^n}\left(\sum_{i\neq j}\left(x_i - (w_{ij}x_j - b_{ij})\right)^2\right) \times \mathbb{I}(\mu(x) = j) f_X(x)dx\right]. \tag{48}$$

The optimal scheduling policy $\mu^\star_\delta(x) = j$ if and only if the following set of inequalities are satisfied:

$$\sum_{i\neq j}\left(x_i - (w_{ij}x_j + b_{ij})\right)^2 < \sum_{i\neq\ell}\left(x_i - (w_{i\ell}x_j + b_{i\ell})\right)^2, \ \ell \neq j. \tag{49}$$

Using this scheduler, we may rewrite the optimization problem as a function of the parameters of the estimators, θ. Thus,

$$J(\mu^\star_\delta, \delta) = \mathbb{E}\left[\min_{j\in\{1,\ldots,n\}}\left\{\sum_{i\neq j}\left(X_i - (w_{ij}X_j + b_{ij})\right)^2\right\}\right] \overset{\text{def}}{=} J(\theta). \tag{50}$$

The following identity holds:

$$\min_j \sum_{i\neq j}\left(x_i - (w_{ij}x_j + b_{ij})\right)^2 = \sum_{\ell=1}^{n}\sum_{i\neq\ell}\left(x_i - (w_{i\ell}x_\ell + b_{i\ell})\right)^2 - \max_j \sum_{\ell\neq j}\sum_{i\neq\ell}\left(x_i - (w_{i\ell}x_\ell + b_{i\ell})\right)^2. \tag{51}$$

■

Remark 4: Notice that the DC decomposition in the broadcast case is not as neat as in the unicast case. The reason is that for each received (j, x_j), the ith estimator uses a different pair of parameters w_{ij}, b_{ij}. However, as we will show next, the decomposition in Theorem 3 is just as useful as the one in Theorem 1. Furthermore, the optimization problem obtained for the unicast case is a particular instance of the one obtained for the broadcast case (if we assume that the weights $w_{ij} = 0$, for all i and j.).

A. Convex–Concave Procedure

For the remainder of this section, we will assume that $n = 2$. The equations for $n > 2$ are presented in Appendix A.

The parameter vector θ which specifies the affine estimators δ_1 and δ_2 is

$$\theta = (w_{21}, b_{21}, w_{12}, b_{12}). \tag{52}$$

Theorem 4: Consider the unconstrained nonconvex optimization problem:

$$\min_{\theta \in \mathbb{R}^4} J(\theta) = F(\theta) - G(\theta) \tag{53}$$

where

$$F(\theta) \overset{\text{def}}{=} \mathbb{E}\left[(X_1 - (w_{12}X_2 + b_{12}))^2 + (X_2 - (w_{21}X_1 + b_{21}))^2\right] \tag{54}$$

and

$$G(\theta) \overset{\text{def}}{=} \mathbb{E}\Big[\max\Big\{(X_1 - (w_{12}X_2 + b_{12}))^2, \\ (X_2 - (w_{21}X_1 + b_{21}))^2\Big\}\Big]. \tag{55}$$

Let g be any subgradient of the function G. One such subgradient is given in eq. (56) shown at the bottom of this page. Let $\mathbf{A}$ and $\mathbf{b}$ be defined as

$$\mathbf{A} \overset{\text{def}}{=} 2\begin{bmatrix} \mathbb{E}[X_1^2] & \mathbb{E}[X_1] & 0 & 0 \\ \mathbb{E}[X_1] & 1 & 0 & 0 \\ 0 & 0 & \mathbb{E}[X_2^2] & \mathbb{E}[X_2] \\ 0 & 0 & \mathbb{E}[X_2] & 1 \end{bmatrix} \tag{57}$$

$$\mathbf{b} \overset{\text{def}}{=} 2\begin{bmatrix} \mathbb{E}[X_1X_2] \\ \mathbb{E}[X_2] \\ \mathbb{E}[X_1X_2] \\ \mathbb{E}[X_1] \end{bmatrix}. \tag{58}$$

The dynamical system described by the recursion

$$\theta^{(k+1)} = \mathbf{A}^{-1}\left(g(\theta^{(k)}) + \mathbf{b}\right) \tag{59}$$

converges to a local minimum of $J(\theta)$.

Remark 5: Under the assumption that the observations at the sensors X_1 and X_2 are (non-deterministic) random variables with finite first and second moments, matrix $\mathbf{A}$ is always invertible.

Proof: Using the CCP to the minimization problem in (53)–(55), we have

$$\theta^{(k+1)} = \arg\min_{\theta \in \mathbb{R}^4}\left\{F(\theta) - G_{\text{affine}}(\theta; \theta^{(k)})\right\} \tag{60}$$

where G_{affine} is defined in (20). The unconstrained convex optimization problem in (60) can be solved by using the first-order optimality condition, which in this case has a unique solution. Computing the gradient at $\theta^\star$ yields

$$\mathbf{A}\theta^\star - \mathbf{b} - g(\theta^{(k)}) = 0. \tag{61}$$

Solving for $\theta^\star$ yields the dynamical system in (59). The convergence to a local minimum is guaranteed by the CCP. $\blacksquare$

Remark 6: The computational bottleneck in our algorithm comes from the fact it requires the computation of two-dimensional integrals with arguments that involve indicator functions. These are numerically hard to deal with and may lead to slow convergence rates. Often, the integral may not converge at all, leading to poor performance. The situation is further complicated when the number of sensor-estimator pairs is large. However, the most crucial observation is that the algorithm's overall structure does not depend on the distribution of the data.

B. Relationship With Subgradient Methods

The algorithm of (56) can also be put in a form that resembles a subgradient method as follows:

$$\theta^{(k+1)} = \theta^{(k)} - \mathbf{A}^{-1}\jmath(\theta^{(k)}). \tag{62}$$

As opposed to the algorithm obtained for unicast networks, there is not a scalar step size. The subgradient $\jmath(\theta^{(k)})$ is instead multiplied by the matrix $\mathbf{A}^{-1}$. Therefore, the "step size" corresponds to the spectral radius of $\mathbf{A}^{-1}$, which is still a constant. However, the inspection of $\mathbf{A}$ suggests that the rate at which the algorithm converges to a local minimum depends on the variances of $X_1, \ldots, X_n$. The larger the variances, the slower the convergence rate.

Corollary 1: The step size α of the algorithm in (56) is the spectral radius of the inverse of $\mathbf{A}$ defined in (57): $\alpha \overset{\text{def}}{=} \rho(\mathbf{A}^{-1})$.

C. Illustrative Example

Consider the observation-driven scheduling in a system with $n = 2$ sensors over a broadcast network. Each sensor observes a component of a bivariate source $X = (X_1, X_2)$. Let X be distributed according to the same mixture of bivariate Gaussians of (34). Running the recursion in (59) for 1000 random initial conditions, $\theta^{(0)}$, and retaining the solution with the best value, we obtain

$$\theta^\star = (0.4238,\ 0.2151,\ -0.2390,\ 0.0624) \tag{63}$$

with $J(\theta^\star) = 0.5276$. Therefore, the optimal scheduler is given by

$$\mu^\star(x) = \begin{cases} 1 & \text{if } |x_1 + 0.2390x_2 - 0.0624| \\ & \qquad \geq |x_2 - 0.4238x_1 - 0.2151| \\ 2 & \text{otherwise.} \end{cases} \tag{64}$$

Comparing the performance of the optimal scheme obtained for a unicast network with the one obtained here for the broadcast network, we observe an improvement of 34.58%. This is possible due to the additional side information provided by the broadcast

$$g(\theta) = -2\mathbb{E}\begin{bmatrix} X_1(X_2 - w_{21}X_1 - b_{21})\mathbf{1}(|X_1 - w_{12}X_2 - b_{12}| < |X_2 - w_{21}X_1 - b_{21}|) \\ (X_2 - w_{21}X_1 - b_{21})\mathbf{1}(|X_1 - w_{12}X_2 - b_{12}| < |X_2 - w_{21}X_1 - b_{21}|) \\ X_2(X_1 - w_{12}X_2 - b_{12})\mathbf{1}(|X_1 - w_{12}X_2 - b_{12}| \geq |X_2 - w_{21}X_1 - b_{21}|) \\ (X_1 - w_{12}X_2 - b_{12})\mathbf{1}(|X_1 - w_{12}X_2 - b_{12}| \geq |X_2 - w_{21}X_1 - b_{21}|) \end{bmatrix} \tag{56}$$

channel to all the estimators at every transmission. However, this comes at the price of a more complex optimization problem involving a larger number of optimization variables.

V. Data-Driven Sensor Scheduling

The main challenge in using the techniques developed in Sections III and IV is that, in practice, we often do not have access to the PDF f_X. Even when the PDF is available, the exact numerical computation of n-dimensional integrals constrains the techniques to small values of n. The examples in the previous section for $n = 2$ were chosen to provide insight on the techniques, and ease the visualization of the landscape of the loss function in the unicast case, as well as allowing for exact computation of the required expectations. In this section, we do not assume any knowledge on f_X. We assume that a dataset $\mathcal{D}$ is available with N i.i.d. samples of the PDF f_X. Instead of solving Problem 1, we solve a version of the problem where the expectations are replaced by their empirical means. We shall refer to this approximation as the empirical risk minimization (ERM) problem [26].

A. Convergence Results

Consider the expected cost

$$J(\theta) \overset{\text{def}}{=} \mathbb{E}\left[J(\theta, X)\right] \tag{65}$$

and its associated empirical mean approximation

$$J_{\mathcal{D}}(\theta) \overset{\text{def}}{=} \frac{1}{N}\sum_{k=1}^{N} J(\theta, x(k)) \tag{66}$$

where $\mathcal{D} = \{x(k)\}_{k=1}^{N}$ are i.i.d. samples from f_X. From here on, the functions $J(\theta, x)$ are called sample functions. When the size N of the dataset $\mathcal{D}$ is large, we would like the optimal value of the approximated function J_N to converge to that of the true objective function J. The critical condition for this convergence is a uniform version of the strong law of large numbers (ULLN), which we state below:

$$\sup_{\theta\in\Theta} |J_{\mathcal{D}}(\theta) - J(\theta)| \xrightarrow{\text{a.s.}} 0, \quad N \to \infty \tag{67}$$

where $\Theta = \mathbf{dom}\ J$.

To determine if the empirical mean approximation is appropriate, we need to prove that the objective functions in Section III and IV satisfy the ULLN.

Definition 3: The function $J(\theta, x)$, $\theta \in \Theta$, is dominated by an integrable function if there exists a non-negative valued measurable function $T(x)$ such that $\mathbb{E}[T(X)] < +\infty$ and for every $\theta \in \Theta$ the inequality $|J(\theta, x)| \le T(x)$ holds with probability one.

Proposition 2 (Proposition 7, p. 363 in [10]): Let Θ be a nonempty compact subset of $\mathbb{R}^d$ and suppose that

1) $J(\theta, x)$ is continuous on Θ for almost every $x \in \mathbb{R}^n$;
2) $J(\theta, x)$, $\theta \in \Theta$, is dominated by an integrable function;
3) $\mathcal{D} = \{x(k)\}_{k=1}^{N}$ is i.i.d. according to the PDF f_X.

Then, the expected cost function $J(\theta)$ is finite valued and continuous on Θ. Moreover,

$$\mathbb{P}\left(\sup_{\theta\in\Theta} |J_{\mathcal{D}}(\theta) - J(\theta)| \to 0\right) = 1. \tag{68}$$

Remark 7: Although both optimization problems are unconstrained, they can always be constrained to a compact Θ. For example, we may let $\Theta = \{\theta \in \mathbb{R}^d \mid \|\theta\|_2 \le C\}$, with a very large $C < +\infty$.

Theorem 5: Let the objective function $J(\theta)$ be defined as in (14). Let the sample function be defined as follows:

$$J(\theta, x) \overset{\text{def}}{=} \|x - \theta\|_2^2 - \|x - \theta\|_\infty^2. \tag{69}$$

If the moments of first and second order of the random vector $X \sim f_X$ are finite, then the ULLN in (67) is satisfied.

Proof: The function (14) can be expressed as $J(\theta) = \mathbb{E}[J(\theta, X)]$, where $J(\theta, x)$ is given in (69). The sample function $J(\theta, x)$ is the difference of the squares of the 2-norm and the ∞-norm, each of which is continuous. Therefore, the sample function $J(\theta, x)$ is continuous in θ. Furthermore,

$$J(\theta, x) \le \|x - \theta\|_2^2. \tag{70}$$

Let $\Theta \overset{\text{def}}{=} \{\theta \in \mathbb{R}^n \mid \|\theta\|_2 \le C\}$, where $C < +\infty$. From the triangle inequality applied to the right-hand side of (70), we have

$$J(\theta, x) \le (\|x\|_2 + C)^2 \overset{\text{def}}{=} T(x). \tag{71}$$

Under the assumption that the moments of first and second order of the random vector $X \sim f_X$ are finite, we have $\mathbb{E}[T(X)] < +\infty$. Therefore, under the i.i.d. assumption on the dataset $\mathcal{D}$, Proposition 2 implies that ULLN holds. ■

Theorem 6: Let the objective function $J(\theta)$ be defined as in (50). Let the sample function be defined as

$$J(\theta, x) \overset{\text{def}}{=} \sum_{\ell=1}^{n}\sum_{i\neq\ell}(x_i - (w_{ij}x_\ell + b_{i\ell}))^2 - \max_{j\in\{1,\ldots,n\}} \sum_{\ell\neq j}\sum_{i\neq\ell}(x_i - (w_{ij}x_\ell + b_{i\ell}))^2, \tag{72}$$

where $\{w_{i\ell}, b_{i\ell}\}$ are components of the parameter vector $\theta \in \mathbb{R}^d$ defined in (46). If the moments of first and second order of the random vector $X \sim f_X$ are finite, then the ULLN in (67) is satisfied.

Proof: The continuity of the sample function (72) can be established by expressing $J(\theta, x)$ as the difference of squares of a Frobenius norm and the ∞-norm of a particular linear map, and their respective continuities. We omit this step for brevity. Define $w_{\ell\ell} = 1$ and $b_{\ell\ell} = 0$, $\ell = \{1, \ldots, n\}$. Then,

$$J(\theta, x) \le \|x \cdot \mathbf{1}^{\mathsf{T}} - \left(\mathbf{W} \circ (\mathbf{1} \cdot x^{\mathsf{T}}) + \mathbf{B}\right)\|_F^2 \tag{73}$$

where the operation $\circ$ denotes the Schur product between two matrices, and

$$\mathbf{W} \overset{\text{def}}{=} \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{n1} & \cdots & w_{nn} \end{bmatrix}, \mathbf{B} \overset{\text{def}}{=} \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix}. \tag{74}$$

Bounding the right-hand side of (73) using the triangle inequality twice, we have

$$J(\theta, x) \leq \left(\|x \cdot \mathbf{1}^\mathsf{T}\| + \|\mathbf{W} \circ (\mathbf{1} \cdot x^\mathsf{T})\|_F + \|\mathbf{B}\|_F \right)^2. \tag{75}$$

The following inequality for the Schur product of two matrices holds [27, Fact: 9.14.33, p. 675]:

$$\|\mathbf{W} \circ (\mathbf{1} \cdot x^\mathsf{T})\|_F \leq \|\mathbf{W}\|_F \cdot \|\mathbf{1} \cdot x^\mathsf{T}\|_F. \tag{76}$$

Suppose that $\Theta \overset{\text{def}}{=} \{\theta \in \mathbb{R}^d \mid \|\mathbf{W}\|_F \leq C_1, \|\mathbf{B}\|_F \leq C_2\}$, with $C_1, C_2 < +\infty$. Then,

$$J(\theta, x) \leq (n(1 + C_1)\|X\|_2 + C_2)^2 \overset{\text{def}}{=} T(X). \tag{77}$$

From the assumption on the moments of first and second order of the random vector $X \sim f_X$, we have $\mathbb{E}[T(X)] < +\infty$. Therefore, under the i.i.d. assumption on the dataset $\mathcal{D}$, Proposition 2 implies that ULLN holds. ■

Remark 8 (Sample complexity): Theorems 5 and 6 are important because they allow us to estimate the optimal solution to Problem 1 using the solutions to the approximate problem when the number of samples in the dataset N is large enough. An ERM problem solved to δ-optimality gives the ϵ-optimal solution to the corresponding true problem with probability at least $1 - \alpha$ if the sample size N satisfies the following inequality [10]:

$$N \geq \frac{12\sigma^2}{(\epsilon - \delta)^2} \left(d \log \left(\frac{2DL}{\epsilon - \delta} \right) - \log \alpha \right) \tag{78}$$

where D is the diameter of set Θ, $\delta < \epsilon$, the objective function $J(\theta)$ is assumed to be L-Lipschitz continuous on Θ, d is the dimension of the parameter vector θ, and σ^2 is the maximal variance of certain differences between values of the approximate objective function $J_\mathcal{D}(\theta)$. This sample complexity bound is overly conservative because it holds under very general assumptions on the cost function, and does not yield practical values of N. Under very modest values of ϵ and δ (e.g. $\epsilon \approx 10^{-3}$ and $\delta \approx 10^{-4}$), the right-hand side of (78) could easily reach the hundreds of millions samples. Moreover, estimating the Lipschitz constant of J and the variance σ^2 is a challenging problem on their own right. However, we empirically observed in Section V-D that very good approximate solutions can be found using relatively small training datasets.

B. Approximate CCP

Consider a dataset $\mathcal{D}$ where

$$\mathcal{D} = \{x_1(k), \ldots, x_n(k)\}_{k=1}^N \tag{79}$$

with N i.i.d. samples from an unknown PDF f_X. Define the ERM problem:

$$\underset{\theta \in \Theta}{\text{minimize}} \;\; J_\mathcal{D}(\theta) \overset{\text{def}}{=} \frac{1}{N} \sum_{k=1}^N J(\theta, x(k)) \tag{80}$$

where

$$J(\theta, x) = \begin{cases} \|x - \theta\|_2^2 - \|x - \theta\|_\infty^2 & \text{(unicast)} \\ \sum\limits_{\ell=1}^n \sum\limits_{i \neq \ell} (x_i - (w_{ij} x_\ell + b_{i\ell}))^2 - \\ \quad \max\limits_{j \in \{1,\ldots,n\}} \sum\limits_{\ell \neq j} \sum\limits_{i \neq \ell} (x_i - (w_{ij} x_\ell + b_{i\ell}))^2 & \text{(broadcast).} \end{cases} \tag{81}$$

When applied to the ERM problem, the CCP operates exactly the same as before, but with the advantage that computing a subgradient involves evaluating empirical means instead of computing n-dimensional integrals. The approximate CCP recursions become

$$\theta^{(k+1)} = \begin{cases} \frac{1}{2} g_\mathcal{D}(\theta^{(k)}) + \frac{1}{N} \sum\limits_{k=1}^N x(k) & \text{(unicast)} \\ \mathbf{A}_\mathcal{D}^{-1} \left(g_\mathcal{D}(\theta^{(k)}) + \mathbf{b}_\mathcal{D} \right) & \text{(broadcast).} \end{cases} \tag{82}$$

In the 2-D broadcast case, the matrix $\mathbf{A}_\mathcal{D}$ and vector $\mathbf{b}_\mathcal{D}$ are given by

$$\mathbf{A}_\mathcal{D} = \frac{2}{N} \sum_{k=1}^N \begin{bmatrix} x_1^2(k) & x_1(k) & 0 & 0 \\ x_1(k) & 1 & 0 & 0 \\ 0 & 0 & x_2^2(k) & x_2^2(k) \\ 0 & 0 & x_2(k) & 1 \end{bmatrix} \tag{83}$$

$$\mathbf{b}_\mathcal{D} = \frac{2}{N} \sum_{k=1}^N \begin{bmatrix} x_1(k) x_2(k) \\ x_2(k) \\ x_1(k) x_2(k) \\ x_1(k) \end{bmatrix}. \tag{84}$$

The expressions for the n-dimensional case are given in Appendix A. Finally, $g_\mathcal{D}$ is a subgradient of appropriate $G_\mathcal{D}$ (unicast or broadcast) computed as follows:

$$g_\mathcal{D}(\theta) = \frac{1}{N} \sum_{k=1}^N \texttt{subgrad}(\theta; x(k)). \tag{85}$$

The algorithm above converges to a local minimum $\bar{\theta}_\mathcal{D}$ of the ERM objective function $J_\mathcal{D}$, and not of the original cost J. However, due to the ULLN proved in Theorems 5 and 6, when N is sufficiently large, $J_\mathcal{D}$ is approximately equal to J, and the point $\bar{\theta}_\mathcal{D}$ will be a good estimate of a locally optimal solution to the original problem.

C. Learning Framework

The approximate CCP algorithm described in the previous section is a heuristic, i.e., since the sample functions are nonconvex in θ, we cannot guarantee that a given candidate solution $\bar{\theta}_\mathcal{D}$ for Problem 1 is a global minimizer. However, it is possible to produce a confidence interval on the optimality gap with respect to any candidate solution by solving instances of the ERM problem using global optimization solvers. For sample functions that admit a DC decomposition, the branch-and-bound method [28] can be used to solve the ERM problem to a prescribed accuracy.

Suppose that we have access to a training dataset $\mathcal{D}$ and M validation datasets $\mathcal{T}^m$, $m = \{1, \ldots, M\}$, each with N i.i.d. samples from f_X. From the training dataset $\mathcal{D}$, we compute a candidate solution $\bar{\theta}_{\mathcal{D}}$ using the CCP on the ERM problem in (80). Let the *optimality gap* be defined as

$$\mathrm{gap}(\bar{\theta}_{\mathcal{D}}) \overset{\mathrm{def}}{=} \mathbb{E}[J(\bar{\theta}_{\mathcal{D}}, X)] - J^\star \tag{86}$$

where $J^\star$ is the unknown global minimum of Problem 1. Mak *et al.* [29] have shown that

$$\mathrm{gap}(\bar{\theta}) \leq \mathbb{E}\left[\frac{1}{N}\sum_{k=1}^{N} J\left(\bar{\theta}_{\mathcal{D}}, X(k)\right) - \min_{\theta} \frac{1}{N}\sum_{k=1}^{N} J\left(\theta, X(k)\right)\right] \tag{87}$$

where $X(k)$ are i.i.d. random variables with density f_X.

Define the random variable U_N as a function of $\{X(k)\}_{k=1}^N$ as follows:

$$U_N \overset{\mathrm{def}}{=} \frac{1}{N}\sum_{k=1}^{N} J\left(\bar{\theta}_{\mathcal{D}}, X(k)\right) - \min_{\theta} \frac{1}{N}\sum_{k=1}^{N} J\left(\theta, X(k)\right). \tag{88}$$

From M i.i.d. batches of data $\mathcal{T}^m = \{x^m(k)\}_{k=1}^N$, where $m = \{1, \ldots, M\}$, we form an estimate of the upper bound to optimality gap as follows:

$$\hat{u}_N^M = \frac{1}{M}\sum_{m=1}^{M} u_N^m \tag{89}$$

where

$$u_N^m \overset{\mathrm{def}}{=} \frac{1}{N}\sum_{k=1}^{N} J\left(\bar{\theta}_{\mathcal{D}}, x^m(k)\right) - \min_{\theta} \frac{1}{N}\sum_{k=1}^{N} J\left(\theta, x^m(k)\right). \tag{90}$$

From the Central Limit Theorem, we have

$$\sqrt{M}\left(\hat{u}_N^M - \mathbb{E}[U_N]\right) \xrightarrow{\mathrm{d}} \mathcal{N}(0, \sigma_U^2), \quad \text{as} \quad M \to \infty \tag{91}$$

where $\sigma_U^2 = \mathrm{Var}(U_N)$. Based on the asymptotic normality of the estimator $\hat{u}_N^M$, we have the following high probability upper bound on the optimality gap:

$$\mathbb{P}\left(\mathrm{gap}(\bar{\theta}_{\mathcal{D}}) \leq \hat{u}_N^M + \frac{t_{M-1,\alpha} \cdot \hat{\sigma}_U}{\sqrt{M}}\right) \geq 1 - \alpha \tag{92}$$

where $t_{M-1,\alpha}$ is the α-critical value of the t-distribution with $M - 1$ degrees of freedom, and $\hat{\sigma}_U$ is the sample variance estimator computed based on $\{u_N^m\}_{m=1}^M$.

To validate the solution $\bar{\theta}_{\mathcal{D}}$, we first choose a confidence level α, then we must solve M global optimization problems

$$J^\star_{\mathcal{T}^m} = \min_{\theta} \frac{1}{N}\sum_{k=1}^{N} J\left(\theta, x^m(k)\right), \quad m = \{1, \ldots, M\}. \tag{93}$$

Computing the value of each of the empirical mean approximate objective functions at $\bar{\theta}_{\mathcal{D}}$, we have

$$J_{\mathcal{T}^m}(\bar{\theta}_{\mathcal{D}}) = \frac{1}{N}\sum_{k=1}^{N} J\left(\bar{\theta}_{\mathcal{D}}, x^m(k)\right), \quad m = \{1, \ldots, M\}. \tag{94}$$

TABLE I
TRAINING AND VALIDATION RESULTS FOR THE EMPIRICAL RISK MINIMIZATION FROM N SAMPLES FOR UNICAST NETWORKS

N	$\bar{\theta}_1$	$\bar{\theta}_2$	$J_{\mathcal{D}}(\bar{\theta})$	$\mathrm{gap}(\bar{\theta})$	$T_{\mathrm{t}}(\mathrm{s})$	$T_{\mathrm{v}}(\mathrm{s})$
10^2	-0.1343	$+1.6010$	0.7915	4.4×10^{-2}	0.08	0.08
10^3	$+0.1507$	$+1.6322$	0.7824	1.3×10^{-2}	0.33	0.51
10^4	$+0.0076$	$+1.5964$	0.8059	3.6×10^{-4}	2.80	6.45
10^5	$+0.0068$	$+1.5623$	0.8057	2.1×10^{-4}	26.4	58.3

TABLE II
TRAINING AND VALIDATION RESULTS FOR THE EMPIRICAL RISK MINIMIZATION FROM N SAMPLES FOR BROADCAST NETWORKS

N	$\bar{\theta}_1$	$\bar{\theta}_2$	$J_{\mathcal{D}}(\theta)$	$\mathrm{gap}(\bar{\theta})$	$T_{\mathrm{t}}(\mathrm{s})$	$T_{\mathrm{v}}(\mathrm{s})$
10^2	$\begin{bmatrix}+0.4614\\-0.0238\end{bmatrix}$	$\begin{bmatrix}-0.5577\\+0.2260\end{bmatrix}$	0.4354	9.9×10^{-2}	0.11	0.27
10^3	$\begin{bmatrix}+0.4703\\+0.0906\end{bmatrix}$	$\begin{bmatrix}-0.3133\\-0.0302\end{bmatrix}$	0.5603	1.1×10^{-2}	0.63	1.65
10^4	$\begin{bmatrix}+0.4263\\+0.2044\end{bmatrix}$	$\begin{bmatrix}-0.2272\\+0.0778\end{bmatrix}$	0.5387	6.5×10^{-4}	3.93	13.9
10^5	$\begin{bmatrix}+0.4142\\+0.2527\end{bmatrix}$	$\begin{bmatrix}-0.1960\\+0.1036\end{bmatrix}$	0.5304	6.9×10^{-4}	30.2	168.0

Finally, we compute the upper bound on the optimality gap

$$\hat{u}_N^M + \frac{t_{M-1,\alpha} \cdot \hat{\sigma}_U}{\sqrt{M}}. \tag{95}$$

If this upper bound is within a target margin of error tolerance, we declare $\bar{\theta}_{\mathcal{D}}$ as the learned the optimal parameter for Problem 1, and use them to recover the jointly optimal scheduler and estimators. Otherwise, we must increase N or decrease α, and repeat the process. However, the necessity of solving M global optimization problems for nonconvex, nonsmooth ERM problems, limits our ability to increase N due to computational complexity issues.

D. Illustrative Examples

1) Synthetic Data: Consider a dataset $\mathcal{D}$ consisting of N i.i.d. samples from the bivariate Gaussian mixture model of (34). We use the approximate CCP to compute candidate solutions to Problem 1 in the unicast and the broadcast cases. Then, we use the validation framework described in the previous section to compute the probabilistic bound on the optimality gap between the candidate solution found using CCP, and the unknown exact optimal solution to Problem 1.[4] We have chosen $\alpha = 0.05$. Our numerical results are shown in Tables I and II. For each row of the tables, we have used $M = 100$ validation datasets. We recorded the time in seconds to train via the CCP in T_{t}, and the average time to validate (globally solving the ERM based on the validation sets $\mathcal{T}^m$, $m = \{1, \ldots, M\}$), via the generalized pattern search method [30] in T_{v}.

2) Real Data: In this example, we use our algorithms and learning framework to solve Problem 1 using a real dataset containing measurements of temperature and humidity sensors collected by a wireless sensor network of a smart-home. This dataset is publicly available,[5] and more details and analysis can

[4] The code used to obtain the results in this section is available at https://github.com/mullervasconcelos/DDSS.

[5] The complete dataset can be dowloaded from https://github.com/LuisM78/Appliances-energy-prediction-data.

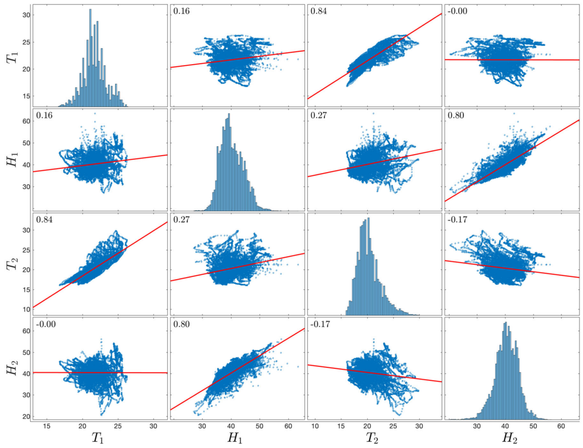

Fig. 2. Empirical distribution and correlation analysis of the temperature and relative humidity in rooms 1 and 2 of the dataset in [31].

be found in [31]. The relevant portion of the dataset to our article consists of $n = 16$ variables corresponding to the temperature (in Celsius) and relative humidity in each of the home's eight rooms. The empirical distribution and the correlation structure of the observations, and the fact that the dimension of the dataset is of moderate size, provide a more realistic application scenario than the previous example. Fig. 2 shows the empirical distributions and correlations for the temperature and relative humidity in rooms 1 and 2. In this dataset, there is a total of 19 735 samples, which we randomly permuted and split into 21 batches of $N = 936$ samples. One batch is used for training, and $M = 20$ batches are used for the validation analysis. It is noted that the data was collected as a time-series and the samples are not i.i.d.; however, since our problem is akin to a one-shot regression, the temporal dependence does not play a significant role. The random permutation used before the batch splitting allows each batch to have approximately the same empirical distribution.

For the training phase, we used the recursion in (82) of the CPP for the unicast and broadcast networks. In both cases, we have initialized the CCP at $\theta^{(0)} = (0, \ldots, 0)$. The convergence criterion used is the function tolerance $\Delta = 10^{-4}$, where $\Delta \overset{\text{def}}{=} J_{\mathcal{D}}(\theta^{(k)}) - J_{\mathcal{D}}(\theta^{(k+1)})$.

Unicast: In the unicast case, the dimension of the parameter vector is $d = n = 16$. The CCP was used to train our model based on the training data batch $\mathcal{D}$, and we obtained a tentative solution $\bar{\theta}_{\mathcal{D}}$ in $T_{\text{t}} = 0.41$ s, with an associated cost of $J_{\mathcal{D}}(\bar{\theta}_{\mathcal{D}}) = 143.44$. The blind strategy consists of transmitting the data from the sensor whose observations have the largest sample variance, which has a cost of $J_{\mathcal{D}}^{\text{blind}} = 167.45$. Our observation-driven scheduling and estimation strategy provides a gain of 14.34% over the blind strategy. More importantly, the dimension $d = 16$ allows us to compute the following high probability bound on the optimality gap: $\mathbb{P}\left(\text{gap}(\bar{\theta}_{\mathcal{D}}) \leq 0.5573\right) \geq 0.95$. This bound was computed using the procedure described in the previous section. To compute the upper bound, we used Matlab's global optimization function `patternsearch`, which can efficiently handle nonsmooth objective functions. The average validation time is $T_{\text{v}} = 11.81$ s, with a standard deviation of 1.07 s. Based on the optimality gap above, we claim that we have effectively learned the optimal parameters for Problem 1 in the unicast case.

Broadcast: In the broadcast case, we used the CCP to obtain a tentative solution $\bar{\theta}_{\mathcal{D}}$ with a value $J_{\mathcal{D}}(\bar{\theta}_{\mathcal{D}}) = 31.2196$. The total training time was $T_{\text{t}} = 180$ s. Note that the dimension of the broadcast problem is $d = 2 \times 16 \times 15 = 480$, and that this is a nonconvex, nonsmooth unconstrained optimization problem. The number of variables renders the optimization with any of the built-in global optimization solvers in Matlab, such as the Genetic Algorithm, Simulated Annealing, and Pattern Search, infeasible. Therefore, it is currently impossible to implement the validation analysis in this case. We propose to use $\bar{\theta}_{\mathcal{D}}$ as a warm start to the CCPs on $J_{\mathcal{T}^m}$, $m \in \{1, \ldots, M\}$. If each $\bar{\theta}_{\mathcal{T}^m}$ is a global minimum, we have the following high-confidence upper bound on the optimality gap: $\mathbb{P}\left(\text{gap}(\bar{\theta}_{\mathcal{D}}) \leq 5.3172\right) \geq 0.95$.

Notice that $J_{\mathcal{D}}(\bar{\theta}_{\mathcal{D}}) = 31.22$ corresponds to a gain of approximately 81% over blind scheduling, and 78% relative to the performance over the unicast network. This large gain is due to the high correlation among temperatures and humidity in different rooms. Obviously, the performance gain comes at the price in computational complexity, and the current inability of performing a proper validation analysis using global optimization solvers on nonconvex, nonsmooth large-scale objective functions. The average validation time using the CCP heuristics is $T_{\text{v}} = 131$ s. The recursion converges faster in this case due to the warm start at $\bar{\theta}_{\mathcal{D}}$.

VI. Conclusion

This article aimed at establishing the foundations for scheduling and estimation of sensor measurements when information about the probabilistic model of the problem is imprecise, missing, or incomplete. We considered the design of observation-driven schedulers for a remote sensing system for which the random measurements at the sensors were jointly distributed according to an unknown PDF. Such situations occur in many practical applications where the probabilistic model is not known *a priori* or whose underlying physical processes that generate the data are difficult to obtain. We first derived results and accompanying algorithms that hold for an arbitrary joint PDF, and later we used them in a data-driven framework where training and test datasets were available to design the parameters of a scheduler with performance close to the optimal ones with high probability.

The framework proposed herein assumed that the wireless network can be of two types: unicast or broadcast. For each case, we showed that the optimization problem is nonconvex, but admits a useful DC decomposition, which allowed us to use the CCP to obtain very efficient descent algorithms that were guaranteed to converge to a local minimum of the objective function. The structure of both algorithms was independent of the measurements' joint PDF and can be approximated using data by replacing expectations with their corresponding empirical means. We proved that the two empirical mean approximations converge to the expected costs uniformly almost surely, which is the critical condition for learning the optimal model parameters from data. Moreover, both algorithms can be interpreted as subgradient methods with constant step sizes with guaranteed convergence properties. Such methods are not necessarily convergent if used on nonconvex objective functions.

There are many opportunities for future research that branch out from this work. One possible problem is to devise an online learning scheme where the data becomes available one sample at a time to the system designer, which adaptively reconfigures the scheduling and estimation decision rules over time, instead of using batches of data as it was done here. In the online setting, important research questions arise from the possible lack of synchronization and different timescales across system components. It would also be interesting to assume other classes of parametrizable nonlinear estimators for the optimization problem over broadcast networks. For example, we are interested in the following question: can we train neural networks to serve as estimation policies at the estimators? Moreover, can we find neural network architectures that will preserve a DC decomposition and take advantage of the CCP? The problem of scheduling k out of n sensors can also be posed and solved by augmenting the objective function with a regularizer which enforces k-sparsity. Remarkably, such regularizers admit a DC decomposition [9]. Finally, we suggest an entirely new framework where data is used in a distributionally robust framework, where a set of PDFs consistent with the observed data is constructed and a minimax optimization problem is solved as in [32]. We are particularly interested in using techniques from robust sample average approximation [33], which has both finite sample and asymptotic performance guarantees for a broad class of problems.

APPENDIX A

General Broadcast Case

The results in Section IV hold for an arbitrary number of sensors. In this Appendix, we show the structure of the matrices and vectors that defines the CCP algorithm in the general case. Recall that $\theta^{(k+1)} = \mathbf{A}^{-1}(g(\theta^{(k)}) + \mathbf{b})$, where

$$\mathbf{A} = 2 \cdot \mathrm{diag}(\underbrace{\mathbf{A}_1, \ldots, \mathbf{A}_1}_{n-1}, \ldots, \underbrace{\mathbf{A}_n, \ldots, \mathbf{A}_n}_{n-1}) \tag{98}$$

where

$$\mathbf{A}_j = \begin{bmatrix} \mathbb{E}[X_j^2] & \mathbb{E}[X_j] \\ \mathbb{E}[X_j] & 1 \end{bmatrix} \text{ and } \mathbf{b} = \mathrm{vec}(\mathbf{b}_1, \ldots, \mathbf{b}_n) \tag{99}$$

with

$$\mathbf{b}_j = 2 \cdot \mathrm{vec}\left(\left\{ \begin{bmatrix} \mathbb{E}[X_i X_j] \\ \mathbb{E}[X_i] \end{bmatrix}, i \neq j \right\}\right). \tag{100}$$

The subgradient $g(\theta)$ is computed by $g(\theta) = \mathbb{E}[g(\theta; X)]$, where $g(\theta; x)$ can be computed using Algorithm 1 as $g(\theta; x) = \texttt{subgrad}(\theta; x)$ by substituting $\nabla_\theta G_j(\theta; x)$ with

$$\nabla_\theta G_j(\theta; x) = \mathrm{vec}(\mathbf{k}_1, \ldots, \mathbf{k}_{j-1}, \mathbf{0}, \mathbf{k}_{j+1}, \ldots, \mathbf{k}_n) \tag{101}$$

where

$$\mathbf{k}_\ell = -2 \cdot \mathrm{vec}\left(\left\{ \begin{bmatrix} x_\ell(x_i - (w_{i\ell} x_\ell + b_{i\ell})) \\ (x_i - (w_{i\ell} x_\ell + b_{i\ell})) \end{bmatrix}, i \neq \ell \right\}\right). \tag{102}$$

References

[1] J. Moon and T. Basar, “Static optimal sensor selection via linear integer programming: The orthogonal case,” *IEEE Signal Process. Lett.*, vol. 24, no. 7, pp. 953–957, Jul. 2017.

[2] Y.-C. Ho, “Team decision theory and information structures,” *Proc. IEEE*, vol. 68, no. 6, pp. 644–654, Jun. 1980.

[3] U. Mitra *et al.*, “KNOWME: A case study in wireless body area sensor network design,” *IEEE Commun. Mag.*, vol. 50, no. 5, pp. 116–125, May 2012.

[4] D.-S. Zois, M. Levorato, and U. Mitra, “Energy-efficient, heterogeneous sensor selection for physical activity detection in wireless body area networks,” *IEEE Trans. Signal Process.*, vol. 61, no. 7, pp. 1581–1594, Apr. 2013.

[5] D.-S. Zois, “Sequential decision-making in healthcare IoT: Real-time health monitoring, treatments and interventions,” in *Proc. IEEE 3rd World Forum Internet Things*, 2016, pp. 24–29.

[6] V. Vapnik, *The Nature of Statistical Learning Theory*, 1st ed. New York, NY, USA: Springer-Verlag, 1995.

[7] R. M. Gray, “Quantization in task-driven sensing and distributed processing,” in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2006, pp. 1049–1052.

[8] X. Shen, S. Diamond, Y. Gu, and S. Boyd, “Disciplined convex–concave programming,” in *Proc. IEEE 55th Conf. Decis. Control*, 2016, pp. 1009–1014.

[9] M. Ahn, J.-S. Pang, and J. Xin, “Difference-of-convex learning: Directional stationarity, optimality, and sparsity,” *SIAM J. Optim.*, vol. 27, no. 3, pp. 1637–1665, 2017.

[10] A. Shapiro, “Monte Carlo sampling methods,” in *Handbooks Operations Research and Management Science*, Elsevier, 2003, vol. 10, pp. 353–425.

[11] N. Matni, A. Proutiere, A. Rantzer, and S. Tu, “From self-tuning regulators to reinforcement learning and back again,” in *Proc. IEEE 58th Conf. Decis. Control*, 2019, pp. 3724–3740.

[12] K. Zhou and J. C. Doyle, *Essentials of Robust Control*. Upper Saddle River, NJ, USA: Prentice Hall, 1998, vol. 104.

[13] K. Gatsis and G. J. Pappas, “Statistical learning for analysis of networked control systems over unknown channels,” *Automatica*, vol. 125, 2021, Art. no. 109386.

[14] S. Wu, X. Ren, Q. Jia, K. H. Johansson, and L. Shi, “Learning optimal scheduling policy for remote state estimation under uncertain channel condition,” *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 2, pp. 579–591, Jun. 2020.

[15] A. S. Leong, A. Ramaswamy, D. E. Quevedo, H. Karl, and L. Shi, “Deep reinforcement learning for wireless sensor scheduling in cyber–physical systems,” *Automatica*, vol. 113, 2020, Art. no. 108759.

[16] Y. Li, A. S. Mehr, and T. Chen, “Multi-sensor transmission power control for remote estimation through a SINR-based communication channel,” *Automatica*, vol. 101, pp. 78–86, 2019.

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

[18] M. M. Vasconcelos and U. Mitra, “Observation-driven scheduling for remote estimation of two Gaussian random variables,” *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 1, pp. 232–244, Mar. 2020.

[19] M. M. Vasconcelos, M. Gagrani, A. Nayyar, and U. Mitra, “Optimal scheduling strategy for networked estimation with energy harvesting,” *IEEE Trans. Control Netw. Syst.*, vol. 7, no. 4, pp. 1723–1735, Dec. 2020.

[20] D. Scholz, *Deterministic Global Optimization: Geometric Branch-and-Bound Methods and Their Applications*. Berlin, Germany: Springer, 2012.

[21] A. L. Yuille and A. Rangarajan, “The concave–convex procedure,” *Neural Comput.*, vol. 15, no. 4, pp. 915–936, Apr. 2003.

[22] T. Lipp and S. Boyd, “Variations and extensions of the convex–concave procedure,” *Optim. Eng.*, vol. 17, no. 2, pp. 263–287, Jun. 2016.

[23] B. K. Sriperumbudur and G. R. G. Lanckriet, “On the convergence of the concave–convex procedure,” in *Proc. 22nd Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 1–9.

[24] S. Boyd and J. Park, "Subgradient methods," in *Lecture Notes*, EE 364b. Stanford, CA: Stanford University, 2014.

[25] S. Boyd, J. Duchi, and L. Vandenberghe, "Subgradients," 2018. [Online]. Available: https://stanford.edu/class/ee364b/lectures/subgradients_notes.pdf

[26] S. Mei *et al.*, "The landscape of empirical risk for nonconvex losses," *Ann. Stat.*, vol. 46, no. 6 A, pp. 2747–2774, 2018.

[27] D. S. Bernstein, *Matrix Mathematics: Theory, Facts, and Formulas*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[28] V. Balakrishnan, S. Boyd, and S. Balemi, "Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems," *Int. J. Robust Nonlinear Control*, vol. 1, no. 4, pp. 295–317, 1991.

[29] W.-K. Mak, D. P. Morton, and R. K. Wood, "Monte Carlo bounding techniques for determining solution quality in stochastic programs," *Oper. Res. Lett.*, vol. 24, nos. 1/2, pp. 47–56, 1999.

[30] C. Audet and J. E. Dennis Jr, "Analysis of generalized pattern searches," *SIAM J. Optim.*, vol. 13, no. 3, pp. 889–903, 2003.

[31] L. M. Candanedo, V. Feldheim, and D. Deramaix, "Data driven prediction models of energy use of appliances in a low-energy house," *Energy Build.*, vol. 140, pp. 81–97, 2017.

[32] A. Cherukuri and J. Cortés, "Cooperative data-driven distributionally robust optimization," *IEEE Trans. Autom. Control*, vol. 65, no. 10, pp. 4400–4407, Oct. 2018.

[33] D. Bertsimas, V. Gupta, and N. Kallus, "Robust sample average approximation," *Math. Program.*, vol. 171, no. 1–2, pp. 217–282, 2018.

Marcos M. Vasconcelos received the Ph.D. degree from the University of Maryland, College Park, MD, USA, in 2016.

From 2016 to 2020, he was a Postdoctoral Research Associate with the Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA. He is currently a Research Assistant Professor with the Commonwealth Cyber Initiative at Virginia Tech, Arlington, VA, USA. His research interests include networked control and estimation, multiagent optimization, statistical learning, and systems biology.

Urbashi Mitra received the B.S. and the M.S. degrees from the University of California at Berkeley, Berkeley, CA, USA and the Ph.D. degree from Princeton University, Princeton, NJ, USA.

She is currently the Gordon S. Marshall Professor in Engineering with the University of Southern California, Los Angeles, CA, USA, with appointments in electrical engineering and computer science. Her research interests include wireless communications, communication and sensor networks, biological communication systems, and detection and estimation, and the interface of communication, sensing, and control.

Dr. Mitra is the inaugural Editor-in-Chief for the IEEE TRANSACTIONS ON MOLECULAR, BIOLOGICAL, AND MULTI-SCALE COMMUNICATIONS. She was a member of the IEEE Information Theory Society's Board of Governors (2002–2007, 2012–2017), the IEEE Signal Processing Society's Technical Committee on Signal Processing for Communications and Networks (2012–2017), the IEEE Signal Processing Society's Awards Board (2017–2018), and the Vice Chair of the IEEE Communications Society, Communication Theory Working Group (2017–2018). She has received the 2017 IEEE Communications Society Women in Communications Engineering Technical Achievement Award, a 2016 U.K. Royal Academy of Engineering Distinguished Visiting Professorship, a 2016 US Fulbright Scholar Award, a 2016–2017 U.K. Leverhulme Trust Visiting Professorship, 2015–2016 IEEE Communications Society Distinguished Lecturer, 2012 Globecom Signal Processing for Communications Symposium Best Paper Award, 2012 US National Academy of Engineering Lillian Gilbreth Lectureship, the 2009 DCOSS Applications & Systems Best Paper Award, Texas Instruments Visiting Professor (Fall 2002, Rice University), 2001 Okawa Foundation Award, 2000 OSU College of Engineering Lumley Award for Research, 1997 OSU College of Engineering MacQuigg Award for Teaching, and a 1996 National Science Foundation CAREER Award.