Proximity Without Consensus in Online Multiagent Optimization

Alec Koppel, Brian M. Sadler, and Alejandro Ribeiro

***Abstract*—We consider stochastic optimization problems in multiagent settings, where a network of agents aims to learn parameters that are optimal in terms of a global convex objective, while giving preference to locally observed streaming information. To do so, we depart from the canonical decentralized optimization framework where agreement constraints are enforced, and instead formulate a problem where each agent minimizes a global convex objective while enforcing network proximity constraints. This formulation includes online consensus optimization as a special case, but allows for the more general hypothesis that there is data heterogeneity across the network. To solve this problem, we propose using a stochastic saddle point algorithm inspired by Arrow and Hurwicz. This method yields a decentralized algorithm for processing observations sequentially received at each node of the network. Using Lagrange multipliers to penalize the discrepancy between them, only neighboring nodes exchange model information. We establish that under a constant step-size regime the time-average suboptimality and constraint violation are contained in a neighborhood whose radius vanishes with increasing number of iterations. As a consequence, we prove that the time-average primal vectors converge to the optimal objective while satisfying the network proximity constraints. We apply this method to the problem of sequentially estimating a correlated random field in a sensor network, as well as an online source localization problem, both of which demonstrate the empirical validity of the aforementioned convergence results.**

***Index Terms*—Distributed optimization, online learning, multiagent systems, convex optimization, random field estimation, saddle point method.**

I. Introduction

WE CONSIDER online multi-agent optimization problems, where a group of interconnected agents aim to minimize a global objective $f = \sum_i f_i$ which may be written as a sum of local (non-strongly) convex objectives f_i available at different nodes i of a network $\mathcal{G} = (V, \mathcal{E})$. The problem is online because information upon which the local objectives depend is sequentially and locally received by each agent. We consider the setting where agents aim to keep their decision variables *close* to one another but *not coincide* in order to minimize this global objective while giving preference to possibly distinct local signals. The motivation for this problem comes from the fact that consensus optimization methods implicitly operate on the hypothesis that the distribution of observations at each node is identical, which does not hold for a variety of problems in signal processing [3] and robotics [4], [5].

Distributed optimization has a rich history in wireless communications [6], sensor networks [7], and machine learning [8]. Prior approaches to this problem for the case that objectives are convex require each agent to keep a local copy of the global decision variable, and approximately enforce an agreement constraint between the local copies at each iteration. To do so, various information mixing strategies have been proposed in which agents combine local gradient steps with a weighted average of their neighbors variables [9], [10], variations of primal methods which shuffle the order of weighted averaging and local gradient computations [11], [12], dual reformulations where each agent ascends in the dual domain [13]–[15], and primal-dual methods which combine primal descent with dual ascent [16]–[18]. Stochastic optimization, which allows for sequential processing of observations as they are received, are mostly based upon stochastic approximation [19], [20], and have been successfully applied to extend multi-agent problems to the online domain [4], [21], [22].

In distributed optimization problems, agent agreement may not always be the primary goal. In large-scale settings where one aims to leverage parallel processing architectures to alleviate computational bottlenecks, agreement constraints are suitable. In contrast, if there are different priors on information received at distinct subsets of agents, then requiring the network to reach a common decision may degrade local predictive accuracy. Specifically, if the observations at each node are independent but *not* identically distributed, consensus may yield a sub-optimal solution. Moreover, there are tradeoffs in complexity and communications, and it may be that only a subset of nodes requires a solution.

Various attempts to extend multi-agent optimization techniques to exploit heterogeneous correlation structures among observations received by each agent have been proposed,

Manuscript received June 20, 2016; revised October 17, 2016, January 13, 2017, and February 24, 2017; accepted February 27, 2017. Date of publication March 22, 2017; date of current version April 18, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chandra Murthy. This work was supported in part by the National Science Foundation under Grant CCF-1017454, Grant CCF-0952867, and Grant ONR N00014-12-1-0997, in part by the Association of Research Libraries Micro Autonomous Systems and Technology CTA, and in part by the American Society for Engineering Education Science, Mathematics, And Research For Transformation. This paper was presented in part at the International Conference Acoustics Speech Signal Processing, Shanghai, China, March 20–25, 2016, and in part at the Global Conference on Signal and Information Processing, Washington, DC, USA, December 7–9, 2016. *(Corresponding Author: Alejandro Ribeiro.)*

A. Koppel and A. Ribeiro are with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: akoppel@seas.upenn.edu; aribeiro@seas.upenn.edu).

B. M. Sadler is with the U.S. Army Research Laboratory, Adelphi, MD 20783 USA (e-mail: brian.m.sadler6.civ@mail.mil).

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

Digital Object Identifier 10.1109/TSP.2017.2686368

motivated by multi-task learning [23]. For instance, attempts to extend primal averaging techniques to generic inequality constraints in the online decentralized setting via penalty methods were developed in [24], but require the use of diminishing step-size rules and growing penalty parameters, which are outperformed by constant learning rates in dynamic estimation settings. An alternative primal averaging approach for multi-agent systems with multiple distinct but correlated optima was developed in [25], but only for the square loss. In the later work, Euclidean penalties are added to agents' local objectives to incentivize tracking of multiple interrelated optima, which may or may not capture a generic correlation structure among agents' data streams.

In this paper, we seek to solve problems in which each agent aims to minimize a global cost $\sum_i f_i$ subject to a network proximity constraint, which allows agents the leeway to select actions which are good with respect to a global cost while not ignoring the structure of locally observed information. This setting may correspond to a multi-target tracking problem in a sensor network or a collaborative learning task in a robotic network where each robot is operating in a distinct domain, i.e. instances of multi-task learning [23]. However, we allow for constraints to be generically chosen convex inequalities, rather than a specific Euclidean penalty, as in [25]. We design multi-agent optimization strategies where agents reach a common understanding of global information, while still retaining their local perspectives.

We propose a stochastic variant of the saddle point method [16], [17] to solve online multi-agent optimization problems with network proximity constraints. Our main technical contribution is to demonstrate that saddle point iterates converge in expectation to a primal-dual optimal pair of this problem when a constant algorithm step-size is chosen. We begin the paper in Section II with a discussion of decentralized stochastic optimization problems with network proximity constraints and present an example to decentralized estimation of a correlated random field in a sensor network to illustrate key concepts. The saddle point algorithm is developed in Section III by drawing parallels with deterministic and stochastic optimization, and exploiting factorization properties of the Lagrangian. The method relies on the definition of an augmented stochastic Lagrangian associated with the instantaneous global cost function, and operates by alternating primal descent and dual accept steps. In Section IV, we establish mean convergence properties of this method to a primal-dual optimal pair in the constant step-size regime. We demonstrate the proposed method's utility on a spatially correlated random field estimation problem in a sensor network in Section V, and apply this tool to a source localization problem in Section VI. Finally, we conclude in Section VII.

II. Problem Formulation

We consider agents i of a symmetric, connected, and directed network $\mathcal{G} = (V, \mathcal{E})$ (Assumption 1) with $|V| = N$ nodes and $|\mathcal{E}| = M$ edges and denote as $n_i := \{j : (i,j) \in \mathcal{E}\}$ the neighborhood of agent i. For simplicity we assume that the number of edges M is even. Each of the agents is associated with a (non-strongly) convex loss function $f_i : \mathcal{X} \times \Theta_i \to \mathbb{R}$ that is parameterized by a decision variable $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^p$ and a random vector $\boldsymbol{\theta}_i \in \Theta_i \subset \mathbb{R}^q$ with a proper distribution. The functions $f_i(\mathbf{x}_i, \boldsymbol{\theta}_i)$ for different $\boldsymbol{\theta}_i$ are interpreted as the merit of a particular statistical model $\mathbf{x}_i$, and the random vector $\boldsymbol{\theta}$ may be particularized, for instance, to a random pair $\boldsymbol{\theta} = (\mathbf{z}, \mathbf{y})$. For this case, the random pair correspond to, e.g., feature vectors $\mathbf{z}$ together with their binary labels $\mathbf{y} \in \{-1, 1\}$ or real values $\mathbf{y} \in \mathbb{R}$, respectively, for classification or regression.

In this work, we focus on the case where $\boldsymbol{\theta}_i$ represents data which revealed to node i *sequentially* through realizations $\boldsymbol{\theta}_{i,t}$ at time t, and agents would like to process this information incrementally. Mathematically this is equivalent to the case where the total number of samples T revealed to agent i is not necessarily finite. In the online setting considered here the functions $f_i(\mathbf{x}_i, \boldsymbol{\theta}_i)$ are termed instantaneous because they are observed at particular points in time associated with realizations $\boldsymbol{\theta}_{i,t}$ of the random vector $\boldsymbol{\theta}_i$; see Section III. A possible goal for agent i is the computation of the optimal local estimate,

$$\mathbf{x}_i^{\mathrm{L}} := \arg\min_{\mathbf{x}_i \in \mathcal{X}} F_i(\mathbf{x}_i) := \arg\min_{\mathbf{x}_i \in \mathcal{X}} \mathbb{E}_{\boldsymbol{\theta}_i} \left[f_i(\mathbf{x}_i, \boldsymbol{\theta}_i) \right] . \quad (1)$$

We refer to $F_i(\mathbf{x}_i) := \mathbb{E}_{\boldsymbol{\theta}_i} \left[f_i(\mathbf{x}_i, \boldsymbol{\theta}_i) \right]$ as the local average function at node i. We further assume $\mathcal{X}$ to be a compact convex subset of $\mathbb{R}^p$ associated with the p-dimensional parameter vector of agent i.

When we consider the network as a whole we can define the stacked vector $\mathbf{x} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$, which is an element of the product set $\mathcal{X}^N \subset \mathbb{R}^{Np}$, and the aggregate function $F(\mathbf{x}) := \sum_{i=1}^N \mathbb{E}_{\boldsymbol{\theta}_i} \left[f_i(\mathbf{x}_i, \boldsymbol{\theta}_i) \right]$. It then follows that the set of problems in (1) is equivalent to the aggregate problem

$$\mathbf{x}^{\mathrm{L}} = \arg\min_{\mathbf{x} \in \mathcal{X}^N} F(\mathbf{x}) := \arg\min_{\mathbf{x} \in \mathcal{X}^N} \sum_{i=1}^N \mathbb{E}_{\boldsymbol{\theta}_i} \left[f_i(\mathbf{x}_i, \boldsymbol{\theta}_i) \right] . \quad (2)$$

For convenience, we further define the stacked instantaneous function as $f(\mathbf{x}, \boldsymbol{\theta}) = \sum_i f_i(\mathbf{x}_i, \boldsymbol{\theta}_i)$. That (1) and (2) describe the same problem is true because there is no coupling between the variables $\mathbf{x}_i$ at different agents. In many situations, however, the parameters $\mathbf{x}_i^{\mathrm{L}}$ that different agents want to estimate are related. It then makes sense to couple decisions of different agents as a means of letting agents exploit each others' observations. Consensus optimization problems work on the hypothesis that all agents are interested in learning the same decision parameters $\mathbf{x}_i$ for all $i \in V$. In this case, we modify (2) by introducing consensus constraints of the form

$$\mathbf{x}_i = \mathbf{x}_j, \text{ for all } j \in n_i . \quad (3)$$

For a connected network this constraint makes all variables $\mathbf{x}_i$ equal – hence the definition as a consensus problem. This hypothesis implicitly only makes sense in cases where agents observe information drawn from a common distribution, which may be overly restrictive. In general, parameters of nearby nodes are expected to be close but are not necessarily all equal, as is the situation in, e.g., the estimation of a smooth field that is albeit not uniform. To model this situation we introduce a convex local proximity function with real-valued range of the form $h_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ and a tolerance $\gamma_{ij} \geq 0$. These are used to couple the decisions of agent i to those of its neighbors $j \in n_i$

through the definition of the optimal estimates as the solution of the constrained optimization problem

$$\mathbf{x}^* \in \underset{\mathbf{x}\in\mathcal{X}^N}{\arg\min} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}_i}\left[f_i(\mathbf{x}_i,\boldsymbol{\theta}_i)\right] \quad \text{s.t. } h_{ij}(\mathbf{x}_i,\mathbf{x}_j) \leq \gamma_{ij}, \quad \text{for all } j \in n_i. \tag{4}$$

In the formulation in (4), $\mathbf{x}^*$ belongs to a set of constrained optimizers $\mathcal{X}^*$, i.e., $\mathbf{x}^*$ is not unique, due to the weak convexity of the local objectives $F_i(\mathbf{x})$. Moreover, for this set to be non-empty, we assume that the set of optimizers $\mathcal{X}^*$ has non-empty intersection with the primal feasible set $\mathcal{X}$ – a condition satisfied under Slater's condition (Assumption 4).

We assume that the proximity function $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ that couples node i to node j is equivalent to the proximity function $h_{ji}(\mathbf{x}_j,\mathbf{x}_i)$ that couples node j to node i, i.e., that for all $\mathbf{x}_i$ and $\mathbf{x}_j$ we have $h_{ij}(\mathbf{x}_i,\mathbf{x}_j) = h_{ji}(\mathbf{x}_j,\mathbf{x}_i)$ and $\gamma_{ij} = \gamma_{ji}$. This implies that the constraints $h_{ij}(\mathbf{x}_i,\mathbf{x}_j) \leq \gamma_{ij}$ and $h_{ji}(\mathbf{x}_j,\mathbf{x}_i) \leq \gamma_{ji}$ are redundant. We also define the stacked constraint function $h : \mathcal{X}^N \to \mathbb{R}^M$. We keep them separate to maintain symmetry of the algorithm derived in Section III.

The consensus constraints in (3) are a particular example of a proximity function $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ but so is the norm constraint $\|\mathbf{x}_i - \mathbf{x}_j\|^2 \leq \gamma_{ij}$. This latter choice makes the estimates $\mathbf{x}_i^*$ and $\mathbf{x}_j^*$ of neighboring nodes close to each other but not necessarily equal. Implicitly, this allows i to incorporate the (relevant) information of neighboring nodes without detrimentally incorporating the information of far away nodes that are only weakly correlated with the estimator of node i.

The goal of this paper is to develop an algorithm to solve (4) in distributed online settings where nodes don't know the distribution of the random vector $\boldsymbol{\theta}_i$ but observe local instantaneous functions $f_i(\mathbf{x}_i,\boldsymbol{\theta}_i)$ sequentially. An important observation here is that the workhorse distributed gradient descent (DGD) [9], [10], [22], [26] and dual methods [13]–[15] can't be used to solve (4) because they work only when the constraints $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ are linear. Extensions of DGD to inequality constraints have been considered in [24], but constraints are assumed to be local only, and thus may not capture cross-agent correlations. While penalty-based variants of [24] may be developed for (4), their performance guarantees would hinge on use of attenuating learning rates, which have been found to be empirically inferior to methods based on constant step-sizes. These observations motivate an alternative approach based on Lagrange duality. In particular, we will see that a stochastic saddle point method can be distributed when the functions $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ are not necessarily linear and converges to the solution of (4) when local instantaneous functions $f_i(\mathbf{x}_i,\boldsymbol{\theta}_i)$ are independently sampled over time. Before developing this algorithm, we discuss a representative example to clarify ideas.

Example (LMMSE Estimation of a Random Field): A Gauss-Markov random field is one in which the value of the field at the location of sensor i, denoted by $\mathbf{x}_i$, is of interest. Consider a sequential estimation problem in which the nodes of the sensor network acquire noisy linear transformations of the field's value at their respective positions. Formally, let $\boldsymbol{\theta}_{i,t} \in \mathbb{R}^q$ be the observation collected by sensor i at time t. Observations $\boldsymbol{\theta}_{i,t}$ are noisy linear transformations $\boldsymbol{\theta}_{i,t} = \mathbf{H}_i\mathbf{x}_i + \mathbf{w}_{i,t}$ of a signal $\mathbf{x}_i \in \mathbb{R}^p$ contaminated with Gaussian noise $\mathbf{w}_{i,t} \sim \mathcal{N}(0,\sigma^2\mathbf{I})$ independently distributed across nodes and time. Ignoring neighboring observations, the minimum mean square error local estimation problem at node i can then be written in the form of (1) with $f_i(\mathbf{x}_i,\boldsymbol{\theta}_i) = \|\mathbf{H}_i\mathbf{x}_i - \boldsymbol{\theta}_i\|^2$. The quality of these estimates can be improved using the correlated information of adjacent nodes but would be hurt by trying to make estimates uniformly equal across the network. This problem specification can be captured by the mathematical formulation

$$\mathbf{x}^* := \underset{\mathbf{x}\in\mathcal{X}^N}{\arg\min} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}_i}\left[\|\mathbf{H}_i\mathbf{x}_i - \boldsymbol{\theta}_i\|^2\right] \quad \text{s.t. } (1/2)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \leq \gamma_{ij}, \quad \text{for all } j \in n_i. \tag{5}$$

The constraint $(1/2)\|\mathbf{x}_i - \mathbf{x}_j\|^2 \leq \gamma_{ij}$ makes the estimate $\mathbf{x}_i^*$ of node i close to the estimates $\mathbf{x}_j^*$ of neighboring nodes $j \in n_i$ but not so close to the estimates $\mathbf{x}_k^*$ of nonadjacent nodes $k \notin n_i$. The problem formulation in (5) is a particular case of (4) with $f_i(\mathbf{x}_i,\boldsymbol{\theta}_i) = \|\mathbf{H}_i\mathbf{x}_i - \boldsymbol{\theta}_i\|^2$ and $h_{ij}(\mathbf{x}_i,\mathbf{x}_j) = (1/2)\|\mathbf{x}_i - \mathbf{x}_j\|^2$.

III. Algorithm Development

Recall that a decentralized algorithm is one in which node i has access to local functions $f_i(\mathbf{x}_i,\boldsymbol{\theta}_i)$ and local constraints $h_{ij}(\mathbf{x}_i,\mathbf{x}_j) \leq \gamma_{ij}$ and exchanges information with neighbors $j \in n_i$ only. Recall also that the algorithm is further said to be online if the distribution of θ_i is unknown and agent i has access to independent observations $\boldsymbol{\theta}_{i,t}$ that are acquired sequentially. Our goal is to develop an online decentralized algorithm to solve (4). To achieve this we consider the approximate Lagrangian relaxation of (4) which we state as

$$\mathcal{L}(\mathbf{x},\boldsymbol{\lambda}) = \sum_{i=1}^{N}\Bigg[\mathbb{E}_{\boldsymbol{\theta}_i}\left[f_i(\mathbf{x}_i,\boldsymbol{\theta}_i)\right] + \frac{1}{2}\sum_{j\in n_i}\left(\lambda_{ij}\left(h_{ij}(\mathbf{x}_i,\mathbf{x}_j) - \gamma_{ij}\right) - \frac{\delta\epsilon_t}{2}\lambda_{ij}^2\right)\Bigg], \tag{6}$$

where $\lambda_{ij} \in \mathbb{R}^+$ is a nonnegative Lagrange multiplier associated with the proximity constraint between node i and node j, and the factor of $1/2$ comes from the redundancy of the constraints $h_{ij}(\mathbf{x}_i,\mathbf{x}_j) \leq \gamma_{ij}$ and $h_{ji}(\mathbf{x}_j,\mathbf{x}_i) \leq \gamma_{ij}$, and helps scale the contribution of each when computing gradients (see Proposition 1). Observe that (6) *does not* define the Lagrangian of the optimization problem (4), but instead defines an *augmented Lagrangian* due to the presence of the last term on the right-hand side. This last term $-(\delta\epsilon_t/2)\lambda_{ij}^2$, with scalar parameters δ and ϵ_t, is a regularizer on the dual variable, whose utility arises in controlling the accumulation of constraint violation of the algorithm over time. See Section IV for details.

To solve (4), stochastic approximation is necessary. In particular, the necessity for operating with stochastic gradients rather than true gradients comes from the fact that computing

gradients of the statistical average objective in (4) has complexity that is at least on the order of the sample size T, which in setting considered here may be infinite. Furthermore, due to the online nature of the problem, at time t, each individual agent in the network only has access to random variables $\{\boldsymbol{\theta}_{i,u}\}_{u\leq t}$. Thus, computations involving the average objective involve data $\{\boldsymbol{\theta}_{i,u}\}_{u>t}$ which is not yet observed, and thus is unavailable.

Therefore, we propose applying a stochastic saddle point algorithm to (6) which operates by alternating primal and dual stochastic gradient descent and ascent steps respectively. Consider the stochastic approximation of the augmented Lagrangian evaluated at observed realizations $\boldsymbol{\theta}_{i,t}$ of the random vectors $\boldsymbol{\theta}_i$, which we define as

$$\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}) = \sum_{i=1}^{N}\Big[f_i(\mathbf{x}_i,\boldsymbol{\theta}_{i,t}) + \frac{1}{2}\sum_{j\in n_i}\lambda_{ij}\left(h_{ij}(\mathbf{x}_i,\mathbf{x}_j)-\gamma_{ij}\right) - \frac{\delta\epsilon_t}{2}\lambda_{ij}^2\Big]. \quad (7)$$

Define the stacked dual variable as $\boldsymbol{\lambda} := [\lambda_1;\cdots;\lambda_M]\in\mathbb{R}^M$. Moreover, denote the network aggregate random vector as $\boldsymbol{\theta} = [\boldsymbol{\theta}_1;\cdots;\boldsymbol{\theta}_N]$. The stochastic saddle point method applied to the stochastic Lagrangian stated in (7) takes the form

$$\mathbf{x}_{t+1} = \mathcal{P}_{\mathcal{X}^N}\left[\mathbf{x}_t - \epsilon_t\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\right], \quad (8)$$

$$\boldsymbol{\lambda}_{t+1} = \left[\boldsymbol{\lambda}_t + \epsilon_t\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\right]_+, \quad (9)$$

where $\nabla_{\mathbf{x}}\hat{\mathcal{L}}(\mathbf{x}_t,\boldsymbol{\lambda}_t)$ and $\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}(\mathbf{x}_t,\boldsymbol{\lambda}_t)$, are the primal and dual stochastic gradients of the augmented Lagrangian with respect to $\mathbf{x}$ and $\boldsymbol{\lambda}$, respectively. These stochastic subgradients are approximations of the gradients of (6) evaluated at the current realization of the random vector $\boldsymbol{\theta}$. The notation $\mathcal{P}_{\mathcal{X}}^N(\mathbf{x})$ denotes component-wise orthogonal projection of the individual primal variables $\mathbf{x}_i$ onto the given convex compact set $\mathcal{X}$, and $[\cdot]_+$ denotes the projection onto the M-dimensional nonnegative orthant $\mathbb{R}_+^M$. As an abuse of notation, we also use $[\cdot]_+$ to denote scalar positive projection where appropriate.

The method stated in (8), (9) can be implemented with decentralized computations across the network, as we state in the following proposition.

Proposition 1: Let $\mathbf{x}_{i,t}$ be the ith component of the primal iterate $\mathbf{x}_t$ and $\lambda_{ij,t}$ the i,jth component the dual iterate $\boldsymbol{\lambda}_t$. The primal variable update is equivalent to the set of N parallel local variable updates

$$\mathbf{x}_{i,t+1} = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}_{i,t} - \epsilon_t\Big(\nabla_{\mathbf{x}_i}f_i(\mathbf{x}_{i,t};\boldsymbol{\theta}_{i,t}) + \frac{1}{2}\sum_{j\in n_i}(\lambda_{ij,t}+\lambda_{ji,t})\nabla_{\mathbf{x}_i}h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})\Big)\Big]. \quad (10)$$

Likewise, the dual variable updates in (9) are equivalent to the M parallel updates

$$\lambda_{ij,t+1} = \Big[(1-\epsilon_t^2\delta)\lambda_{ij,t} + \epsilon_t\left(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})-\gamma_{ij}\right)\Big]_+. \quad (11)$$

Proof: See Appendix A. ■

Algorithm 1: SSPM: Stochastic Saddle Point Method.

Require: initialization $\mathbf{x}_0$ and $\boldsymbol{\lambda}_0 = \mathbf{0}$, step-size ϵ_t, regularizer δ

1: **for** $t = 1,2,\ldots,T$ **do**
2: **loop in parallel** agent $i\in V$
3: Send primal and dual variables $\mathbf{x}_{i,t},\boldsymbol{\lambda}_{ij,t}$ to nbhd. $j\in n_i$
4: Receive variables $\mathbf{x}_{j,t},\boldsymbol{\lambda}_{ij,t}$ from neighbors $j\in n_i$
5: Update local parameter $\mathbf{x}_{i,t}$ with (10)

$$\mathbf{x}_{i,t+1} = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}_{i,t} - \epsilon_t\Big(\nabla_{\mathbf{x}_i}f_i(\mathbf{x}_{i,t};\boldsymbol{\theta}_{i,t}) + \sum_{j\in n_i}(\lambda_{ij,t}+\lambda_{ji,t})\nabla_{\mathbf{x}_i}h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})\Big)\Big].$$

6: **end loop**
7: **loop in parallel** communication link $(i,j)\in\mathcal{E}$
8: Update dual variables at network link (i,j) [cf. (11)]

$$\lambda_{ij,t+1} = \Big[(1-\epsilon_t^2\delta)\lambda_{ij,t} + \epsilon_t\left(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})-\gamma_{ij}\right)\Big]_+$$

9: **end loop**
10: **end for**

With primal variables $\mathbf{x}_{i,t}$ and Lagrange multipliers $\lambda_{ij,t}$ maintained and updated by node i, Proposition 1 implies that the saddle point method in (8), (9) can be translated into a decentralized protocol in which: (i) The primal and dual variables variables of distinct agents across the network are decoupled from one another. (ii) The updates require exchanges of information among neighboring nodes only. This protocol is summarized in Algorithm 1.

Indeed, in the primal update in (11) agent i can compute the stochastic gradient $\nabla_{\mathbf{x}_i}f_i(\mathbf{x}_{i,t};\boldsymbol{\theta}_{i,t})$ of its objective function by making use of its local observations $\boldsymbol{\theta}_{i,t}$ and its decision variable $\mathbf{x}_{i,t}$ at the previous time slot t. To compute the gradients of the constraint functions $\nabla_{\mathbf{x}_i}h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})$ the primal variables $\mathbf{x}_{j,t}$ of neighboring nodes $j\in n_i$ are needed on top of the local variables $\mathbf{x}_{i,t}$, but these can be communicated from neighbors. To implement (10) agent i also needs access to the Lagrange multipliers $\lambda_{ij,t}$ associated with the network proximity constraints $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ and the multipliers $\lambda_{ji,t}$ associated with the network proximity constraints $h_{ji}(\mathbf{x}_j,\mathbf{x}_i)$. The multipliers $\lambda_{ij,t}$ are locally available at node i and the multipliers $\lambda_{ji,t}$ can be communicated from neighbors.

To implement the dual update in (11) agent i needs access to its own dual variable $\lambda_{ij,t}$ as well as the local decision variables $\mathbf{x}_{i,t}$. It also needs access to the primal variables $\mathbf{x}_{j,t}$ of neighbors $j\in n_i$ to compute the local dual gradient which is given as the constraint slack $h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})-\gamma_{ij}$. As in the primal, these neighboring variables can be communicated from neighbors. We can then implement (10) after nodes exchange primal and dual variables $\mathbf{x}_{i,t}$ and $\lambda_{ij,t}$, proceed to implement (11) after they exchange updated primal variables $\mathbf{x}_{i,t}$, and conclude with

the exchange of primal and dual variables $\mathbf{x}_{i,t}$ and $\lambda_{ij,t}$ that are needed to implement the primal iteration at time t. These local operations repeated in synchrony by all nodes is equivalent to the centralized operations in (8), (9).

In the following section, we analyze the iterations in (8), (9), which implies convergence of the equivalent iterations in (10), (11). We close here with an example and a remark.

Example (LMMSE Estimation of a Random Field): Revisit the random filed estimation problem of Section II that we summarize in the problem formulation in (5). Recalling the identifications $f_i(\mathbf{x}_i, \boldsymbol{\theta}_i) = \|\mathbf{H}_i\mathbf{x}_i - \boldsymbol{\theta}_i\|^2$ and $h_{ij}(\mathbf{x}_i, \mathbf{x}_j) = (1/2)\|\mathbf{x}_i - \mathbf{x}_j\|^2$ it follows that the local primal update in (10) takes the form

$$\mathbf{x}_{i,t+1} = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}_{i,t} - \epsilon_t\Big[2\mathbf{H}_i^T\left(\mathbf{H}_i\mathbf{x}_{i,t} - \boldsymbol{\theta}_{i,t}\right) + \frac{1}{2}\sum_{j\in n_i}\Big(\lambda_{ij,t} + \lambda_{ji,t}\Big)\Big(\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\Big)\Big]\Big]. \quad (12)$$

Likewise, the specific form of the dual update in (11) is

$$\lambda_{ij,t+1} = \Big[(1-\epsilon_t^2\delta)\lambda_{ij,t} + (\epsilon_t/2)\left(\|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|^2 - \gamma_{ij}\right)\Big]_+. \quad (13)$$

The empirical utility of the decentralized estimation scheme in (12) (13) is studied in Section V. Alternative functional forms for the network proximity constraints are studied for a source localization problem in Section VI . ■

Remark 1: If the proximity constants are $\gamma_{ij} = \gamma_{ji}$ and the initial Lagrange multipliers satisfy $\lambda_{ij,0} = \lambda_{ji,0}$ it follows from (11) that $\lambda_{ij,t} = \lambda_{ji,t}$ for all subsequent times t. This is as it should be because the constraints $h_{ij}(\mathbf{x}_i, \mathbf{x}_j) \leq \gamma_{ij}$ and $h_{ji}(\mathbf{x}_j, \mathbf{x}_i) \leq \gamma_{ji}$ are redundant. If these multipliers are equal for all times, the primal update in (10) does not necessitate exchange of dual variables. This does not save communication cost as it is still necessary to exchange primal variables $\mathbf{x}_{i,t}$.

IV. Convergence Analysis

We turn to establishing that the saddle point algorithm defined by (8), (9) converges to the primal-dual optimal point of the problem stated in (4) when a constant algorithm step-size is used. In particular, we establish bounds on the objective function error sequence $F(\mathbf{x}_t) - F(\mathbf{x}^*)$ and the network-aggregate constraint violation, both in expectation, where $\mathbf{x}^*$ is defined by (4). As a consequence, the time-average primal vector converges to the optimal objective function $F(\mathbf{x}^*)$ at a rate of $\mathcal{O}(1/\sqrt{T})$, while incurring constraint violation on the order of $\mathcal{O}(T^{-1/4})$, both on average, where T is the total number of iterations. To establish these results, we note some facts of the problem setting, and then introduce a few standard assumptions.

First, observe that the dual stochastic gradient is independent of random vectors $\boldsymbol{\theta}_{i,t}$ [cf. (11)], and hence for all t,

$$\nabla_{\boldsymbol{\lambda}}\mathcal{L}(\mathbf{x}_t, \boldsymbol{\lambda}_t) = \nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t). \quad (14)$$

Also pertinent to analyzing the performance of the stochastic saddle point method is the fact that the primal stochastic gradient of the Lagrangian is an unbiased estimator of the true primal gradient. Let $\mathcal{F}_t$ be a sigma algebra that measures the history of the algorithm up until time t, i.e., a collection that contains at least the variables $\{\mathbf{x}_u, \boldsymbol{\lambda}_u, \boldsymbol{\theta}_u\}_{u=1}^t \subseteq \mathcal{F}_t$. That the primal stochastic gradient is an unbiased estimate of the true primal gradients means that,

$$\mathbb{E}\left[\nabla_{\mathbf{x}}\hat{\mathcal{L}}(\mathbf{x}_t, \boldsymbol{\lambda}_t) \,\middle|\, \mathcal{F}_t\right] = \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}_t, \boldsymbol{\lambda}_t)\,. \quad (15)$$

Furthermore, the compactness of the sets $\mathcal{X}$ permits the bounding of the magnitude of the iterates $\mathbf{x}_{i,t}$ by a constant R/N, which in turn implies that the network-wide iterates may be bounded in magnitude as

$$\|\mathbf{x}_t\| \leq R \text{ for all } t\,. \quad (16)$$

To prove convergence of the stochastic saddle point method, some conditions are required of the network, loss functions, and constraints, which we state below.

Assumption 1: (Network connectivity) The network $\mathcal{G}$ is symmetric and connected with diameter D.

Assumption 2: (Smoothness) The stacked instantaneous objective is Lipschitz continuous in expectation with constant L_f, i.e., for distinct primal variables $\mathbf{x}, \tilde{\mathbf{x}} \in \mathcal{X}$ and all $\boldsymbol{\theta}$,

$$\mathbb{E}\left[\|f(\mathbf{x}, \boldsymbol{\theta}) - f(\tilde{\mathbf{x}}, \boldsymbol{\theta})\|\right] \leq L_f\|\mathbf{x} - \tilde{\mathbf{x}}\|\,. \quad (17)$$

Moreover, the stacked constraint function $h(\mathbf{x})$ is Lipschitz continuous with modulus L_h. That is, for distinct primal variables $\mathbf{x}, \tilde{\mathbf{x}} \in \mathcal{X}$, we may write

$$\|h(\mathbf{x}) - h(\tilde{\mathbf{x}})\| \leq L_h\|\mathbf{x} - \tilde{\mathbf{x}}\|. \quad (18)$$

Assumption 3: (Stochastic Gradient Variance) The expected square-magnitudes of the primal gradients of the local objectives are upper bounded, i.e.

$$\max_{i\in V}\mathbb{E}\left[\|\nabla_{\mathbf{x}_i} f(\mathbf{x}_i, \boldsymbol{\theta}_i)\|^2\right] \leq \sigma_{\mathbf{x}}^2 \quad (19)$$

Assumption 4: (Existence of Optima) The set of primal-dual optimal pairs $\mathcal{X}^* \times \boldsymbol{\Lambda}^*$ of the constrained problem (4) has nonempty intersection with the feasible domain $\mathcal{X}^N \times \mathbb{R}_+^M$.

Assumption 1 ensures that the graph is connected and the rate at which information diffuses across the network is finite. This condition is standard in distributed algorithms [13], [22]. Assumption 2 states that the stacked objective and constraints are sufficiently smooth, and have bounded gradients, a stipulation that frequently is required in the analysis of convex optimization methods [27], [28]. Assumption 3 is standard in the analysis of stochastic approximation methods [20]. Moreover, Assumption 4 establishes that the restriction to a finite primal domain $\mathcal{X}$ does not preclude our ability to find a primal-dual optimal pair of (4), and has been used to establish existence of solutions to constrained convex programs [29]. It easily may be guaranteed by the existence of a strictly feasible $\mathbf{x}$, i.e., Slater's condition holds [17, Assumption 2].

Assumption 2 taken with the bound on the primal iterates [cf. (16)] permits the bounding of the expected primal and dual gradients of the Lagrangian by constant terms and terms that depend on the magnitude of the dual variable. In particular, we compute the mean-square-magnitude of the primal gradient of

the stochastic augmented Lagrangian as

$$\begin{aligned}\mathbb{E}[\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda})\|^2] &\leq N\max_i \mathbb{E}\left[\|\nabla_{\mathbf{x}_i} f_i(\mathbf{x}_i,\boldsymbol{\theta}_t)\|^2\right] \\ &\quad + M\|\boldsymbol{\lambda}\|^2 \max_{(i,j)\in\mathcal{E}} \|\nabla_{\mathbf{x}_i} h_{ij}(\mathbf{x}_i,\mathbf{x}_j)\|^2 \\ &\leq N\sigma_{\mathbf{x}}^2 + ML_h^2\|\boldsymbol{\lambda}\|^2 \leq (N+M)L^2(1+\|\boldsymbol{\lambda}\|^2) \qquad (20)\end{aligned}$$

where we have applied the triangle inequality in the first expression and considered the worst-case bounds. The second inequality makes use of the smoothness properties defined in (17) and the fact that the constraint $h_{ij}(\mathbf{x}_i,\mathbf{x}_j)$ is independent of $\boldsymbol{\theta}$. On the right-hand side of (20) we have defined $L := \max(\sigma_{\mathbf{x}}, L_h)$ to simplify the expression. We further may derive a bound on the expected magnitude of the dual stochastic gradient of the augmented Lagrangian by making use of Assumption 2. That is,

$$\begin{aligned}\mathbb{E}\left[\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda})\|^2\right] &\leq M\max_{(i,j)\in\mathcal{E}}(h_{ij}(\mathbf{x}_i,\mathbf{x}_j)-\gamma_{ij})^2 + \delta^2\epsilon_t^2\|\boldsymbol{\lambda}\|^2 \\ &\leq ML_h^2\|\mathbf{x}\|^2 + \delta^2\epsilon_t^2\|\boldsymbol{\lambda}\|^2 \leq ML_h^2R^2 + \delta^2\epsilon_t^2\|\boldsymbol{\lambda}\|^2. \qquad (21)\end{aligned}$$

The first inequality makes use of the triangle inequality and a worst-case bound on the constraint slack, whereas the second uses the Lipschitz continuity of the constraint (Assumption 2), and the last is an application of the compactness of the primal domain $\mathcal{X}^N$. We proceed with a remark.

Remark 2: Rather than bound the primal and dual gradients of the Lagrangian by constants, as is conventionally done in the analysis of primal-dual algorithms, we instead consider upper estimates in terms of the magnitude of the dual variable $\boldsymbol{\lambda}$. In doing so, we alleviate the need for the dual variable to be restricted to a compact subset of the nonnegative real numbers $\mathbb{R}_+^M$. The use of unbounded Lagrange multipliers allow us to mitigate the growth of constraint violation over time using the dual regularization term $(\delta\epsilon_t/2)\|\boldsymbol{\lambda}\|^2$ in (6).

The following lemma is used in the proof of the main theorem, and bounds the Lagrangian difference $\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}_t)$ by a telescopic quantity involving the primal and dual iterates, as well as the magnitude of the primal and dual gradients.

Lemma 1: Denote as $(\mathbf{x}_t,\boldsymbol{\lambda}_t)$ the sequence generated by the saddle point algorithm in (8) and (9) with stepsize ϵ_t. If Assumptions 1–4 hold, the instantaneous Lagrangian difference sequence $\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}_t)$ satisfies the decrement property

$$\begin{aligned}&\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}) - \hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}_t) \\ &\leq \frac{1}{2\epsilon_t}\Big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\Big) \\ &\quad + \frac{\epsilon_t}{2}\Big(\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2 + \|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2\Big). \qquad (22)\end{aligned}$$

Proof: See Appendix B ■

Lemma 1 exploits the fact that the stochastic augmented Lagrangian is convex-concave with respect to its primal and dual variables to obtain an upper bound for the difference $\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}_t)$ in terms of the difference between the primal and dual iterates to a fixed primal-dual pair $(\mathbf{x},\boldsymbol{\lambda})$ at the next and current time, as well as the square magnitudes of the primal and dual gradients. This property is the basis for establishing the convergence of the primal iterates to their constrained optimum given by (4) in terms of objective function evaluation and constraint violation, when a specific constant step-size is chosen, as we state next.

Theorem 1: Denote $(\mathbf{x}_t,\boldsymbol{\lambda}_t)$ as the sequence generated by the saddle point algorithm in (8), (9) and suppose Assumptions 1–4 hold. Suppose the algorithm is run for T iterations with a constant step-size selected as $\epsilon_t = \epsilon = 1/\sqrt{T}$, then the average time aggregation of the objective function error sequence $\mathbb{E}F(\mathbf{x}_t) - F(\mathbf{x}^*)$, with $\mathbf{x}^*$ defined as in (4), grows sublinearly with the final iteration index T as

$$\sum_{t=1}^{T}\mathbb{E}\left[F(\mathbf{x}_t) - F(\mathbf{x}^*)\right] \leq \mathcal{O}(\sqrt{T}). \qquad (23)$$

Moreover, the time-aggregation of the average constraint violation of the algorithm grows sublinearly in final time T as

$$\sum_{(i,j)\in\mathcal{E}}\mathbb{E}\left[\sum_{t=1}^{T}\Big(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})-\gamma_{ij}\Big)\right]_+ \leq \mathcal{O}(T^{3/4}). \qquad (24)$$

Proof: We first consider the expression in (22), and expand the left-hand side using the definition of the augmented Lagrangian in (7). Doing so yields the following expression,

$$\begin{aligned}&\sum_{i=1}^{N}[f_i(\mathbf{x}_{i,t},\boldsymbol{\theta}_{i,t}) - f_i(\mathbf{x}_i,\boldsymbol{\theta}_{i,t})] + \frac{\delta\epsilon_t}{2}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\ &\quad + \sum_{(i,j)\in\mathcal{E}}[\lambda_{ij}\left(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})\right. \\ &\quad \left.-\gamma_{ij}\right) - \lambda_{ij,t}\left(h_{ij}(\mathbf{x}_i,\mathbf{x}_j)-\gamma_{ij}\right)] \\ &\leq \frac{1}{2\epsilon_t}\Big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\Big) \\ &\quad + \frac{\epsilon_t}{2}\Big(\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2 + \|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2\Big), \qquad (25)\end{aligned}$$

after gathering like terms. Compute the expectation of (25) conditional on $\mathcal{F}_0$, the sigma algebra that measures the *entire* algorithm history, and substitute in the bounds for the mean-square-magnitude of the primal and dual gradients of the stochastic augmented Lagrangian given in (20) and (21), respectively, into the right-hand side to obtain

$$\begin{aligned}&\mathbb{E}\Bigg[F(\mathbf{x}_t) - F(\mathbf{x}) + \frac{\delta\epsilon_t}{2}\left(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2\right) \\ &\quad + \sum_{(i,j)\in\mathcal{E}}\left(\lambda_{ij}\left(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t})-\gamma_{ij}\right) - \lambda_{ij,t}\right. \\ &\quad \left.\left(h_{ij}(\mathbf{x}_i,\mathbf{x}_j)-\gamma_{ij}\right)\right)\Bigg] \\ &\leq \mathbb{E}\Bigg[\frac{1}{2\epsilon_t}\Big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 \\ &\quad - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\Big) \\ &\quad + \frac{\epsilon_t}{2}\Big((N+M)L^2(1+\|\boldsymbol{\lambda}_t\|^2) + ML_h^2R^2 + \delta^2\epsilon_t^2\|\boldsymbol{\lambda}_t\|^2\Big)\Bigg], \qquad (26)\end{aligned}$$

where we have also used the fact that the constraint functions $h_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ appearing as the third term on the left-hand side are independent of $\boldsymbol{\theta}$, and noting that the right-hand side of (26) is equal to its expectation. Observe that $\mathbf{x} \in \mathcal{X}$ is an arbitrary feasible point, which implies that $h_{ij}(\mathbf{x}_i, \mathbf{x}_j) \leq \gamma_{ij}$ for all $(i,j) \in \mathcal{E}$. Making use of this property to annihilate the last term on the left-hand side of (26) and subtracting $(\delta\epsilon_t/2)\|\boldsymbol{\lambda}_t\|^2$ from both sides yields

$$\begin{aligned}
&\mathbb{E}\Bigg[F(\mathbf{x}_t) - F(\mathbf{x}) + \sum_{(i,j)\in\mathcal{E}} \big(\lambda_{ij}\left(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij}\right) \\
&\quad - \frac{\delta\epsilon_t}{2}\lambda_{ij}^2\big)\Bigg] \\
&\leq \mathbb{E}\Bigg[\frac{1}{2\epsilon_t}\Big(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2 \\
&\quad - \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2\Big) \\
&\quad + \frac{\epsilon_t}{2}\Big(K + ((N+M)L^2 + \delta^2\epsilon_t^2 - \delta)\|\boldsymbol{\lambda}_t\|^2\Big)\Bigg]. \qquad (27)
\end{aligned}$$

after reordering terms, and defining the constant $K := (N + M)L^2 + ML_h^2R^2$. Now sum the expression (27) over times $t = 1, \ldots, T$ for a fixed T, and select the constant δ to satisfy $(N + M)L^2 + \delta^2\epsilon_t^2 \leq \delta$ for a constant step-size $\epsilon_t = \epsilon$ to drop the term involving $\|\boldsymbol{\lambda}_t\|^2$ from the right-hand side as

$$\begin{aligned}
&\mathbb{E}\Bigg[\sum_{t=1}^{T}[F(\mathbf{x}_t) - F(\mathbf{x})] + \sum_{(i,j)\in\mathcal{E}} \lambda_{ij}\left(\sum_{t=1}^{T}(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij})\right) \\
&\quad - \frac{\delta\epsilon T}{2}\|\boldsymbol{\lambda}\|^2\Bigg] \\
&\leq \frac{1}{2\epsilon}\Big(\|\mathbf{x}_1 - \mathbf{x}\|^2 + \|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}\|^2\Big) + \frac{\epsilon TK}{2}. \qquad (28)
\end{aligned}$$

In (28), we exploit the telescopic property of the summand over differences in the magnitude of primal and dual iterates to a fixed primal-dual pair $(\mathbf{x}, \boldsymbol{\lambda})$ which appears as the first term on right-hand side of (27), and the fact that the resulting expression is deterministic. By assuming the dual variable is initialized as $\boldsymbol{\lambda}_1 = \mathbf{0}$ and subtracting the resulting $(1/2\epsilon)\|\boldsymbol{\lambda}\|^2$ term to the other side, the expression in (28) becomes

$$\begin{aligned}
&\mathbb{E}\Bigg[\sum_{t=1}^{T}[F(\mathbf{x}_t) - F(\mathbf{x})] + \sum_{(i,j)\in\mathcal{E}} \lambda_{ij}\left(\sum_{t=1}^{T}(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij})\right) \\
&\quad - \left(\frac{\delta\epsilon T}{2} + \frac{1}{2\epsilon}\right)\|\boldsymbol{\lambda}\|^2\Bigg] \leq \frac{1}{2\epsilon}\|\mathbf{x}_1 - \mathbf{x}\|^2 + \frac{\epsilon TK}{2}. \qquad (29)
\end{aligned}$$

At this point, we note that the left-hand side of the expression in (29) consists of two terms. The first is the accumulation over time of the global loss, which is a sum of all local losses at each node as defined in (2). The second term is the inner product of the an arbitrary Lagrange multiplier $\boldsymbol{\lambda}$ with the time-aggregation of constraint violation, and the last is a term which depends on the magnitude of this multiplier. We may use these later terms to define an "optimal" Lagrange multiplier to control the growth of the long-term constraint violation of the algorithm. This technique is inspired by the approach in [30], [31]. To do so, define the *augmented* dual function $\tilde{g}(\boldsymbol{\lambda})$ using the later two terms on the left-hand side of (29)

$$\begin{aligned}
\tilde{g}(\boldsymbol{\lambda}) = \mathbb{E}\Bigg[&\sum_{(i,j)\in\mathcal{E}} \lambda_{ij}\left(\sum_{t=1}^{T}\Big(\left(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij}\right)\Big)\right) \\
&- \left(\frac{\delta\epsilon T}{2} + \frac{1}{2\epsilon}\right)\|\boldsymbol{\lambda}\|^2\Bigg]. \qquad (30)
\end{aligned}$$

Computing the gradient of (30) and solving the resulting stationary equation over the range $\mathbb{R}_+^M$ yields

$$\tilde{\lambda}_{ij} = \mathbb{E}\left[\left(\frac{1}{2(T\delta\epsilon + 1/\epsilon)}\right)\sum_{t=1}^{T}\left[h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij}\right]_+\right] \qquad (31)$$

for all $(i,j) \in \mathcal{E}$. Substituting the selection $\boldsymbol{\lambda} = \tilde{\boldsymbol{\lambda}}$ defined by (31) into (29) results in the following expression

$$\begin{aligned}
&\mathbb{E}\Bigg[\sum_{t=1}^{T}[F(\mathbf{x}_t) - F(\mathbf{x})] + \sum_{(i,j)\in\mathcal{E}} \\
&\quad \times \frac{\left[\sum_{t=1}^{T}\left(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij}\right)\right]_+^2}{2(T\delta\epsilon + 1/\epsilon)}\Bigg] \\
&\leq \frac{1}{2\epsilon}\|\mathbf{x}_1 - \mathbf{x}\|^2 + \frac{\epsilon TK}{2}. \qquad (32)
\end{aligned}$$

Now select the constant step-size $\epsilon = 1/\sqrt{T}$, and substitute the result into (32), using the formula for K defined following expression (27), to obtain

$$\begin{aligned}
&\mathbb{E}\Bigg[\sum_{t=1}^{T}[F(\mathbf{x}_t) - F(\mathbf{x})] + \sum_{(i,j)\in\mathcal{E}} \\
&\quad \times \frac{\left[\sum_{t=1}^{T}\left(h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij}\right)\right]_+^2}{2\sqrt{T}(\delta + 1)}\Bigg] \\
&\leq \frac{\sqrt{T}}{2}\left(\|\mathbf{x}_1 - \mathbf{x}\|^2 + (N+M)L^2 + ML_h^2R^2\right). \qquad (33)
\end{aligned}$$

The expression in (33) allows us to derive both the convergence of the global objective and the feasibility of the stochastic saddle point iterates.

We first consider the average objective error sequence $\mathbb{E}[F(\mathbf{x}_t) - F(\mathbf{x}^*)]$. To do so, subtract the last term on the left-hand side of (33) from both sides, and note that the resulting term is non-positive. This observation allows us to omit the constraint slack term in (33), which taken with the selection $\mathbf{x} = \mathbf{x}^*$ [cf. (4)] and pulling the expectation inside the summand, yields

$$\sum_{t=1}^{T}\mathbb{E}[(F(\mathbf{x}_t) - F(\mathbf{x}^*))] \leq \frac{\sqrt{T}}{2}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 + K\right) = \mathcal{O}(\sqrt{T}), \qquad (34)$$

which is as stated in (23).

Now we turn to establishing a sublinear growth of the constraint violation in T, using the expression in (33). First, observe that the objective function error sequence is bounded above as

$$F(\mathbf{x}_t) - F(\mathbf{x}^*) \leq L_f \|\mathbf{x}_t - \mathbf{x}^*\| \leq 2L_f R \tag{35}$$

An immediate implication of (35) is the relation $F(\mathbf{x}_t) - F(\mathbf{x}^*) \geq -2L_f R$, which may be obtained by switching the order, and again applying the Lipschitz continuity of F with the compactness of $\mathcal{X}^N$. Substituting this lower bound for the objective function error sequence into the first term on the left-hand side of (33) and adding the result to both sides yields

$$\mathbb{E}\left[\sum_{(i,j)\in\mathcal{E}} \frac{\left[\sum_{t=1}^T \Big(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij}\Big)\right]_+^2}{2\sqrt{T}(\delta+1)}\right]$$
$$\leq \frac{\sqrt{T}}{2}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 + K\right) + 2TL_f R. \tag{36}$$

which, after multiplying both sides by $2\sqrt{T}(\delta+1)$ yields

$$\mathbb{E}\left[\sum_{(i,j)\in\mathcal{E}} \left[\sum_{t=1}^T \Big(h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij}\Big)\right]_+^2\right]$$
$$\leq \Big(2\sqrt{T}(\delta+1)\Big)\left(\frac{\sqrt{T}}{2}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 + K\right) + 2TL_f R\right). \tag{37}$$

We complete the proof by noting that the square of the network-in-aggregate constraint violation $\sum_{(i,j)}[\sum_{t=1}^T (h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij})]_+^2$ upper bounds the square of individual proximity constraint violations since it is a sum of positive squared terms, i.e.,

$$\mathbb{E}\left[\sum_{(i,j)_\varepsilon}\left[\sum_{t=1}^T (h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij})\right]_+^2\right]$$
$$\geq \left[\left[\sum_{t=1}^T (h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij})\right]_+^2\right]. \tag{38}$$

Thus the right-hand side of (38) may be used in place of the left-hand side of (37), implying that

$$\mathbb{E}\left[\left[\sum_{t=1}^T (h_{ij}(\mathbf{x}_{i,t},\mathbf{x}_{j,t}) - \gamma_{ij})\right]_+^2\right]$$
$$\leq \Big(2\sqrt{T}(\delta+1)\Big)\left(\frac{\sqrt{T}}{2}\left(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 + K\right) + 2TL_f R\right). \tag{39}$$

Compute the square root of both sides of (39), and sum the resulting expression over all $(i,j) \in \mathcal{E}$ to conclude (24). ■

Theorem 1 establishes that the stochastic saddle point method, when run with a fixed algorithm step-size, yields an objective function error sequence whose difference is bounded by a constant strictly less times than T, the final iteration index. Moreover, the time-accumulation of the constraint violation incurred by the algorithm is strictly smaller than T, the final iteration index. Thus, for larger T, the iterate average difference between $F(\mathbf{x}_t)$ and $F(\mathbf{x}^*)$ goes to null in expectation, as does the average constraint violation in expectation.

This result is comparable to results for stochastic gradient method for unconstrained weakly convex problems with constant step-sizes with no smoothness assumptions, provided the data domain and feasible set are compact. In this setting, convergence to a neighborhood on the order of $T\epsilon$ is standard—see [32, Sec. 2.2, eq. (2.19)] or [33, Sec. 4], for instance. In such cases, convergence to a neighborhood of size $\mathcal{O}(\epsilon T)$ is attained in terms of primal sub-optimality for the time-average vector, and the step-size is chosen as $\epsilon = \mathcal{O}(1/\sqrt{T})$ to balance the growth of constant terms with the minimizing of neighborhood size. It must be noted, however, that the neighborhood for accumulation of constraint violation $\mathcal{O}(\epsilon T^{5/4})$ is larger than the primal sub-optimality, yielding the larger accumulation of constraint violation over T as $\mathcal{O}(T^{3/4})$ for this step-size choice. The reason we present results in this way is to draw the connection with regret analysis in online learning [34].

Theorem 1 also allows us to establish convergence of the average iterates to a specific level of accuracy dependent on the total number of iterations T, as we subsequently state.

Corollary 1: Let $\bar{\mathbf{x}}_T = (1/T)\sum_{t=1}^T \mathbf{x}_t$ be the vector formed by averaging the primal iterates $\mathbf{x}_t$ over times $t = 1,\ldots,T$. Under Assumptions 1 4, with constant algorithm step-size $\epsilon_t = 1/\sqrt{T}$, the objective function evaluated at $\bar{\mathbf{x}}_T$ satisfies

$$\mathbb{E}\left[F(\bar{\mathbf{x}}_T) - F(\mathbf{x}^*)\right] \leq \mathcal{O}(1/\sqrt{T}) \tag{40}$$

Moreover, the constraint violation evaluated at the average vector $\bar{\mathbf{x}}_T$ satisfies

$$\mathbb{E}\left[\sum_{(i,j)\in\mathcal{E}} \left[h_{ij}(\bar{\mathbf{x}}_{i,T},\bar{\mathbf{x}}_{j,T}) - \gamma_{ij}\right]_+\right] = \mathcal{O}(T^{-\frac{1}{4}}). \tag{41}$$

Proof: Consider the expressions in Theorem 1. In particular, to prove (40), we consider the expression in (23), divide the expression by T, and use the definition of convexity of the expected objective $\mathbb{E}[F(\mathbf{x})]$ which says that the average of average function values upper bounds the average function evaluated at the average vector, i.e.

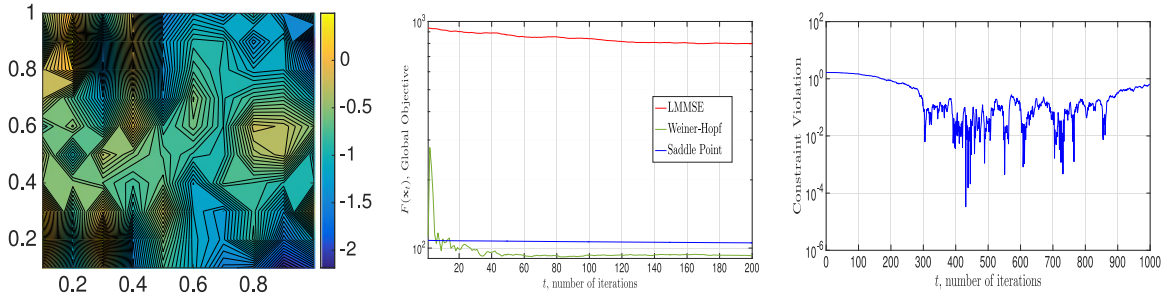
$$\mathbb{E}\left[F(\bar{\mathbf{x}}_T)\right] \leq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T F(\mathbf{x}_t)\right] \tag{42}$$

and similarly for the average of the expected constraint functions $\mathbb{E}[h_{ij}(\mathbf{x}_i,\mathbf{x}_i)]$,

$$\mathbb{E}[h_{ij}(\bar{\mathbf{x}}_{i,T},\bar{\mathbf{x}}_{j,T})] \leq \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^T h_{ij}(\bar{\mathbf{x}}_{i,t},\bar{\mathbf{x}}_{j,t})\right] \tag{43}$$

Apply the relation (42) to the expressions in (23) divided by T to obtain (40). To conclude, (41) we apply (43) to each term in the summand (24) divided by T. ■

Corollary 1 shows that the average saddle point primal iterates $\bar{\mathbf{x}}_T$ converge to within a margin $\mathcal{O}(1/\sqrt{T})$ in terms of objective function evaluation to the optimal objective $F(\mathbf{x}^*)$ on



(a) Snapshot of field across $\mathcal{A}$ (b) Global objective $F(\mathbf{x}_t)$ vs. iteration t (c) Constraint Violation vs. iteration t

Fig. 1. Saddle point algorithm applied to the problem of estimating a correlated random field. Nodes are deployed uniformly in a square region of size 200×200 meters in a grid formation (at the integer lattice within the Cartesian plane), and node estimators are correlated according to the distance-based model $\rho(\mathbf{x}_i, \mathbf{x}_j) = e^{-\|l_i - l_j\|}$, where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the decisions of nodes i and j, and l_j are their respective locations. A normalized snapshot of the field at time $t = 45$ is given in Fig. 1(a) – observe that nearby values are similar. The saddle point method achieves comparable accuracy to the Weiner-Hopf filter, and far outperforms a simple LMMSE estimator which ignores observation correlation. The saddle point method achieves this performance by satisfying proximity constraints that encode sensor correlations (Fig. 1(c)).

average, where T is the number of iterations. Moreover, the primal average vector also yields the bound in expectation on the network proximity constraint violation as $\mathcal{O}(T^{-1/4})$. We note that for a fixed T, this result amounts to convergence to a neighborhood on average. The radius of this neighborhood crucially depends on using the expressions in (20) and (21), which are the variances of the primal and dual gradients of the stochastic augmented Lagrangian (7), respectively.

After cancelling out key terms in the proof, the remaining constant $(\|\mathbf{x}_1 - \mathbf{x}^*\|^2 + (N+M)L^2 + ML_h^2R^2)$ on the right-hand side of (34) determines the radius of convergence, for a fixed T, where we have substituted in the definition of $K = (N+M)L^2 + ML_h^2R^2$. This expression depends on initialization $\mathbf{x}_1$, the size of the network, the Lipschitz modulus of continuity of the objective and constraints (Assumption 2), and the diameter of set $\mathcal{X}$ (16). Similarly, for the constraint violation, the constant in front terms involving T on the right-hand side of (39) depend on the initialization, the dual regularization constant δ, $K = (N+M)L^2 + ML_h^2R^2$, as discussed above, the objective Lipschitz constant L_f, and the diameter R of set $\mathcal{X}$.

The convergence of recursively averaged saddle point iterates with constant step-size has appeared in the deterministic setting in [17] and in the context of regret for online learning in [30]. Theorem 1 and Corollary 1 are the first attempts at translating this type of result into the constrained stochastic programming case with weakly convex objectives. In doing so, we attain comparable rates to stochastic gradient method for the weakly convex unconstrained stochastic case [32], [33] for the primal sub-optimality, but slower rates for the reduction of constraint violation.

V. Random Field Estimation

Consider the task of estimating a planar spatially correlated Gaussian random field in a specified region $\mathcal{A} \subset \mathbb{R}^2$. A planar random field is a random function of spatial components u and v, which index the value of the field across region $\mathcal{A}$ (u for x-axis, v for y-axis). The random field is further parameterized by the correlation matrix $\mathbf{R}_x$, which is assumed to follow a spatial correlation structure of the form $\rho(\mathbf{x}_i, \mathbf{x}_j) = e^{-\|l_i - l_j\|}$, where $l_i \in \mathcal{A}$ and $l_j \in \mathcal{A}$ are the respective locations of sensor i and sensor j in the deployed region, see, e.g., [35]. Observe that now each node has a unique signal-to-noise ratio based upon its location and that more distant nodes are less important; however, their contribution to the aggregate objective $F(\mathbf{x})$ still incentivizes global coordination.

We consider making use of a sensor network (the example in Section II and III). Sensors collect observations $\boldsymbol{\theta}_{i,t}$ which are noisy linear transformations of the value of the field $\mathbf{x}_{u,v}(t) \in \mathbb{R}^p$ they would like to estimate at time t. That is, we consider the observation model $\boldsymbol{\theta}_{it} = \mathbf{H}_i \mathbf{x}_{u,v}(t) + \mathbf{w}_{i,t}$ with Gaussian noise $\mathbf{w}_{i,t} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_q)$ that is i.i.d across time and node, with $\sigma^2 = 2$. The goal is for each sensor to sequentially minimize its local estimation error, which amounts to online maximum likelihood estimation where the estimators of distinct sensors depend on one another.

To solve this problem, we deploy $N = 100$ sensors in a grid formation along the scaled positive integer lattice, where neighboring nodes have a constant distance from one another in a 200×200 meter square region $\mathcal{A} = \{(x, y) : 200 \geq x \geq 0, 200 \geq y \geq 0\}$. At each instantaneous time, then, the observations across the network are given by $\mathbf{x}_t = \boldsymbol{\mu} + \mathbf{C}^T \mathbf{z}_t$, where $\boldsymbol{\mu}$ is a fixed mean vector of length N chosen uniformly at random from the fractions $\{1/N, 2/N, \ldots, 1\}$, $\mathbf{C}$ denotes the Cholesky factorization of the correlation matrix $\mathbf{R}_x$, and $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ is a Gaussian random vector – see, for instance, [36]. An example instance of the field values observed by the deployed grid network (rescaled within the unit box) are displayed in Fig. 1(a). Observe that nearby values are similar to each other.

We make use of the saddle point algorithm [cf. (10), (11)], whose updates for the random field estimation problem are given by the explicit expressions in (12) and (13), respectively. We select $\gamma_{ij} = \rho(\mathbf{x}_i, \mathbf{x}_j)$. Besides the local and global losses which on average converge to a neighborhood of the constrained optima depending on the final iteration index T when a constant step-size is used (Theorem 1), we also study the amount of constraint violation over time, stated as $\sum_{j \in n_i} (\|\mathbf{x}_{i,t} - \mathbf{x}_{j,t}\|^2 - \gamma_{ij})$.

To compute $\mathbf{x}^*$ for a single time slot, stack observations $\boldsymbol{\theta} = [\boldsymbol{\theta}_1; \cdots; \boldsymbol{\theta}_N]$ and observation models $\mathbf{H} = [\mathbf{H}_1; \cdots; \mathbf{H}_N]$. Then the optimal estimator is the one that solves the weighted least-squares estimate derived from the Weiner-Hopf equations $\mathbf{x}^* = (\mathbf{H}\mathbf{R}_x\mathbf{H}^T + \frac{1}{\sigma^2}\mathbf{I})^{-1}(\mathbf{H} + \frac{1}{\sigma^2}\mathbf{I})\mathbf{R}_x\boldsymbol{\theta}$. The optimal estimator $\mathbf{x}^*$ is the one that would dictate stacking signals $\boldsymbol{\theta}_{i,t}$ for all nodes i and times t at a centralized location into one large linear system and substituting the sample variance $\hat{\sigma}^2$ in the prior computation. We consider an incremental variant of such a strategy, similar to the Levinson-Durbin recursion [37].

We consider problem instances where observations and signal estimates are scalar ($p = q = 1$), the scalar $\mathbf{H} = 1$, and the field is set a vector of ones, and run the algorithm for for $T = 1000$ iterations with a constant step-size strategy $\epsilon_t = 10^{-2.75}$. We further select the dual regularization parameter $\delta = 10^{-5}$. The noise level is set to $\sigma^2 = 10$. We compare the performance of the algorithm with that of a simple LMMSE estimator strategy which does not take advantage of the correlation structure of the sensor network, as well as the sequential implementation of a Weiner-Hopf estimator which optimally exploits correlation.

In Fig. 1, we plot the results of this numerical experiment. Fig. 1(b) shows the global objective $(1/N)\sum_i \mathbb{E}_{\boldsymbol{\theta}_i}[f_i(\mathbf{x}_{i,t}, \boldsymbol{\theta}_i)]$ versus iteration t. We note that the numerical behavior of the local objective is similar to the global objective, and is thus omitted. We see that when nodes incorporate the correlation structure of the random field into their estimation strategy via the quadratic proximity constraint with γ_{ij} chosen according to the correlation of node i and its neighbors $j \in n_i$, the estimation performance improves. We observe that for small t, the saddle point method outperforms the Weiner-Hopf estimator in terms of estimation accuracy, but after a burn in period the later performs more favorably. In contrast, the LMMSE estimator which ignores correlation does not appear to yield an effective tool for this context.

In Fig. 1(c), we plot the local constraint violation of an arbitrarily chosen sensor $i \in V$, and observe that the algorithm successsfully keeps the estimates of node i close to those of its neighbors, where the closeness constraint is given by the correlation structure of the random field. Thus by using proximity constraints, individual sensors are successfully able to incorporate spatial information about the random field into their estimation.

VI. Source Localization

We now consider the use of the stochastic saddle point method given in (8) (9) to solve an online source localization problem. In particular, we consider an array of N sensors, where $\mathbf{l}_i \in \mathbb{R}^p$ denotes the position of the sensor i in some deployed environment $\mathcal{A} \subset \mathbb{R}^p$. Each node seeks to learn the location of a source signal $\mathbf{x} \in \mathbb{R}^p$ through its access to noisy range observations of the form $r_{i,t} = \|\mathbf{x} - \mathbf{l}_i\| + \varepsilon_{i,t}$ where $\varepsilon_t = [\varepsilon_{1,t}; \cdots; \varepsilon_{N,t}]$ is some unknown noise vector. The goal of each sensor i in the network is, given access to sequentially observed range measurements $r_{i,t}$, to learn the position of the source $\mathbf{x}$, assuming it is aware of its location $\mathbf{l}_i$ in the deployed region. Range-based source localization has been studied in a variety of fields, from wireless communications to geophysics [38], [39].

Rather than considering a range-based least squares problem, which is nonconvex and may be solved approximately using semidefinite relaxations [40], we consider the squared range-based least squares (SR-LS) problem, stated as

$$\mathbf{x}^* := \operatorname*{arg\,min}_{\mathbf{x}\in\mathbb{R}^p} \sum_{i=1}^{N} \mathbb{E}_{\mathbf{r}_i}\left(\|\mathbf{l}_i - \mathbf{x}\|^2 - r_i^2\right)^2. \tag{44}$$

Although this problem is also nonconvex (due to, for instance, the fact that when the outer square is expanded, a quartic term appears), it may be solved approximately in a lower-complexity manner as a quadratic program – see [41], Section II-B and references therein. To do so, expand the square in the first term in the objective stated in (44) and consider the modified argument inside the expectation $(\alpha - 2\mathbf{l}_i^T\mathbf{x} + \|\mathbf{l}_i\|^2 - r_i^2)^2$ with the constraint $\|\mathbf{x}\|^2 = \alpha$. Proceeding as in [41], Section II-B, approximate this transformation by a convex unconstrained problem by defining matrix $\mathbf{A} \in \mathbb{R}^{N\times(p+1)}$ whose ith row associated with sensor i is given as $\mathbf{A}_i = [-2\mathbf{l}_i^T; 1]$, and vector $\mathbf{b} \in \mathbb{R}^N$ with ith entry as $\mathbf{b}_i = r_i^2 - \|\mathbf{l}_i\|^2$ and relaxing the constraint $\|\mathbf{x}\|^2 = \alpha$. Further define $\mathbf{y} = [\mathbf{x}; \alpha] \in \mathbb{R}^{p+1}$. Then, by dropping a quadratic equality constraint induced by this change of variables, (44) may be approximated as

$$\mathbf{y}^* := \operatorname*{arg\,min}_{\mathbf{y}\in\mathbb{R}^{p+1}} \sum_{i=1}^{N} \mathbb{E}_{\mathbf{b}_i}\left(\|\mathbf{A}_i\mathbf{y} - \mathbf{b}_i\|^2\right); \tag{45}$$

which is a least mean-square error problem. We note that the techniques in [41] to solve this problem exactly do not apply to the online setting [42].

We propose solving (45) in decentralized settings which more effectively allow for each sensor to operate based on real-time observations. To do so, each sensor keeps a local copy $\mathbf{y}_i$ of the global source estimate $\mathbf{y}$ based on information that is available with local information only and via message exchange with neighboring sensors. However, each sensor would still like to attain the greater estimation accuracy associated with aggregating range observations over the entire network. We proceed to illustrate how this may be achieved by using the proximity constrained optimization in Section II.

In application domains such as wireless communications or acoustics [3], the quality of the observed range measurements is better for sensors which are in closer proximity to the source. Motivated by this fact, we consider the case where sensor i weights the importance of neighboring sensors $j \in n_i$ by aiming to keep its estimate $\mathbf{x}_i$ within an ℓ_2 ball centered at its neighbors estimate $\mathbf{x}_j$, whose radius is given by the pairwise minimum of the estimated distance to the source. This goal may be achieved via the quadratic inequality constraint

$$\|\mathbf{x}_i - \mathbf{x}_j\|^2 \leq \min\{\|\mathbf{x}_i - \mathbf{l}_i\|^2, \|\mathbf{x}_j - \mathbf{l}_j\|^2\} \text{ for all } j \in n_i. \tag{46}$$

Observe that (45) with the constraint (46) is a non-convex variant of a QCQP due to the minimum on the right-hand side (see, for instance, [43]). We may convexify the constraint by rearranging the right-hand side of (46) and replacing the resulting maximum

by the log-sum-exp function – see [44, Ch. 2]. Thus we obtain

$$(1/2)\Big(\|\mathbf{x}_i - \mathbf{x}_j\|^2 + \log\Big(e^{\|\mathbf{x}_i - \mathbf{l}_i\|^2} + e^{\|\mathbf{x}_j - \mathbf{l}_j\|^2}\Big)\Big) \leq 0\,, \tag{47}$$

which is a convex constraint, since the later term is a composition of a monotone function with a convex function. Taking (45) together with the constraint (47), and noting that a constraint on $\mathbf{x}_i$ is equivalent to a constraint on the first p entries of $\mathbf{y}_i$ after appending a 0 to the $p+1$-th entry of $\mathbf{l}_i$, we may write

$$\begin{aligned}&\min_{\mathbf{y}\in\mathbb{R}^{N(p+1)}} \sum_{i=1}^{N} \mathbb{E}_{\mathbf{b}_i}\Big(\|\mathbf{A}_i\mathbf{y}_i - \mathbf{b}_i\|^2\Big),\\ &\text{s.t.}\quad (1/2)\Big(\|\mathbf{y}_i - \mathbf{y}_j\|^2 + \log\Big(e^{\|\mathbf{y}_i - \mathbf{l}_i\|^2} + e^{\|\mathbf{y}_j - \mathbf{l}_j\|^2}\Big)\Big) \leq 0,\end{aligned} \tag{48}$$

where the constraint for node i is with respect to all of its neighbors $j \in n_i$. Observe that the problem in (48) is of the form (4). Define $g(\mathbf{y}_i, \mathbf{y}_j)$ as the constraint function the left-hand side of (47). Then primal update of the saddle point method stated in (8) specialized to this problem setting for node i is stated as

$$\begin{aligned}\mathbf{y}_{i,t+1} = \mathbf{y}_{i,t} - \epsilon_t\Big(&2\mathbf{A}_{i,t}^T\big(\mathbf{A}_{i,t}\mathbf{y}_{i,t} - \mathbf{b}_{i,t}\big)\\ &+ \sum_{j\in n_i}\boldsymbol{\lambda}_{ij,t}\Big(\frac{e^{\|\mathbf{y}_{i,t}-\mathbf{l}_i\|^2}(\mathbf{y}_{i,t}-\mathbf{l}_i)}{e^{\|\mathbf{y}_{i,t}-\mathbf{l}_i\|^2} + e^{\|\mathbf{y}_{j,t}-\mathbf{l}_j\|^2}} + (\mathbf{y}_{i,t} - \mathbf{y}_{j,t})\Big),\end{aligned} \tag{49}$$

where we omit the use of set projections for simplicity, while the dual update [cf. (9)] executed at the link layer of the sensor network is

$$\boldsymbol{\lambda}_{ij,t+1} = \big[(1-\delta\epsilon_t^2)\boldsymbol{\lambda}_{ij,t} + \epsilon_t g(\mathbf{y}_{i,t}, \mathbf{y}_{j,t})\big]_+\,. \tag{50}$$

We turn to analyzing the empirical the performance of the saddle point updates (49) (50) to solve localization problems in a decentralized manner, such that nodes more strongly weight the importance of sensors in closer proximity to the source in the sense of (47). Besides the local objective $\mathbb{E}_{\mathbf{b}_i}\|\mathbf{A}_i\mathbf{y}_i - \mathbf{b}_i\|^2$, which we know converges to its contained optimal value, we also study the standard error to the source signal $\mathbf{x}^*$, denoted as $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$. Recall that we recover $\mathbf{x}_{i,t}$ from $\mathbf{y}_{i,t}$ by taking its first p elements. We further consider the magnitude of the constraint violation for this problem, which when considering the proximity constrained problem in (48), is given by

$$\begin{aligned}\sum_{j\in n_i}(1/2)g(\mathbf{y}_{i,t}, \mathbf{y}_{j,t}) = \sum_{j\in n_i}(1/2)\Big(&\|\mathbf{y}_{i,t} - \mathbf{y}_{j,t}\|^2\\ &+ \log\Big(e^{\|\mathbf{y}_{i,t}-\mathbf{l}_{i,t}\|^2} + e^{\|\mathbf{y}_{j,t}-\mathbf{l}_j\|^2}\Big)\Big)\,,\end{aligned} \tag{51}$$

and when implementing consensus methods, is given by

$$\sum_{j\in n_i} h(\mathbf{y}_{i,t}, \mathbf{y}_{j,t}) = \sum_{j\in n_i}\|\mathbf{y}_{i,t} - \mathbf{y}_{j,t}\| \tag{52}$$

for a randomly chosen sensor in the network.

Throughout the rest of this section, we fix the dual regularization parameter $\delta = 10^{-7}$, and study the performance of the saddle point method with proximity constraints as compared with two methods which attempt to satisfy consensus constraints. We further analyze the saddle point method in (49), (50) for a variety of network sizes to understand the practical effect of the learning rate on the number of sensors, and for different spatial deployment strategies which induce different network topologies.

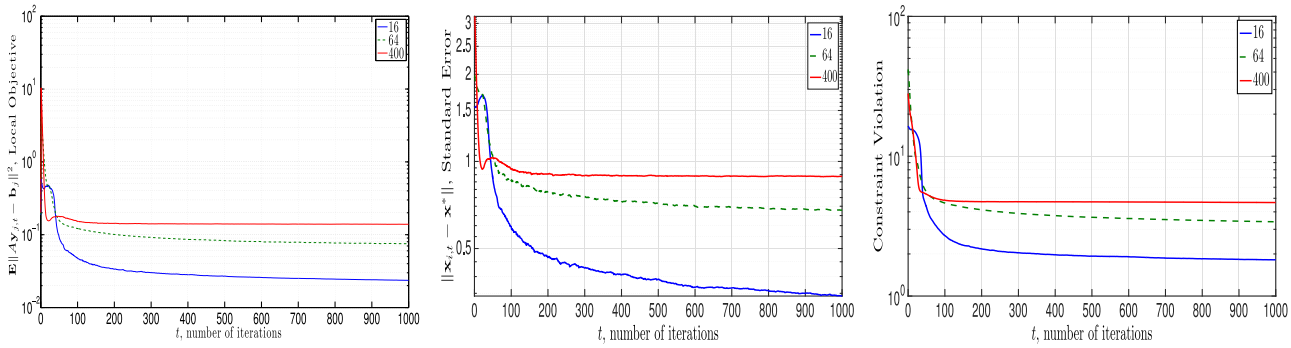
A. Consensus Comparison

In this subsection, we compare the saddle point method on a proximity constrained problem as compared with methods which implement variations of the consensus protocol. In particular, we run the saddle point method for the localization problem given in (49) (50) with proximity constraints, as compared with the same primal-dual scheme when the consensus constraint in (3) is used. We further compare these instantiations of the saddle point method with distributed online gradient descent (DOGD) [45], which is a scheme that operates by having each node selecting its next iterate by taking a weighted average of its neighbors and descending through the negative of the local stochastic gradient. For each of these methods, we run the localization procedure for a total of 1000 iterations for $\tilde{T} = 100$ different runs when each node initializes its local variable $\mathbf{y}_{i,0}$ uniformly at random from the unit interval, and plot the sample mean of the results.

We consider problem instances of (44) when the number of sensors is fixed at $N = 64$, and are spatially deployed in a grid formation as an 8×8 square in a planar ($p = 2$) region of size 1000×1000. Moreover, the noise perturbing the observations at node i is zero-mean Gaussianan, with a variance proportional its distance to the source as $\sigma^2 = 2\|\mathbf{l}_i - \mathbf{x}^*\|$, where $\mathbf{l}_i$ is the location of node i. The true source signal $\mathbf{x}^*$ is located at the average location of the sensors. For the saddle point methods, we find a hybrid step-size strategy to be most effective, and hence set $\epsilon_t = \min(\epsilon, \epsilon t_0/t)$ with $t_0 = 100$ and $\epsilon = 10^{-1.5}$. For DOGD, we find best performance to correspond to using a constant outer step-size $\epsilon = 10^{-1.5}$, along with a halving scheme step-size in the inner recursive averaging loop [45].

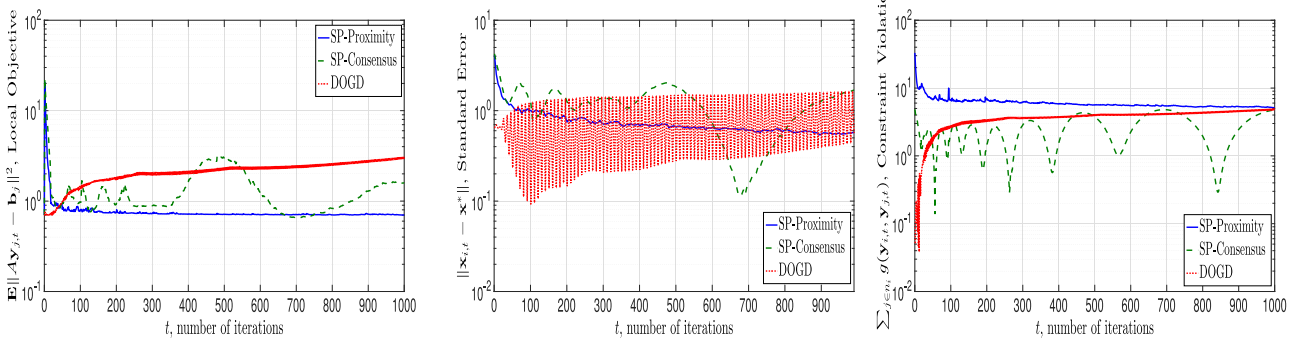
We plot the results Fig. 3 for an arbitrarily chosen node $i \in V$. Observe that the saddle point method which implements the network proximity constraints method yields the best performance in terms of objective convergence. In particular, by $t = 500$ iterations, in Fig. 3(a) we observe that the saddle point algorithm implemented with proximity constraints (SP-Proximity) achieves objective convergence to a neighborhood, i.e., $\mathbb{E}_{\mathbf{b}_i}\|\mathbf{A}_i\mathbf{y}_{i,t} - \mathbf{b}_i\|^2 \leq 1$. In contrast, the saddle point with consensus constraints (SP-Consensus) and DOGD respectively experience numerical oscillations and divergent behavior after a burn-in period of $t = 100$.

This trend is confirmed in the plot of the standard error to the optimizer $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ of the original problem (44) in Fig. 3(b). We see that SP-Proximity yields convergence to a neighborhood between 10^{-1} and 1 by $t = 200$ iterations, whereas SP-Consensus and DOGD experience numerical oscillations and do not appear to localize the source signal $\mathbf{x}^*$. While SP-Proximity exhibits superior behavior in terms of objective and standard error convergence, it incurs larger levels of constraint



(a) Local objective vs. iteration t (b) Standard error $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ vs. iteration t (c) Constraint violation vs. iteration t

Fig. 2. Comparison of the saddle point method with proximity constraints [cf. (49) (50)] with dual regularization $\delta = 10^{-7}$ and hybrid step-size strategy $\epsilon_t = \min(\epsilon, \epsilon t_0/t)$ with $t_0 = 100$ and $\epsilon = 10^{-1.5}$ on the source localization problem stated in (44) using the convex approximation (45). We fix the network topology as a grid and vary the number of sensors as $N = 16$, $N = 64$, and $N = 400$ which are deployed in a square region of size 1000×1000 meters. The noise perturbing observations at sensor i is zero-mean Gaussian, with a variance proportional its distance to the source as $\sigma^2 = 0.5\|\mathbf{l}_i - \mathbf{x}^*\|$, where $\mathbf{l}_i$ is the location of node i. Observe that in larger networks, the rate at which nodes are able to localize the source is slower in terms of objective convergence and standard error to the true source. Moreover, we see that the level of constraint violation is larger with increasing N.



(a) Local objective vs. iteration t (b) Standard error $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ vs. iteration t (c) Constraint violation vs. iteration t

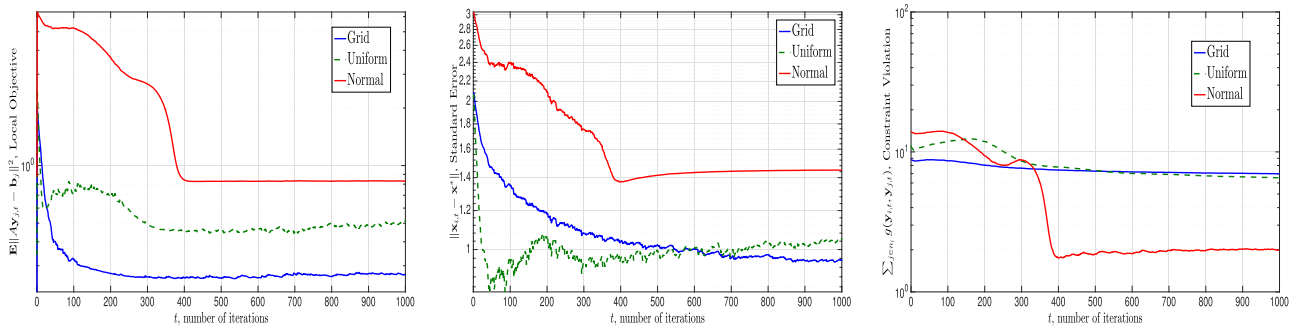
Fig. 3. Comparison of proximity and consensus algorithms on the source localization problem stated in (44) using the convex approximation (45) for an $N = 64$ node grid network deployed as an 8×8 square in a 1000×1000 meter region for the case that the noise perturbing observations at node i is zero-mean Gaussian, with a variance proportional its distance to the source as $\sigma^2 = 2\|\mathbf{l}_i - \mathbf{x}^*\|$, where $\mathbf{l}_i$ is the location of node i. We run the saddle point method with proximity constraints [cf (49) (50)] given in (48) using dual regularization $\delta = 10^{-7}$, as compared with the saddle point method which executes a consensus constraint (3), as well as Distributed Online Gradient Descent (DOGD) [45], which is a weighted averaging gradient method. For the former two, we use hybrid step-size strategy $\epsilon_t = \min(\epsilon, \epsilon t_0/t)$ with $t_0 = 100$ and $\epsilon = 10^{-1.5}$, and for DOGD we use constant step-size $10^{-1.5}$. We observe that the proximity-constrained saddle point method yields the best performance in terms of objective convergence and standard error, although it incurs higher levels of constraint violation.

violation than its consensus counterpoints, as may be observed in Fig. 3(c). To be specific, SP-Proximity on average experiences constraint violation [cf. (51)] on average an order of magnitude larger than SP-Consensus and DOGD [cf. (52)] for the first $t = 400$ iterations. After this benchmark, the magnitude of the constraint of the different methods converges to around 5. Thus, we see that achieving smaller constraint violation and implementing consensus constraints may lead to inferior source localization accuracy.

B. Impact of Network Size

In this subsection, we study the effect of the size of the deployed sensor network on the ability of the proximity-constrained saddle point method to effectively localize the source signal. We fix the topology of the deployed sensors as a grid network, and again set the source signal $\mathbf{x}^*$ to be the average of node positions in a planar ($p = 2$) spatial region $\mathcal{A}$ of size 1000×1000 meters. We set the noise distribution which perturbs the range measurements of node i to be zero-mean Gaussian with variance $.5\|\mathbf{l}_i - \mathbf{x}^*\|$. We run the algorithm stated in (49), (50) with hybrid step-size strategy $\epsilon_t = \min(\epsilon, \epsilon t_0/t)$ with $t_0 = 100$ and $\epsilon = 10^{-1.5}$ for a total of $T = 1000$ iterations for $\tilde{T} = 100$ total runs where each node initializes its local variable $\mathbf{y}_{i,0}$ uniformly at random from the unit interval, and plot the sample mean results for problem instances of (48) when the network size is varied as $N = 16$, $N = 64$, $N = 400$, which correspond respectively to 4×4, 8×8, and 20×20 grid sensor formations.

We plot the results of this numerical setup in Fig. 2 for a randomly chosen sensor in the network. Observe that in Fig. 2(a), which shows the convergence behavior in terms of the local objective $\mathbb{E}_{\mathbf{b}_i}\|\mathbf{A}_i\mathbf{y}_{i,t} - \mathbf{b}_i\|^2$ versus iteration t, that the rate at which sensors are able to localize the source is comparable across the different network sizes; however, the convergence accuracy is higher in smaller networks. In particular, by $t = 1000$, the objective converges to respective values 0.03, 0.08, and 0.14 for the $N = 16$, $N = 64$, $N = 400$ node networks. This



(a) Local objective vs. iteration t (b) Standard error $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ vs. iteration t (c) Constraint violation vs. iteration t

Fig. 4. Numerical results on the localization problem stated in (44) using the convex approximation (45) for the saddle point method with proximity constraints [cf. (49) (50)] with hybrid step-size strategy $\epsilon_t = \min(\epsilon, \epsilon t_0/t)$ with $t_0 = 100$ and $\epsilon = 10^{-1.5}$. We run the algorithm for a variety of spatial deployment strategies, which induce different network topologies. We consider a square region of size 1000×1000 meters, and deploy nodes in grid formations, uniformly at random, and according to a two-dimensional Normal distribution. In the later two cases, sensors which are a distance of 50 meters or less are connected by an edge. The noise perturbing observations at sensor i is zero-mean Gaussian, with a variance proportional to its distance to the source as $\sigma^2 = 0.5\|\mathbf{l}_i - \mathbf{x}^*\|$, where $\mathbf{l}_i$ is the location of node i. We see that while the Normal configuration yields the worst localization performance, it achieves the lowest levels of constraint violation. In contrast, Uniform and Grid configurations both are effective spatial deployment strategies to localize the source in terms of local objective convergence and standard error.

relationship between convergence accuracy and number of sensors in the network is corroborated in the plot of the standard error $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ to the true source location $\mathbf{x}^*$ in Fig. 2(b). We see that the standard error across the different networks converges to within a radius of 1 to the optimum, but the rate at which convergence is exhibited is comparable across the different network sizes. In particular, by $t = 400$, we observe the standard error benchmarks 0.41, 0.74, and 0.9 for the $N = 16$, $N = 64$, and $N = 400$ node networks.

A similar pattern may be gleaned from Fig. 2(c), in which we plot the magnitude of the constraint violation $\sum_{j \in n_i} g(\mathbf{y}_{i,t}, \mathbf{y}_{j,t})$ as given in (51) with iteration t. Observe that for the networks with $N = 16$, $N = 64$, and $N = 400$ sensors, respectively, we have the constraint violation benchmarks 2.1, 4, and 4.74 by $t = 300$. Moreover, the rate at which benchmarks are achieved is comparable across the different network sizes, such that the primary difference in the dual domain is the asymptotic magnitude of constraint violation, but not dual variable convergence rate.

C. Effect of Spatial Deployment

We turn to studying the impact of the way in which sensors are spatially deployed on their ability to localize the source signal, which implicitly is an analysis of the impact of the network topology on the empirical convergence behavior. To do so, we consider a problem instance in which the source signal $\mathbf{x}^*$ is located at the average of sensor positions in the network in a planar ($p = 2$) spatial region $\mathcal{A}$ of size 1000×1000 meters. The noise distribution which perturbs the range measurements received at node i is fixed as zero-mean Gaussian with variance $.5\|\mathbf{l}_i - \mathbf{x}^*\|$, implying that nodes which are closer to the source receive observations with higher SNR. Each node initializes its local variable $\mathbf{y}_{i,0}$ uniformly at random from the unit interval, and then executes the saddle point method stated in (49) (50) for a total of $T = 1000$ iterations for $\tilde{T} = 100$ total runs. We consider the sample mean results of the $\tilde{T} = 100$ for problem instances of (48) when the sensor deployment strategy is either a grid formation, uniformly at random, or according to a two-dimensional Gaussian distribution. In the later two cases, sensors which are closer than a distance of 50 meters are connected. Since in general random networks of these types will not be connected, we repeatedly generate such networks until we obtain the first one which has the a comparable Fiedler number (second-smallest eigenvalue of the graph Laplacian matrix) as the grid network, which is a standard measure of network connectivity – see [46, Ch. 1].

We display the results of this localization experiment in Fig. 4. In Fig. 4 (a), we plot the local objective as compared with iteration t across these different sensor deployment strategies. We see that sensor localization performance is best in terms of objective convergence in the grid network, followed by network topologies generated from uniform and Normal spatial deployment strategies. In particular, by $t = 400$, the grid, Uniform, and Normal sensor networks achieve the objective ($\mathbb{E}_{\mathbf{b}_i} \|\mathbf{A}_i \mathbf{y}_{i,t} - \mathbf{b}_i\|^2$) benchmarks 0.26, 0.45, and 0.83. This trend is not corroborated by our analysis of these sensor networks' ability to learn the true source $\mathbf{x}^*$ as measured by the standard error $\|\mathbf{x}_{i,t} - \mathbf{x}^*\|$ (Fig. 4 b). In particular, to achieve the benchmark $\|\mathbf{x}_{i,t} - \mathbf{x}^*\| \leq 1$ we see that the Uniform topology requires $t = 26$ iterations, whereas the grid network requires $t = 557$ iterations, and the Normal network does not achieve the error bound by $t = 1000$. However, we observe that the grid network experiences more stable convergence behavior in terms of its error sequence, as compared with the other two networks.

In Fig. 4(c), we display the constraint violation [cf. (51)] incurred by the proximity-constrained saddle point method when we vary the sensor deployment strategy. Observe that the grid and Uniform network topologies incur comparable levels of constraint violation, whereas the sensor network induced by choosing spatial locations according to a two-dimensional Gaussian distribution is able to maintain closer levels of network proximity by nearly an order of magnitude.

VII. Conclusion

We considered multi-agent stochastic optimization problems where the hypothesis that all agents are trying to learn common parameters may be violated. In doing so, agents to make decisions which give preference to locally observed information while incorporating the relevant information of others. This problem class incorporates sequential estimation problems in multi-agent settings where observations are independent but *not identically* distributed. We formulated this task as a decentralized stochastic program with convex proximity constraints which incentivize distinct nodes to make decisions which are close to one another. We considered an augmented Lagrangian relaxation of the problem, to which we apply a stochastic variant of the saddle point method of Arrow and Hurwicz to solve it. We established that under a constant step-size regime the time-average suboptimality and constraint violation are contained in a neighborhood whose radius vanishes with increasing number of iterations (Theorem 1). As a consequence, we obtain in Corollary 1 that the average primal vectors converge to the optimum while satisfying the network proximity constraints.

Numerical analysis on a random field estimation problem in a sensor network illustrated the benefits of using the saddle point method with proximity constraints as compared with a simple LMMSE estimator scheme. We find that these benefits are more pronounced in problem instances with lower SNR and larger spatial regions in which sensors are deployed, e.g., instances where the correlation structure of the information across the network plays a larger role. We further considered a source localization problem in a sensor network, where sensors collect noisy range estimates whose SNR is proportional to their distance to the true source signal. In this problem setting, the proximity-constrained saddle point method outperforms methods which attempt to execute consensus constraints.

Appendix A
Proof of Proposition 1

To compute the primal stochastic gradient of the Lagrangian in (6), observe that in the instantaneous Lagrangian in (7) only a few summands depend on $\mathbf{x}_i$. In the first sum only the one associated with the local objective $f_i(\mathbf{x}_i, \boldsymbol{\theta}_{i,t})$ depends on $\mathbf{x}_i$. In the second sum the terms that depend on $\mathbf{x}_i$ include the local constraints $h_i(\mathbf{x}_i, \mathbf{x}_j) - \gamma_{ij}$ and the neighboring constraints $h_j(\mathbf{x}_j, \mathbf{x}_i) - \gamma_{ji}$. Taking gradients of these terms yields,

$$\nabla_{\mathbf{x}_i} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) = \nabla_{\mathbf{x}_i} f_i(\mathbf{x}_{i,t}; \boldsymbol{\theta}_{i,t}) + \sum_{j \in n_i} (\boldsymbol{\lambda}_{ij,t} + \boldsymbol{\lambda}_{ji,t})^T \nabla_{\mathbf{x}_i} h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}). \tag{53}$$

Writing (8) componentwise and substituting $\nabla_{\mathbf{x}_i} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)$ for its expression in (53), the result in (10) follows.

To prove (11) we just need to compute the gradient $\hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)$ of the stochastic Lagrangian with respect to the Lagrange multipliers associated with edge (i, j). By noting that only one summand in (7) depends on this multiplier we conclude that

$$\nabla_{\lambda_{ij}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) = h_{ij}(\mathbf{x}_{i,t}, \mathbf{x}_{j,t}) - \gamma_{ij} - \epsilon_t \delta\lambda_{ij,t} \ . \tag{54}$$

After gathering terms in (54) and substituting the result into (9), we obtain (11). ■

Appendix B
Proof of Lemma 1

Consider the squared 2-norm of the difference between the iterate $\mathbf{x}_{t+1}$ at time $t+1$ and an arbitrary feasible point $\mathbf{x} \in \mathcal{X}^N$ and use (8) to express $\mathbf{x}_{t+1}$ in terms of $\mathbf{x}_t$,

$$\|\mathbf{x}_{t+1} - \mathbf{x}\|^2 = \|\mathcal{P}_{\mathcal{X}^N}[\mathbf{x}_t - \epsilon_t \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)] - \mathbf{x}\|^2. \tag{55}$$

Since $\mathbf{x} \in \mathcal{X}^N$, the distance between the projected vector $\mathcal{P}_{\mathcal{X}^N}[\mathbf{x}_t - \epsilon_t \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)]$ and $\mathbf{x}$ is smaller than the distance before projection. Use this fact in (55) and expand the square

$$\begin{aligned}\|\mathbf{x}_{t+1} - \mathbf{x}\|^2 &\leq \|\mathbf{x}_t - \epsilon_t \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) - \mathbf{x}\|^2 \\ &= \|\mathbf{x}_t - \mathbf{x}\|^2 - 2\epsilon_t \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \lambda_t)^T (\mathbf{x}_t - \mathbf{x}) \\ &\quad + \epsilon_t^2 \|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2. \end{aligned} \tag{56}$$

We reorder terms of the above expression such that the gradient inner product is on the left-hand side, yielding

$$\nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)^T (\mathbf{x}_t - \mathbf{x}) \leq \frac{1}{2\epsilon_t} \left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 \right) + \frac{\epsilon_t}{2} \|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2. \tag{57}$$

Observe now that since the functions $f_{i,t}(\mathbf{x}_i, \boldsymbol{\theta})$ and $h_{ij}(\mathbf{x}_i, \mathbf{x}_j)$ are convex, the online Lagrangian is a convex function of $\mathbf{x}$ [cf. (6)]. Thus, it follows from the first order convexity condition that

$$\hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_t(\mathbf{x}, \boldsymbol{\lambda}_t) \leq \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)^T (\mathbf{x}_t - \mathbf{x}). \tag{58}$$

Substituting the upper bound in (57) for the right hand side of the inequality in (58) yields

$$\hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_t(\mathbf{x}, \boldsymbol{\lambda}_t) \leq \frac{1}{2\epsilon_t} \left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 \right) + \frac{\epsilon_t}{2} \|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2. \tag{59}$$

We set this analysis aside and proceed to repeat the steps in (55), (59) for the distance between the iterate $\boldsymbol{\lambda}_{t+1}$ at time $t+1$ and an arbitrary multiplier $\boldsymbol{\lambda}$.

$$\|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2 = \|[\boldsymbol{\lambda}_t + \epsilon_t \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)]_+ - \boldsymbol{\lambda}\|^2, \tag{60}$$

where we have substituted (9) to express $\boldsymbol{\lambda}_{t+1}$ in terms of $\boldsymbol{\lambda}_t$. Using the non-expansive property of the projection operator in (60) and expanding the square, we obtain

$$\begin{aligned}\|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2 &\leq \|\boldsymbol{\lambda}_t + \epsilon_t \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) - \boldsymbol{\lambda}\|^2. \\ &= \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2 + 2\epsilon_t \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \lambda_t)^T (\boldsymbol{\lambda}_t - \boldsymbol{\lambda}) \\ &\quad + \epsilon_t^2 \|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2. \end{aligned} \tag{61}$$

Reorder terms in the above expression such that the gradient-iterate inner product term is on the left-hand side as

$$\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)^T(\boldsymbol{\lambda}_t-\boldsymbol{\lambda}) \geq \frac{1}{2\epsilon_t}\left(\|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2-\|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2\right)-\frac{\epsilon_t}{2}\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2. \tag{62}$$

Note that the online Lagrangian [cf. (6)] is a concave function of its Lagrange multipliers, which implies that instantaneous Lagrangian differences for fixed $\mathbf{x}_t$ satisfy

$$\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)-\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}) \geq \nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)^T(\boldsymbol{\lambda}_t-\boldsymbol{\lambda}). \tag{63}$$

By using the lower bound stated in (62) for the right hand side of (63), we may write

$$\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)-\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}) \geq \frac{1}{2\epsilon_t}\left(\|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2-\|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2\right)-\frac{\epsilon_t}{2}\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2. \tag{64}$$

We now turn to establishing a telescopic property of the instantaneous Lagrangian by combining the expressions in (59) and (64). To do so observe that the term $\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)$ appears in both inequalities. Thus, subtraction in inequality (64) from those in (59) followed by reordering terms yields

$$\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}_t) \leq \frac{1}{2\epsilon_t}\left(\|\mathbf{x}_t-\mathbf{x}\|^2-\|\mathbf{x}_{t+1}-\mathbf{x}\|^2+\|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2-\|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\right) + \frac{\epsilon_t}{2}\left(\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2+\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2\right),$$

which is as stated in (22). ■

REFERENCES

[1] A. Koppel, B. M. Sadler, and A. Ribeiro, "Proximity without consensus in online multi-agent optimization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 20–25, 2016, pp. 3726–3730.

[2] A. Koppel, B. M. Sadler, and A. Ribeiro, "Decentralized online learning with heterogeneous data sources," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 7–9, 2016.

[3] A. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: Challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 24–40, Jul. 2005.

[4] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "D4l: Decentralized dynamic discriminative dictionary learning," *IEEE Trans. Signal Inf. Process. Netw.*, Jun. 2018. [Online]. Available: http://www.seas.upenn.edu/ aribeiro/wiki

[5] A. Koppel, J. Fink, G. Warnell, E. Stump, and A. Ribeiro, "Online learning for characterizing," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2016, pp. 626–633.

[6] A. Ribeiro, "Optimal resource allocation in wireless communication and networking," *EURASIP J. Wireless Commun. Netw.*, vol. 2012, no. 1, pp. 1–19, 2012.

[7] M. Rabbat and R. Nowak, "Decentralized source localization and tracking [wireless sensor networks]," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, May 2004, vol. 3, pp. iii-921–iii-924.

[8] K. Tsianos, S. Lawlor, and M. Rabbat, "Consensus-based distributed optimization: Practical issues and applications in large-scale machine learning," in *Proc. 50th Annu. Allerton Conf. Commun. Control, Comput.*, Oct. 2012, pp. 1543–1550.

[9] D. Jakovetic, J. M. F. Xavier, and J. M. F. Moura, "Fast distributed gradient methods," *CoRR*, vol. abs/1112.2972, Apr. 2011.

[10] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," ArXiv e-prints 1310.7063, Oct. 2013.

[11] J. Chen and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[12] J. Chen and A. H. Sayed, "On the learning behavior of adaptive networks?Part i: Transient analysis," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3487–3517, Jun. 2015.

[13] M. Rabbat, R. Nowak, and J. Bucklew, "Generalized consensus computation in networked systems with erasure links," in *Proc. IEEE 6th Workshop Signal Process. Adv. Wireless Commun. Process.*, Jun. 5–8, 2005, pp. 1088–1092.

[14] F. Jakubiec and A. Ribeiro, "D-map: Distributed maximum a posteriori probability estimation of dynamic systems," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 450–466, Feb. 2013.

[15] E. Wei and A. Ozdaglar, "Distributed alternating direction method of multipliers," in *Proc. IEEE 51st Annu. Conf. Decision Control*, 2012, pp. 5445–5450.

[16] K. Arrow, L. Hurwicz, and H. Uzawa, *Studies in Linear and Non-Linear Programming (Series Stanford Mathematical Studies in the Social Sciences)*, vol. II. Stanford, CA, USA: Stanford Univ. Press, Dec. 1958.

[17] A. Nedic and A. Ozdaglar, "Subgradient methods for saddle-point problems," *J Optim. Theory Appl.*, vol. 142, no. 1, pp. 205–228, Aug. 2009.

[18] Q. Ling and A. Ribeiro, "Decentralized dynamic optimization through the alternating direction method of multipliers," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1185–1197, Mar. 2014.

[19] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. AMS-22, no. 3, pp. 400–407, Sep. 1951.

[20] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. AC-31, no. 9, pp. 803–812, Sep. 1986.

[21] A. Koppel, F. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5149–5164, Oct. 2015.

[22] S. Ram, A. Nedic, and V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *J. Optim. Theory Appl.*, vol. 147, no. 3, pp. 516–545, Sep. 2010.

[23] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997. [Online]. Available: http://dx.doi.org/10.1023/A:1007379606734

[24] Z. J. Towfic and A. H. Sayed, "Adaptive penalty-based distributed stochastic convex optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3924–3938, Aug. 2014.

[25] J. Chen, C. Richard, and A. H. Sayed, "Multitask diffusion adaptation over networks," *IEEE Trans. Signal Process.*, vol. 62, no. 16, pp. 4129–4144, Aug. 2014.

[26] Z. J. Towfic and A. H. Sayed, "Stability and performance limits of adaptive primal-dual networks," *IEEE Trans. Signal Process.*, vol. 63, no. 11, pp. 2888–2903, Jun. 2015.

[27] V. Cevher, S. Becker, and M. Schmidt, "Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics," *IEEE Signal Process. Mag.*, vol. 31, no. 5, pp. 32–43, Sep. 2014.

[28] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.

[29] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Belmont, MA, USA: Athena Scientific, 2003.

[30] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: Online convex optimization with long term constraints," *J. Mach. Learn. Res.*, vol. 13, pp. 2503–2528, Sep. 2012.

[31] R. Jenatton, J. Huang, and C. Archambeau, "Online optimization and regret guarantees for non-additive long-term constraints," arXiv:1602.05394, 2016.

[32] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1574–1609, 2009.

[33] F. R. Bach, "Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 595–627, 2014.

[34] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. 20th Int. Conf. Mach. Learn.*, Washington, DC, USA, Aug. 21–24 2003, vol. 20, no. 2, pp. 928–936.

[35] M. Dong, L. Tong, and B. M. Sadler., "Information retrieval and processing in sensor networks: Deterministic scheduling vs. random access," in *Proc. Int. Symp. Inf. Theory*, Jun. 2004, pp. 79–.

[36] C. E. Powell, "Numerical methods for generating Gaussian random fields," in *Proc. Porous Media-Processes Math. Annu. Meeting*, Oct. 2014.
[37] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.
[38] R. Kozick and B. Sadler, "Accuracy of source localization based on squared-range least squares (sr-ls) criterion," in *Proc. 3rd IEEE Int. Workshop Comput. Adv. Multi-Sensor Adaptive Process.*, Dec. 2009, pp. 37–40.
[39] P. Roux, M. Corciulo, M. Campillo, and D. Dubuq, "Source localization analysis using seismic noise data acquired in exploration geophysics," *AGU Fall Meeting Abstracts*, Dec. 2011, p. C2249.
[40] K. Cheung, W.-K. Ma, and H. So, "Accurate approximation algorithm for TOA-based maximum likelihood mobile location using semidefinite programming," in *Proc. IEEE Int. Conf. Acoust. Speech, Signal Process.*, May 2004, vol. 2, pp. ii-145–ii-148.
[41] A. Beck, P. Stoica, and J. Li, "Exact and approximate solutions of source localization problems," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1770–1778, May 2008.
[42] J. J. Mor, "Generalizations of the trust region problem," *Optim. Methods Softw.*, vol. 2, pp. 189–209, 1993.
[43] K. M. Anstreicher, "On convex relaxations for quadratically constrained quadratic programming," *Math. Program.*, vol. 136, no. 2, pp. 233–251, 2012.
[44] S. Boyd and L. Vanderberghe, *Convex Programming*. New York, NY, USA: Wiley, 2004.
[45] K. I. Tsianos and M. G. Rabbat, "Distributed strongly convex optimization," *CoRR*, vol. abs/1207.3031, Jul. 2012.
[46] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI, USA: American Mathematical Society, 1997.

Alec Koppel received the B.A. degree in mathematics and the M.S. degree in systems science and mathematics from Washington University, St. Louis, MO, USA. He is currently working toward the doctoral degree in the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA and is a participant in the Science, Mathematics, and Research for Transformation (SMART) Scholarship Program sponsored by the American Society of Engineering Education. His sponsoring facility is the U.S. Army Research Laboratory, Adelphi, MD, USA, where he works during doctoral summers. His research interests include the areas of signal processing, optimization, and learning theory. His current work focuses on designing new large-scale or dynamic optimization methods motivated by problems in robotics and computer networks.

Brian M. Sadler received the B.S. and M.S. degrees from the University of Maryland, College Park, MD, USA, and the Ph.D. degree from the University of Virginia, Charlottesville, VA, USA, all in electrical engineering. His research interests include information science, networked and autonomous systems, sensing, and mixed-signal integrated circuit architectures. He is an associate editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and EURASIP *Signal Processing*, was an associate editor of the IEEE SIGNAL PROCESSING LETTERS, and has been a Guest Editor of several journals including the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, the IEEE *Signal Processing Magazine*, and the *International Journal of Robotics Research*. He is a member of the IEEE Signal Processing Society Sensor Array and Multichannel Technical Committee, and the Co-Chair of the IEEE Robotics and Automation Society Technical Committee on Networked Robotics. He is a Fellow of the Army Research Laboratory, Adelphi, MD, USA. He received the Best Paper Awards from the Signal Processing Society in 2006 and 2010. He has received several ARL and Army R&D awards, as well as a 2008 Outstanding Invention of the Year Award from the University of Maryland.

Alejandro Ribeiro received the B.Sc. degree in electrical engineering from the Universidad de la Republica Oriental del Uruguay, Montevideo, Uruguay, in 1998 and the M.Sc. and Ph.D. degrees in electrical engineering from the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, in 2005 and 2007, respectively. From 1998 to 2003, he was a member of the technical staff at Bellsouth Montevideo. After his M.Sc. and Ph.D studies, in 2008, he joined the University of Pennsylvania (Penn), Philadelphia, PA, USA, where he is currently the Rosenbluth Associate Professor at the Department of Electrical and Systems Engineering. His research interests include the applications of statistical signal processing to the study of networks and networked phenomena. His current research focuses on wireless networks, network optimization, learning in networks, networked control, robot teams, and structured representations of networked data structures. He received the 2012 S. Reid Warren, Jr. Award presented by Penn's undergraduate student body for outstanding teaching, the NSF CAREER Award in 2010, and the student paper awards at the 2013 American Control Conference (as adviser), as well as the 2005 and 2006 International Conferences on Acoustics, Speech and Signal Processing. He is a Fulbright scholar and a Penn Fellow.