Asynchronous Decentralized Stochastic Optimization in Heterogeneous Networks

Amrit Singh Bedi*, *Student Member, IEEE,* Alec Koppel†, *Student Member, IEEE,* and Ketan Rajawat*, *Member, IEEE*

***Abstract*—We consider expected risk minimization in multi-agent systems comprised of distinct subsets of agents operating without a common time-scale. Each individual in the network is charged with minimizing the global objective function, which is an average of sum of the statistical average loss function of each agent in the network. Since agents are not assumed to observe data from identical distributions, the hypothesis that all agents seek a common action is violated, and thus the hypothesis upon which consensus constraints are formulated is violated. Thus, we consider nonlinear network proximity constraints which incentivize nearby nodes to make decisions which are close to one another but not necessarily coincide. Moreover, agents are not assumed to receive their sequentially arriving observations on a common time index, and thus seek to learn in an asynchronous manner. An asynchronous stochastic variant of the Arrow-Hurwicz saddle point method is proposed to solve this problem which operates by alternating primal stochastic descent steps and Lagrange multiplier updates which penalize the discrepancies between agents. This tool leads to an implementation that allows for each agent to operate asynchronously with local information only and message passing with neighbors. Our main result establishes that the proposed method yields convergence in expectation both in terms of the primal sub-optimality and constraint violation to radii of sizes $\mathcal{O}(\sqrt{T})$ and $\mathcal{O}(T^{3/4})$, respectively. Empirical evaluation on an asynchronously operating wireless network that manages user channel interference through an adaptive communications pricing mechanism demonstrates that our theoretical results translates well to practice.**

I. Introduction

In emerging technologies such as wireless communications and networks consisting of interconnected consumer devices [2], increased sensing capabilities are leading to new theoretical challenges to classical parameter estimation. These challenges include the fact that data is persistently arriving in a sequential fashion [3], that it is physically decentralized across an interconnected network, and that the nodes of the network may correspond to disparate classes of objects (such as users and a base station) with different time-scale requirements [4]. In this work, we seek to address this class of problems through extensions of online decentralized convex optimization [5] to the case where the agents of the network may be of multiple different classes, and operate on different time-scales [6].

To address the fact that we seek iterative tools for streaming data, we consider stochastic optimization problems [7], [8]. In this setting, the objective function $\mathbb{E}[f(\mathbf{x}, \boldsymbol{\theta})]$ is an expectation over a set of functions parameterized by a random variable $\boldsymbol{\theta}$. The objective function encodes, for example, the quality of a statistical parameter estimate. Through a sequence of realizations of a random variable $\boldsymbol{\theta}_t$, we seek to find parameters that are good with respect to the average objective. The classical method to address this problem is stochastic gradient descent (SGD), which involves descending along the negative of the stochastic gradient in lieu of the true gradient to circumvent the computation of an infinite complexity expectation [9], [10]. SGD forms the foundation of tools considered in this paper for asynchronous multi-agent settings.

Here we seek solutions to stochastic programs in which data is scattered across an interconnected network $\mathcal{G}{=}(\mathcal{V},\mathcal{E})$ of agents, each of which is associated with a unique stream of data $\{\boldsymbol{\theta}_t^i\}_{t\geq 0}$. Agents $i \in \mathcal{V}$ then seek to find a solution based on local computations only which is as good as one at a centralized location: this setting is mathematically defined by introducing a local copy of a global parameter estimate and then having each agent seek to minimize a global sum of all local objectives $\sum_i \mathbb{E}[f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)]$ while satisfying consensus constraints $\mathbf{x}_i = \mathbf{x}_j$ for all node pairs $(i,j) \in \mathcal{E}$. Techniques that are good for distributed convex optimization, for example, those based on penalty method [11], [12] or on Lagrange duality [13], [14], have in most cases translated into the distributed stochastic domain without major hurdles, as in [12], [15], [16].

In the aforementioned works, consensus constraints are enforced in order to estimate a common decision variable while leveraging parallel processing architectures to achieve computational speedup [17], [18]. Contrariwise, when unique priors on information available at distinct group of agents are available as in sensor [19] or robotic [20] networks, enforcing consensus degrades the statistical accuracy of each agent's estimate [19]. Specifically, if the observations at each node are independent but *not* identically distributed, consensus may yield a sub-optimal solution. Motivated by heterogeneous networked settings [20], [21] where each node observes a unique local data stream, we focus on the setting of multi-agent stochastic optimization with nonlinear *proximity* constraints which incentivize nearby nodes to select estimates which are similar but not necessarily equal.

In the setting of nonlinear constraints, penalty methods such as distributed gradient descent do not apply [11], and dual or proximal methods require a nonlinear minimization in an inner-loop of the algorithm [14]. Therefore, we adopt a method which hinges on Lagrange duality that avoids costly argmin computations in the algorithm inner-loop, namely primal-dual method [22], also referred to as saddle point method. Alternative attempts to extend multi-agent optimization techniques to heterogeneously correlated problems have been considered in [23], [24] for special loss functions and correlation models, but the generic problem was online recently solved in [19] with a stochastic variant of the saddle point method.

However, insisting on all agents to operate on a common clock creates a bottleneck for implementation in practical settings because typically nodes may be equipped with different computational capacity due to power and energy design specifications, as well as a difference in the sparsity of each agent's data stream. Therefore, we attempt to extend multi-agent stochastic optimization with nonlinear constraints to asynchronous settings [25]. Asynchrony in online optimization has taken on different forms, such as, for instance, maintaining a local Poisson clock for each agent [26] or a distribution-free generic bounded delay [6],

A. S. Bedi and K. Rajawat are with the Department of Electrical Engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India (e-mail: amritbd@iitk.ac.in; ketan@iitk.ac.in). A. Koppel is with the U.S. Army Research Laboratory, Adelphi, MD, USA. (e-mail: akoppel@seas.upenn.edu.). A part of this work is submitted to IEEE ICC workshops 2018 [1].

[27], the approach considered here.

In this paper, we extend the primal-dual method of [19], [22] for multi-agent stochastic optimization problems with nonlinear network proximity constraints to asynchronous settings. The proposed algorithm allows the gradient to be delayed for the primal and dual updates of the saddle point method. The main technical contribution of this paper is to provide mean convergent results for both the global primal cost and constraint violation, establishing that the Lyapunov stability results of [19] translate successfully to asynchronous computing architectures increasingly important in intelligent communication systems. Empirical evaluation on an asynchronously operating cellular network that manages cross-tier interference through an adaptive pricing mechanism demonstrates that our theoretical results translates well to practice.

The rest of the paper is organized as follows. A multi-agent optimization problem without consensus is formulated in Section II. An asynchronous saddle point algorithm is proposed to solve the problem in Section III. The detailed convergence analysis for the proposed algorithm is presented in Section IV. Next, a practical problem of interference management through pricing is solved in Section V. Section VI concludes the paper.

II. Multi-Agent Optimization without Consensus

We consider agents i of a symmetric, connected, and directed network $\mathcal{G} = (V, \mathcal{E})$ with $|V| = N$ nodes and $|\mathcal{E}| = M$ edges. Each agent is associated with a (non-strongly) convex loss function $f^i : \mathcal{X} \times \Theta_i \to \mathbb{R}$ that is parameterized by a p-dimensional decision variable $\mathbf{x}^i \in \mathcal{X} \subset \mathbb{R}^p$ and a random vector $\boldsymbol{\theta}_i \in \Theta_i \subset \mathbb{R}^q$. The functions $f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)$ for different $\boldsymbol{\theta}^i$ encodes the merit of a particular linear statistical model $\mathbf{x}^i$, for instance, and the random vector $\boldsymbol{\theta}$ may be particularized to a random pair $\boldsymbol{\theta} = (\mathbf{z}, \mathbf{y})$. In this setting, the random pair corresponds to feature vectors $\mathbf{z}$ together with their binary labels $\mathbf{y} \in \{-1, 1\}$ or real values $\mathbf{y} \in \mathbb{R}$, for the respective problems of classification or regression. Here we address the case that the local random vector $\boldsymbol{\theta}^i$ represents data which is revealed to node i *sequentially* as realizations $\boldsymbol{\theta}^i_t$ at time t, and agents would like to process this information on the fly. Mathematically this is equivalent to the case where the total number of samples T revealed to agent i is not necessarily finite. A possible goal for agent i is the solution of the local expected risk minimization problem,

$$\mathbf{x}^{\mathrm{L}}(i) := \operatorname*{argmin}_{\mathbf{x}^i \in \mathcal{X}} F^i(\mathbf{x}^i) := \operatorname*{argmin}_{\mathbf{x}^i \in \mathbb{R}^p} \mathbb{E}_{\boldsymbol{\theta}^i}[f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)] \,. \tag{1}$$

where we define $F^i(\mathbf{x}^i) := \mathbb{E}_{\boldsymbol{\theta}^i}[f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)]$ as the local average function at node i. We also restrict $\mathcal{X}$ to be a compact convex subset of $\mathbb{R}^p$ associated with the p-dimensional parameter vector of agent i. By stacking the problem (1) across the entire network, we obtain the equivalent problem

$$\mathbf{x}^{\mathrm{L}} = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X}^N} F(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X}^N} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}^i}[f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)] \,. \tag{2}$$

where we define the stacked vector $\mathbf{x} = [\mathbf{x}^1, \ldots, \mathbf{x}^N] \in \mathcal{X}^N \subset \mathbb{R}^{Np}$, and the global cost function $F(\mathbf{x}) := \sum_{i=1}^N \mathbb{E}_{\boldsymbol{\theta}^i}[f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)]$. We define the global instantaneous cost similarly: $f(\mathbf{x}, \boldsymbol{\theta}) = \sum_i f^i(\mathbf{x}^i, \boldsymbol{\theta}^i)$.

Note that (1) and (2) describe the same problem since the variables $\mathbf{x}^i$ at different agents are not coupled to one another. In many situations, the parameter vectors of distinct agents are related, and thus there is motivation to couple the estimates of distinct agents to each other such that one agent may take advantage of another's data. Most distributed optimization works, for instance, consensus optimization, hypothesize that all agents seek to learn the common parameters $\mathbf{x}^i$ for all $i \in V$, i.e., $\mathbf{x}^i = \mathbf{x}^j$, for all $j \in n_i$. where n_i denotes the neighborhood of agent i. Making all agents variables equal only makes sense when agents observe information drawn from a common distribution, which is the case for industrial-scale machine learning, but is predominantly not the case for sensor [19] and robotic networks [20]. As noted in [19], generally, nearby nodes observe similar but not identical information, and thus to incentivize collaboration without enforcing consensus, we introduce a convex local proximity function with real-valued range of the form $h^{ij}(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i, \boldsymbol{\theta}^j)$ that depends on the observations of neighboring agents and a tolerance $\gamma_{ij} \geq 0$. These stochastic constraints then couple the decisions of agent i to those of its neighbors $j \in n_i$ as the solution of the constrained stochastic program

$$\begin{aligned} \mathbf{x}^* \in \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X}^N} & \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}_i}[f^i(\mathbf{x}_i, \boldsymbol{\theta}_i)] \\ \text{s.t.} \quad & \mathbb{E}_{\boldsymbol{\theta}^i, \boldsymbol{\theta}^j}\left[h^{ij}(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i, \boldsymbol{\theta}^j)\right] \leq \gamma_{ij}, \text{ for all } j \in n_i. \end{aligned} \tag{3}$$

Examples of the constraint in the above formulation include approximate consensus constraints $\left\|\mathbf{x}^i - \mathbf{x}^j\right\| \leq \gamma_{ij}$, quality of service $\mathbf{SINR}(\mathbf{x}^i, \mathbf{x}^j) \geq \gamma_{ij}$ where SINR is the signal-to-interference-plus-noise function, relative entropy $D(\mathbf{x}^i \,||\, \mathbf{x}^j) \leq \gamma_{ij}$, or budget $\gamma_{ij}^{\min} \leq x^i + x^j \leq \gamma_{ij}^{\max}$ constraints. In this work, we seek decentralized online solutions to the constrained problem (3) *without* the assumption that agents operate on a common time index, motivated by the fact that asynchronous computing settings are common in large distributed wireless networks. In the next section we turn to developing an algorithmic solution that meets these criteria.

Remark 1 The pairwise stochastic constraints in (3) can be readily generalized to arbitrary neighborhood constraints:

$$\mathbb{E}\left[\mathbf{h}^i(\{\mathbf{x}^j, \boldsymbol{\theta}^j\}_{j \in n_i'})\right] \leq \mathbf{0} \tag{4}$$

where the set $n_i' := n_i \cup \{i\}$ denotes the set of all neighbors of node i inlcuding the node i itself. It can be seen that constraint in (3) is a special case of that in (4), with the j-th entry of the $|n_i| \times 1$ vector function $\mathbf{h}^i(\cdot)$ defined as

$$\left[\mathbf{h}^i(\{\mathbf{x}^j, \boldsymbol{\theta}^j\}_{j \in n_i'})\right]_j := h^{ij}(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i, \boldsymbol{\theta}^j) - \gamma_{ij}. \tag{5}$$

More generally, (4) allows the consensus constraints to be imposed on the entire neighborhood of a node. For instance the approximate version of the consensus constraint $\mathbf{x}^i = (1/|n_i|) \sum_{j \in n_i} \mathbf{x}^j$ takes the form

$$\left\|\mathbf{x}^i - (1/|n_i|) \sum_{j \in n_i} \mathbf{x}^j\right\| \leq \gamma_i. \tag{6}$$

Such general constraints also arise in communication systems in form of SINR constraints. For instance, consider a communication system where the interference at node i from j can be written as $p^j(\mathbf{x}^j, g^{ij})$ with g^{ij} denoting the channel gain from node j to

node i. Then the SINR constraint at node i is of the form in (4), and is given by

$$h^i(\{\mathbf{x}^j, \boldsymbol{\theta}^j\}_{j\in n_i'}) = \gamma_{ij} - \frac{p^i(\mathbf{x}^i, g^{ii})}{\sigma^2 + \sum_{j\in n_i} p^j(\mathbf{x}^j, g^{ij})} \tag{7}$$

where σ^2 denotes the noise power. Hence, the generalized stochastic problem can be expressed as

$$\begin{aligned} \mathbf{x}^* \in \underset{\mathbf{x}\in\mathcal{X}^N}{\operatorname{argmin}} & \sum_{i=1}^{N} \mathbb{E}[f^i(\mathbf{x}_i, \boldsymbol{\theta}_i)] \\ \text{s.t.} \quad & \mathbb{E}\left[\mathbf{h}^i\left(\{\mathbf{x}^j, \boldsymbol{\theta}^j\}_{j\in n_i'}\right)\right] \leq \mathbf{0} \ \text{ for all } i. \end{aligned} \tag{8}$$

It is mentioned that the convergence analysis is performed for the generalized problem in (8) for clarity of exposition since in this more general setting we may vectorize the constraints, while the main results in Section IV are presented for simpler problem of (3) [see Appendix 0] for increased interpretability.

III. Asynchronous Saddle Point Method

Methods based upon distributed gradient descent and penalty methods more generally [28]–[30] are inapplicable to settings with nonlinear constraints, with the exception of [31], which requires attenuating learning rates to attain constraint satisfaction. On the other hand, the dual methods proposed in [32]–[34] require a non-linear minimization computation at each algorithm iteration, and thus is impractically costly. Therefore, in this section we develop a computationally light weight method based on primal-dual method that may operate in decentralized online asynchronous settings with constant learning rates that are better suited to changing environments.

For a decentralized algorithm, each node i can access the information from its neighbors $j \in n_i$ only. For an online algorithm, the stochastic i.i.d quantities of unknown distribution are observed sequentially $\boldsymbol{\theta}_t^i$ at each time instant t. In addition to these properties, an algorithm is called asynchronous, if parameter updates may be executed with out-of-date information and the requirement that distinct nodes operate on a common time-scale is omitted. To develop an algorithm which meets these specifications, begin by considering the approximate Lagrangian relaxation of (3) stated as

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^{N} \Bigg[\mathbb{E}\Bigg[& f^i(\mathbf{x}^i, \boldsymbol{\theta}^i) \\ & + \sum_{j\in n_i} \lambda^{ij} \left(h^{ij}\left(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i, \boldsymbol{\theta}^j\right) - \gamma_{ij}\right) - \frac{\delta\epsilon}{2}(\lambda^{ij})^2 \Bigg] \Bigg], \end{aligned} \tag{9}$$

where λ^{ij} is a non-negative Lagrange multiplier associated with the non-linear constraint in (3). Here, $\boldsymbol{\lambda}$ defines the collection of all dual variables λ^{ij} into a single vector $\boldsymbol{\lambda}$. Observe that (9) is not the standard Lagrangian of the (3) but instead an augmented Lagrangian due to the presence of the term $-(\delta\epsilon/2)(\lambda^{ij})^2$. This terms acts like a regularizer on the dual variable with associated parameters δ and ϵ that allow us to control the accumulation of constraint violation of the algorithm over time, as is discussed in the following section and proofs in the appendices.

The stochastic saddle point algorithm, when applied to (9), operates by alternating primal and dual stochastic gradient descent and ascent steps, respectively. We consider the stochastic saddle point method as a template upon which we construct an asynchronous protocol. Begin then by defining the stochastic approximation of the augmented Lagrangian evaluated at observed realizations $\boldsymbol{\theta}_t^i$ of the random vectors $\boldsymbol{\theta}^i$ for each $i \in \mathcal{V}$:

$$\begin{aligned} \hat{\mathcal{L}}_t(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^{N} \Bigg[& f^i(\mathbf{x}^i, \boldsymbol{\theta}_t^i) \\ & + \sum_{j\in n_i} \lambda^{ij} \Big(h^{ij}\left(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}_t^i, \boldsymbol{\theta}_t^j\right) - \gamma_{ij}\Big) - \frac{\delta\epsilon}{2}(\lambda^{ij})^2 \Bigg]. \end{aligned} \tag{10}$$

The stochastic saddle point method applied to the stochastic Lagrangian (10) takes the following form similar to [19] as

$$\mathbf{x}_{t+1} = \mathcal{P}_{\mathcal{X}}\left[\mathbf{x}_t - \epsilon \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\right], \tag{11}$$

$$\boldsymbol{\lambda}_{t+1} = \left[\boldsymbol{\lambda}_t + \epsilon \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\right]_+, \tag{12}$$

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[1] of the augmented Lagrangian with respect to $\mathbf{x}$ and $\boldsymbol{\lambda}$, respectively. These are not the actual gradients of (9) rather are stochastic gradients calculated at the current realization of the random vectors $\boldsymbol{\theta}_t^i$ for all i. The component wise projection for a vector $\mathbf{x}$ on to the given compact set $\mathcal{X}$ is here denoted by $\mathcal{P}_{\mathcal{X}}(\mathbf{x})$. Similarity, $[\cdot]_+$ represents the component wise projection on to the positive orthant $\mathbb{R}_+^M$. An important point here is that the method stated in (11) - (12) can be implemented with decentralized computations across the network, as stated in [19, Proposition 1]. Here $\epsilon > 0$ is a constant positive step-size.

Observe that the implementation of the [19, Algorithm 1], which is defined by (11)-(12), it is mandatory to perform the primal and dual updates at each node with a common time index t. The update of primal variable at node i requires the current gradient of its local objective function $\nabla_{\mathbf{x}^i} f^i(\mathbf{x}_t^i, \boldsymbol{\theta}_t^i)$ and current gradient from all the neighbors $j \in n_i$ of node i as $\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}_t^i, \mathbf{x}_t^j, \boldsymbol{\theta}_t^i, \boldsymbol{\theta}_t^j\right)$. This availability of the gradients from the neighbors on a common time-scale is a strong assumption that insists upon perfect communications, similarity of computational capability of distinct nodes, and similar levels of sparsity among agents' data that are oftentimes violated in large heterogeneous systems. This limitation of synchronized methods motivates the subsequent development of asynchronous decentralized variants of (11)-(12)

In particular, to ameliorate the computational bottleneck associated with synchronized computation and communication rounds among the nodes, we consider situations in which observations and updates are subject to stochastic delays, i.e., an *asynchronous processing architecture*. These delays take the form of random delays on the gradients which are used for the algorithm updates. We associate to each node i in the network a time-dependent delay $\tau_i(t)$ for its stochastic gradient. Since the gradient corresponding to node i are delayed by $\tau_i(t)$, it implies that the received gradient corresponds to $t - \tau_i(t)$ time slot which we denote as $[t]_i$. Rather than waiting for the current gradient at time t, agent i instead uses the delayed gradient from the neighboring nodes at time $[t]_j$ for its update at time $[t]_i$. This leads to the following asynchronous

[1]Note that these may be subgradients if the objective/ constraint functions are non-differentiable. The proof is extendable to non-differentiable cases.

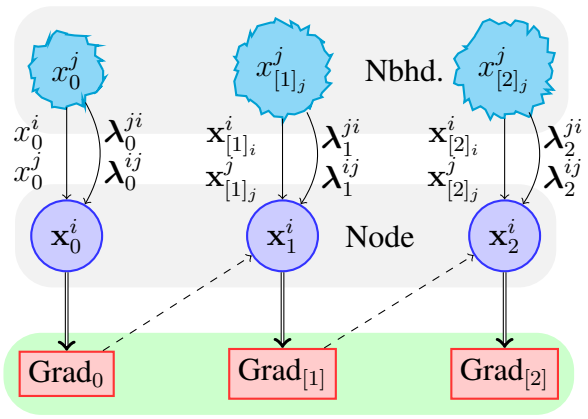


Fig. 1: Message passing in proposed algorithm. The red block represents the gradients required as mentioned in step 4 of the Algorithm 1 which depends upon the realizations $\boldsymbol{\theta}^i_{[t]_i}$ and $\boldsymbol{\theta}^j_{[t]_j}$ at time step t.

primal update for stochastic online saddle point algorithm at each node i

$$\mathbf{x}^i_{t+1} = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}^i_t - \epsilon\Big(\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i}) + \sum_{j\in n_i}\Big(\lambda^{ij}_t + \lambda^{ji}_t\Big)\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)\Big)\Big]. \tag{13}$$

Likewise, the dual update for each edge $(i,j) \in \mathcal{E}$ is

$$\lambda^{ij}_{t+1} = \Big[(1-\epsilon^2\delta)\lambda^{ij}_t + \epsilon\left(h^{ij}\Big(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\Big)\right)\Big]_+. \tag{14}$$

Note that to perform the asynchronous primal updates at node i in (13), delayed primal gradients $\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i})$ and $\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)$ are utilized. Similarity, the dual delayed gradient is utilized for the update in (14). For the consistency in the algorithm implementation, it is assumed that at each node i, only the recent received copy of the gradient is kept and used for the update. Equivalently, this condition can be mentioned as $[t]_i \geq [t-1]_i$ which implies that $\tau_i(t) \leq \tau_i(t-1)+1$. For brevity, we will use the notation $[\mathbf{t}]$ as a collective notation for all the delayed time instances as $[\mathbf{t}] := [[t]_1; \cdots; [t]_N]$. The asynchronous algorithm is summarized in Algorithm 1. The practical implementation of the proposed asynchronous algorithm is explanied with the help of diagram in Fig.1. As described in figure, each node receives delayed parameters, gradients and carries out the updated accordingly. The convergence guarantees for the proposed algorithm are shown to hold as $t-[t]_i \leq \tau$ is finite (assumption 6) for all i and t. The convergence results presented in [19] can be obtained as a special case with $\tau_i(t) = 0$ from the results developed in this paper. This shows the generalization of the existing results in literature.

Before proceeding with the convergence analysis, in order to have a tractable derivation, let us define the compact notation for the primal and dual delayed gradient as follows

$$\nabla_{\mathbf{x}^i}\hat{\mathcal{L}}^i_{[\mathbf{t}]}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\lambda}^i_{[t]_i}, \boldsymbol{\lambda}^j_{[t]_j}\right) := \Big(\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i}) + \sum_{j\in n_i}(\lambda^{ij}_t + \lambda^{ji}_t)\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)\Big) \tag{15}$$

which follows from (13) and

$$\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}^i_{[\mathbf{t}]}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\lambda}^i_{[t]_i}\right) := h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right) \tag{16}$$

follows from (14). These definitions of (15) and (16) will be used in rest of the places in this paper.

Algorithm 1 ASSP: Asynchronous Stochastic Saddle Point

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

1: **for** $t = 1, 2, \ldots, T$ **do**
2: **loop in parallel** agent $i \in \mathcal{V}$
3: Send dual vars. $\boldsymbol{\lambda}_{ij,t}$ to nbhd. $j \in n_i$
4: Observe delayed gradients $\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i})$, $\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)$ and constraint function $h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)$.
5: Update $\mathbf{x}^i_{t+1}$ using (13) local parameter $\mathbf{x}^i_t$

$$\mathbf{x}^i_{t+1} = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}^i_t - \epsilon\Big(\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i}) + \sum_{j\in n_i}\Big(\lambda^{ij}_t + \lambda^{ji}_t\Big)\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)\Big)\Big]$$

6: Update dual variables at each agent i [cf. (14)]

$$\lambda^{ij}_{t+1} = \Big[(1-\epsilon^2\delta)\lambda^{ij}_t + \epsilon\left(h^{ij}\left(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)\right)\Big]_+.$$

7: **end loop**
8: **end for**

IV. Convergence in Expectation

In this section, we establish convergence in expectation of the proposed asynchronous technique in (13)-(14) to a primal-dual optimal pair of the problem formulated in (3) when constant step sizes are used. Specifically, a sublinear bound on the average objective function optimality gap $F(\mathbf{x}_t) - F(\mathbf{x}^*)$ and the network-aggregate delayed constraint violation is established, both on average. The optimal feasible vector $\mathbf{x}^*$ is defined by (3). It is shown that the time-average primal objective function $F(\mathbf{x}_t)$ converges to the optimal value $F(\mathbf{x}^\star)$ at a rate of $\mathcal{O}(1/\sqrt{T})$. Similarly, the time-average aggregated delayed constraint violation over network vanishes with the order of $\mathcal{O}(T^{-1/4})$, both in expectation, where T is the final iteration index. To prove convergence of the stochastic asynchronous saddle point method, some assumptions related to the system model and parameters are required which we state as follows.

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

Assumption 2 *(*Existence of Optima*) The set of primal-dual optimal pairs $\mathcal{X}^* \times \boldsymbol{\Lambda}^*$ of the constrained problem* (3) *has non-empty intersection with the feasible domain $\mathcal{X}^N \times \mathbb{R}^M_+$.*

Assumption 3 *(*Stochastic Gradient Variance*) The instantaneous objective and constraints for all i and t satisfy*

$$\mathbb{E}\left\|\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i, \boldsymbol{\theta}^i_t)\right\|^2 \leq \sigma_f^2 \tag{17}$$

$$\mathbb{E}\left\|\nabla_{\mathbf{x}^i} h^{ij}\left(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}\right)\right\|^2 \leq \sigma_h^2 \tag{18}$$

which states that the second moment of the norm of objective and constraint function gradients are bounded above.

Assumption 4 *(*Constraint Function Variance*) For the instantaneous constrain function for all pairs $(i,j) \in \mathcal{E}$ and t over the compact set $\mathcal{X}$, it holds that*

$$\max_{(\mathbf{x}^i, \mathbf{x}^j)\in\mathcal{X}} \mathbb{E}\left[\left(h^{ij}\left(\mathbf{x}^i, \mathbf{x}^j, \boldsymbol{\theta}^i_t, \boldsymbol{\theta}^j_t\right)^2\right)\right] \leq \sigma^2_{\boldsymbol{\lambda}} \tag{19}$$

which implies that the maximum value the constraint function can take is bounded by some finite scalar $\sigma_{\boldsymbol{\lambda}}^2$ in expectation.

Assumption 5 *(Lipschitz continuity) The expected objective function defined in* (2) *satisfies*

$$\|F(\mathbf{x}) - F(\mathbf{y})\| \leq L_f \|\mathbf{x} - \mathbf{y}\|. \tag{20}$$

for any $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{Np}$.

Assumption 6 *(Bounded Delay) The delay $\tau_i(t)$ associated with each node i is upper bounded: $\tau_i(t) \leq \tau$ for some $\tau < \infty$.*

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 [29], [32]. Moreover, Assumption 2 is a Slater's condition which makes sure the existence of an optimal primal-dual pair within the feasible sets onto which projections occur which are necessary for various quantities to be bounded. It has appeared in various forms to guarantee existence of solutions in constrained settings [35]. Assumption 3 assumes an upper bound on the mean norm of the primal and dual stochastic gradients, which is crucial to developing the gradient bounds for the Lagrangian used in the proof. Assumption 4 yields an upper bound on the maximum possible value of the constraint function in expectation similar to that of [36], and is guaranteed to hold when $\mathcal{X}$ is compact and h^{ij} is Lipschitz. Assumption 5 is related to the Lipschitz continuity of the primal objective function. Assumption 6 ensures that the delay is always bounded by τ, which holds in most wireless communications problems and autonomous multi-agent networks [37].

For the analysis to follow, we first derive bounds on the mean square-norms of the stochastic gradients of the Lagrangian. Thus, consider the mean square-norm of the primal stochastic gradient of the Lagrangian, stated as:

$$\begin{aligned}\mathbb{E}\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda})\|^2] \leq & 2N\Big[\max_i \mathbb{E}\left\|\nabla_{\mathbf{x}^i} f^i(\mathbf{x}^i,\boldsymbol{\theta}_t^i)\right\|^2\Big] \\ & + 2M^2\|\boldsymbol{\lambda}\|^2 \max_{(i,j)\in\mathcal{E}} \mathbb{E}\Big\|\nabla_{\mathbf{x}^i} h^{ij}\Big(\mathbf{x}^i,\mathbf{x}^j,\boldsymbol{\theta}_t^i,\boldsymbol{\theta}_t^j\Big)\Big\|^2\end{aligned} \tag{21}$$

Now apply Assumptions in 3 to the mean square-norm terms in (21) to obtain

$$\begin{aligned}\mathbb{E}[\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda})\|^2] &\leq 2N\sigma_f^2 + 2M^2\|\boldsymbol{\lambda}\|^2\sigma_h^2 \\ &\leq 2(N+M^2)L^2(1+\|\boldsymbol{\lambda}\|^2)\end{aligned} \tag{22}$$

where, $L^2 = \max(\sigma_f^2, \sigma_h^2)$. Similarly for the gradient with respect to the dual variable $\boldsymbol{\lambda}$, we have

$$\begin{aligned}\mathbb{E}\left[\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda})\|^2\right] &\leq 2M\max_{(i,j)\in\mathcal{E}}\mathbb{E}[(h^{ij}(\mathbf{x}^i,\mathbf{x}^j,\boldsymbol{\theta}_t^i,\boldsymbol{\theta}_t^j))^2] + 2\delta^2\epsilon^2\|\boldsymbol{\lambda}\|^2 \\ &\leq 2M\sigma_{\boldsymbol{\lambda}}^2 + 2\delta^2\epsilon^2\|\boldsymbol{\lambda}\|^2\end{aligned} \tag{23}$$

The bounds developed in (22) and (23) are in terms of the norm of the dual variable and utilizes the Assumptions 3 and 4. It is important to note that these bounds are for arbitrary $\mathbf{x}$ and $\boldsymbol{\lambda}$, therefore holds for any realization of the primal $\mathbf{x}_t$ and dual variables $\boldsymbol{\lambda}_t$. Before proceeding towards the main lemmas and theorem of this work, a remark on the importance of the bounds in (22) and (23) is due.

Remark 2 Conventionally, in the analysis of primal-dual methods, the primal and dual gradients are bounded by constants. In contrast, here our upper-estimates depend upon the magnitude of the dual variable $\boldsymbol{\lambda}$ which allows us to avoid dual set projections onto a compact set, and instead operate with unbounded dual sets $\mathbb{R}_+^M$. We are able to do so via exploitation of the dual regularization term $-(\delta\epsilon/2)\|\boldsymbol{\lambda}\|^2$ that allows us to control the growth of the constraint violation.

Subsequently, we establish a lemma for the instantaneous Lagrangian difference $\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x},\boldsymbol{\lambda}_t)$ by a telescopic quantity involving the primal and dual iterates, as well as the magnitude of the primal and dual gradients. This lemma is crucial to the proof of our main result at the end of this section.

Lemma 1 *Under the Assumptions 1 - 6 , the sequence $(\mathbf{x}_t, \boldsymbol{\lambda}_t)$ generated by the proposed asynchronous stochastic saddle point algorithm in* (13)-(14) *is such that for a constant step size ϵ, the instantaneous Lagrangian difference sequence $\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x},\boldsymbol{\lambda}_t)$ satisfies the decrement property*

$$\begin{aligned}&\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x},\boldsymbol{\lambda}_t) \\ &\leq \frac{1}{2\epsilon}\big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\big) \\ &\quad + \frac{\epsilon}{2}\left(\left\|\nabla_{\boldsymbol{\lambda}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}_t)\right\|^2 + \|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}_t)\|^2\right) \\ &\quad + \langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}_t),(\mathbf{x}_{[\mathbf{t}]}-\mathbf{x}_t)\rangle\end{aligned} \tag{24}$$

Proof: See Appendix A. ■

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 $\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]},\boldsymbol{\lambda}) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x},\boldsymbol{\lambda}_t)$ in terms of the difference between the current and the next primal and dual iterates to a fixed primal-dual pair $(\mathbf{x},\boldsymbol{\lambda})$, as well as the square magnitudes of the primal and dual gradients. Observe that here, relative to [19][Proposition 1], an additional term is present which appears that represents the directional error caused by asynchronous updates of Algorithm 1. This contractive property is the basis for establishing the convergence of the primal iterates to their constrained optimum given by (3) in terms of mean objective function evaluation and mean constraint violation with constant step-size selection. Before proceeding towards our main theorem, we establish an additional lemma which simplifies its proof and clarifies ideas.

Lemma 2 *Denote as $(\mathbf{x}_t, \boldsymbol{\lambda}_t)$ the sequence generated by the asynchronous saddle point algorithm in* (13)-(14) *with stepsize ϵ. If Assumptions 1 - 6 holds, then it holds that*

$$\begin{aligned}&\mathbb{E}\Big[\sum_{t=1}^{T}[F(\mathbf{x}_t)-F(\mathbf{x})] + \sum_{(i,j)\in\mathcal{E}}\lambda^{ij}\left(\sum_{t=1}^{T}h^{ij}\Big(\mathbf{x}^i_{[t]_i},\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^i_{[t]_i},\boldsymbol{\theta}^j_{[t]_j}\Big)\right) \\ &\quad - \Big(\frac{\delta\epsilon T}{2} + \frac{1}{2\epsilon}\Big)(\lambda^{ij})^2\Big] \leq \frac{1}{2\epsilon}\|\mathbf{x}_1-\mathbf{x}\|^2 + \frac{\epsilon TK}{2}\end{aligned} \tag{25}$$

where the constant K is defined in terms of system parameters as

$$K := M\sigma_{\boldsymbol{\lambda}}^2 + (N+M)L[(1/2)L + \tau^2(2L+L_f)]. \tag{26}$$

Proof: See Appendix B. ■

Lemma 2 is derived by considering Lemma 1, computing expectations, and applying (22) and (23). This lemma describes the global behavior of the augmented Lagrangian when following Algorithm 1, and may be used to establish convergence in terms of primal objective optimality gap and aggregated network constraint violation, as we state in the following theorem.

Theorem 1 *Under the Assumptions 1 - 6 , denote* $(\mathbf{x}_t, \boldsymbol{\lambda}_t)$ *as the sequence of primal-dual variables generated by Algorithm 1 [cf.* (13)-(14)*]. When the algorithm is run for* T *total iterations with constant step size* $\epsilon = 1/\sqrt{T}$*, the average time aggregation of the sub-optimality sequence* $\mathbb{E}\left[F(\mathbf{x}_t) - F(\mathbf{x}^*)\right]$*, with* $\mathbf{x}^*$ *defined by* (3)*, grows sublinearly with* T *as*

$$\sum_{t=1}^{T} \mathbb{E}[F(\mathbf{x}_t) - F(\mathbf{x}^*)] \leq \mathcal{O}(\sqrt{T}). \tag{27}$$

Likewise, the delayed time aggregation of the average constrain violation also grows sublinearly in T *as*

$$\sum_{(i,j)\in\mathcal{E}} \mathbb{E}\left[\left[\sum_{t=1}^{T}\left(h^{ij}(\mathbf{x}^i_{[t]_i}, \mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^i_{[t]_i}, \boldsymbol{\theta}^j_{[t]_j}) - \gamma_{ij}\right)\right]_+\right] \leq \mathcal{O}(T^{3/4}). \tag{28}$$

Proof: See Appendix C. ■

Theorem 1 presents the behavior of the Algorithm 1 when run for T total iterations with a constant step size. Specifically, the average aggregated objective function error sequence is upper bounded by a constant time $\sqrt{T}$ sequence. This establishes that the expected value of the objective function $\mathbb{E}\left[F(\mathbf{x}_t)\right]$ will become closer to the optimal $F(\mathbf{x}^\star)$ for larger T. Similar behavior is shown by the average delayed aggregated network constraint violation term. The key innovation establishing this result is the bound on the directional error caused by asynchrony. We achieve this by bounding it in terms of the gradient norms and dual variable $\boldsymbol{\lambda}$ and exploiting the fact that the delay is at-worst bounded.

These results are similar to those for the unconstrained convex optimization problems with sub-gradient descent approach and constant step size. For most of the algorithms in this context [38, Section 2.2, eqn. 2.19], or [39, Section 4], convergence to the neighborhood of size $\mathcal{O}(\epsilon T)$ is well known. For such algorithms, the primal sub-optimality is shown of the order $\mathcal{O}(\epsilon T)$ is shown and the radius of suboptimality is optimally controlled by selecting $\epsilon = 1/\sqrt{T}$ [40]. The bound $\mathcal{O}(T^{3/4})$ on the constraint violation aggregation is comparable to existing results for synchronized multi-agent online learning [36] and stochastic approximation [19].

Moreover, tt is possible to extend the result in (27) to show that the convergence results of the average objective function error sequence also holds for the running average of primal iterates $\hat{x}_T := \frac{1}{T}\sum_{t=1}^{T} \mathbf{x}_t$ as in [19, corollary 1]. However, obtaining such a result for the constraint violation in (28) is not straightforward due to the presence of delayed primal variables. Subsequently, we turn to studying the empirical performance of Algorithm 1 for developing intelligent interference management in communication systems.

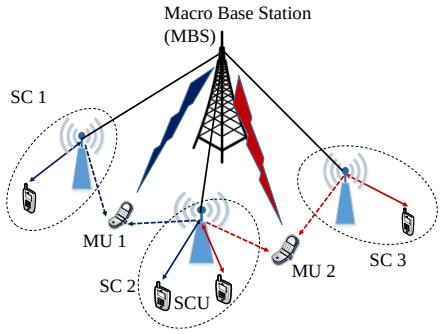


Fig. 2: Heterogeneous cellular network with one MBS, two MUs, and three SCBSs with each serving one, two, and one SCU, respectively.

V. Interference Management through Pricing

The rising number of cellular users has fueled the increase in infrastructure spending by cellular operators towards better serving densely populated areas. In order to circumvent the near-absolute limits on spectrum availability, the current and future generations rely heavily on frequency reuse via small cells and associated interference management techniques [41]–[43]. This work builds upon the pricing-based interference management framework proposed in [43]. We consider heterogeneous networks with multiple autonomous small cell users. Under heavy load situations, the macro base station (MBS) may assign the same operating frequency to multiple but geographically disparate small cell base stations (SCBS) and macro cell users (MU). The base station regulates the resulting cross-tier interference (from SCBS to MU) by penalizing the received interference power at the MUs. Consequently, the SCBSs coordinate among themselves and employ power control to limit their interference at the MUs. This section considers the pricing problem from the perspective of the BS that seeks to maximize its revenue.

A. Problem formulation

Consider the network depicted in Fig. 2, consisting of a MBS serving M MU users and N SCBSs [43]. Each MU is assigned a unique subchannel, indexed by $i \in \{1, \ldots, M\}$. At times of high traffic, the BS also allows the SCBSs to use these M subchannels, so that the n-th SCBS may serve $K_n \leq M$ SCUs. In other words, at each time slot, a particular subchannel is used by MU i and a non-empty set of SCBSs $\mathcal{N}_i \subset \{1, \ldots, N\}$. Denoting the channel gains between the n-th SCBS and i-th MU by g_{ni}, it follows that the total interference at the MU i is given by $I_i := \sum_{n\in\mathcal{N}_i} g_{ni} p^i_n$ where p^i_n is the transmit power of SCBS n while using subchannel assigned to MU i. The BS regulates this cross-tier interference by imposing a penalty x^i_n on the SCBSs $n \in \mathcal{N}_i$. The total revenue generated by the BS is therefore given by

$$\sum_{i=1}^{M} \sum_{n\in\mathcal{N}_i} x^i_n g_{ni} p^i_n \tag{29}$$

which the BS seeks to maximize. The BS also adheres to the constraint that the total penalty imposed on each SCBS is within certain limit, i.e.,

$$C_{\min} \leq \sum_{i:n\in\mathcal{N}_i} x^i_n \leq C_{\max}. \tag{30}$$

The limit on the maximum and minimum penalties can also be viewed as a means for BS to be fair to all small cell operators.

The power allocation at the SCBSs is governed by their local transmission costs, denoted by c per unit transmit power, and the interference prices levied by the BS. As in [43], each SCBS solves a penalized rate minimization subproblem, resulting in the power allocation

$$p_n^i = \left((W/(c\mu_n + \nu_n x_n^i)) - (1/h_n^i)\right)_+ \tag{31}$$

for all $n \in \mathcal{N}_i$ and $1 \leq i \leq M$. Here, h_n^i is the channel gain between n-th SCBS and its scheduled user, μ_n and ν_n represent SCBS-specific parameters used to trade-off the achieved sum rate against the transmission costs and W is the bandwidth per subcarrier. Finally, the channel gains g_{ni} and h_n^i are not known in advance, so the BS seeks to solve the following stochastic optimization problem:

$$\max_{\{x_n^i\}} \sum_{i=1}^{M} \sum_{n\in\mathcal{N}_i} \mathbb{E}\left[x_n^i g_{ni} p_n^i(x_n^i, h_n^i)\right] \tag{32a}$$

$$\text{s. t.} \sum_{n\in\mathcal{N}_i} \mathbb{E}\left[g_{ni} p_n^i(x_n^i, h_n^i)\right] \leq \gamma_i \;\; 1 \leq i \leq M \tag{32b}$$

$$C_{\min} \leq \sum_{i:n\in\mathcal{N}_i} x_n^i \leq C_{\max} \;\; 1 \leq n \leq N \tag{32c}$$

Here, γ_i is the interference power margin and observe that the interference constraint is required to hold only on an average, while the limits on the interference penalties are imposed at every time slot. It is remarked that the similar pricing based interference management scheme is considered is [43] but with the assumption that the distribution of the random variables is known. For this work, we omit such assumption and propose an online solution to stochastic optimization problem in (32).

Algorithm 2 Online interference management through pricing

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

1: **for** $t = 1, 2, \ldots, T$ **do**

2: **loop in parallel** for all MU and SCBS user

3: Send dual vars. λ_t^m to nbhd.

4: Observe the delayed primal and dual (sub)-gradients

5: Update the price $x_{n,t+1}^i$ at SCBS n as

$$x_{n,t+1}^i = \mathcal{P}_{\mathcal{X}_n}\Big[x_{n,t}^i + \epsilon\Big(g_{ni,[t]}\left[\frac{W(c\mu_n + \nu_n\lambda_t^i)}{(c\mu_n + \nu_n x_{n,[t]}^i)^2} - \frac{1}{h_{n,[t]}^i}\right]\cdot\mathbf{1}(x_{n,[t]}^i)\Big)\Big]$$

6: Update dual variables at each MU i [cf. (14)]

$$\lambda_{t+1}^i = \left[(1+\delta\epsilon^2)\lambda_t^i - \epsilon\left(\gamma_i - \sum_{n\in\mathcal{N}_i} g_{ni,[t]}\left(\frac{W}{c\mu_n + \nu_n x_{n,[t]}^i} - \frac{1}{h_{n,[t]}^i}\right)_+\right)\right]_+ .$$

7: **end loop**

8: **end for**

B. Solution using stochastic saddle point algorithm

It can be seen that the stochastic optimization problem formulated in (32) is of the form required in (3) with $\mathcal{X}$ capturing the constraint in (32c). Since the random variables h_n^i and $g_{ni,t}$ have bounded moments, the assumptions in Section IV can be readily verified. Further, the saddle point method may be applied for solving (32). To do so, we use the preceding definition of the power function p_n^i defined as in (31), and associating dual variable λ^i with the i-th constraint in (32), the stochastic augmented Lagrangian is given by

$$\hat{\mathcal{L}}_t(\mathbf{x}, \boldsymbol{\lambda}) = \sum_{i=1}^{M}\sum_{n\in\mathcal{N}_i} x_n^i g_{ni,t} p_n^i(x_n^i, h_{n,t}^i) + \sum_{i=1}^{M}\lambda^i[\gamma_i - \sum_{n\in\mathcal{N}_i} g_{ni,t} p_n^i(x_n^i, h_{n,t}^i)] - \frac{\delta\epsilon}{2}\|\boldsymbol{\lambda}\|^2 \tag{33}$$

where $\mathbf{x}$ collects the variables $\{x_n^i\}_{i=1,n\in n_i}^M$ and $\boldsymbol{\lambda}$ collects the dual variables $\{\lambda^i\}_{i=1}^M$.

The asynchronous saddle point method for pricing-based interference management in wireless systems then takes the form of Algorithm 2 with the modified projection defined as

$$P_{\mathcal{X}_n}(\mathbf{u}) := \min_y \|\mathbf{y} - \mathbf{u}\| \quad \text{s. t.} \;\; C_{\min} \leq \langle \mathbf{1}, \mathbf{y}\rangle \leq C_{\max}. \tag{34}$$

In order to get the primal and dual updates of step 5 and 6, note that the subgradient of the Lagrangian in (33) with respect to primal variables is given by

$$\partial_{x_n^i}\mathcal{L}_t(x_n^i, \lambda^i) := g_{ni,t}\left[\frac{W(c\mu_n + \nu_n\lambda^i)}{(c\mu_n + \nu_n x_n^i)^2} - \frac{1}{h_{n,t}^i}\right]\cdot\mathbf{1}(x_n^i)$$

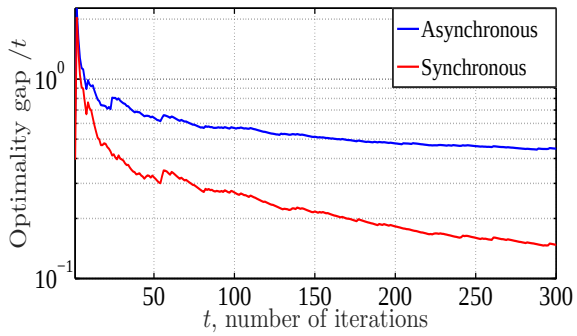
and the gradient of the Lagrangian with respect to the dual variable $\boldsymbol{\lambda}^i$ is given by

$$\nabla_{\lambda^i}\mathcal{L}_t(x_n^i, \lambda^i) = \gamma_i - \sum_{n\in\mathcal{N}_i} g_{ni,t}\left(\frac{W}{c\mu_n + \nu_n x_{n,t}^i} - \frac{1}{h_{n,t}^i}\right)_+ - \delta\epsilon\lambda^i.$$

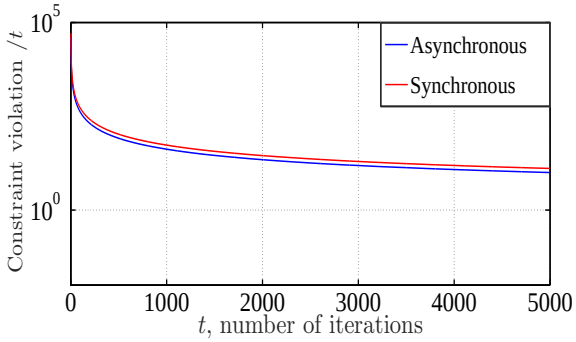
Observe here that the implementation in Algorithm 2 allows the primal updates to be carried out in a decentralized manner. On the other hand, the base station carries out the dual updates. Consequently both, the SCBSs and the BSs may utilize old price iterates $x_{n,[t]}^i$.

For the simulation purposes, we considered a cellular network with $M = 2$ MBSs and $N = 3$ SCBSs with index $\{m1, m2\}$ and $\{s1, s2, s3\}$. The scenario considered is similar to as shown in Fig. 2, means that $\{$s1, s2$\}$ are in the neighborhood of MU $m1$ and $\{s2, s3\}$ constitutes the neighborhood of MU $m2$. The random channel gain g_{ni} and h_n^i are assumed to be exponentially distributed with mean $\mu = 3$. The minimum and maximum values $C_{\min} = 0.9$ and $C_{\max} = 20$. The other parameter values are $W = 1MHz$, $\gamma_i = -3$ dB, $\delta = 10^{-5}$, $c = 0.1$, $\mu_n = \nu_n = 1$, and $\epsilon = 0.01$. The maximum delay parameter is $\tau = 10$.

Fig. 3a shows the difference of running average of primal objective from its optimal value. It is important to note that the difference goes to zero as $t \to \infty$. The result for both synchronous and asynchronous algorithm algorithms are plotted. The optimal value to plot Fig. 3a is obtained by running the synchronous algorithm for long duration of time and utilizing the converged value as the optimal one. We observe that running the saddle point method without synchrony breaks the bottleneck associated with heterogeneous computing capabilities of different nodes, although it attains slightly slower learning than its synchronized counterpart. Fig. 3b shows the behavior of constraint violation term derived in (28) for a randomly chosen MBS. In the sample path of the empirical average constraint violation, the trend of sublinear growth of objective sub-optimality from Fig. 3a is further



(a) Average objective sub-optimality vs. iteration t



(b) Average constraint violation vs. iteration t

Fig. 3: Algorithm 2 applied to a 5G cellular network with two MBS user and three SCBSs. The y axis of first figure is $\frac{1}{t}\sum_{u=1}^{t}\mathbb{E}[F(\mathbf{x}_u)-F(\mathbf{x}^*)]$ and of second figure is $(1/t)\mathbb{E}\Big[\Big[\sum_{u=1}^{t}\Big(h^i\left(\{\mathbf{x}^j,\boldsymbol{\theta}_t^j\}_{j\in n_i'}\right)\Big)\Big]_+\Big]$. Observe that both the asynchronous and synchronous implementations attain convergence but the asynchronous method settles to a higher level of sub-optimality. Thus, we may solve decentralized online learning problems without a synchronized clock.

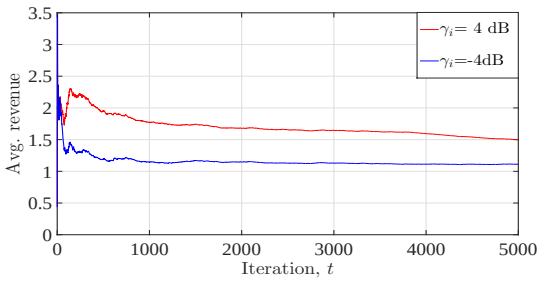


Fig. 4: Average revenue for different interference power margin.

User	Algo. 2	Naive approach
MU 1	29 dB	22 dB
MU 2	28 dB	22 dB

TABLE I: Comparison of Average SINR at MU

substantiated by convergence in expectation of the constraint violation as the iteration index t increases. We observe that the performance reduction of asynchronous operations relative to synchronous ones is smaller with respect to constraint violation as compared with primal-suboptimality, corroborating the rate analysis of Theorem 1.

Fig. 4 shows that the average value of the revenue generated by MBS converges to a higher value for higher interference power margin $\gamma_i = 4$ dB. It shows the advantage of proposed interference management scheme which exploits the higher allowed interference as a resource to generate revenue for the MBS. Next, the proposed technique is compared with a 'naive' approach in which each SCBS user transmits at unity power all the times irrespective of the allowed interference power margin and channel conditions. The result in Table I shows that the SINR achieved at both the MUs is higher for the proposed technique than that of the naive approach. It is due to the fact that proposed technique takes care of current channel conditions and therefore limits the interference caused to MU due to SCBS user transmission.

VI. Conclusion

We considered multi-agent stochastic optimization problems where the hypothesis that all agents are trying to learn common parameters may be violated, with the additional stipulation that agents do not even operate on a synchronized time-scale. To solve this problem such that agents give preference to locally observed information while incorporating the relevant information of others, 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 derived an asynchronous stochastic variant of the Arrow-Hurwicz saddle point method to solve this problem through the use of a dual-augmented Lagrangian.

We established that in expectation, 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). This result extends existing results for multi-agent convex stochastic programs with inequality constraints to asynchronous computing architectures which are important for wireless systems.

We then considered an empirical evaluation for the task of pricing-based interference management in large distributed wireless networks. For this application setting, we observe that the theoretical convergence rates translate into practice, and further that the use of asynchronous updates breaks the computational bottleneck associated with requiring devices to operate on a common clock. In particular, the asynchronous saddle point method learns slightly more slowly its synchronous counterpart while yielding a substantial complexity reduction on a real wireless application, and thus holds promise for other online multi-agent settings where synchronous operations and consensus are overly restrictive.

Appendix 0: Derivation of Generalized Algorithm

Consider the generalized the problem in (8). The stochastic augmented Lagrangian takes the form

$$\hat{\mathcal{L}}_t(\mathbf{x},\boldsymbol{\lambda}) = \sum_{i=1}^{N}\Bigg[f^i(\mathbf{x}^i,\boldsymbol{\theta}_t^i) + \Big(\langle\boldsymbol{\lambda}^i,\mathbf{h}^i\left(\{\mathbf{x}^j,\boldsymbol{\theta}_t^j\}_{j\in n_i'}\right)\rangle - \frac{\delta\epsilon}{2}\left\|\boldsymbol{\lambda}^i\right\|^2\Big)\Bigg]. \quad (35)$$

The primal update for the generalized asynchronous stochastic saddle point algorithm at each node i is given by

$$\mathbf{x}_{t+1}^i = \mathcal{P}_{\mathcal{X}}\Big[\mathbf{x}_t^i - \epsilon\Big(\nabla_{\mathbf{x}^i} f^i(\mathbf{x}_{[t]_i}^i,\boldsymbol{\theta}_{[t]_i}^i) + \sum_{k\in n_i'}\langle\boldsymbol{\lambda}_t^k,\nabla_{\mathbf{x}^i}\mathbf{h}_k\left(\{\mathbf{x}_{[t]_j}^j,\boldsymbol{\theta}_{[t]_j}^j\}_{j\in n_k'}\right)\rangle\Big)\Big]. \quad (36)$$

through applying comparable logic to Section III. Likewise, the dual variable updates at each node i is given by

$$\boldsymbol{\lambda}_{t+1}^i = \left[(1-\epsilon^2\delta)\boldsymbol{\lambda}_t^i + \epsilon\left(\mathbf{h}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\theta}_{[t]_j}^j\}_{j\in n_i'}\right)\right)\right]_+. \quad (37)$$

Before proceeding with the convergence analysis, to have a tractable derivation, let us define the compact notation for the primal and dual delayed gradient as follows

$$\nabla_{\mathbf{x}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_{[t]_j}^j\}_{j\in n_i'}\right) := \Big(\nabla_{\mathbf{x}^i} f^i(\mathbf{x}_{[t]_i}^i, \boldsymbol{\theta}_{[t]_i}^i) + \sum_{k\in n_i'}\langle \boldsymbol{\lambda}_t^k, \nabla_{\mathbf{x}^i}\mathbf{h}_k\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\theta}_{[t]_j}^j\}_{j\in n_k'}\right)\rangle\Big) \quad (38)$$

which generalizes (13) to the setting of (8) and the notation

$$\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_{[t]_j}^j\}_{j\in n_i'}\right) := \mathbf{h}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\theta}_{[t]_j}^j\}_{j\in n_i'}\right) \quad (39)$$

follows from (14). These generalized algorithm updates are used to simplify the notation in the subsequent analysis.

Appendix A: Proof of Lemma 1

Consider the squared 2-norm of the difference between the iterate $\mathbf{x}_t^i$ at time $t+1$ and an arbitrary feasible point $\mathbf{x}^i \in \mathcal{X}^N$ and use (36) to express $\mathbf{x}_{t+1}^i$ in terms of $\mathbf{x}_t^i$,

$$\|\mathbf{x}_{t+1}^i - \mathbf{x}^i\|^2 = \|\mathcal{P}_{\mathcal{X}}[\mathbf{x}_t^i - \epsilon\nabla_{\mathbf{x}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)] - \mathbf{x}^i\|^2. \quad (40)$$

where, we have utilized the compact notation defined in (38) to substitute (36) into (40). Since $\mathbf{x}^i \in \mathcal{X}$, utilizing non-expansive property of the projection operator in (40) and expanding the square

$$\begin{aligned}\|\mathbf{x}_{t+1}^i - \mathbf{x}^i\|^2 &\le \|\mathbf{x}_t^i - \epsilon\nabla_{\mathbf{x}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right) - \mathbf{x}^i\|^2 \\ &= \|\mathbf{x}_t^i - \mathbf{x}^i\|^2 \\ &\quad - 2\epsilon\langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right), (\mathbf{x}_t^i - \mathbf{x}^i)\rangle \\ &\quad + \epsilon^2\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)\|^2. \end{aligned} \quad (41)$$

We reorder terms of the above expression such that the gradient inner product is on the left-hand side and then take summation over $i \in \mathcal{V}$, yielding Take the summation over nodes $i \in \mathcal{V}$ on both sides, we get

$$\begin{aligned}&\sum_{i=1}^N \langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right), (\mathbf{x}_t^i - \mathbf{x}^i)\rangle \\ &\le \frac{1}{2\epsilon}\sum_{i=1}^N\left(\|\mathbf{x}_t^i - \mathbf{x}^i\|^2 - \|\mathbf{x}_{t+1}^i - \mathbf{x}^i\|^2\right) \\ &\quad + \frac{\epsilon}{2}\sum_{i=1}^N\left\|\nabla_{\mathbf{x}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)\right\|^2. \end{aligned} \quad (42)$$

Let us use the following notation,

$$\mathbf{x}_{[\mathbf{t}]} := \left[\mathbf{x}_{[\mathbf{t}]_1}^1; \cdots \mathbf{x}_{[\mathbf{t}]_N}^N\right], \text{ and } \boldsymbol{\lambda}_t := \left[\boldsymbol{\lambda}_t^1; \cdots; \boldsymbol{\lambda}_t^N\right]. \quad (43)$$

Utilizing this notation, we can write

$$\begin{aligned}&\langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_t - \mathbf{x})\rangle \\ &\le \frac{1}{2\epsilon}\left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2\right) + \frac{\epsilon}{2}\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2. \end{aligned} \quad (44)$$

Add and subtract $\langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), \mathbf{x}_{[\mathbf{t}]}\rangle$ to left hand side of (44) to obtain

$$\begin{aligned}&\langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x})\rangle \\ &\le \frac{1}{2\epsilon}\left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2\right) + \frac{\epsilon}{2}\left\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|^2 \\ &\quad + \langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\rangle. \end{aligned} \quad (45)$$

Observe now that since the functions $f^i(\mathbf{x}^i, \boldsymbol{\theta}_t^i)$ and $\mathbf{h}^i(\{\mathbf{x}^j, \boldsymbol{\theta}_t^j\}_{j\in n_i'})$ are convex with respect to optimization variables for any given realization of the associated random variables, therefore the stochastic Lagrangian is a convex function of $\mathbf{x}^i$ and $\mathbf{x}^j$ [cf. (9)]. Hence, from the first order convex inequality and the definition of Lagrangian in (10), it holds that

$$\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}, \boldsymbol{\lambda}_t) \le \langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x})\rangle. \quad (46)$$

Substituting the upper bound in (45) into the right hand side of (46) yields

$$\begin{aligned}&\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}, \boldsymbol{\lambda}_t) \\ &\le \frac{1}{2\epsilon}\left(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2\right) + \frac{\epsilon}{2}\left\|\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|^2 \\ &\quad + \langle\nabla_{\mathbf{x}}\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\rangle. \end{aligned} \quad (47)$$

We set this analysis aside and proceed to repeat the steps in (40)-(47) for the distance between the iterate $\boldsymbol{\lambda}_{t+1}^i$ at time $t+1$ and an arbitrary multiplier $\boldsymbol{\lambda}^i$.

$$\|\boldsymbol{\lambda}_{t+1}^i - \boldsymbol{\lambda}^i\|^2 = \|[\boldsymbol{\lambda}_t^i + \epsilon\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)]_+ - \boldsymbol{\lambda}^i\|^2 \quad (48)$$

where we have substituted (37) to express $\boldsymbol{\lambda}_{t+1}^i$ in terms of $\boldsymbol{\lambda}_t^i$. Using the non-expansive property of the projection operator in (48) and expanding the square, we obtain

$$\begin{aligned}\|\boldsymbol{\lambda}_{t+1}^i - \boldsymbol{\lambda}^i\|^2 &\le \|\boldsymbol{\lambda}_t^i + \epsilon\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right) - \boldsymbol{\lambda}^i\|^2 \\ &= \|\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}\|^2 \\ &\quad + 2\epsilon\langle\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right), (\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}^i)\rangle \\ &\quad + \epsilon^2\|\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)\|^2. \end{aligned} \quad (49)$$

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

$$\begin{aligned}&\langle\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right), (\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}^i)\rangle \\ &\ge \frac{1}{2\epsilon}\left(\|\boldsymbol{\lambda}_{t+1}^i - \boldsymbol{\lambda}^i\|^2 - \|\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}^i\|^2\right) \\ &\quad - \frac{\epsilon}{2}\|\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)\|^2. \end{aligned} \quad (50)$$

Take the summation over nodes $i \in \mathcal{V}$ so that we may write

$$\begin{aligned}&\sum_{i=1}^N \langle\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right), (\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}^i)\rangle \\ &\ge \frac{1}{2\epsilon}\sum_{i=1}^N\left(\|\boldsymbol{\lambda}_{t+1}^i - \boldsymbol{\lambda}^i\|^2 - \|\boldsymbol{\lambda}_t^i - \boldsymbol{\lambda}^i\|^2\right) \\ &\quad - \sum_{i=1}^N\frac{\epsilon}{2}\|\nabla_{\boldsymbol{\lambda}^i}\hat{\mathcal{L}}_{[\mathbf{t}]}^i\left(\{\mathbf{x}_{[t]_j}^j, \boldsymbol{\lambda}_t^j\}_{j\in n_i'}\right)\|^2. \end{aligned} \quad (51)$$

Utilizing the notation defined in (43), we write (51) as follows

$$\langle \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\boldsymbol{\lambda}_t - \boldsymbol{\lambda}) \rangle \geq \frac{1}{2\epsilon}\big(\|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2\big) - \frac{\epsilon}{2}\|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2. \quad (52)$$

Note that the online Lagrangian [cf. (9)] is a concave function of its Lagrange multipliers, which implies that instantaneous Lagrangian differences for fixed $\mathbf{x}_{[\mathbf{t}]}$ satisfy

$$\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}) \geq \nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)^T (\boldsymbol{\lambda}_t - \boldsymbol{\lambda}). \quad (53)$$

By using the lower bound stated in (52) for the right hand side of (53), we can write

$$\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t) - \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}) \geq \frac{1}{2\epsilon}\big(\|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2\big) - \frac{\epsilon}{2}\|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2. \quad (54)$$

We now turn to establishing a telescopic property of the instantaneous Lagrangian by combining the expressions in (47) and (54). To do so observe that the term $\hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)$ appears in both inequalities. Thus, subtracting the inequality (54) from those in (47) followed by reordering terms yields the required result in (24). ■

Appendix B: Proof of Lemma 2

We first consider the expression in (24), and expand the left-hand side using the definition of the augmented Lagrangian in (10). Doing so yields the following expression,

$$\begin{aligned}
&\sum_{i=1}^{N}[f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i}) - f^i(\mathbf{x}^i, \boldsymbol{\theta}^i_{[t]_i})] + \frac{\delta\epsilon}{2}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+ \sum_{i=1}^{N}\Big[\langle \boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\lambda}^j_t\}_{j\in n'_i}\right)\rangle - \langle \boldsymbol{\lambda}^i_t, \mathbf{h}^i\left(\{\mathbf{x}^j, \boldsymbol{\lambda}^j_t\}_{j\in n'_i}\right)\rangle\Big] \\
&\leq \frac{1}{2\epsilon}\big(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2\big) \\
&\quad + \frac{\epsilon}{2}\left(\left\|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|^2 + \|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2\right) \\
&\quad + \langle \nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\rangle. \quad (55)
\end{aligned}$$

Let $\mathcal{F}_t$ denotes the sigma field collecting the algorithm history which collects the information for all random quantities as $\{\boldsymbol{\theta}_u, \mathbf{x}_u, \boldsymbol{\lambda}_u\}_{u<t}$. Note that this notation is slightly different from the standard notation in literature and the maximum value u can take is $t-1$ which is for the synchronous case. Note that the conditional expectation of the following term for given sigma algebra $\mathcal{F}_{[\mathbf{t}]}$ is equal to

$$\mathbb{E}\Big[\sum_{i=1}^{N}[f^i(\mathbf{x}^i_{[t]_i}, \boldsymbol{\theta}^i_{[t]_i}) - f^i(\mathbf{x}^i, \boldsymbol{\theta}^i_{[t]_i})] \mid \mathcal{F}_{[\mathbf{t}]}\Big] = F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x}) \quad (56)$$

Taking the total expectation of (55) and utilizing the simplified expression in (56), we get

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x})] + \frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+ \sum_{i=1}^{N}\mathbb{E}\left[\langle \boldsymbol{\lambda}^i, \mathbf{h}^i\Big(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\rangle - \langle \boldsymbol{\lambda}^i_t, \mathbf{h}^i\Big(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\rangle\right] \\
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2\big) \\
&\quad + \frac{\epsilon}{2}\left(\mathbb{E}\left\|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|^2 + \mathbb{E}\|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2\right) \\
&\quad + \mathbb{E}\left[\langle \nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\rangle\right]. \quad (57)
\end{aligned}$$

Let us develop the upper bound on the term $\mathbb{E}\left[\langle \boldsymbol{\lambda}^i_t, \mathbf{h}^i\left(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right]$. Note that

$$\begin{aligned}
\mathbb{E}\left[\langle \boldsymbol{\lambda}^i_t, \mathbf{h}^i\Big(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\rangle\right] &= \mathbb{E}\left[\mathbb{E}\left[\langle \boldsymbol{\lambda}^i_t, \mathbf{h}^i\Big(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\rangle \mid \mathcal{F}_{[\mathbf{t}]}\right]\right] \\
&= \mathbb{E}\left[\langle \boldsymbol{\lambda}^i_t, \mathbb{E}\left[\left(\mathbf{h}^i\left(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\right) \mid \mathcal{F}_{[\mathbf{t}]}\right]\rangle\right] \leq 0, \quad (58)
\end{aligned}$$

where the first equality in (58) holds from the law of iterated averages. The second equality holds since $\boldsymbol{\lambda}^i_t$ is deterministic for given $\mathcal{F}_{[\mathbf{t}]}$, and the third is due to the fact that for any feasible $\{\mathbf{x}^j\}_{j\in n'_i}$, $\mathbb{E}[\mathbf{h}^i\left(\{\mathbf{x}^j, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)] \leq 0$ due to the Slater's conditions (Assumption 2), where here we use the generalized constraint definition given in (4) which subsumes (5). Now, let's use (58) in the left hand side of (57) to obtain

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x})] + \frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \quad (59) \\
&+ \sum_{i=1}^{N}\mathbb{E}\left[\langle \boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right] \\
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_t - \mathbf{x}\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t - \boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\lambda}\|^2\big) \\
&\quad + \frac{\epsilon}{2}\left(\mathbb{E}\left\|\nabla_{\boldsymbol{\lambda}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|^2 + \mathbb{E}\|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\|^2\right) + I,
\end{aligned}$$

where in (59) we have defined I on the right-hand side as

$$I := \langle \nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t), (\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\rangle. \quad (60)$$

Observe that (60) is the directional error of the stochastic gradient caused by asynchronous updates. To proceed further, an upper bound on the term I is required, which we derive next. Using Cauchy Schwartz inequality, we obtain

$$I \leq \left\|\nabla_{\mathbf{x}} \hat{\mathcal{L}}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\| \left\|(\mathbf{x}_{[\mathbf{t}]} - \mathbf{x}_t)\right\|. \quad (61)$$

Since the maximum delay is τ, we can write the difference as $\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|$ as sum of τ intermediate difference using triangular inequality as follows

$$\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\| \leq \sum_{s=t-\tau}^{t-1}\|\mathbf{x}_{s+1} - \mathbf{x}_s\| \leq \epsilon\sum_{s=t-\tau}^{t-1}\left\|\nabla_{\mathbf{x}}\mathcal{L}_{[s]}(\mathbf{x}_{[s]}, \boldsymbol{\lambda}_s)\right\|. \quad (62)$$

where the inequality in (62) follows from the primal update in (36). For brevity, let us denote $B_t := \left\|\nabla_{\mathbf{x}}\mathcal{L}_{[\mathbf{t}]}(\mathbf{x}_{[\mathbf{t}]}, \boldsymbol{\lambda}_t)\right\|$. Note that in the definition of B_t, only t index is emphasized since the dual variable involved is $\boldsymbol{\lambda}_t$. Substituting upper bound obtained in (62) into (61), simplyfying the notation using the definition of B_t and then taking expectation, we get

$$\begin{aligned}
\mathbb{E}[I] \leq \epsilon\sum_{s=t-\tau}^{t-1}\mathbb{E}[B_t B_s] &\leq \frac{\epsilon}{2}\sum_{s=t-\tau}^{t-1}\mathbb{E}\left[B_t^2 + B_s^2\right] \\
&= \frac{\epsilon}{2}\Big[\tau\cdot\mathbb{E}\left[B_t^2\right] + \sum_{s=t-\tau}^{t-1}\mathbb{E}\left[B_s^2\right]\Big]. \quad (63)
\end{aligned}$$

The second inequality in (63) follows directly using $ab \leq \frac{a^2+b^2}{2}$. The last equality of (63) is obtained by expanding the summation.

Next, applying the gradient norm square upper bound of (22) and (23) into (63), we obtain

$$\mathbb{E}\left[I\right] \leq \frac{\epsilon}{2}\Big[\Big(2\tau(N+M^2)L^2(1+\mathbb{E}\left[\|\boldsymbol{\lambda}_t\|^2\right])\Big) + \sum_{s=t-\tau}^{t-1}\Big(2(N+M^2)L^2(1+\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2])\Big)\Big]. \quad (64)$$

Rearranging and collecting the like terms, we get

$$\mathbb{E}\left[I\right] \leq \epsilon\Big[2\tau(N+M^2)L^2 + \tau(N+M^2)L^2\mathbb{E}\left[\|\boldsymbol{\lambda}_t\|^2\right] + (N+M^2)L^2\sum_{s=t-\tau}^{t-1}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big]. \quad (65)$$

Multiplying the last term of (65) by τ, we get

$$\mathbb{E}\left[I\right] \leq \epsilon\Big[2\tau(N+M^2)L^2 + \tau(N+M^2)L^2\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2] + \tau(N+M^2)L^2\sum_{s=t-\tau}^{t-1}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big] \quad (66)$$

Let us define $K_1 := (N+M^2)L^2$, expression in (66) can be expressed as

$$\mathbb{E}\left[I\right] \leq \epsilon\tau K_1\Big[2+\sum_{s=t-\tau}^{t}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big]. \quad (67)$$

Substitute the bounds developed in (22), (23) and (67) back into (59), we get

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x})] + \frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+\sum_{i=1}^{N}\mathbb{E}\left[\langle\boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right] \\
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\big) \\
&+\frac{\epsilon}{2}\Big(2M\sigma^2_{\boldsymbol{\lambda}} + 2\delta^2\epsilon^2\mathbb{E}\left[\|\boldsymbol{\lambda}_t\|^2\right] + 2(N+M^2)L^2(1+\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2])\Big) \\
&+\epsilon\tau K_1\Big[2+\sum_{s=t-\tau}^{t}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big]. \quad (68)
\end{aligned}$$

Collecting the like terms together and defining $K_2 := M\sigma^2_{\boldsymbol{\lambda}} + (N+M^2)L^2 + \tau K_1$ and $K_3 := \delta^2\epsilon^2 + (N+M^2)L^2$, we get

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x})] + \frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+\sum_{i=1}^{N}\mathbb{E}\left[\langle\boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right] \\
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\big) \\
&+\epsilon\Big[K_2 + K_3\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2] + \frac{\tau K_1}{2}\sum_{s=t-\tau}^{t}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big]. \quad (69)
\end{aligned}$$

Adding $\mathbb{E}\left[F(\mathbf{x}_{[\mathbf{t}]}) - F(\mathbf{x}_t)\right]$ to the both sides of (69) and apply Lipschitz continuity of the objective, yields

$$\begin{aligned}
&\mathbb{E}[F(\mathbf{x}_t) - F(\mathbf{x})] + \frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+\sum_{i=1}^{N}\mathbb{E}\left[\langle\boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right] \quad (70)
\end{aligned}$$

$$\begin{aligned}
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_t-\mathbf{x}\|^2 - \|\mathbf{x}_{t+1}-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_t-\boldsymbol{\lambda}\|^2 - \|\boldsymbol{\lambda}_{t+1}-\boldsymbol{\lambda}\|^2\big) \\
&+\epsilon\Big[K_2 + K_3\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2] + \frac{\tau K_1}{2}\sum_{s=t-\tau}^{t}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big] + L_f\mathbb{E}\left[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|\right].
\end{aligned}$$

Now, we proceed to analyze the resulting term $L_f\mathbb{E}\left[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|\right]$. Further note that using $[\mathbb{E}\left[X\right]]^2 \leq \mathbb{E}\left[X^2\right]$ for any random variable X, we can write

$$\mathbb{E}[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|] \leq \sqrt{\mathbb{E}\left[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|^2\right]} \leq \Big(\mathbb{E}\big[\big(\sum_{s=t-\tau}^{t-1}\|\mathbf{x}_{k+1} - \mathbf{x}_k\|\big)^2\big]\Big)^{1/2} \quad (71)$$

Inequality in (71) follows from the triangular inequality using comparable analysis to that which yields (62). Further utilizing the result $\Big(\sum_{i=1}^{U} a_i\Big)^2 \leq U\sum_{i=1}^{U} a_i^2$, we get

$$\mathbb{E}[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|] \leq \Big(\tau\sum_{s=t-\tau}^{t-1}\mathbb{E}\left[\|\mathbf{x}_{k+1} - \mathbf{x}_k\|^2\right]\Big)^{1/2}. \quad (72)$$

Now using the upper bound for single iterate different in terms of gradient as in (62), we get

$$\begin{aligned}
\mathbb{E}[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|] &\leq \Big(\tau\epsilon^2\sum_{s=t-\tau}^{t-1}\mathbb{E}[\left\|\nabla_{\mathbf{x}}\mathcal{L}_{[s]}(\mathbf{x}_{[s]}, \boldsymbol{\lambda}_s)\right\|^2]\Big)^{1/2} \\
&\leq \epsilon\Big(2\tau(N+M^2)L^2\sum_{s=t-\tau}^{t-1}[1+\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]]\Big)^{1/2}. \quad (73)
\end{aligned}$$

Inequality in (73) holds due to application of gradient norm bounds. From the standard inequality of $\sqrt{1+Z} \leq (1+Z)$ for all $Z \geq 0$, we can write

$$\mathbb{E}[\left\|\mathbf{x}_t - \mathbf{x}_{[\mathbf{t}]}\right\|] \leq 2\epsilon\tau\sqrt{K_1}\sum_{s=t-\tau}^{t-1}[1+\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]]. \quad (74)$$

where we pull 2τ out of square root because the product is either zero or grater than 1. Utilizing the upper bound of (74) for the last term in right hand side of (70) and taking the summation over $t = 1$ to T, we get

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[F(\mathbf{x}_t) - F(\mathbf{x})] + \sum_{t=1}^{T}\frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2 - \|\boldsymbol{\lambda}\|^2) \\
&+\sum_{t=1}^{T}\sum_{i=1}^{N}\mathbb{E}\left[\langle\boldsymbol{\lambda}^i, \mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j}, \boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right] \\
&\leq \frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_1-\mathbf{x}\|^2 + \|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}\|^2\big) \\
&+\epsilon\Big[TK_2 + K_3\sum_{t=1}^{T}\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2] + \frac{\tau K1}{2}\sum_{t=1}^{T}\sum_{s=t-\tau}^{t}\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]\Big] \\
&+2\epsilon\tau L_f\sqrt{K_1}\sum_{t=1}^{T}\sum_{s=t-\tau}^{t-1}[1+\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]] \quad (75)
\end{aligned}$$

In (75), 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 (76), and the fact that the resulting expression is deterministic. By assuming the dual variable is initialized as $\boldsymbol{\lambda}_1 =$

$\mathbf{0}$ and then combining the like terms together, we can upper bound the right hand side of (75), yielding

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}[F(\mathbf{x}_t)-F(\mathbf{x})]+\sum_{t=1}^{T}\frac{\delta\epsilon}{2}\mathbb{E}(\|\boldsymbol{\lambda}_t\|^2-\|\boldsymbol{\lambda}\|^2)\\
&+\sum_{t=1}^{T}\sum_{i=1}^{N}\mathbb{E}\left[\langle\boldsymbol{\lambda}^i,\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right]\\
&\leq\frac{1}{2\epsilon}\mathbb{E}\big(\|\mathbf{x}_1-\mathbf{x}\|^2+\|\boldsymbol{\lambda}\|^2\big)\\
&\quad+\epsilon TK_2+\epsilon K_3\sum_{t=1}^{T}\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2]+2\epsilon\tau TL_f\sqrt{K_1}\\
&\quad+\epsilon\tau(\frac{K_1}{2}+2L_f\sqrt{K_1})\sum_{t=1}^{T}\sum_{s=t-\tau}^{t}[\mathbb{E}[\|\boldsymbol{\lambda}_s\|^2]]. \qquad (76)
\end{aligned}$$

Note that in (76), in order to gather terms, an extra $\|\boldsymbol{\lambda}_t\|^2$ is added to the right-hand side. We upper bound the last term on the right-hand side of (76) by considering

$$\begin{aligned}
\sum_{t=1}^{T}\sum_{s=t-\tau}^{t}\|\boldsymbol{\lambda}_s\|^2=&\|\boldsymbol{\lambda}_1\|^2+(\|\boldsymbol{\lambda}_1\|^2+\|\boldsymbol{\lambda}_2\|^2)\\
&+(\|\boldsymbol{\lambda}_1\|^2+\|\boldsymbol{\lambda}_2\|^2+\|\boldsymbol{\lambda}_3\|^2)+\cdots\\
&+(\|\boldsymbol{\lambda}_{T-\tau}\|^2+\|\boldsymbol{\lambda}_{T-\tau+1}\|^2+\cdots+\|\boldsymbol{\lambda}_T\|^2). \qquad (77)
\end{aligned}$$

The relationship in (77) then simplifies to

$$\sum_{t=1}^{T}\sum_{s=t-\tau}^{t}\|\boldsymbol{\lambda}_s\|^2\leq(\tau+1)\sum_{t=1}^{T}\mathbb{E}\left[\|\boldsymbol{\lambda}_t\|^2\right]. \qquad (78)$$

Utilizing this on the right hand side of (76), we get

$$\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}[F(\mathbf{x}_t)-F(\mathbf{x})]+\sum_{i=1}^{N}\langle\boldsymbol{\lambda}^i,\sum_{t=1}^{T}\mathbf{h}^i\Big(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\rangle\\
&\quad-\Big(\frac{\delta\epsilon T}{2}+\frac{1}{2\epsilon}\Big)\|\boldsymbol{\lambda}\|^2\Big]\leq\frac{1}{2\epsilon}\|\mathbf{x}_1-\mathbf{x}\|^2+\frac{\epsilon TK}{2}\\
&\qquad+(\epsilon/2)(K_4-\delta)\sum_{t=1}^{T}\mathbb{E}[\|\boldsymbol{\lambda}_t\|^2]. \qquad (79)
\end{aligned}$$

where $K:=2K_2+4\tau L_f\sqrt{K_1}$ and $K_4:=(2K_3+(\tau+1)\tau(K_1+4L_f\sqrt{K_1}))$. Now selecting δ such that $(K_4-\delta)\leq 0$ makes the last term on the right-hand side of the preceding expression null, so that we may write

$$\begin{aligned}
&\mathbb{E}\Big[\sum_{t=1}^{T}[F(\mathbf{x}_t)-F(\mathbf{x})]+\sum_{i=1}^{N}\left[\langle\boldsymbol{\lambda}^i,\sum_{t=1}^{T}\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle\right]\\
&\quad-\Big(\frac{\delta\epsilon T}{2}+\frac{1}{2\epsilon}\Big)\|\boldsymbol{\lambda}\|^2\Big]\leq\frac{1}{2\epsilon}\|\mathbf{x}_1-\mathbf{x}\|^2+\frac{\epsilon TK}{2}. \qquad (80)
\end{aligned}$$

Observe that Lemma 2 is a special case of (80) with simplified constraint functions that only allow for pair-wise coupling of the decisions of distinct nodes (see Remark 1). ■

Appendix C: Proof of Theorem 1

At this point, we note that the left-hand side of the expression in (80), and hence (25), consists of three terms. The first is the accumulation over time of the global sub-optimality, which is a sum of all local losses at each node as defined in (2); the second is the inner product of the an arbitrary Lagrange multiplier $\boldsymbol{\lambda}$ with the time-aggregation of constraint violation; and the last 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 [36], [44]. To do so, define the *augmented* dual function $\tilde{g}(\boldsymbol{\lambda})$ using the later two terms on the left-hand side of (79)

$$\tilde{g}(\boldsymbol{\lambda})=\sum_{i=1}^{N}\langle\boldsymbol{\lambda}^i,\sum_{t=1}^{T}\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\rangle-\Big(\frac{\delta\epsilon T}{2}+\frac{1}{2\epsilon}\Big)\|\boldsymbol{\lambda}\|^2.$$

Computing the gradient and solving the resulting stationary equation over the range $\mathbb{R}^M_+$ yields

$$\tilde{\boldsymbol{\lambda}}^i=Z(\epsilon)\Big[\sum_{t=1}^{T}\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\Big]_+ \qquad (81)$$

for all $(i,z)\in\mathcal{E}$, where $Z(\epsilon):=\frac{1}{(T\delta\epsilon+1/\epsilon)}$. Substituting the selection $\boldsymbol{\lambda}^i=\tilde{\boldsymbol{\lambda}}^i$ defined by (81) into (80) results in the following expression

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T}[F(\mathbf{x}_t)-F(\mathbf{x})]+Z(\epsilon)\sum_{i=1}^{N}\left\|\Big[\sum_{t=1}^{T}\Big(\mathbf{h}^i\Big(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\Big)\Big)\Big]_+\right\|^2\right]\\
&\quad\leq\frac{1}{2\epsilon}\|\mathbf{x}_1-\mathbf{x}\|^2+\frac{\epsilon TK}{2}\leq\frac{\sqrt{T}}{2}\big(\|\mathbf{x}_1-\mathbf{x}\|^2+K\big). \qquad (82)
\end{aligned}$$

The second inequality in (82) is obtained by selecting the constant step-size $\epsilon=1/\sqrt{T}$. This result 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 (82) from both sides, and note that the resulting term is non-positive. This observation allows us to omit the constraint slack term in (82), which taken with the selection $\mathbf{x}=\mathbf{x}^*$ [cf. (3)] and pulling the expectation inside the summand, yields

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

which is as stated in (27). Now we turn to establishing a sublinear growth of the constraint violation in T, using the expression in (82). Note that from the Lipschitz continuity of the objective function, we have $|F(\mathbf{x}_t)-F(\mathbf{x}^\star)|\leq L_f\|\mathbf{x}_t-\mathbf{x}^\star\|$. An immediate consequence of this inequality is that $F(\mathbf{x}_t)-F(\mathbf{x}^\star)\geq-2L_fR$, using this in (82) yields

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{\sqrt{T}(\delta+1)}\sum_{i=1}^{N}\left\|\Big[\sum_{t=1}^{T}\Big(\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\Big)\Big]_+\right\|^2\right]\\
&\qquad\leq\frac{\sqrt{T}}{2}\big(\|\mathbf{x}_1-\mathbf{x}^*\|^2+K\big)+2TL_fR. \qquad (83)
\end{aligned}$$

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

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i=1}^{N}\left\|\Big[\sum_{t=1}^{T}\Big(\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\Big)\Big]_+\right\|^2\right] \qquad (84)\\
&\qquad\leq\Big(2\sqrt{T}(\delta+1)\Big)\Big(\frac{\sqrt{T}}{2}\big(\|\mathbf{x}_1-\mathbf{x}^*\|^2+K\big)+2TL_fR\Big).
\end{aligned}$$

We complete the proof by noting that the square of the network-in-aggregate constraint violation $\sum_{i=1}^{N}\left\|\left[\sum_{t=1}^{T}\left(\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\right)\right]_+\right\|^2$ upper bounds the square of individual proximity constraint violations since it is a sum of positive squared terms, i.e.,

$$\mathbb{E}\left[\sum_{i=1}^{N}\left\|\left[\sum_{t=1}^{T}\left(\mathbf{h}^i\left(\{\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^j_{[t]_j}\}_{j\in n'_i}\right)\right)\right]_+\right\|^2\right] \geq \mathbb{E}\left[\left[\sum_{t=1}^{T}\left(h_{ij}(\mathbf{x}^i_{[t]_i},\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^i_{[t]_i},\boldsymbol{\theta}^j_{[t]_j})-\gamma_{ij}\right)\right]_+^2\right] \tag{85}$$

where we utilized the notation defined in (5) and the inequality that the norm square of a vector is always greater than square of each element of the vector. The inequality in (85) is true for any arbitrary ij in the right hand side.

Thus the right-hand side of (85) may be used in place of the left-hand side of (84), implying that

$$\mathbb{E}\left[\left[\sum_{t=1}^{T}h_{ij}\left(\mathbf{x}^i_{[t]_i},\mathbf{x}^j_{[t]_j},\boldsymbol{\theta}^i_{[t]_i},\boldsymbol{\theta}^j_{[t]_j}\right)\right]_+^2\right] \leq\left(2\sqrt{T}(\delta+1)\right)\left(\frac{\sqrt{T}}{2}(\|\mathbf{x}_1-\mathbf{x}^*\|^2+K)+2TL_fR\right). \tag{86}$$

In order to present the results for the special case discussed in (3), compute the square root of both sides of (86) and take the summation over all $(i,j)\in\mathcal{E}$ to conclude (28). ■

REFERENCES

[1] A. S. Bedi, A. Koppel, and K. Rajawat, "Asynchronous saddle point method: Interference management through pricing," *IEEE ICC workshops 2018 (submitted)*.

[2] L. Chen, S. Low, M. Chiang, and J. Doyle, "Cross-Layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks," in *in Proc. IEEE INFOCOM*, April 2006, pp. 1–13.

[3] A. Koppel, J. Fink, G. Warnell, E. Stump, and A. Ribeiro, "Online learning for characterizing unknown environments in ground robotic vehicle models," in *Proc. of IEEE IROS*, 2016, pp. 626–633.

[4] D. Palomar and M. Chiang, "A tutorial on decomposition methods for network utility maximization," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1439–1451, Aug 2006.

[5] S. Shalev-Shwartz, "Online learning and online convex optimization," *Found. Trends Mach. Learn.*, vol. 4, no. 2, pp. 107–194, Feb. 2012.

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

[7] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[8] O. Bousquet and L. Bottou, "The tradeoffs of large scale learning," in *Adv. Neural Inf. Process. Syst.*, 2008, pp. 161–168.

[9] H. Robbins and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, no. 3, pp. 400–407, 09 1951.

[10] L. Bottou, "Online algorithms and stochastic approximations," in *Online Learning and Neural Networks*, D. Saad, Ed. Cambridge, UK: Cambridge University Press, 1998.

[11] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multiagent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, 2009.

[12] K. I. Tsianos and M. G. Rabbat, "Distributed dual averaging for convex optimization under communication delays," in *Proc. of IEEE ACC, 2012*, 2012, pp. 1067–1072.

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

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

[15] A. Koppel, F. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," *IEEE Trans. Signal Process.*, p. 15, Oct 2015.

[16] M. G. Rabbat and K. I. Tsianos, "Asynchronous decentralized optimization in heterogeneous systems," in *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014, pp. 1125–1130.

[17] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.

[18] B. Recht, C. Re, S. Wright, and F. Niu, "Hogwild: A lock-free approach to parallelizing stochastic gradient descent," in *Adv. Neural Inf. Process. Syst.*, 2011, pp. 693–701.

[19] A. Koppel, B. Sadler, and A. Ribeiro, "Proximity without consensus in online multi-agent optimization," *IEEE Trans. Signal Process.*, 2017.

[20] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "D4l: Decentralized dynamic discriminative dictionary learning," *IEEE Trans. Signal Inf. Process. over Netw.*, 2017.

[21] A. S. Bedi and K. Rajawat, "Asynchronous incremental stochastic dual descent algorithm for network resource allocation," *arXiv preprint arXiv:1702.08290*, 2017.

[22] K. Arrow, L. Hurwicz, and H. Uzawa, *Studies in linear and non-linear programming*, ser. Stanford Mathematical Studies in the Social Sciences. Stanford University Press, Stanford, Dec. 1958, vol. II.

[23] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, Jul. 1997.

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

[25] J. C. Duchi, S. Chaturapruek, and C. Ré, "Asynchronous stochastic convex optimization," *arXiv preprint arXiv:1508.00882*, 2015.

[26] K. Srivastava and A. Nedić, "Distributed asynchronous constrained stochastic optimization," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 4, pp. 772–790, 2011.

[27] A. H. Sayed and X. Zhao, "Asynchronous adaptive networks," *arXiv preprint arXiv:1511.09180*, 2015.

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

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

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

[31] 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.

[32] M. Rabbat, R. Nowak, and J. Bucklew, "Generalized consensus computation in networked systems with erasure links," in *6th IEEE SPAWC*, Jun. 5-8 2005, pp. 1088–1092.

[33] 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.

[34] E. Wei and A. Ozdaglar, "Distributed alternating direction method of multipliers," in *51st IEEE CDC*. IEEE, 2012, pp. 5445–5450.

[35] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar, *Convex Analysis and Optimization*. Athena Scientific, 2003.

[36] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: online convex optimization with long term constraints," *J. Mach. Learn. Res.*, vol. 13, no. Sep, pp. 2503–2528, 2012.

[37] A. Nedić, D. P. Bertsekas, and V. S. Borkar, "Distributed asynchronous incremental subgradient methods," *Studies in Computational Mathematics*, vol. 8, no. C, pp. 381–407, 2001.

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

[39] 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.

[40] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. of the ICML*, vol. 20, no. 2, Washington DC, USA, Aug. 21-24 2003, pp. 928–936.

[41] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, 2005.

[42] D. López-Pérez, A. Valcarce, G. De La Roche, and J. Zhang, "OFDMA femtocells: A roadmap on interference avoidance," *IEEE Commun. Mag.*, vol. 47, no. 9, 2009.

[43] S. Bu, F. R. Yu, and H. Yanikomeroglu, "Interference-aware energy-efficient resource allocation for ofdma-based heterogeneous networks with incomplete channel state information," *IEEE Trans. Veh. Technol.*, vol. 64, no. 3, pp. 1036–1050, 2015.

[44] R. Jenatton, J. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints," in *Proc. of the ICML*, 2016, pp. 402–411.