Conservative Multi-agent Online Kernel Learning in Heterogeneous Networks

Hrusikesha Pradhan*, Amrit Singh Bedi†, Alec Koppel†, and Ketan Rajawat*

***Abstract*—In this paper, we consider a decentralized heterogeneous network of agents that seek to cooperatively estimate a function belonging to reproducing kernel Hilbert space (RKHS), motivated by supervised learning. Since the network is assumed to be heterogeneous, imposing consensus constraints fails to allow agents to retain their local differences in their estimates, which motivates the use of nonlinear constraints to incentivize coordination. We develop a decentralized functional stochastic variant of the primal-dual (Arrow-Hurwicz) method. We use a dynamically regularized Lagrangian relaxation to derive convergence in expectation of this scheme to a optimality gap of $\mathcal{O}(\sqrt{T})$ with zero constraint violation when used with a constant step-size, where T is the final iteration index. Further, the decentralized algorithms performance is validated on real world data set for estimating temperature and salinity from depth measurements of the Gulf of Mexico.**

I. Introduction

In distributed online learning, agents in a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ seek to estimate a function which minimizes an objective which is common to the whole network while only exchanging information with neighbors and using their locally available data stream. The global objective is the sum of local convex objectives across different nodes of the network, which depends on distinct locally observed data. This framework arises networked control [1], [2], distributed signal processing [3], federated machine learning [4], robotics [5], communications [6], and social media [7].

Distributed methods typically operate by introducing a local copy of the global variable, and impose consensus, i.e., all the nodes are pushed to a common decision. Then, they may be solved with distributed versions of gradient descent [2] or methods based upon Lagrangian relaxation [8]–[10]. By contrast, we focus on the case where agents' local information may be distinct, but still cooperation is advantageous, which arises in settings with spatial correlation [6]. This behavior may be mathematically modeled by a nonlinear convex local proximity constraint, for which primal-dual methods are appropriate – see [11].

Our focus is on settings where data arrives sequentially [12], which makes evaluation of gradients unavailable. Instead, gradients must be estimated using samples. This may be accomplished with stochastic approximation [13], which substitutes full gradients with stochastic gradients, and therefore is able to operate on samples as they arrive. Stochastic variants of primal-dual methods have been successfully applied to online multi-agent settings when one seeks to estimate vector-valued parameters in [14], [15].

*Department of Elect. Engg., Indian Institute of Technology Kanpur, Kanpur, Uttar Pradesh, India. Email: {hpradhan, ketan}@iitk.ac.in.

†US Army Research Laboratory, Adelphi, MD, USA. Email: {amrit0714}@gmail.com, {alec.e.koppel.civ}@mail.mil.

However, the restriction to vector parameters precludes state of the art techniques based upon universal function approximators, i.e., deep neural networks [16] and kernel methods [17]. We focus on the later, motivated by the fact they are able to gain universality while preserving convexity, via the "kernel trick" [18]. More specifically, the Representer theorem implies that nonlinearity may be subsumed into a weighted basis expansions over kernel evaluations over N training points [19]. Unfortunately for online settings, N grows to infinity. To address this memory issue, hard thresholding projections of the function into the subspaces defined by the past training samples may be employed to carefully balance complexity and convergence [20].

Online learning with constraints has been considered extensively in recent years. In the case of nonlinear statistical models, primal-dual schemes have been employed for linear [21] and nonlinear constraints [22], whose convergence has been established in terms of expected sub-optimality $\mathcal{O}(\sqrt{T})$ and constraint violation $\mathcal{O}(T^{3/4})$ in terms of a final iteration index T results. This matches primal-only results for unconstrained online optimization [23]. In this work, we improve upon these results, preserving the *same sub-optimality rates* while ensuring strict feasibility. More specifically, we tighten existing primal sub-optimality rates while ensuring strict feasibility [24] through the use of a dynamic regularization of the dual update, generalizing [25] to nonparametric settings.

The main contributions of this work then are to formulate a nonparametric proximity-constrained optimization problem for multi-agent networks, akin to [26]. Relative to [26], our main contributions are to (i) propose a conservative version of the algorithm in [26] which improves the convergence of time-aggregation of the constraint violation from $\mathcal{O}(T^{3/4})$ to zero without hampering the optimality of $\mathcal{O}(\sqrt{T})$; (ii) we provide a bound on the number of points (model order) used to learn the model; and (iii) implement the algorithm on a real data set obtained from World Oceanic Database [27].

II. Problem Formulation

We consider an expected risk minimization problem where we want to solve the constrained optimization problem over function f minimizing the global loss averaged over observed data samples. We define a symmetric, connected, and directed network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = V$ nodes and $|\mathcal{E}| = M$ edges and denote as $n_i := \{j : (i, j) \in \mathcal{E}\}$ the neighborhood of agent i. Each agent $i \in \mathcal{V}$ observes a local data sequence as realizations $(\mathbf{x}_{i,t}, y_{i,t})$ at time instant t from random pair $(\mathbf{x}_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, where $\mathbf{x}_i \in \mathbb{R}^p$ and $y_i \in \mathbb{R}$, and tries to learn local optimum function f_i. This multi-agent setting can be realized by associating to each node i a convex loss functional $\ell_i : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ that measures the merit of the estimator $f_i(\mathbf{x}_i)$ evaluated at feature vector $\mathbf{x}_i$.

The goal for each node is the minimization of the common global loss which is the sum of local objectives. Motivated by the differences in the local data we introduce a convex local proximity constraint $h_{ij}(f_i, f_j)$ with tolerance parameter $\gamma_{ij} \geq 0$ and thereby avoiding making common decisions. We also assume that $h_{ij}(f_i, f_j) = h_{ji}(f_j, f_i)$. Thus, now we write our main problem as in [26] as follows

$$\mathbf{f}^\star = \underset{\{f_i\}\in\mathcal{H}}{\operatorname{argmin}} \sum_{i\in\mathcal{V}} \left(\mathbb{E}_{\mathbf{x}_i,y_i}\Big[\ell_i(f_i(\mathbf{x}_i), y_i)\Big] + \frac{\lambda}{2}\|f_i\|^2_{\mathcal{H}} \right)$$
$$\text{s.t. } \mathbb{E}_{\mathbf{x}_i}\Big[h_{ij}(f_i(\mathbf{x}_i), f_j(\mathbf{x}_i))\Big] \leq \gamma_{ij}, \text{ for all} j \in n_i, \quad (1)$$

where $\mathcal{H}$ represents RKHS and the expectation is over the data samples $(\mathbf{x}_i, y_i)$. For compactness, denote $\mathcal{H}^V$ as functions aggregated over the network whose elements are stacked functions $\mathbf{f}(\cdot) = [f_1(\cdot); \cdots; f_V(\cdot)]$ that yield vectors of length V when evaluated at local random vectors $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}_1); \cdots; f_V(\mathbf{x}_V)] \in \mathbb{R}^V$. Moreover, denote $\mathbf{x} = [\mathbf{x}_1; \cdots; \mathbf{x}_V] \in \mathcal{X}^V \subset \mathbb{R}^{Vp}$ and $\mathbf{y} = [y_1; \cdots y_V] \in \mathbb{R}^V$.

The optimization problem in (1), is intractable in general, since it defines a variational inference problem integrated over the unknown joint distribution $\mathbb{P}(\mathbf{x}, y)$. However, when $\mathcal{H}$ is equipped with a *reproducing kernel* $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ (see [18]), a function estimation problem of the form (1) may be reduced to a parametric form via the Representer Theorem [19]. Thus, we restrict the Hilbert space to be RKHS, i.e., for $\tilde{f} : \mathcal{X} \to \mathbb{R}$ in $\mathcal{H}$, it holds that (i) $\langle \tilde{f}, \kappa(\mathbf{x}_i, \cdot)\rangle_{\mathcal{H}} = \tilde{f}(\mathbf{x}_i)$, (ii) $\mathcal{H} = \text{span}\{\kappa(\mathbf{x}_i, \cdot)\}$ for all $\mathbf{x}_i \in \mathcal{X}$ and $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the Hilbert inner product for $\mathcal{H}$. Further assume that the kernel is positive semidefinite, i.e., $\kappa(\mathbf{x}_i, \mathbf{x}_i') \geq 0$ for all $\mathbf{x}_i, \mathbf{x}_i' \in \mathcal{X}$. The optimal function in (1) can be represented in terms of linear combinations of kernels evaluated at training points via the Representer theorem generalized to constrained muti-agent settings given in [26, Corollary 1].

We now present the conservative version of (1) to vanish the long term constraint violation. To achieve this, we add ν to the constraint in (1) to reformulate the problem as follows

$$\mathbf{f}^\star_\nu = \underset{\{f_i\}\in\mathcal{H}}{\operatorname{argmin}} \sum_{i\in\mathcal{V}} \left(\mathbb{E}_{\mathbf{x}_i,y_i}[\ell_i(f_i(\mathbf{x}_i), y_i)] + \frac{\lambda}{2}\|f_i\|^2_{\mathcal{H}} \right)$$
$$\text{s.t. } \mathbb{E}_{\mathbf{x}_i}[h_{ij}(f_i(\mathbf{x}_i), f_j(\mathbf{x}_i))] + \nu \leq \gamma_{ij}, \text{ for all } j \in n_i, \quad (2)$$

By the modification in (2), we consider a more stricter constraint than (1). This strict constraint makes the resulting sequence of solutions satisfy the long term constraints $\sum_{t=1}^{T} \mathbb{E}[h_{ij}(f_i(\mathbf{x}_{i,t}), f_j(\mathbf{x}_{i,t}))] \leq \gamma_{ij}$ for all (i, j) even though the original constraint of (1) is violated on many instances. This reformulation is able to achieve the optimality gap of $\mathcal{O}(\sqrt{T})$ while satisfying the constraints in the long run. This is an interesting result as in the process of satisfying the constraints on the long run, we don't compromise on the optimality gap as opposed to $\mathcal{O}(T^{3/4})$ which we achieve when $\nu = 0$ achieved in [24].

III. Decentralized Algorithm

This section develops an online and decentralized algorithm for solving (2) when $\{f_i\}_{i\in\mathcal{V}}$ are elements of a RKHS. To do so, we consider the stochastic estimate of the augmented Lagrangian of (2) evaluated at sample $(\mathbf{x}_{i,t}, y_{i,t})$,

$$\hat{\mathcal{L}}_t(\mathbf{f}, \boldsymbol{\mu}) \coloneqq \sum_{i\in\mathcal{V}} \Bigg[\ell_i(f_i(\mathbf{x}_{i,t}), y_{i,t}) + \frac{\lambda}{2}\|f_i\|^2_{\mathcal{H}} \quad (3)$$
$$+ \sum_{j\in n_i} \left\{ \mu_{ij}\Big(h_{ij}(f_i(\mathbf{x}_{i,t}), f_j(\mathbf{x}_{i,t})) + \nu - \gamma_{ij}\Big) - \frac{\delta\eta}{2}\mu_{ij}^2 \right\} \Bigg],$$

where $\delta, \eta > 0$ for the dual variable μ_{ij}. The regularization term in (3) controls the violation of non-negative constraints on the dual variable over time t. The alternating primal/dual stochastic descent/ascent update steps corresponding to (3) are:

$$f_{i,t+1} = f_{i,t}(1 - \eta\lambda) - \eta\Big[\ell_i'(f_{i,t}(\mathbf{x}_{i,t}), y_{i,t}) \quad (4)$$
$$+ \sum_{j\in n_i} \mu_{ij,t} h_{ij}'(f_{i,t}(\mathbf{x}_{i,t}), f_{j,t}(\mathbf{x}_{i,t}))\Big]\kappa(\mathbf{x}_{i,t}, \cdot).$$
$$\mu_{ij,t+1} = \Big[\mu_{ij,t}(1 - \delta\eta^2) + \eta\Big(h_{ij}(f_{i,t}(\mathbf{x}_{i,t}), f_{j,t}(\mathbf{x}_{i,t})) - \gamma_{ij} + \nu\Big)\Big]_+.$$

The detailed calculation of the updates in (4) are provided in [28]. Further, the V-fold stacking of $f_{i,t+1}$ across all nodes at instant t is denoted as $\mathbf{f}_{t+1}$ and similarly the M-fold stacking of $\mu_{ij,t+1}$ across all the edges is denoted as $\boldsymbol{\mu}_t$. The sequence of $(\mathbf{f}_t, \boldsymbol{\mu}_t)$ is initialized by $\mathbf{f}_0 = 0 \in \mathcal{H}^V$ and $\boldsymbol{\mu} = 0 \in \mathbb{R}_+^M$. Using the Representer theorem given in [26], $f_{i,t}$ can be written in terms of kernels evaluated at past observations as

$$f_{i,t}(\mathbf{x}) = \sum_{n=1}^{t-1} w_{i,n}\kappa(\mathbf{x}_{i,n}, \mathbf{x}) = \mathbf{w}_{i,t}^T \boldsymbol{\kappa}_{\mathbf{X}_{i,t}}(\mathbf{x}) \,. \quad (5)$$

We define $\mathbf{X}_{i,t} = [\mathbf{x}_{i,1}, \ldots, \mathbf{x}_{i,t-1}] \in \mathbb{R}^{p\times(t-1)}$, $\boldsymbol{\kappa}_{\mathbf{X}_{i,t}}(\cdot) = [\kappa(\mathbf{x}_{i,1}, \cdot), \ \ldots \ , \kappa(\mathbf{x}_{i,t-1}, \cdot)]^T$, and $\mathbf{w}_{i,t} = [w_{i,1}, \ \ldots \ w_{i,t-1}]^T \in \mathbb{R}^{t-1}$ on the right-hand side of (5). Combining the update in (4) along with the kernel expansion in (5), implies that the primal functional stochastic descent step in $\mathcal{H}^V$ results in the following V parallel parametric updates on both kernel dictionaries $\mathbf{X}_i$ and $\mathbf{w}_i$:

$$\mathbf{X}_{i,t+1} = [\mathbf{X}_{i,t}, \ \ \mathbf{x}_{i,t}] \,, \quad (6)$$
$$[\mathbf{w}_{i,t+1}]_u = \begin{cases} (1 - \eta\lambda)[\mathbf{w}_{i,t}]_u & \text{for } 0 \leq u \leq t-1 \\ -\eta\Big(\ell_i'(f_{i,t}(\mathbf{x}_{i,t}), y_{i,t}) & \\ + \sum_{j\in n_i} \mu_{ij,t} h_{ij}'(f_{i,t}(\mathbf{x}_{i,t}), f_{j,t}(\mathbf{x}_{i,t}))\Big), & \text{for } u = t \end{cases}$$

From (6) we note that each time one more column gets added to the columns in $\mathbf{X}_{i,t}$, an instance of the curse of kernelization. We define the number of data points, i.e., the number of columns of $\mathbf{X}_{i,t}$ at time t as the *model order* $(M_{i,t})$. We note that for the update in (4), the model order is $t - 1$ and it grows unbounded with iteration index t. To alleviate the aforementioned memory bottleneck, we project the function sequence (4) onto a lower dimensional subspace such that $\mathcal{H}_{\mathbf{D}} \subseteq \mathcal{H}$, where $\mathcal{H}_{\mathbf{D}}$ is represented by a dictionary $\mathbf{D} = [\mathbf{d}_1, \ \ldots, \ \mathbf{d}_M] \in \mathbb{R}^{p\times M}$. Similarly, we define dictionaries $\mathbf{D}_{i,t}$ and subspace $\mathcal{H}_{\mathbf{D}_{i,t}}$ for each agent at time t. Now, we project the update in (4) to a lower dimensional subspace $\mathcal{H}_{\mathbf{D}_{i,t+1}} = \text{span}\{\kappa(\mathbf{d}_{i,n}, \cdot)\}_{n=1}^{M_{t+1}}$ as $f_{i,t+1} = \mathcal{P}_{\mathcal{H}_{\mathbf{D}_{i,t+1}}}\Big[\tilde{f}_{i,t+1}\Big]$, where we define the projection operator $\mathcal{P}_{\mathbf{D}_{i,t+1}}$ onto subspace $\mathcal{H}_{\mathbf{D}_{i,t+1}} \subset \mathcal{H}$. The function $\tilde{f}_{i,t+1}$ is the un-projected functional update given in (4). The

Algorithm 1 Conservative Learning with Kernels

Require: $\{\mathbf{x}_t, \mathbf{y}_t\}_{t=0,1,2,\ldots}$, ϵ, η, ν and δ
  initialize $f_{i,0}(\cdot) = 0, \mathbf{D}_{i,0} = [], \mathbf{w}_0 = []$, i.e. initial dictionary, coefficients are empty for each $i \in \mathcal{V}$
  for $t = 0, 1, 2, \ldots$ **do**
    loop in parallel for agent $i \in \mathcal{V}$
      Observe local training example realization $(\mathbf{x}_{i,t}, y_{i,t})$
      Send $\mathbf{x}_{i,t}$ to neighbors $j \in n_i$ and receive $f_{j,t}(\mathbf{x}_{i,t})$
      Receive $\mathbf{x}_{j,t}$ from neighbors $j \in n_i$ and send $f_{i,t}(\mathbf{x}_{j,t})$
    Update primal variable $\tilde{f}_{i,t+1}$ using (4)
    Update dual variables for $j \in n_i$ using (4)
    Update params: $\tilde{\mathbf{D}}_{i,t+1} = [\mathbf{D}_{i,t}, \ \mathbf{x}_{i,t}]$, $\tilde{\mathbf{w}}_{i,t+1}$ [cf. (7)]
      Greedily compress function using matching pursuit
    $(f_{i,t+1}, \mathbf{D}_{i,t+1}, \mathbf{w}_{i,t+1}) = \textbf{KOMP}(\tilde{f}_{i,t+1}, \tilde{\mathbf{D}}_{i,t+1}, \tilde{\mathbf{w}}_{i,t+1}, \epsilon)$
    end loop
  end for

update $f_{i,t+1} = \mathcal{P}_{\mathcal{H}_{\mathbf{D}_{i,t+1}}}\Big[\tilde{f}_{i,t+1}\Big]$, is equivalent to finding coefficients of kernels evaluated at points of fixed dictionary $\mathbf{D}_{i,t+1} \in \mathbb{R}^{p \times M_{t+1}}$. To notice this, we first form the original dictionary $\tilde{\mathbf{D}}_{i,t+1}$ and coefficient vector $\tilde{\mathbf{w}}_{i,t+1}$

$$\tilde{\mathbf{D}}_{i,t+1} = [\mathbf{D}_{i,t}, \ \mathbf{x}_{i,t}] \,, \tag{7}$$

$$[\tilde{\mathbf{w}}_{i,t+1}]_u = \begin{cases} (1-\eta\lambda)[\mathbf{w}_{i,t}]_u, & \text{for } 0 \le u \le t-1 \\ -\eta\Big(\ell_i'(f_{i,t}(\mathbf{x}_{i,t}), y_{i,t}) \\ \quad +\sum_{j \in n_i} \mu_{ij,t} h_{ij}'(f_{i,t}(\mathbf{x}_{i,t}), f_{j,t}(\mathbf{x}_{i,t}))\Big), & \text{for } u = t \end{cases}$$

from the un-projected functional update step in (4). Now, similar to [20], we handle the uncontrollable memory growth in the dictionary by using a variant of kernel orthogonal matching pursuit algorithm (KOMP). This algorithm projects the function sequences onto subspaces defined by a compressed dictionary $\tilde{\mathbf{D}}_{i,t+1}$ for ith node at instant t for a compression budget ϵ and obtains dictionary $\mathbf{D}_{i,t+1}$ as given in Algorithm 1. The number of points in $\mathbf{D}_{i,t}$ is denoted by $M_{i,t}$.

IV. Convergence Results

Now with the Algorithm 1 stated, we now present the convergence in expectation result of the optimality gap and constraint violation for a constant step size rule. Before doing so, we state the key assumptions required to analyze the convergence behavior and present the main result.

[A1] The feature space $\mathcal{X} \subset \mathbb{R}^p$ and target domain $\mathcal{Y} \subset \mathbb{R}$ are compact, and the kernel map may be bounded as $\sup_{\mathbf{x} \in \mathcal{X}} \sqrt{\kappa(\mathbf{x}, \mathbf{x})} = X < \infty$.

[A2] The local losses $\ell_i(f_i(\mathbf{x}), y)$ are convex and differentiable with respect to the first (scalar) argument $f_i(\mathbf{x})$ on $\mathbb{R}$ for all $\mathbf{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$. Moreover, the instantaneous losses $\ell_i : \mathcal{H} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ are C_i-Lipschitz continuous

$$|\ell_i(z, y) - \ell_i(z', y)| \le C_i |z - z'| \ \text{ for all } z \text{ with } y \text{ fixed}, \tag{8}$$

where $C := \max_i C_i$ is the largest modulus of continuity.

[A3] The constraint functions h_{ij} for all $(i, j) \in \mathcal{E}$ are all uniformly L_h-Lipschitz continuous in its first (scalar) argument; i.e., there exist constant L_h, such that

$$|h_{ij}(z, y) - h_{ij}(z', y)| \le L_h |z - z'|, \ \text{ for all } \ z, z' \in \mathbb{R}, \tag{9}$$

and is also convex w.r.t the first argument z.

[A4] There exists $\mathbf{f}^\dagger$ such that for all $(i, j) \in \mathcal{E}$, we have $h_{ij}(f_i^\dagger, f_j^\dagger) + \xi \le \gamma_{ij}$, for some $\xi > 0$, which implies that the constraint is strictly satisfied.

[A5] The functions $f_{i,t+1}$ output from KOMP have Hilbert norm bounded by $R_{\mathcal{B}} \le \infty$. Also, the optimal $f_i^\star$ lies in the ball $\mathcal{B}$ with radius $R_{\mathcal{B}}$.

Now using Assumption IV, we bound the gap between optimal of problem (1) and (2) and is presented as Lemma 1.

Lemma 1 *Under Assumption **A2**, **A4** and **A5**, for $0 \le \nu \le \xi/2$, it holds that:*

$$S(\mathbf{f}_\nu^*) - S(f^*) \le \frac{4VR_{\mathcal{B}}(CX + \lambda R_{\mathcal{B}})}{\xi}\nu \tag{10}$$

where $S(\mathbf{f}) := \sum_{i \in \mathcal{V}} \big[\ell_i(f_i(\mathbf{x}_{i,t}), y_{i,t}) + \frac{\lambda}{2}\|f_i\|_{\mathcal{H}}^2\big]$.

The proof of Lemma 1 is provided in [28]. The importance of Lemma 1 is that it establishes the fact that the gap between the solutions of the problem (1) and (2) is $\mathcal{O}(\nu)$. For the completeness of Theorem 1, we define constant $\zeta \ge R_{\mathcal{B}}^2 + (1+\delta)\Big[2 + 2\Big(\frac{4VR_{\mathcal{B}}(CX+\lambda R_{\mathcal{B}})}{\xi}\Big)^2\Big] + 4VPR_{\mathcal{B}} + 8VX^2C^2 + 4V\lambda^2 \cdot R_{\mathcal{B}}^2 + 2MK_1 + 2ML_h^2X^2 \cdot R_{\mathcal{B}}^2$, which will be used for defining ν below in Theorem 1 and the detailed derivation of ζ is shown in the proof of Theorem 1 in [28].

Theorem 1 *Suppose Assumptions **A1**-**A5** hold and $\nu = \zeta T^{-1/2}$, and $(\mathbf{f}_t, \boldsymbol{\mu}_t)$ be the sequence of Algorithm 1 under constant step-size $\eta = T^{-1/2}$ and compression budget ϵ .*

i The time-aggregation of the expected sub-optimality grows sub-linearly with horizon T as

$$\sum_{t=1}^{T} \mathbb{E}[S(\mathbf{f}_t) - S(\mathbf{f}^\star)] \le \mathcal{O}(\sqrt{T}) + V\epsilon T^{3/2}\Big(2R_{\mathcal{B}} + \epsilon\Big).$$

where $\mathbf{f}^\star$ is defined by (1).

ii Moreover, the aggregate constraint is met, i.e.,

$$\sum_{t=1}^{T} \mathbb{E}\Big[h_{ij}(f_i(\mathbf{x}_{i,t}), f_j(\mathbf{x}_{i,t})) - \gamma_{ij}\Big] \le 0, \ \text{ for all } (i, j) \in \mathcal{E}.$$

Theorem 1 bounds the optimality gap in terms of number of iterations (T) and compression budget, ϵ. Specifically if we consider ϵ in terms of step size, $\eta = T^{-1/2}$ and write $\epsilon = P\eta^2$, then the optimality gap stated in Theorem 1 can be written as

$$\sum_{t=1}^{T} \mathbb{E}[S(\mathbf{f}_t) - S(\mathbf{f}^\star)] \le \mathcal{O}(\sqrt{T}). \tag{11}$$

Thus, Theorem 1 establishes that the time-aggregation of the sub-optimality sequence associated with Algorithm 1 when run with fixed step-size $\eta = T^{-1/2}$ is bounded by constant less than T, i.e., $\mathcal{O}(\sqrt{T})$. Moreover for a particular choice of ν, we also satisfy the constraints on average. The proof of Theorem 1 is presented in [28, Appendix A].

The radius of these error neighborhood may be reduced by appropriately adjusting the step-size $\eta = 1/\sqrt{T}$. Here, we achieve the same primal suboptimality of $\mathcal{O}(\sqrt{T})$ but

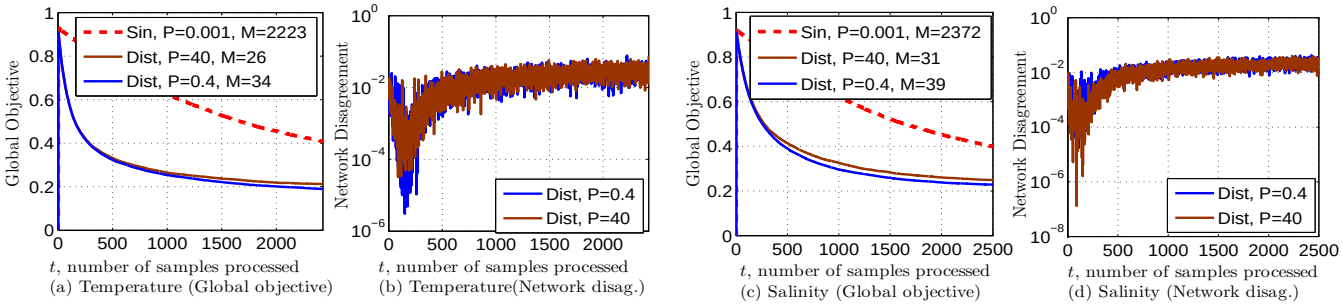

Fig. 1: In Fig. 1(a) we show the convergence of global objective $\sum_i \mathbf{E}_{\mathbf{x}_i,y_i}[\ell(f_{i,t}(\mathbf{x}_i),y_i)]$ versus the number of samples processed and in Fig. 1(b) we plot the constraint violation for temperature. Similarly in Fig. 1(c) and (d) we show the convergence of global objective and constraint violation for salinity field. We choose $\lambda=\delta=10^{-5}$, and $\eta=0.01$ for measurement of both temperature and salinity field. The M in legends denotes the final model order.

with zero constraint violation as compared to $\mathcal{O}(T^{3/4})$ in [24] and $\mathcal{O}(\sqrt{T})$ in [25]. In [24], the authors achieved zero average constraint violation but with inferior optimality gap of $\mathcal{O}(T^{3/4})$. In this paper we also generalized the result of sublinear convergence of objective error sequence in [22] to multi-agent settings with proximity constraints but with aggregated average constraint violation going to zero.

Now, we establish a upper bound on the memory order of function $f_{i,t}$ obtained from Algorithm 1 in terms of the absolute value of the dual variable update. Thus, using Assumption **A2** and **A3** we present the model order theorem.

Theorem 2 *Let $f_{i,t}$ denote the function sequence of agent i at tth instant generated from Algorithm 1 with dictionary $\mathbf{D}_{i,t}$. Denote $M_{i,t}$ as the model order representing the number of dictionary elements in $\mathbf{D}_{i,t}$. Then with constant step size $\eta=1/\sqrt{T}$ and compression budget ϵ, for a Lipschitz Mercer kernel κ on a compact set $\mathcal{X}\subset\mathbb{R}^p$, there exists a constant β such that for any training set $\{\mathbf{x}_{i,t}\}_{t=1}^{\infty}$, $M_{i,t}$ satisfies*

$$M_{i,t}\leq\beta\left(\frac{\eta R_M}{\epsilon}\right)^p, \tag{12}$$

where $R_M=C+L_hMR_{i,t}$ and $R_{i,t}=\max_{j\in n_i}|\mu_{ij,t}|$. The total model order, M_t of the network consisting of N nodes can be written as

$$M_t\leq N\max_i M_{i,t}. \tag{13}$$

The result in Theorem 2 presents a quantitative aspect of the compression algorithm running in Algorithm 1, and presents a bound on the model order of an individual gent i and as well of the multi-agent network. The proof of Theorem 2 is presented in [28, Appendix B].

V. Numerical Results

We run Algorithm 1 on the data obtained from multiple underwater sensors in the Gulf of Mexico from World Oceanic Database [27] for estimating temperature and salinity parameters at different locations with varying depths, during the winter time-period. The readings of the climatological fields are obtained for a particular latitude and longitude at standard depths starting from 0 meters to 5000 meters. The latitude and longitude specifies the node (sensor) location. The experiment is carried out considering 50 nodes, where edges are determined by measuring the distance between two nodes, and drawing an edge to a particular node if its distance is less than 1000 kilometers away. The proximity parameter γ_{ij} between two nodes is obtained by evaluating $\exp(-\text{dist}(i,j)/1000)$, where $\text{dist}(i,j)$ measures the distance between two nodes in kilometers. We solve problem (1) by minimizing the regularized quadratic loss between estimated climatological field $f_i(d_i)$ and observed climatological field y_i over function f_i using Algorithm 1. Thus we predict the statistical mean of the temperature and salinity field of different nodes at varying depths and compare it with the real data obtained from the World Oceanic database. We run the the algorithm for a constant step-size $\eta=0.01$ and regularizers λ, δ set to 10^{-5}. The bandwidth parameter of the Gaussian kernel is set at $\sigma=50$, and parsimony constant is fixed at two values, $P=0.4$ and 40. The parsimony constant for learning a single function f (centralized approach:all data at a single location) for all the data obtained across different locations is set at $P=0.001$ to ensure a fair comparison of comparably sized models between the decentralized and centralized methods.

Fig. 1 (a) demonstrates that the prediction error for test cases reduces with increasing samples for temperature field. In Fig. 1 (a), the centralized function (denoted as "Sin") for all the data obtained across different locations gives a poor fit resulting in high loss relative to the distributed case (denoted as "Dist"). Fig. 1 (a) shows the final model order for the distributed case with $P=0.4$ for all the 50 nodes is 34 times 50, i.e., 1700, less compared to the complexity of the centralized case, i.e., 2223. Thus, the centralized function fails to fit the data with higher number of points in the dictionary compared to the distributed case. Moreover, increasing the parsimony constant from 0.4 to 40, i.e., increasing the error tolerance, reduces model complexity at the cost of degrading model fitness. In Fig. 1 (b), we display the constraint violation over time (for decentralized case only). Note that it approaches null across choice of parsimony constant. Moreover, Fig. 1 (a) demonstrates that the complexity of nodes' functions does not grow, but rather remains stable even in the face of experimental oceanic data, settling to $M_{i,t}=34$ (resulting from KOMP compression algorithm) compared to sample size 2430. Similar performance is also observed for salinity field data in Fig. 1 (c) and (d). The complexity for salinity data also settles to around 39, which is orders of magnitude smaller than the 2500 sample size.

REFERENCES

[1] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," vol. 49, no. 9, pp. 1520–1533, 2004.

[2] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," vol. 54, no. 1, pp. 48–61, 2009.

[3] A. H. Sayed and C. G. Lopes, "Distributed processing over adaptive networks," in *9th Int. Sym. on Signal Process. and Its Appl.* IEEE, 2007, pp. 1–3.

[4] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[5] M. Schwager, P. Dames, D. Rus, and V. Kumar, "A multi-robot control policy for information gathering in the presence of unknown hazards," in *Robotics Research*. Springer, 2017, pp. 455–472.

[6] R. J. Kozick and B. M. Sadler, "Source localization with distributed sensor arrays and partial spatial coherence," vol. 52, no. 3, pp. 601–616, 2004.

[7] S.-W. Seong, J. Seo, M. Nasielski, D. Sengupta, S. Hangal, S. K. Teh, R. Chu, B. Dodson, and M. S. Lam, "Prpl: a decentralized social networking infrastructure," in *Proc. of the 1st ACM Workshop on Mob. Cloud Comp. & Services: Social Netw. and Beyond*. ACM, 2010, p. 8.

[8] A. Koppel, F. Y. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," vol. 63, no. 19, pp. 5149–5164, 2015.

[9] E. Wei and A. Ozdaglar, "Distributed alternating direction method of multipliers," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 5445–5450.

[10] S. Hosseini, A. Chapman, and M. Mesbahi, "Online distributed optimization via dual averaging," in *Decision and Control (CDC), 2013 IEEE 52nd Annual Conf. on*. IEEE, 2013, pp. 1484–1489.

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

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

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

[14] A. Koppel, F. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," p. 15, Oct 2015.

[15] A. Koppel, B. M. Sadler, and A. Ribeiro, "Proximity without consensus in online multiagent optimization," vol. 65, no. 12, pp. 3062–3077, 2017.

[16] S. Haykin, "Neural networks: A comprehensive foundation," 1994.

[17] B. Schölkopf, A. J. Smola, F. Bach *et al.*, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, 2002.

[18] K. Slavakis, P. Bouboulis, and S. Theodoridis, "Online learning in reproducing kernel hilbert spaces," *Sig. Process. Theory and Mach. Learn.*, pp. 883–987, 2013.

[19] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," *Subseries of Lect. Notes in Comput. Sci. Edited by JG Carbonell and J. Siekmann*, p. 416, 2001.

[20] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "Parsimonious online learning with kernels via sparse projections in function space," *The Journal of Mach. Learn. Research*, vol. 20, no. 1, pp. 83–126, 2019.

[21] A. Koppel, S. Paternain, C. Richard, and A. Ribeiro, "Decentralized online learning with kernels," vol. 66, no. 12, pp. 3240–3255, June 2018.

[22] A. Koppel, K. Zhang, H. Zhu, and T. Basar, "Projected stochastic primal-dual method for constrained online learning with kernels," pp. 1–1, 2019.

[23] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003, pp. 928–936.

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

[25] A. N. Madavan and S. Bose, "Subgradient methods for risk-sensitive optimization," *arXiv preprint arXiv:1908.01086*, 2019.

[26] H. Pradhan, A. S. Bedi, A. Koppel, and K. Rajawat, "Exact nonparametric decentralized online optimization," in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 643–647.

[27] T. P. Boyer, M. Biddle, M. Hamilton, A. V. Mishonov, C. Paver, D. Seidov, and M. . Zweng, "Gulf of mexico regional climatology (NCEI Accession 0123320)," *Version 1.1. NOAA National Centers for Environmental Inf.*.

[28] H. Pradhan, A. S. Bedi, A. Koppel, and K. Rajawat, "Adaptive kernel learning in heterogeneous networks," *arXiv preprint arXiv:1908.00510*, 2019.