DECENTRALIZED BUNDLE METHOD FOR NONSMOOTH CONSENSUS OPTIMIZATION

Zifeng Wang[*] *Qing Ling*[§†] *Wotao Yin*[‡]

[*] School of the Gifted Young, University of Science and Technology of China, Hefei, China
[§] School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China
[†] Department of Automation, University of Science and Technology of China, Hefei, China
[‡] Department of Mathematics, University of California, Los Angeles, USA

ABSTRACT

In this paper, we propose a decentralized bundle method (DBM) to solve a convex but nonsmooth consensus optimization problem defined over a network. At each iteration, each agent first refines a piecewise linear approximation of its local cost function by sampling a new subgradeint. Then the agent solves a quadratic program (QP) of minimizing the approximated cost function, regularized with a quadratic term that is constructed from its neighbors' iterates. A serious step test is conducted to determine whether to accept the solution of the QP as the agent's new iterate or not, and thus guarantees sufficient descent. To reduce costs of storing subgradients and solving the QP subproblems, we further apply a subgradient aggregation technique to bound the number of stored subgradients and thus the dimension of the subproblems. Numerical experiments demonstrate the superior convergence speed of the proposed decentralized bundle method over existing algorithms.

Index Terms— Decentralized consensus optimization, nonsmooth optimization, bundle method

1. INTRODUCTION

Consider a bidirectionally connected network with n agents that solves a decentralized consensus optimization problem

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}). \tag{1}$$

Here $f_i : \mathbb{R}^d \to \mathbb{R}$ is a convex but nonsmooth local cost function that is only available to agent i, and we do not assume any special structure of f_i. Our goal is to develop a decentralized first-order algorithm such that all the agents obtain an optimal consensual solution to (1), while every agent is only allowed to communicate with its neighbors.

A general nonsmooth optimization problem, whose cost function is without any special structure, is challenging even in the centralized setting. Most subgradient-based methods suffer from slow convergence. To develop an efficient decentralized algorithm that only evaluates cost functions and their subgradients, we resort to the bundle method, which uses multiple subgradients from previous iterations and is practically much faster than other subgradient-based methods. However, the original bundle method is centralized and is not implementable over a decentralized network. Therefore, we make simple yet necessary changes and develop a novel decentralized bundle method (DBM) to solve (1).

1.1. Related works

Existing algorithms for solving (1) include distributed subgradient method (DSM) [9], distributed dual averaging (DDA) [10], decentralized alternating direction method of multipliers (DADMM) [11], and PG-EXTRA [12]. At each iteration of DSM, each agent calculates the weighted average of its local iterate and its neighbors', and then performs a local subgradient step [9]. In DDA, each agent maintains a dual variable, which is updated by adding a new subgradient to the weighted average of its local dual variable and its neighbors'. The new iterate is computed by the projection of the dual variable on a predefined proximal function. To obtain an exact consensual solution, both DSM and DDA have to use diminishing stepsizes such that their convergence rates are unfavorable. DADMM rewrites (1) to a constrained form and solves it in the primal-dual domain [11]. Though DADMM has fast and exact convergence to the optimal solution, at each iteration, each agent must minimize its local cost function regularized by a quadratic proximal term, which is often time-consuming. PG-EXTRA is designed for the case that every local cost function is the summation of a smooth term and a nonsmooth but proximable term [12]. At each iteration, each agent performs an EXTRA step [13] on the smooth term, and then computes a proximal mapping on the nonsmooth term. PG-EXTRA is simple, fast and exact, but is unable to handle the setting that all the local cost functions are general nonsmooth.

The bundle method is a celebrated approach to solving centralized nonsmoooth optimization problems [2]. The algorithm iteratively generates a piecewise linear approximation of the cost function, called as the cutting plane model from historic subgradient samples, and then minimizes the cutting plane model plus a quadratic regularization term to find the next iterate. To address the issue that the model size increases linearly with the number of the historic subgradients, the subgradient aggregation strategy proposes to select and store only a limited number of affinely independent subgradients [8]. It is theoretically proved that under mild conditions, the bundle method has a cluster point which is optimal [3]. Variants to improve the bundle method include the one that tunes the weight of the quadratic regularization term so as to utilize second-order information of the cost function [4], and the one that replaces the linear cutting plane model by a quadratic model [5].

The power of the bundle method to handle nonsmooth problems has not been explored in the decentralized setting. A relevant work is the bundle-based decomposition algorithm that minimizes a separable cost function with linear constraints [6]. The constrained primal problem is dualized to a possibly nonsmooth dual form, which is solved by the classic bundle method. This algorithm is naturally implementable in a master-slave computing network, other than the decentralized network considered here [7].

1.2. Our contribution and paper organization

This paper proposes a decentralized bundle method (DBM) to solve decentralized consensus optimization problems where the cost func-

978-1-5090-5990-4/17/$31.00 ©2017 IEEE

tions are general nonsmooth. To address the challenges of decentralized computation, we introduce novel cutting plane models and serious step tests, which include not only local information, but also neighboring iterates. Comparing with DSM and DDA, DBM has faster convergence according to numerical experiments. Comparing with DADMM, at every iteration of DSM, every agent only needs to solve a limited-size quadratic program (QP), other than an often complicated optimization problem. Since DBM does not require any special structure of the cost functions, it is able to solve problems that PG-EXTRA cannot handle.

2. ALGORITHM DEVELOPMENT

2.1. Centralized Bundle Method Revisited

We begin with introducing the centralized bundle method that minimizes a convex but possibly nonsmooth function $f : \mathbb{R}^d \to \mathbb{R}$. The algorithm generates two sequences of variables, an iterate sequence $\{\mathbf{x}^k\}$ and a sample sequence $\{\mathbf{y}^k\}$, as well as a sequence of functions $\{\hat{f}^k\}$ containing piecewise linear approximations of f. At the k-th iteration, the bundle method has the following steps.

Step 1. Sample a subgradient $\mathbf{s}^k \in \partial f(\mathbf{y}^k)$ of the cost function f at the current sampling point $\mathbf{y}^k$, and then update the piecewise linear approximation as

$$\hat{f}^k(\mathbf{y}) = \max\{\hat{f}^{k-1}(\mathbf{y}), f(\mathbf{y}^k) + \left\langle \mathbf{s}^k, \mathbf{y} - \mathbf{y}^k \right\rangle\}. \tag{2}$$

Observe that $\hat{f}^k$ is a lower bound for f. The bound is tight at $\{\mathbf{y}^t, t = 0, 1, \cdots, k\}$, and gets tighter when k increases.

Step 2. Obtain the next sampling point by solving

$$\mathbf{y}^{k+1} \in \arg\min_{\mathbf{y}\in\mathbb{R}^d} \hat{f}^k(\mathbf{y}) + \frac{\mu}{2}||\mathbf{y} - \mathbf{x}^k||^2, \tag{3}$$

where $\mu > 0$ is a constant parameter. The cost function of (3) is the summation of the piecewise linear approximation and a quadratic proximal term defined at the current iterate $\mathbf{x}^k$. Since $\hat{f}^k$ can be unbounded (to $-\infty$) even if f is lower bounded, we use proximal minimization of $\hat{f}^k$, instead of directly minimizing it. The computational complexity of solving the quadratic program (QP) (3) is determined by the structure of the piecewise linear approximation $\hat{f}^k$, which is tightly connected with k, the number of historic sampling points.

Step 3. Define

$$\delta^k := f(\mathbf{x}^k) - \left[\hat{f}^k(\mathbf{y}^{k+1}) + \frac{\mu}{2}||\mathbf{y}^{k+1} - \mathbf{x}^k||^2\right]. \tag{4}$$

If δ^k is smaller than a threshold $\bar{\delta}$, the algorithm is terminated and outputs $\mathbf{x}^k$. Otherwise, the algorithm makes the following test

$$f(\mathbf{x}^k) - f(\mathbf{y}^{k+1}) \geq m\delta^k, \tag{5}$$

where $m \in (0, 1)$ is a constant. If (5) holds, the algorithm performs a serious step and the iterate $\mathbf{x}^{k+1} = \mathbf{y}^{k+1}$. Otherwise, the algorithm performs a null step and the iterate stays at $\mathbf{x}^{k+1} = \mathbf{x}^k$.

To see the implications of the stopping criterion $\delta^k < \bar{\delta}$, observe

$$\begin{aligned}\delta^k =& f(\mathbf{x}^k) - \left[\hat{f}^k(\mathbf{y}^{k+1}) + \frac{\mu}{2}||\mathbf{y}^{k+1} - \mathbf{x}^k||^2\right] \\ \geq& f(\mathbf{x}^k) - \left[\hat{f}^k(\mathbf{x}^k) + \frac{\mu}{2}||\mathbf{x}^k - \mathbf{x}^k||^2\right] \geq 0.\end{aligned} \tag{6}$$

Here the first inequality is because that $\mathbf{y}^{k+1}$ minimizes $\hat{f}^k(\mathbf{y}) + (\mu/2)||\mathbf{y}-\mathbf{x}^k||^2$, and the second inequality comes from that $\hat{f}^k(\mathbf{x}^k)$ is a lower bound of $f(\mathbf{x}^k)$. Thus, the necessary condition for $\delta^k = 0$ is that $\mathbf{x}^k$ has already been a minimizer of the approximation $\hat{f}^k$ (also the minimizer of its proximal mapping) and that the approximation $\hat{f}^k$ is exact at $\mathbf{x}^k$; namely, $\mathbf{x}^k$ is a minimizer of f. Thus, δ^k is a metric to characterize the distance from the current iterate to the optimal solution. The stopping criterion $\delta^k < \bar{\delta}$ guarantees that x^k is close enough to the optimal solution that we are looking for.

If $\delta^k \geq \bar{\delta}$, the proximal mapping of $\hat{f}^k$ denoted as $\mathbf{y}^{k+1}$ does not necessarily make sufficient progress in minimizing f. For example, somewhat surprisingly, even if $\mathbf{x}^k$ is already a minimizer of f, it is not necessarily the proximal mapping of $\hat{f}^k$ if $\hat{f}^k$ is not a good approximation to f at x^k. Thus, the serious step test (5) directly compares f at $\mathbf{y}^{k+1}$ and $\mathbf{x}^k$. The threshold is selected so that $\mathbf{y}^{k+1}$ makes sufficient descent in f compared to it is at $\mathbf{x}^k$. If the test succeeds, we set $\mathbf{x}^{k+1} = \mathbf{y}^{k+1}$ and no longer need $\mathbf{x}^k$. But even if the test fails, $\mathbf{y}^{k+1}$ still helps improve the bounding quality of the piecewise linear approximation by inserting another linear function into $\hat{f}^{k+1}$ that equals f at $\mathbf{y}^{k+1}$.

2.2. Decentralized bundle method (DBM)

Now we move to the decentralized setting where each agent i has access to a local nonsmooth function f_i. The network is modeled as an undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of agents and $\mathcal{E}$ is the set of edges. Agent i's neighbor set is denoted as $\mathcal{N}_i = \{j|(i,j) \in \mathcal{E}\}$. Slightly different to its centralized counterpart, for each agent i, DBM generates three sequences of variables, an iterate sequence $\{\mathbf{x}_i^k\}$, an sample sequence $\{\mathbf{y}_i^k\}$ and a dual sequence $\{\mathbf{p}_i^k\}$, as well as one sequence of functions $\{\hat{f}_i^k\}$ containing piecewise linear approximations of f_i. At the k-th iteration, agent i works as follows.

Step 1. Sample a subgradient $\mathbf{s}_i^k \in \partial f_i(\mathbf{y}_i^k)$ of the cost function f_i at the current sampling point $\mathbf{y}_i^k$, and then update the piecewise linear approximation as

$$\hat{f}_i^k(\mathbf{y}_i) = \max\{\hat{f}_i^{k-1}(\mathbf{y}_i), f_i(\mathbf{y}_i^k) + \left\langle \mathbf{s}_i^k, \mathbf{y}_i - \mathbf{y}_i^k \right\rangle\}. \tag{7}$$

This step is exactly the same as the one in the centralized algorithm.

Step 2. Obtain the next sampling point by solving

$$\mathbf{y}_i^{k+1} \in \arg\min_{\mathbf{y}_i\in\mathbb{R}^d} \hat{f}_i^k(\mathbf{y}_i) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2. \tag{8}$$

where $\mu_i > 0$ is a constant parameter, $\mathbf{p}_i^k$ is a dual variable, and

$$\mathbf{z}_i^k = \sum_{j\in\mathcal{N}_i\cup\{i\}} w_{ij}\mathbf{x}_j^k \tag{9}$$

is the weighted sum of agent i's neighboring iterates with $\mathbf{W} = [w_{ij}]$ being the weight matrix that we will discuss below. The dual variable $\mathbf{p}_i^k$ is updated from

$$\mathbf{p}_i^k = \mathbf{p}_i^{k-1} + \mu_i(\mathbf{x}_i^k - \mathbf{z}_i^k). \tag{10}$$

Observe that (8) is different to (3) in two aspects. First, agent i essentially handles the approximated local Lagrangian function $\hat{f}_i^k(\mathbf{y}_i) + \langle \mathbf{p}_i^k, \mathbf{y}_i \rangle$, other than $\hat{f}_i^k(\mathbf{y}_i)$ itself. Since solely minimizing $\hat{f}_i^k(\mathbf{y}_i)$ often biases from the optimal consensual solution, the dual variable term is appended to compromise among the agents. Second, the proximal point $\mathbf{x}_i^k$ is replaced by $\mathbf{z}_i^k$. This is also for combining neighboring iterates so as to guide the algorithm to the optimum.

Similar to DSM, DDA and PG-EXTRA, DBM can use a symmetric weight matrix $\mathbf{W}$ following the Laplacian-based constant

Algorithm 1 Decentralized Bundle Method (DBM) at Agent i

```
Initialize constants $\bar{\delta} > 0$, $m \in (0,1)$, $\mu_i > 0$, $\{w_{ij}\}$, as well as variables $\mathbf{x}_i^0$ and $\mathbf{p}_i^0$. Define function $\hat{f}_i^{-1}(\mathbf{y}_i) = f_i(\mathbf{x}_i^0)$.
for $k = 0, 1, \ldots$ do
    Compute the auxiliary variable $\mathbf{z}_i^k$ by (9).
    Update the dual variable $\mathbf{p}_i^k$ by (10).
    Pick $\mathbf{s}_i^k \in \partial f_i(\mathbf{y}_i^k)$, form $\hat{f}_i^k(\mathbf{y})$ by (7), solve $\mathbf{y}_i^{k+1}$ by (8).
    If $\delta_i^k$ in (12) is less than $\bar{\delta}$, STOP and RETURN $\mathbf{x}_i^k$.
    if (13) holds, then
        Run the serious step $\mathbf{x}_i^{k+1} = \mathbf{y}_i^{k+1}$.
    else
        Run the null step $\mathbf{x}_i^{k+1} = \mathbf{x}_i^k$.
    end if
end for
```

edge or Metropolis-Hastings rules [1]. However, we find easy-to-implement asymmetric weight matrices, as long as $Ker(\boldsymbol{I}_n - \mathbf{W}) = span(\mathbf{1})$ also perform well in practice. For example, we can use

$$w_{ij} = \frac{1}{2} \text{ if } j = i; \quad \frac{1}{2|\mathcal{N}_i|} \text{ if } j \in \mathcal{N}_i; \quad 0 \text{ otherwise}. \tag{11}$$

Step 3. Define

$$\begin{aligned}\delta_i^k =& f_i(\mathbf{x}_i^k) + \left\langle \mathbf{p}_i^k, \mathbf{x}_i^k \right\rangle + \frac{\mu_i}{2}||\mathbf{x}_i^k - \mathbf{z}_i^k||^2 \\ &- \left[\hat{f}_i(\mathbf{y}_i^{k+1}) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i^{k+1} \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i^{k+1} - \mathbf{z}_i^k||^2 \right].\end{aligned} \tag{12}$$

If δ_i^k is smaller than $\bar{\delta}$, agent i terminates the algorithm and outputs $\mathbf{x}_i^k$. Observe that this stopping criterion includes the iterates of agent i's neighbors. If $\delta_i^k \geq \bar{\delta}$, agent i makes the following test

$$f_i(\mathbf{x}_i^k) + \left\langle \mathbf{p}_i^k, \mathbf{x}_i^k \right\rangle - \left[f_i(\mathbf{y}_i^{k+1}) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i^{k+1} \right\rangle \right] \geq m\delta_i^k, \tag{13}$$

where $m \in (0,1)$ is a constant. If the inequality (13) holds such that $\mathbf{y}_i^{k+1}$ brings sufficient descent on the local Lagrangian function $f_i(\mathbf{y}_i) + \langle \mathbf{p}_i^k, \mathbf{y}_i \rangle$, then agent i performs a serious step $\mathbf{x}_i^{k+1} = \mathbf{y}_i^{k+1}$. Otherwise, agent i performs a null step: $\mathbf{x}_i^{k+1} = \mathbf{x}_i^k$.

DBM run by agent i is outlined in Algorithm 1. When there is only one agent, it reduces to the centralized bundle method.

2.3. Stopping Criterion

To understand the stopping criterion, rewrite (1) to

$$\min_{\{x_i \in \mathbb{R}^d\}} \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x}_i), \quad \text{s.t. } \mathbf{x}_i = \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij}\mathbf{x}_j, \ \forall\, i. \tag{14}$$

The equivalence between (1) and (14) when the network is connected and the weight matrix $\mathbf{W}$ satisfies $Ker(\boldsymbol{I}_n - \mathbf{W}) = span(\mathbf{1})$. The augmented Lagrangian of (14) is

$$\begin{aligned}L_{\{\mu_i\}}(\{\mathbf{x}_i\}, \{\mathbf{p}_i\}) =& \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x}_i) + \sum_{i=1}^n \langle \mathbf{p}_i, \mathbf{x}_i - \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij}\mathbf{x}_j \rangle \\ &+ \sum_{i=1}^n \frac{\mu_i}{2} ||\mathbf{x}_i - \sum_{j \in \mathcal{N}_i \cup \{i\}} w_{ij}\mathbf{x}_j||^2,\end{aligned} \tag{15}$$

If the stopping criterion $\delta_i^k < \bar{\delta}$ is met at agent i, we have

$$\begin{aligned}& f_i(\mathbf{x}_i^k) + \left\langle \mathbf{p}_i^k, \mathbf{x}_i^k \right\rangle + \frac{\mu_i}{2}||\mathbf{x}_i^k - \mathbf{z}_i^k||^2 \\ <& \bar{\delta} + \left[\hat{f}_i(\mathbf{y}_i^{k+1}) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i^{k+1} \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i^{k+1} - \mathbf{z}_i^k||^2 \right] \\ =& \bar{\delta} + \min_{\mathbf{y}_i} \{ \hat{f}_i^k(\mathbf{y}_i) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \} \\ \leq& \bar{\delta} + \min_{\mathbf{y}_i} \{ f_i^k(\mathbf{y}_i) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \}.\end{aligned} \tag{16}$$

The second line is from the definition of δ_i^k in (12), the third line is from the definition of $\mathbf{y}_i^{k+1}$ in (8), and the last line holds since $\hat{f}_i^k$ is a lower bound for f_i^k.

When the stopping criteria are met at all agents, taking the average of (16) over $i = 1, 2, \cdots, n$, we derive that

$$\begin{aligned}& \frac{1}{n} \sum_{i=1}^n \left(f_i(\mathbf{x}_i^k) + \left\langle \mathbf{p}_i^k, \mathbf{x}_i^k \right\rangle + \frac{\mu_i}{2}||\mathbf{x}_i^k - \mathbf{z}_i^k||^2 \right) \\ <& \bar{\delta} + \frac{1}{n} \sum_{i=1}^n \min_{\mathbf{y}_i} \{ f_i^k(\mathbf{y}_i) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \} \\ \leq& \bar{\delta} + \min_{\{\mathbf{y}_i\}} \{ \frac{1}{n} \sum_{i=1}^n \left(f_i^k(\mathbf{y}_i) + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \right) \}.\end{aligned} \tag{17}$$

According to the definition of the augmented Lagrangian defined by (15) and that of $\mathbf{z}_i^k$ defined by (9), (17) indicates

$$L_{\{\mu_i\}}(\{\mathbf{x}_i^k\}, \{\mathbf{p}_i^k\}) < \bar{\delta} + \min_{\mathbf{y}_i} L_{\{\mu_i\}}(\{\mathbf{y}_i\}, \{\mathbf{p}_i^k\}), \tag{18}$$

meaning that $\{\mathbf{x}_i^k\}$ are close enough to the minimizers of the augmented Lagrangian at the current dual variables $\{\mathbf{p}_i^k\}$.

3. IMPLEMENTATION ISSUES

3.1. Solution of (8)

The main computational burden of the decentralized bundle method occurs in solving $\mathbf{y}_i^{k+1}$ from (8). Below we will derive its dual form, which is a standard QP. For $t \leq k$, define the error between $f_i(\mathbf{y}_i)$ and the t-th linear piece of $\hat{f}_i^k$, which is $f(\mathbf{y}_i^k) + \langle \mathbf{s}_i^k, \mathbf{y}_i - \mathbf{y}_i^k \rangle$, evaluated at the point $\mathbf{y}_i = \mathbf{x}_i^k$ as

$$e_i^t = f_i(\mathbf{x}_i^k) - \left[f_i(\mathbf{y}_i^t) + \left\langle \mathbf{s}_i^t, \mathbf{x}_i^k - \mathbf{y}_i^t \right\rangle \right] \geq 0. \tag{19}$$

Then we can rewrite $\hat{f}_i^k(\mathbf{y}_i)$ as

$$\begin{aligned}\hat{f}_i^k(\mathbf{y}_i) =& \max_{t=0,\ldots,k} \{ f_i(\mathbf{y}_i^t) + \langle \mathbf{s}_i^t, \mathbf{y}_i - \mathbf{y}_i^t \rangle \} \\ =& \max_{t=0,\ldots,k} \{ f_i(\mathbf{x}_i^k) - e_i^t - \left\langle \mathbf{s}_i^t, \mathbf{x}_i^k - \mathbf{y}_i^t \right\rangle + \langle \mathbf{s}_i^t, \mathbf{y}_i - \mathbf{y}_i^t \rangle \} \\ =& \max_{t=0,\ldots,k} \{ f_i(\mathbf{x}_i^k) - e_i^t + \left\langle \mathbf{s}_i^t, \mathbf{y}_i - \mathbf{x}_i^k \right\rangle \}.\end{aligned} \tag{20}$$

Therefore, (8) is equivalent to

$$\begin{aligned}& \min_{\mathbf{y}_i, r_i} r_i + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \\ & \text{s.t. } r_i \geq f_i(\mathbf{x}_i^k) - e_i^t + \left\langle \mathbf{s}_i^t, \mathbf{y}_i - \mathbf{x}_i^k \right\rangle, t = 0, 1, \ldots, k.\end{aligned} \tag{21}$$

Define $\alpha_i^t \in \mathbb{R}_+$ as the optimal Lagrange multiplier of (21) that corresponds to the constraint at time t and collect all the multipliers in a vector $\boldsymbol{\alpha}_i \in \mathbb{R}_+^{k+1}$. The Lagrangian function of (21) is

$$\begin{aligned}L(\mathbf{y}_i, r_i, \boldsymbol{\alpha}_i) =& \left(1 - \sum_{t=0}^k \alpha_i^t \right) r_i + \left\langle \mathbf{p}_i^k, \mathbf{y}_i \right\rangle + \frac{\mu_i}{2}||\mathbf{y}_i - \mathbf{z}_i^k||^2 \\ &+ \sum_{t=0}^k \alpha_i^t \left[f_i(\mathbf{x}_i^k) - e_i^t + \left\langle \mathbf{s}_i^t, \mathbf{y}_i - \mathbf{x}_i^k \right\rangle \right].\end{aligned} \tag{22}$$

```
Algorithm 2 Update of $J_i^{k+1}$
1: Solve (26) to obtain $\{\alpha_i^t\}_{t\in J_i^k}$.
2: Let $J_i^{k'} = \{t \in J_i^k : \alpha_i^t > 0\}$.
3: Use Wolfe's algorithm to find a set of affinely independent subgradients $\{\mathbf{s}_i^{t'}, t' \in J_i^{k'}\}$ and weights $\{\beta_i^{t'}, t' \in J_i^{k'}\}$ such that
       $\sum_{t'\in J_i^{k'}} \beta_i^{t'} \mathbf{s}_i^{t'} = \sum_{t\in J_i^k} \alpha_i^t \mathbf{s}_i^t$.
4: Update $J_i^{k+1} = \{t' \in J_i^{k'} : \beta_i^{t'} > 0\} \cup \{k+1\}$.
```

Denote $(\mathbf{y}_i^{k+1}, r_i^k)$ as the optimal primal solution of (21). According to the KKT condition, for the optimal dual variable $\boldsymbol{\alpha}_i$, we have both $\partial L(\mathbf{y}_i^{k+1}, r_i^k, \boldsymbol{\alpha}_i)/\partial r_i^k = 0$ and $\partial L(\mathbf{y}_i^{k+1}, r_k^k, \boldsymbol{\alpha}_i)/\partial \mathbf{y}_i^{k+1} = \mathbf{0}$. Thus, we have

$$\sum_{t=0}^{k} \alpha_i^t = 1 \quad \text{and} \quad \alpha_i^t \geq 0, \tag{23}$$

$$\mathbf{y}_i^{k+1} = \mathbf{z}_i^k - \frac{1}{\mu_i}\left(\sum_{t=0}^{k} \alpha_i^t \mathbf{s}_i^t + \mathbf{p}_i^k\right). \tag{24}$$

Since $(\mathbf{y}_i^{k+1}, r_i^k)$ is the optimal primal solution, the optimal dual variable $\boldsymbol{\alpha}_i$ is the minimizer of the Lagrangian $L(\mathbf{y}_i^{k+1}, r_i^k, \boldsymbol{\alpha}_i)$ under the condition (23). Using this fact and substituting (24) into the Lagrangian, we know that $\boldsymbol{\alpha}_i$ is the minimizer of

$$\min_{\boldsymbol{\alpha}_i \in \Delta^{k+1}} \frac{1}{2\mu_i} \| \sum_{t=0}^{k} \alpha_i^t \mathbf{s}_i^t + \mathbf{p}_i^k + \mu_i(\mathbf{x}_i^k - \mathbf{z}_i^k) \|^2 + \sum_{t=0}^{k} \alpha_i^t e_i^t, \tag{25}$$

where $\Delta^{k+1} = \{\boldsymbol{\alpha}_i \in \mathbb{R}_+^{k+1} | \sum_{t=0}^{k} \alpha_i^t = 1\}$ is a simplex. Observe that (25) is a standard QP and can be solved with various toolboxes, given that $k+1$, the dimension of $\boldsymbol{\alpha}_i$, is limited. After solving (25), the optimal solution $\mathbf{y}_i^{k+1}$ of (8) is obtained by substituting the optimal $\boldsymbol{\alpha}_i$ into (24).

3.2. Subgradient Aggregation

In the decentralized bundle method, the problem scale of (25) grows with the number of iterations. Meanwhile, it requires to save all the previous subgradients $\{\mathbf{s}_i^t\}$ and linearization errors $\{e_i^t\}$ for $t = 0, 1, \cdots, k$. To address these issues, we adopt the subgradient aggregation technique that has been proved to be efficient in the centralized bundle method [8]. The basic idea of the subgradient aggregation technique is that most of the subgradients and linearization errors are redundant when the number of sampling points is large enough. Thus, instead of solving (25), we consider

$$\min_{\boldsymbol{\alpha}_i \in \Delta^{|J_i^k|}} \frac{1}{2\mu_i} \| \sum_{t\in J_i^k} \alpha_i^t \mathbf{s}_i^t + \mathbf{p}_i^k + \mu_i(\mathbf{x}_i^k - \mathbf{z}_i^k) \|^2 + \sum_{t\in J_i^k} \alpha_i^t e_i^t, \tag{26}$$

where $J_i^k \subset \{0, 1, \cdots, k\}$ and $\Delta^{|J_i^k|}$ is the simplex defined within a $|J_i^k|$-dimensional space, with $|J_i^k| \leq d + 2$.

The update of J_i^{k+1} from J_k^k is given by Algorithm 2. First, solve (26) to obtain $\{\alpha_i^t\}_{t\in J_i^k}$. Second, throw away those $\alpha_i^t = 0$ and construct $J_i^{k'} = \{t \in J_i^k : \alpha_i^t > 0\}$. Third, find a set of affinely independent subgradients $\{\mathbf{s}_i^{t'}, t' \in J_i^{k'}\}$ and the corresponding weights $\{\beta_i^{t'}, t' \in J_i^{k'}\}$ such that $\sum_{t'\in J_i^{k'}} \beta_i^{t'} \mathbf{s}_i^{t'} = \sum_{t\in J_i^k} \alpha_i^t \mathbf{s}_i^t$. This can be done by using Wolfe's algorithm to find the zero within the convex hull of $\{\mathbf{s}_i^{t'} - \sum_{t\in J_i^k} \alpha_i^t \mathbf{s}_i^t, t' \in J_i^{k'}\}$, where the solution is represented by $\{\mathbf{s}_i^{t'}, t' \in J_i^{k'}\}$ and $\{\beta_i^{t'}, t' \in J_i^{k'}\}$ [14]. Since the subgradients fall in the d-dimensional space, we know that $|J_i^{k'}| \leq d+1$. Finally, J_i^{k+1} is set as $\{t' \in J_i^{k'} : \beta_i^{t'} > 0\} \cup \{k+1\}$, and its cardinality is no larger than $d+2$.

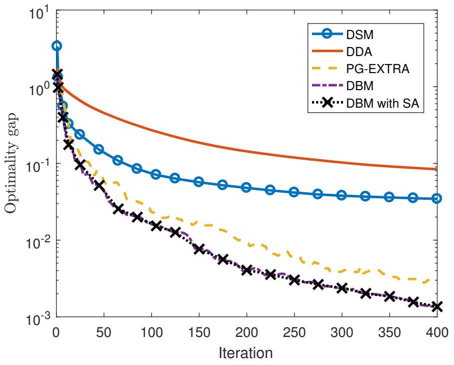

Fig. 1. Optimality gap versus the number of iterations.

With the subgradient aggregation technique, the problem scale of (25) is limited to $d + 2$, the cardinality of J_i^k. Meanwhile, agent i only needs to store historic $\mathbf{s}_i^t$ and e_i^t for all $t \in J_i^k$. This way, the computation and storage burdens become affordable.

4. NUMERICAL EXPERIMENTS

We consider the binary classification problem in the numerical experiments. Given n pairs of data $\{\mathbf{a}_i, y_i\}$, where $\mathbf{a}_i \in \mathbb{R}^d$ is a feature vector and $y_i \in \{-1, 1\}$ is the associated label. Define $f_i(x) = \max\{0, 1 - y_i \langle \mathbf{a}_i, \mathbf{x} \rangle\}$ as the nondifferentiable hinge loss function associated with i-th pair, which is possessed by agent i. To generate the experimental data, the feature vectors $\{\mathbf{a}_i\}_{i=1}^n$ are randomly sampled from a unit ball in $\mathbb{R}^d$ and the ground truth $\mathbf{x}_0 \sim N(\mathbf{0}, \mathbf{I}_d)$. The labels are generated from $\{y_i = sign(\langle \mathbf{a}_i, \mathbf{x}_0 \rangle)\}_{i=1}^n$, followed by randomly flipping 5% signs.

We compare the proposed decentralized bundle method (DBM) and that with subgradient aggregation (DBM with SA) with the existing algorithms DSM, DDA and PG-EXTRA to minimize $f(x) = (1/n)\sum_{i=1}^n f_i(x)$. We let $n = 100$ and the network is a 10×10 grid; $d = 3$. For DBM and DBM with SA, we set $\mu_i = 2$, $\bar{\delta} = 0$, $m = 0.8$, and the weight matrix $[w_{ij}]$ as in (11). DSM, DSA and PG-EXTRA use the Laplacian-based constant edge weight matrix. The diminishing stepsizes of DSM and DSA, as well as the constant stepsize of PG-EXTRA, are hand-tuned to the best.

Fig. 1 plots the optimality gap defined by $\max_i\{f(x_i^k) - f^*\}$, where $f^* = \min_x f(x)$, versus the number of iterations k. Observe that DBM and DBM with SA demonstrate much faster convergence than the three existing algorithms. The solutions of the two DBM algorithms are two-magnitude more accurate than those of DSM and DSA. PG-EXTRA also works well in this particular problem; however, as we have indicated, it is unable to handle general nonsmooth cases. The curves of DBM and DBM with SA are almost identical, showing the effectiveness of the subgradient aggregation technique.

Acknowledgement. Qing Ling is supported by China NSF 61573331 and Anhui NSF 1608085QF130. Wotao Yin is supported by NSF DMS-1720237 and ONR N000141712162.

5. REFERENCES

[1] S. Boyd, P. Diaconis, and L. Xiao, “Fastest mixing Markov chain on a graph,” SIAM Review 46.4 (2004): 667-89

[2] C. Lemarechal, “Nonsmooth optimization and descent methods,” Research Report RR-78-4, International Institute of Applied Systems Analysis (Laxenburg, Austria, 1977).

[3] K.C. Kiwiel, *Methods of Descent for Nondifferentiable Optimization*, Vol. 1133. Springer, 2006.

[4] K.C. Kiwiel, “Proximity control in bundle methods for convex nondifferentiable minimization,” Mathematical programming 46.1 (1990): 105-122.

[5] L. Lukšan, and J. Vlček. “A bundle-Newton method for nonsmooth unconstrained minimization,” Mathematical Programming 83.1-3 (1998): 373-391.

[6] S. Robinson, “Bundle-based decomposition: Description and preliminary results,” System Modelling and Optimization. Springer Berlin Heidelberg, 1986. 751-756.

[7] D. Medhi, “Parallel bundle-based decomposition for large-scale structured mathematical programming problems,” Annals of Operations Research 22.1 (1990): 101-127.

[8] K.C. Kiwiel, “An aggregate subgradient method for nonsmooth convex minimization,” Mathematical Programming 27.3 (1983): 320-341.

[9] A. Nedic, and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” IEEE Transactions on Automatic Control 54.1 (2009): 48-61.

[10] J. Duchi, A. Agarwal, and M. Wainwright, “Dual averaging for distributed optimization: Convergence analysis and network scaling,” IEEE Transactions on Automatic control 57.3 (2012): 592-606.

[11] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, “On the linear convergence of the ADMM in decentralized consensus optimization,” IEEE Transactions on Signal Processing 62.7 (2014): 1750-1761.

[12] W. Shi, Q. Ling, G. Wu and W. Yin, “A proximal gradient algorithm for decentralized composite optimization,” IEEE Transactions on Signal Processing 63.22 (2015): 6013-6023.

[13] W. Shi, Q. Ling, G. Wu and W. Yin, “Extra: An exact first-order algorithm for decentralized consensus optimization,” SIAM Journal on Optimization 25.2 (2015): 944-966.

[14] P. Wolfe, “Finding the nearest point in a polytope,” Mathematical Programming 11.1 (1976): 128-149.