ANZIAM J. 56(2014), 160–178
doi:10.1017/S1446181114000273

DISTRIBUTED PROXIMAL-GRADIENT METHOD FOR CONVEX OPTIMIZATION WITH INEQUALITY CONSTRAINTS

JUEYOU LI[1,2], CHANGZHI WU[3], ZHIYOU WU✉[1], QIANG LONG[4] and XIANGYU WANG[3,5]

(Received 17 September, 2013; revised 26 May, 2014; first published online 18 November 2014)

Abstract

We consider a distributed optimization problem over a multi-agent network, in which the sum of several local convex objective functions is minimized subject to global convex inequality constraints. We first transform the constrained optimization problem to an unconstrained one, using the exact penalty function method. Our transformed problem has a smaller number of variables and a simpler structure than the existing distributed primal–dual subgradient methods for constrained distributed optimization problems. Using the special structure of this problem, we then propose a distributed proximal-gradient algorithm over a time-changing connectivity network, and establish a convergence rate depending on the number of iterations, the network topology and the number of agents. Although the transformed problem is nonsmooth by nature, our method can still achieve a convergence rate, $O(1/k)$, after k iterations, which is faster than the rate, $O(1/\sqrt{k})$, of existing distributed subgradient-based methods. Simulation experiments on a distributed state estimation problem illustrate the excellent performance of our proposed method.

2010 *Mathematics subject classification*: 90C25.

Keywords and phrases: distributed algorithm, proximal-gradient method, exact penalty function method, convex optimization.

[1]School of Mathematics, Chongqing Normal University, Chongqing 400047, PR China; e-mail: zywu@cqnu.edu.cn.
[2]School of SITE, Federation University Australia, VIC 3353, Australia; e-mail: lijueyou@163.com.
[3]Australasian Joint Research Centre for Building Information Modelling, School of Built Environment, Curtin University, WA 6102, Australia; e-mail: c.wu@curtin.edu.au.
[4]School of Science, Southwest University of Science and Technology, Sichuan 621010, PR China; e-mail: longqiang@swust.edu.cn.
[5]Department of Housing and Interior Design, Kyung Hee University, Seoul, Korea; e-mail: x.wang@curtin.edu.au.

1. Introduction

In the modern age, our world has become more and more connected through infrastructures such as wireless networks and the Internet. A common feature of these systems is that they are composed of many subsystems that are interconnected through certain protocols. This class of system is referred to as a network system, and their subsystems are referred to as agents. Many real problems arising from these networks are too large for classical decision making to be helpful. There may be a multitude of agents who are decision makers, yet none of these possess all relevant knowledge. In addition, there may be limitations on the amount of communications allowed between distinct agents, so that it is impractical to exchange all available information and convert the problem to a centralized one (see for example the technical report by Tsitsiklis [24]). In fact, even if the communications are perfect between different agents, the centralized approach is still difficult to apply, because no agent may have the capability of tackling the overall problem by itself. Motivated by these reasons, there is a trend to study distributed optimization in recent years (see for example the book and articles [5, 11, 28] and references therein).

Optimizing the sum of several local objective functions is ubiquitous in many application fields. For example, problems arise in resource allocation and network utility maximization [23], state estimation and optimal control for systems [9, 12, 15, 26]. In the literature, several useful techniques are proposed to solve these related problems in a distributed manner. In terms of update strategies, they are classified as the incremental-based approach and the consensus-based approach. In the incremental approach, a cyclic path is defined over the nodes, and data are processed in a cyclic manner through the network until optimization is achieved [16]. The drawback of this method is that it has a slow asymptotic convergence rate. To improve its convergence, the proximal point method is incorporated in the incremental method [4]. On the other hand, in the consensus-based approach, the nodes achieve the minimizer globally through sharing the information locally (that is, the node only shares information with its neighbours). Nedic and Ozdaglar [17] showed that every node generates and maintains estimates of the optimal solution of the global optimization problem through the subgradient-based methods. These estimates were communicated to other nodes synchronously and over a time-varying connectivity structure. They established the explicit error bounds between the objective function values of the estimates at each node and the optimal value of the global optimization problem. However, only unconstrained optimization was discussed in that paper. Duchi et al. [8] proposed a distributed optimization based on dual subgradient averaging, and established that the number of iterations required by their algorithm scales inversely in the spectral gap of the network.

In practical applications, however, a wide variety of problems ranging from urban traffic networks [7] to interconnected chemical processes [25], subject to certain physical constraints, are modelled as distributed optimization problems with equality and/or inequality constraints. There are a few methods available for solving constrained distributed optimization problems. Nedic et al. [19] developed

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

a distributed projection subgradient method under closed convex constraint sets, and established the convergence with a diminishing step-size rule. But there was no estimate on the convergence rate. In the paper by Zhu and Martinez [31], a distributed primal–dual subgradient method was proposed to solve a distributed optimization problem with equality and inequality constraints by using Lagrangian duality. A drawback of this method was the increase in the number of variables due to the Lagrange multiplier.

In this paper, we propose a new distributed algorithm for the distributed optimization problem with inequality constraints. Instead of Lagrangian duality, we add the inequality constraints to the objective function through the exact penalty method (see for example the articles [2, 13, 14, 30]). Using Slater's condition [3], we establish the equivalence between the original problem and the transformed problem. Since the exact penalty term is nondifferentiable, a proximal-gradient method is introduced together with a multi-step consensus scheme to accelerate the convergence rate. In this case, we establish that the convergence rate of our proposed method is of order $O(1/k)$, in terms of the iteration counter k, which is faster than that of the standard distributed subgradient method of order $O(1/\sqrt{k})$. Furthermore, our established convergence rate is not only dependent on the number of iterations, but it is also dependent on the network topology and number of agents. Compared to the existing primal–dual subgradient methods used by Zhu and Martinez [31], our proposed method can achieve a faster convergence rate with a simpler communication scheme in the sense that no Lagrange multiplier is involved.

The rest of this paper is organized as follows. In Section 2, we introduce some concepts and results on which our subsequent analysis relies. In Section 3, we first formulate the problem to be solved, and transform it to an equivalent unconstrained problem by using the exact penalty method. Based on the particular structure of the penalizing problem, we then propose a novel distributed proximal-gradient algorithm with multi-consensus to solve it. In Section 4, we prove the convergence of the algorithm. An explicit convergence rate is given in terms of the number of iterations, the network size and its topology. To demonstrate the performance of the proposed algorithm, numerical simulations on a distributed state estimation problem are reported in Section 5. Finally, some concluding remarks are given in Section 6.

2. Preliminaries

2.1. Notation and definitions The standard inner product of two vectors $x, y \in \mathbb{R}^d$ is denoted by $\langle x, y\rangle = x^T y$. For $x \in \mathbb{R}^d$, its Euclidean norm is $\|x\| = \sqrt{\langle x, x\rangle}$, and the l_∞ norm is $\|x\|_\infty = \max_l |x(l)|$, where $x(l)$ is its lth entry. Let $\mathbb{R}^d_+$ represent the nonnegative orthant on $\mathbb{R}^d$.

For a matrix W, we denote its entry at the ith row and jth column as W_{ij}. Sometimes, we also write $[W_{ij}]$ to represent a matrix W. A matrix W is said to be *stochastic* if the entries in each row sum up to 1, and it is *doubly stochastic* if W and its transpose W^T are both stochastic.

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

We write $a(k) = O(b(k))$ if and only if there exist a positive real number M and a real number $k_0 > 0$ such that $|a(k)| \le M|b(k)|$ for all $k \ge k_0$. A vector $S_h(x) \in \mathbb{R}^d$ is called a *subgradient* of a convex function $h : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ at $x \in \mathrm{dom}(h)$ if, for all $y \in \mathrm{dom}(h)$,

$$h(y) \ge h(x) + \langle S_h(x), y - x\rangle,$$

where $\mathrm{dom}(h) = \{x \in \mathbb{R}^d \mid h(x) < \infty\}$. For any $x \in \mathrm{dom}(h)$, the set of all subgradients of h at x is denoted by $\partial h(x)$.

2.2. Inexact proximal-gradient method Now we discuss some properties of the proximal operator, and then summarize results of convergence rate for an inexact centralized proximal-gradient method.

For a closed proper convex function $h : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ and a scalar $\alpha > 0$, we define the proximal operator with respect to h as

$$\mathrm{Prox}_h^\alpha\{x\} = \arg\min_{z\in\mathbb{R}^d}\left\{h(z) + \frac{\alpha}{2}\|z - x\|^2\right\}.$$

The proximal operator has the following useful properties.

Proposition 2.1 [1]. *Let $h : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a closed proper convex function. For a scalar $\alpha > 0$ and $x \in \mathbb{R}^d$, let $y = \mathrm{Prox}_h^\alpha\{x\}$; we have:*

(i) *$\alpha(x - y) \in \partial h(y)$, and y can be represented as $y = x - z/\alpha, z \in \partial h(y)$;*
(ii) *$\|\mathrm{Prox}_h^\alpha\{u\} - \mathrm{Prox}_h^\alpha\{v\}\| \le \|u - v\|$ for all $u, v \in \mathbb{R}^d$.*

Our distributed proximal-gradient algorithm is to cast it as an inexact centralized proximal-gradient method, in which the errors are controlled by multi-step consensus at each iteration. This enables us to utilize recent results [21] on the convergence rate of an inexact centralized proximal-gradient method to establish the convergence rate of our distributed algorithm. An advantage of this method is that its convergence rate is of order $O(1/k)$ for nonsmooth optimization problems, while classical subgradient-based methods only achieve a rate of order $O(1/\sqrt{k})$ after k iterations (see for example the articles [1, 20]).

Proposition 2.2 [21]. *Let $g : \mathbb{R}^d \to \mathbb{R}$ be a convex function which has a Lipschitz continuous gradient with Lipschitz constant L, and let $h : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ be a lower semi-continuous proper convex function. Suppose that the function $f = g + h$ attains its minimum at a point $x^* \in \mathbb{R}^d$. Given two sequences $\{e^{(k)}\}_{k=1}^\infty$ and $\{\varepsilon^{(k)}\}_{k=1}^\infty$, where $e^{(k)} \in \mathbb{R}^d$ and $\varepsilon^{(k)} \in \mathbb{R}$, consider the inexact proximal-gradient method, which iterates as follows:*

$$z^{(k)} = x^{(k-1)} - \frac{1}{L}[\nabla g(x^{(k-1)}) + e^{(k)}],$$
$$x^{(k)} \in \mathrm{Prox}_{h,\varepsilon^{(k)}}^L\{z^{(k)}\},$$

where

$$\mathrm{Prox}_{h,\varepsilon}^L\{z\} = \left\{x \in \mathbb{R}^d \mid h(x) + \frac{L}{2}\|x - z\|^2 \le \min_{y\in\mathbb{R}^d}\left(h(y) + \frac{L}{2}\|y - z\|^2\right) + \varepsilon\right\}$$

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

is the set of all ε-optimal solutions for the proximal operator. Then, for all $k \geq 1$,

$$f(x^{(k)}) - f(x^*) \leq \frac{L}{2}(\|x^{(0)} - x^*\| + A^{(k)} + B^{(k)})^2 \frac{1}{k}, \tag{2.1}$$

where $A^{(k)} = (2/L)\sum_{i=1}^{k}(\|e^{(i)}\| + \sqrt{2L\varepsilon^{(i)}})$, $B^{(k)} = \{(2/L)\sum_{i=1}^{k}\varepsilon^{(i)}\}^{1/2}$.

Remark 2.3. Proposition 2.2 implies that the inexact centralized proximal-gradient method achieves the convergence rate of $O(1/k)$ when the sequences $\{A^{(k)}\}$ and $\{B^{(k)}\}$ are both bounded. Schmidt et al. [21] have shown that the estimate (2.1) also holds for the average of $x^{(i)}$, that is,

$$f\left(\frac{1}{k}\sum_{i=1}^{k} x^{(i)}\right) - f(x^*) \leq \frac{L}{2}(\|x^{(0)} - x^*\| + A^{(k)} + B^{(k)})^2 \frac{1}{k}.$$

3. Problem and algorithm

In this section, we formulate the problem to be solved, and describe the proposed algorithm.

3.1. Problem Consider a multi-agent optimization problem over a network. Let $G = (V, E)$ be an undirected graph over the vertex set $V = \{1, \ldots, N\}$ with edge set $E \subset V \times V$. Each vertex of the graph is referred to as an agent. The network objective is to minimize the sum of several objective functions which are distributed among the multiple agents in the network, subject to global inequality constraints. More specifically, it can be expressed as

$$\begin{aligned} &\min_{x\in\mathbb{R}^d} \; F(x) = \sum_{i=1}^{N} f_i(x) \\ &\text{such that} \;\; g_s(x) \leq 0, \quad s = 1, 2, \ldots, p, \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^d$ is a global decision vector; $f_i : \mathbb{R}^d \to \mathbb{R}$ is the objective function of agent i, only known by agent i; $g_s : \mathbb{R}^d \to \mathbb{R}$, $s = 1, 2, \ldots, p$ are the global inequality constraints known by all the agents in the network.

We adopt the following assumptions on functions $f_i(x)$ and $g_s(x)$. These assumptions are standard in the analysis of first-order methods (see the articles [1, 20, 21]).

Assumption 3.1.

(a) *For every i, $f_i : \mathbb{R}^d \to \mathbb{R}$ is convex, continuously differentiable and its gradient ∇f_i is L-Lipschitz with respect to the norm $\|\cdot\|$. And, there exists a scalar $M_f > 0$ such that for every $i \in V$ and for every $x \in \mathbb{R}^d$, $\|\nabla f_i(x)\| \leq M_f$.*

(b) *For every s ($s = 1, 2, \ldots, p$), $g_s : \mathbb{R}^d \to \mathbb{R}$ is convex, and there exists a scalar $M_g > 0$ such that for every $x \in \mathbb{R}^d$, $\|S_{g_s}(x)\| \leq M_g$ for all $S_{g_s}(x) \in \partial g_s(x)$.*

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

(c) *Slater's condition holds, that is, there exists a vector* $\bar{x} \in \mathbb{R}^d$ *such that* $g_s(\bar{x}) < 0$, $s = 1, 2, \ldots, p$.

(d) *The problem* (3.1) *attains its minimum at a point* $x^* \in \mathbb{R}^d$, *and its optimal value* $F(x^*)$ *is finite.*

3.2. Exact penalty function approach To handle the inequality constraints, we impose the inequality constraints on the objective function through the exact penalty method. Then the problem (3.1) becomes

$$\min_{x\in\mathbb{R}^d} F_c(x) = \sum_{i=1}^{N} f_i(x) + cP(x), \tag{3.2}$$

where c is a penalty parameter and

$$P(x) = \max\{0, g_1(x), g_2(x), \ldots, g_p(x)\}.$$

Obviously, $P(x)$ is convex but not differentiable on $\mathbb{R}^d$, and $\|S_P(x)\| \le M_g$ for all $S_P(x) \in \partial P(x)$.

Under certain conditions [2], the solutions of the penalized problem (3.2) are also the solutions of the constrained problem (3.1). For a detailed explanation of problem (3.1), we introduce the Lagrangian function

$$\mathcal{L}(x, \lambda) = \sum_{i=1}^{N} f_i(x) + \lambda^T g(x) = \sum_{i=1}^{N} \mathcal{L}^i(x, \lambda), \tag{3.3}$$

where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)^T \in \mathbb{R}^p_+$ is the vector of dual variables, $g(x) = (g_1(x), g_2(x), \ldots, g_p(x))^T$ and $\mathcal{L}^i(x, \lambda) = f_i(x) + \lambda^T g(x)/N$. The dual problem of problem (3.1) is

$$\max_{\lambda\in\mathbb{R}^p_+} d(\lambda) \quad \text{with } d(\lambda) = \inf_{x\in\mathbb{R}^d} \mathcal{L}(x, \lambda). \tag{3.4}$$

It can be verified that there is no duality gap between the primal problem (3.1) and its dual (3.4) if Slater's condition is satisfied (see the article by Bertsekas [3, Proposition 5.3.1]). Furthermore, the set of dual optimal solutions is nonempty and bounded. Thus, according to [2, Proposition 1], there exists a penalty parameter that satisfies $c > \sum_{s=1}^{p} \lambda_s$ such that the solutions of the penalized problem (3.2) coincide with the solutions of the constrained problem (3.1).

Letting $g_c(x) = cP(x)/N$, problem (3.2) is equivalent to

$$\min_{x\in\mathbb{R}^d} F_c(x) = \sum_{i=1}^{N} [f_i(x) + g_c(x)]. \tag{3.5}$$

Now the transformed unconstrained optimization problem (3.5) is ready for distributed computations among the agents over a network, since the function $f_i(x) + g_c(x)$ can be

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

interpreted as a private objective function associated with agent i. Furthermore, the local objectives $f_i(x) + g_c(x)$ have distinct differentiable components $f_i(x)$, but they share a common nondifferentiable component $g_c(x)$, which has a favourable structure suitable for effective computation of the proximal operator.

3.3. How to choose penalty parameter Based on the above analysis, the choice of penalty parameter is very important. If the penalty parameter is greater than a threshold, the equivalence of solutions of problem (3.1) and its exact penalized problem (3.2) holds. If we choose the parameter $c > \sum_{s=1}^{p} \lambda_s$, we have to solve the dual problem (3.4). The dual problem itself is impractical, since it is hard to solve. Here we offer an alternative way to find the upper bound on the norm of dual optimal solutions for the dual problem (3.4), since the set of dual optimal solutions is nonempty and bounded. The following proposition shows that Slater's condition guarantees the boundedness of the dual optimal set. We denote the dual optimal value of the dual problem (3.4) by d^* and its dual optimal set by $D^* = \{\lambda \in \mathbb{R}_+^p \mid d(\lambda) \geq d^*\}$.

Proposition 3.2 [18]. *Let Slater's condition of problem* (3.1) *hold. Then, for any dual optimal solution* $\lambda^* \in D^*$,

$$\|\lambda^*\| \leq \frac{F(\bar{x}) - d^*}{\delta},$$

where $\delta = \min_{1 \leq s \leq p}\{-g_s(\bar{x})\}$ *and* $\bar{x}$ *is a Slater vector.*

In practice, the dual optimal value d^* is not readily available. However, using Proposition 3.2, we can still provide an upper bound on the norm of any dual optimal solution. In particular, due to $d^* \geq d(\bar{\lambda})$, for all $\bar{\lambda} \in \mathbb{R}_+^p$ and $\|\cdot\|_\infty \leq \|\cdot\|$,

$$\|\lambda^*\|_\infty \leq \frac{F(\bar{x}) - d(\bar{\lambda})}{\delta}$$

from Proposition 3.2. Such upper bounds play a key role in finding the penalty parameter. Since

$$\begin{aligned}\|\lambda^*\|_\infty \leq \frac{F(\bar{x}) - d(\bar{\lambda})}{\delta} &< \frac{1}{\delta}\Big[\sum_{i=1}^{N} f_i(\bar{x}) - \inf_{x \in \mathbb{R}^d}\Big\{\sum_{i=1}^{N} \mathcal{L}^i(x, \bar{\lambda})\Big\}\Big] + \tau \\ &\leq \frac{1}{\delta}\sum_{i=1}^{N}\Big[f_i(\bar{x}) - \inf_{x \in \mathbb{R}^d}\{\mathcal{L}^i(x, \bar{\lambda})\}\Big] + \tau,\end{aligned}$$

where τ is a small positive constant, we set

$$\Lambda = \frac{1}{\delta}\sum_{i=1}^{N}\Big[f_i(\bar{x}) - \inf_{x \in \mathbb{R}^d}\{\mathcal{L}^i(x, \bar{\lambda})\}\Big] + \tau. \tag{3.6}$$

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

Note that the vectors $\bar{x} \in \mathbb{R}^d$, $\bar{\lambda} \in \mathbb{R}^p_+$ and $\tau > 0$ are predetermined, and $\delta = \min_{1 \le s \le p}\{-g_s(\bar{x})\}$ can be calculated easily. For given $\bar{x}$ and $\bar{\lambda}$, (3.6) is an unconstrained convex problem with the sum of several local convex functions. It is interesting that Λ can be calculated effectively in a distributed way by adopting the average consensus algorithm [27]. In terms of the results from Sections 3.2 and 3.3, now we can choose the penalty parameter $c \ge \Lambda$.

3.4. Distributed proximal-gradient algorithm Next we focus on solving the penalized problem (3.5). We first present the existing information exchange model [17, 24], and then propose our distributed proximal-gradient algorithm under this model.

Assumption 3.3. *For a time-varying weight matrix $W(t) = [w_{ij}(t)]$, $t = 1, 2, \ldots$ on the communication graph $G = (V, E)$, we assume the following properties.*

(a) *(Weights rule) There exists a scalar $\eta \in (0, 1)$ such that, for all $i \in V$, $w_{ii}(t) \ge \eta$ and, for $j \ne i$, either $w_{ij}(t) \ge \eta$, in which case j is said to be a neighbour of i, and j receives the estimate of i, at time t; or $w_{ij}(t) = 0$, in which case j is not a neighbour of i at time t.*
(b) *(Double stochasticity) For every t, $W(t)$ is doubly stochastic.*
(c) *(Connectivity and bounded intercommunication intervals) The graph (V, E_∞) is connected and there exists an integer $B \ge 1$ such that $(j, i) \in E_t \cup E_{t+1} \cup \cdots \cup E_{t+B-1}$ for all $(j, i) \in E_\infty$ and $t \ge 0$, where $E_t = \{(j, i) | w_{ij}(t) > 0\}$, $E_\infty = \{(j, i) | (j, i) \in E_t \text{ for infinitely many } t\}$.*

Assumption 3.3(a) ensures that each agent gives significant weight to its current estimate and the estimates received from its neighbours. Assumption 3.3(b) guarantees that each agent's estimate imposes an equal influence on the estimates of others in the network. Assumption 3.3(c) shows that the overall communication network is capable of exchanging information between any pair of agents in finite time. This implies that $E_t \cup E_{t+1} \cup \cdots \cup E_{t+B_0-1} = E_\infty$ with $B_0 = (N - 1)B$.

Next we present a result on the convergence property of the matrix $\Phi(t, s)$, which is important in establishing the convergence of our algorithm in Section 4.

Proposition 3.4 [17]. *Let Assumption 3.3 hold and, for $t \ge s$, let $\Phi(t, s) = W(t)W(t - 1) \cdots W(s + 1)W(s)$. Then, for all $i, j \in V$ and all t, s with $t \ge s$, $|[\Phi(t, s)]_{ij} - N^{-1}| \le \Gamma\gamma^{t-s}$, where $\Gamma = 2(1 + \eta^{-B_0})/(1 - \eta^{B_0})$ and $\gamma = (1 - \eta^{B_0})^{1/B_0}$.*

To utilize the distributed method, we assume that the penalty parameter is chosen such as $c \ge \Lambda$. Thus, the solutions of the penalized problem (3.5) are the same as the solutions of the constrained problem (3.1). Since the local objective function f_i can be accessed only by the agent i itself, the traditional centralized optimization methods do not work for problem (3.5). To overcome this difficulty, we propose a distributed proximal-gradient algorithm with multi-step consensus (DPGMC).

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

Algorithm 1: Algorithm DPGMC for solving problem (3.5)

Initialization: Given $x_i^{(0)} \in \mathbb{R}^d$, the Lipschitz constant L and the penalty parameter c.

Iteration : For each agent $i \in V$ and $k \geq 1$, do

1 Local state update

$$z_i^{(k)} = x_i^{(k-1)} - \frac{1}{L}\nabla f_i(x_i^{(k-1)}), \tag{3.7}$$

2 Multi-step consensus update

$$\hat{z}_i^{(k)} = \sum_{j=1}^{N} \chi_{ij}^{(k)} z_j^{(k)}, \tag{3.8}$$

3 Proximal step update

$$x_i^{(k)} = \mathrm{Prox}_{g_c}^{L}\{\hat{z}_i^{(k)}\}, \tag{3.9}$$

where the weights $\chi_{ij}^{(k)}$, $i, j \in V$, are given by $\chi_{ij}^{(k)} = [\Phi(\mathcal{K}^{(k)} + k, \mathcal{K}^{(k)})]_{ij}$ with $\mathcal{K}^{(k)}$ as the total number of consensus steps before iteration k.

At each iteration k, the algorithm maintains N pairs of vectors $(z_i^k, \hat{z}_i^k, x_i^k)$ with the ith pair associated with agent i. Each agent $i \in V$ first makes a local gradient update (see (3.7)), then receives information about $\{z_j^k \mid j \in \mathcal{N}_i^k\}$ associated with agents j in its neighborhood $\mathcal{N}_i^k := \{j \in V \mid (j, i) \in E \text{ and } j \neq i\}$ and makes a convex combination among them to obtain $\hat{z}_i^k$ (see (3.8)). Finally, the current estimated solution x_i^k for agent i is given by a local proximal step update related to the function g_c (see (3.9)). Based on the computational mechanism of this algorithm, it can be implemented in distributed fashion.

Note that the idea of multi-step consensus was derived from the work of Johansson et al. [10] and, later, it was developed by Chen and Ozdaglar [6] and Li et al. [11]. Here we introduce it to ensure the convergence of the algorithm with a constant step size. We replace the Lipschitz constant L appearing in (3.7) and (3.9) with a constant, α, such that $\alpha \geq L$. For the computation of the proximity operator in (3.9), we refer the reader to the articles [1, 22, 29].

4. Main results

We analyse the convergence rates of Algorithm 1 by considering the evolution of the global averages at iteration k:

$$\bar{x}^{(k)} = \frac{1}{N}\sum_{i=1}^{N} x_i^{(k)} \quad \text{and} \quad \bar{z}^{(k)} = \frac{1}{N}\sum_{i=1}^{N} z_i^{(k)}.$$

We first present some useful recursive estimates.

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

PROPOSITION 4.1. *Let* $\{x_i^{(k)}\}_{k=1}^{\infty}, \{\hat{z}_i^{(k)}\}_{k=1}^{\infty}$ *and* $\{z_i^{(k)}\}_{k=1}^{\infty}$ *be the sequences generated by Algorithm* 1. *For all* $k \geq 2$, *we have:*

(i) $\|\hat{z}_i^{(k)} - \bar{z}^{(k)}\| \leq \Gamma\gamma^k \sum_{j=1}^N \|z_j^{(k)}\|$;

(ii) $\sum_{j=1}^N \|z_j^{(k)}\| \leq \sum_{j=1}^N \|z_j^{(k-1)}\| + (1/L)(NM_f + cM_g)$.

PROOF. (i) From step (3.8) and Proposition 3.4,

$$\|\hat{z}_i^{(k)} - \bar{z}^{(k)}\| = \left\|\sum_{j=1}^N \chi_{ij}^{(k)} z_j^{(k)} - \frac{1}{N}\sum_{j=1}^N z_j^{(k)}\right\| \leq \sum_{j=1}^N \left|\chi_{ij}^{(k)} - \frac{1}{N}\right| \|z_j^{(k)}\| \leq \Gamma\gamma^k \sum_{j=1}^N \|z_j^{(k)}\|.$$

(ii) Using Proposition 2.1(i), (3.9) can be written as

$$x_i^{(k)} = \hat{z}_i^{(k)} - \frac{1}{L} v_i^{(k)}, \quad v_i^{(k)} \in \partial g_c(x_i^{(k)}).$$

Since the subgradient of g_c is bounded,

$$\|x_i^{(k)}\| \leq \|\hat{z}_i^{(k)}\| + \frac{cM_g}{NL}.$$

Taking the sum over i for the above inequality, double stochasticity of $W(t)$ yields

$$\sum_{i=1}^N \|x_i^{(k)}\| \leq \sum_{i=1}^N \|\hat{z}_i^{(k)}\| + \frac{cM_g}{L} \leq \sum_{i=1}^N \|z_i^{(k)}\| + \frac{cM_g}{L}.$$

Integrating (3.7), the above inequality gives rise to

$$\sum_{i=1}^N \|z_i^{(k)}\| = \sum_{i=1}^N \left\|x_i^{(k-1)} - \frac{1}{L}\nabla f_i(x_i^{(k-1)})\right\| \leq \sum_{i=1}^N \|x_i^{(k-1)}\| + \frac{NM_f}{L} \leq \sum_{i=1}^N \|z_i^{(k-1)}\| + \frac{1}{L}(NM_f + cM_g),$$

and the proof is complete. □

PROPOSITION 4.2. *Let* $\{x_i^{(k)}\}_{k=1}^{\infty}, \{\hat{z}_i^{(k)}\}_{k=1}^{\infty}$ *and* $\{z_i^{(k)}\}_{k=1}^{\infty}$ *be the sequences generated by Algorithm* 1*; then the algorithm can be expressed as*

$$\begin{cases} \bar{z}^{(k)} = \bar{x}^{(k-1)} - \dfrac{1}{L}[\nabla f(\bar{x}^{(k-1)}) + e^{(k)}], \\ \bar{x}^{(k)} \in \operatorname{Prox}_{g_c, \varepsilon^{(k)}}^{L}\{\bar{z}^{(k)}\}, \end{cases}$$

where $\nabla f(\bar{x}^{(k-1)}) = N^{-1}\sum_{i=1}^N \nabla f_i(\bar{x}^{(k-1)})$ *and error sequences* $\{e^{(k)}\}_{k=1}^{\infty}$ *and* $\{\varepsilon^{(k)}\}_{k=1}^{\infty}$ *satisfy*

$$\|e^{(k)}\| \leq 2L\Gamma\gamma^{k-1} \sum_{j=1}^N \|z_j^{(k-1)}\| \tag{4.1}$$

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

and

$$\varepsilon^{(k)} \le \frac{2cM_g}{N}\Gamma\gamma^k \sum_{j=1}^{N} \|z_j^{(k)}\| + \frac{L}{2}\Big[\Gamma\gamma^k \sum_{j=1}^{N} \|z_j^{(k)}\|\Big]^2. \tag{4.2}$$

Proof. From step (3.7),

$$\bar{z}^{(k)} = \bar{x}^{(k-1)} - \frac{1}{L}[\nabla f(\bar{x}^{(k-1)}) + e^{(k)}],$$

where $\nabla f(\bar{x}^{(k-1)}) = N^{-1}\sum_{i=1}^{N} \nabla f_i(\bar{x}^{(k-1)})$ and $e^{(k)} = N^{-1}\sum_{i=1}^{N}[\nabla f_i(x_i^{(k-1)}) - \nabla f_i(\bar{x}^{(k-1)})]$. Since the gradient of f_i is L-Lipschitz continuous,

$$\begin{aligned}\|e^{(k)}\| &\le \frac{L}{N}\sum_{i=1}^{N}\|x_i^{(k-1)} - \bar{x}^{(k-1)}\| \le \frac{L}{N}\sum_{i=1}^{N}\Big\|x_i^{(k-1)} - \frac{1}{N}\sum_{j=1}^{N} x_j^{(k-1)}\Big\| \\ &\le \frac{L}{N}\sum_{i=1}^{N}\Big[\frac{1}{N}\sum_{j=1}^{N}\|x_i^{(k-1)} - x_j^{(k-1)}\|\Big],\end{aligned}$$

where the last inequality follows from the convexity of the norm $\|\cdot\|$. The combination of (3.9) and Proposition 2.1(ii) yields

$$\begin{aligned}\|x_i^{(k-1)} - x_j^{(k-1)}\| &= \|\mathrm{Prox}_{g_c}^{L}\{\hat{z}_i^{(k-1)}\} - \mathrm{Prox}_{g_c}^{L}\{\hat{z}_j^{(k-1)}\}\| \\ &\le \|\hat{z}_i^{(k-1)} - \hat{z}_j^{(k-1)}\| \le \|\hat{z}_i^{(k-1)} - \bar{z}^{(k-1)}\| + \|\hat{z}_j^{(k-1)} - \bar{z}^{(k-1)}\| \\ &\le 2\Gamma\gamma^{k-1}\sum_{j=1}^{N}\|z_j^{(k-1)}\|.\end{aligned}$$

Combined with the above inequalities, it gives rise to inequality (4.1).

Let $u^{(k)} = \mathrm{Prox}_{g_c}^{L}\{\bar{z}^{(k)}\} = \arg\min_{x\in\mathbb{R}^d}\{g_c(x) + \|x - \bar{z}^{(k)}\|^2 L/2\}$; then $\bar{x}^{(k)} = N^{-1}\sum_{i=1}^{N} x_i^{(k)}$ $= N^{-1}\sum_{i=1}^{N}\mathrm{Prox}_{g_c}^{L}\{\hat{z}_i^{(k)}\}$ can be regarded as an approximation of $u^{(k)}$. Next we link $u^{(k)}$ with $\bar{x}^{(k)}$ by formulating the latter as an inexact proximal step with the error $\varepsilon^{(k)}$. Considering the convexity of $g_c(x)$ and the boundedness of $\partial g_c(x)$,

$$\begin{aligned}g_c(\bar{x}^{(k)}) + \frac{L}{2}\|\bar{x}^{(k)} - \bar{z}^{(k)}\|^2 &\le g_c(u^{(k)}) + \frac{cM_g}{N}\|\bar{x}^{(k)} - u^{(k)}\| + \frac{L}{2}[\|\bar{x}^{(k)} - u^{(k)}\|^2 \\ &\quad + 2\langle\bar{x}^{(k)} - u^{(k)}, u^{(k)} - \bar{z}^{(k)}\rangle + \|u^{(k)} - \bar{z}^{(k)}\|^2] \\ &\le \min_{x\in\mathbb{R}^d}\Big\{g_c(x) + \frac{L}{2}\|x - \bar{z}^{(k)}\|^2\Big\} \\ &\quad + \|\bar{x}^{(k)} - u^{(k)}\|\Big(\frac{cM_g}{N} + L\|u^{(k)} - \bar{z}^{(k)}\|\Big) + \frac{L}{2}\|\bar{x}^{(k)} - u^{(k)}\|^2,\end{aligned}$$

where the last inequality above follows from the fact that $u^{(k)}$ is the unique minimizer of $g_c(x) + \|x - \bar{z}^{(k)}\|^2 L/2$. Now we can write $\bar{x}^{(k)} \in \mathrm{Prox}_{g_c,\varepsilon^{(k)}}^{L}\{\bar{z}^{(k)}\}$, where $\varepsilon^{(k)} = \|\bar{x}^{(k)} - u^{(k)}\|(cM_g/N + L\,\|u^{(k)} - \bar{z}^{(k)}\|) + \|\bar{x}^{(k)} - u^{(k)}\|^2 L/2$. In light of Proposition 2.1(i), $u^{(k)} = \mathrm{Prox}_{g_c}^{L}\{\bar{z}^{(k)}\}$ implies $L(\bar{z}^{(k)} - u^{(k)}) \in \partial g_c(\bar{z}^{(k)})$. Thus, by the boundedness of $\partial g_c(x)$,

$$\varepsilon^{(k)} \le \frac{2cM_g}{N}\|\bar{x}^{(k)} - u^{(k)}\| + \frac{L}{2}\|\bar{x}^{(k)} - u^{(k)}\|^2.$$

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

By the nonexpansiveness of the proximal operator and Proposition 4.1(i),

$$\begin{aligned}\|\bar{x}^{(k)} - u^{(k)}\| &= \left\| \frac{1}{N}\sum_{i=1}^{N} \mathrm{Prox}_{g_c}^{L}\{\hat{z}_i^{(k)}\} - \mathrm{Prox}_{g_c}^{L}\{\bar{z}^{(k)}\} \right\| \\ &\le \frac{1}{N}\sum_{i=1}^{N} \|\mathrm{Prox}_{g_c}^{L}\{\hat{z}_i^{(k)}\} - \mathrm{Prox}_{g_c}^{L}\{\bar{z}^{(k)}\}\| \\ &\le \frac{1}{N}\sum_{i=1}^{N} \|\hat{z}_i^{(k)} - \bar{z}^{(k)}\| \le \Gamma\gamma^k \sum_{j=1}^{N} \|z_j^{(k)}\|.\end{aligned}$$

Therefore, the desired relation (4.2) can be obtained. □

Proposition 4.2 indicates that Algorithm 1 can be viewed as an inexact centralized proximal-gradient method. Moreover, error sequences $\{\|e^{(k)}\|\}$ and $\{\varepsilon^{(k)}\}$ can be upper bounded by the term $\sum_{j=1}^{N} \|z_j^{(k)}\|$, which are in turn controlled by the multi-step consensus. According to Proposition 2.2, if $\{A^{(k)}\}$ and $\{B^{(k)}\}$ are both finite, then the inexact centralized proximal-gradient method achieves the convergence rate of $O(1/k)$ after iteration k. Next we shall see that in our proposed Algorithm 1 this is indeed the case.

By using Proposition 4.1, we find a first-order polynomial in k as an upper boundary for $\sum_{i=1}^{N} \|z_i^{(k)}\|$.

LEMMA 4.3. *Let $\{x_i^{(k)}\}_{k=1}^{\infty}$, $\{\hat{z}_i^{(k)}\}_{k=1}^{\infty}$ and $\{z_i^{(k)}\}_{k=1}^{\infty}$ be the sequences generated by Algorithm 1. Then there exist scalars $c_0 = \sum_{j=1}^{N} \|x_j^{(0)}\|$ and $c_1 = (NM_f + cM_g)/L$ such that for all $k \ge 1$,*

$$\sum_{j=1}^{N} \|z_j^{(k)}\| \le c_0 + c_1 k. \tag{4.3}$$

PROOF. From Proposition 4.1(ii),

$$\begin{aligned}\sum_{j=1}^{N} \|z_j^{(k)}\| &\le \sum_{j=1}^{N} \|z_j^{(k-1)}\| + \frac{1}{L}(NM_f + cM_g) \\ &\le \sum_{j=1}^{N} \|z_j^{(k-2)}\| + \frac{2}{L}(NM_f + cM_g) \le \cdots \\ &\le \sum_{j=1}^{N} \|z_j^{(1)}\| + \frac{k-1}{L}(NM_f + cM_g) \\ &\le \sum_{j=1}^{N} \|x_j^{(0)}\| + \frac{k}{L}(NM_f + cM_g).\end{aligned}$$

This completes the proof. □

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

We next prove that both sequences $\{A^{(k)}\}$ and $\{B^{(k)}\}$ are bounded.

LEMMA 4.4. *Let*

$$A^{(k)} = \frac{2}{L}\sum_{i=1}^{k}(\|e^{(i)}\| + \sqrt{2L\varepsilon^{(i)}}) \quad \text{and} \quad B^{(k)} = \left(\frac{2}{L}\sum_{i=1}^{k}\varepsilon^{(i)}\right)^{1/2},$$

where the estimates for $\|e^{(i)}\|$ and $\varepsilon^{(i)}$ are given in (4.1) and (4.2), respectively. Then, for all $k \geq 1$,

$$A^{(k)} < 4\sqrt{\frac{cM_g\Gamma}{NL}}\frac{\sqrt{c_0}+\sqrt{c_1}}{(1-\sqrt{\gamma})^2} + 6\Gamma\frac{c_0+c_1}{(1-\sqrt{\gamma})^2} \tag{4.4}$$

and

$$B^{(k)} < 2\sqrt{\frac{cM_g\Gamma}{NL}}\frac{\sqrt{c_0}+\sqrt{c_1}}{(1-\sqrt{\gamma})^2} + 2\Gamma\frac{c_0+c_1}{(1-\sqrt{\gamma})^2}. \tag{4.5}$$

PROOF. Since $\gamma \in (0,1)$, as in Proposition 3.4,

$$\sum_{i=0}^{\infty}\gamma^i = \frac{1}{1-\gamma}, \quad \sum_{i=0}^{\infty}i\gamma^i = \frac{\gamma}{(1-\gamma)^2}, \quad \sum_{i=0}^{\infty}\gamma^2\gamma^i = \frac{\gamma+\gamma^2}{(1-\gamma)^2}. \tag{4.6}$$

Thus, from (4.1), Lemma 4.3 and (4.6),

$$\begin{aligned}\sum_{i=1}^{k}\|e^{(i)}\| &\leq \sum_{i=1}^{k}2L\Gamma\gamma^{i-1}[c_0+c_1(i-1)] < 2L\Gamma\left[\frac{c_0}{1-\gamma}+\frac{c_1\gamma}{(1-\gamma)^2}\right]\\ &< 2L\Gamma\frac{c_0+c_1}{(1-\gamma)^2} < 2L\Gamma\frac{c_0+c_1}{(1-\sqrt{\gamma})^2}.\end{aligned} \tag{4.7}$$

Similarly, (4.2), Lemma 4.3 and (4.6) yield

$$\begin{aligned}\sum_{i=1}^{k}\varepsilon^{(i)} &\leq \sum_{i=1}^{k}\frac{2cM_g}{N}\Gamma\gamma^i(c_0+c_1i) + \frac{L}{2}[\Gamma\gamma^i(c_0+c_1i)]^2\\ &< \frac{2cM_g}{N}\Gamma\frac{c_0+c_1}{(1-\gamma)^2} + \frac{L}{2}\Gamma^2[\gamma^i(c_0^2+2c_0c_1i+c_1^2i^2)]\\ &< \frac{2cM_g}{N}\Gamma\frac{c_0+c_1}{(1-\gamma)^2} + L\Gamma^2\frac{(c_0+c_1)^2}{(1-\gamma)^2}.\end{aligned} \tag{4.8}$$

Since $\sqrt{a+b} \leq \sqrt{a}+\sqrt{b}$ for any $a, b \in \mathbb{R}_+$, from inequality (4.8) it follows that

$$\begin{aligned}B^{(k)} = \left(\frac{2}{L}\sum_{i=1}^{k}\varepsilon^{(i)}\right)^{1/2} &< 2\sqrt{\frac{cM_g\Gamma}{NL}}\frac{\sqrt{c_0}+\sqrt{c_1}}{1-\gamma} + 2\Gamma\frac{c_0+c_1}{1-\gamma}\\ &< 2\sqrt{\frac{cM_g\Gamma}{NL}}\frac{\sqrt{c_0}+\sqrt{c_1}}{(1-\sqrt{\gamma})^2} + 2\Gamma\frac{c_0+c_1}{(1-\sqrt{\gamma})^2},\end{aligned}$$

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

which proves the inequality in (4.5). Also, since $\sqrt{k} \le k$ for all $k \ge 1$ and $\varepsilon^{(k)} \le N^{-1}2cM_g\Gamma\gamma^k(c_0 + c_1k) + [\Gamma\gamma^k(c_0 + c_1k)]^2L/2$,

$$\sqrt{\varepsilon^{(k)}} \le \sqrt{\frac{2cM_g\Gamma}{N}}\sqrt{\gamma}^k(\sqrt{c_0} + \sqrt{c_1}k) + \sqrt{\frac{L}{2}}\Gamma\gamma^k(c_0 + c_1k),$$

which yields

$$\sum_{i=1}^{k}\sqrt{\varepsilon^{(i)}} < \sqrt{\frac{2cM_g\Gamma}{N}}\frac{\sqrt{c_0} + \sqrt{c_1}}{(1-\sqrt{\gamma})^2} + \sqrt{\frac{L}{2}}\Gamma\frac{c_0 + c_1}{(1-\sqrt{\gamma})^2}. \tag{4.9}$$

The inequality in (4.4) now follows from (4.7) and (4.9). □

Using Lemmas 4.3 and 4.4, we establish the convergence rate of our proposed Algorithm 1 for solving problem (3.5).

THEOREM 4.5. *Suppose that Assumptions 3.1 and 3.3 hold. Let* $\{x_i^{(k)}\}_{k=1}^{\infty}$, $\{\hat{z}_i^{(k)}\}_{k=1}^{\infty}$ *and* $\{z_i^{(k)}\}_{k=1}^{\infty}$ *be the sequences generated by Algorithm DPGMC. Let* x_c^* *be the optimal solution of problem* (3.5) *for a given penalty parameter* $c > 0$. *Then, for all* $k \ge 1$, *there exists a scalar* $C(N, c, L, \Gamma, M_f, M_g) = [\|\bar{x}^{(0)} - x_c^*\|L/2 + 8(\sqrt{cM_g\Gamma/(NL)}(\sqrt{c_0} + \sqrt{c_1}) + \Gamma(c_0 + c_1))]^2$ *such that*

$$F_c(\bar{x}^{(k)}) - F_c(x_c^*) \le C(N, c, L, \Gamma, M_f, M_g)\frac{1}{(1-\sqrt{\gamma})^4}\frac{1}{k}.$$

Furthermore, for all $\mathcal{K} \ge 1$,

$$F_c(\bar{x}^{(\mathcal{K})}) - F_c(x_c^*) \le C(N, c, L, \Gamma, M_f, M_g)\frac{1}{(1-\sqrt{\gamma})^4}\frac{1}{\sqrt{\mathcal{K}}},$$

where $\mathcal{K}$ *is the total number of consensus steps taken.*

PROOF. According to Proposition 4.2, Algorithm 1 can be formulated as an inexact centralized proximal-gradient method in the framework of Proposition 2.2. Thus, the conclusion of Proposition 2.2 holds. Next, from Lemma 4.4, we get an estimation for $(L/2)(\|\bar{x}^{(0)} - x_c^*\| + A^{(k)} + B^{(k)})^2$ as follows:

$$\begin{aligned}
&\frac{L}{2}(\|\bar{x}^{(0)} - x_c^*\| + A^{(k)} + B^{(k)})^2 \\
&\quad < \frac{L}{2}\left[\|\bar{x}^{(0)} - x_c^*\| + \left(6\sqrt{\frac{cM_g\Gamma}{NL}}(\sqrt{c_0} + \sqrt{c_1}) + 8\Gamma(c_0 + c_1)\right)\frac{1}{(1-\sqrt{\gamma})^2}\right]^2 \\
&\quad \le \frac{L}{2}\left[\|\bar{x}^{(0)} - x_c^*\| + 8\left(\sqrt{\frac{cM_g\Gamma}{NL}}(\sqrt{c_0} + \sqrt{c_1}) + \Gamma(c_0 + c_1)\right)\right]^2\frac{1}{(1-\sqrt{\gamma})^4}.
\end{aligned}$$

Thus, we establish the first assertion of this theorem by using the constant $C(N, c, L, \Gamma, M_f, M_g)$.

The second assertion follows from the multi-step consensus of Algorithm DPGMC, since it takes k consensus steps to complete the kth iteration. □

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

REMARK 4.6. Since $c_1 = (NM_f + cM_g)/L$, we have $c_1 = O(N)$ by neglecting the other constants. Then $C(N, c, L, \Gamma, M_f, M_g) = O(N^2)$ and

$$F_c(\bar{x}^{(k)}) - F_c(x_c^*) = O\Big(\frac{N^2}{(1-\sqrt{\gamma})^4}\frac{1}{k}\Big). \tag{4.10}$$

Thus, the optimality gap obtained above depends on the number of iterations, k, the number of agents, N, and the parameter of network topology, γ. If the communication matrix, $W(t) \equiv W$ for $t = 1, 2, \ldots$; then $\gamma = \sigma_2(W)$, where $\sigma_2(W)$ is the second largest singular value of W and characterizes the connectivity of the network considered.

REMARK 4.7. Theorem 4.5 together with (4.10) implies that at most

$$K(\epsilon, \gamma, N) = O\Big(\frac{1}{\epsilon}\frac{N^2}{(1-\sqrt{\gamma})^4}\Big) \tag{4.11}$$

iterations are required to achieve an ϵ-accurate solution if the communication matrix $W(t)$ satisfies Assumption 3.3. It is clear that $K(\epsilon, \gamma, N)$ in (4.11) is a increasing function of γ. This shows that the more well connected the underlying network is, the fewer the number of iterations that we need to run in Algorithm 1. In addition, $K(\epsilon, \gamma, N)$ is also increasing in terms of the number of agents, N. Thus, the more agents in the network, the more iterations are required to achieve the given accuracy.

The next theorem now follows from Theorem 4.5 and Proposition 3.2.

THEOREM 4.8. *Assume that the conditions of Theorem 4.5 hold. Let* $\{x_i^{(k)}\}_{k=1}^{\infty}$, $\{\hat{z}_i^{(k)}\}_{k=1}^{\infty}$ *and* $\{z_i^{(k)}\}_{k=1}^{\infty}$ *be the sequences generated by Algorithm 1. Let* x^* *be the optimal solution of problem (3.1). If the parameter* $c > \sum_{s=1}^{p} \lambda_s$ *(for example, taking* $c = \Lambda$*), then, for all* $k \geq 1$, $\bar{x}^{(k)}$ *is the solution to problem (3.1) and*

$$F(\bar{x}^{(k)}) - F(x^*) \leq C(N, c, L, \Gamma, M_f, M_g)\frac{1}{(1-\sqrt{\gamma})^4}\frac{1}{k}.$$

5. Numerical simulation

To demonstrate the effectiveness of the proposed Algorithm 1, we consider a distributed state estimation problem (see the article by Necoara et al. [15, Example 2.1]). Mathematically, it is formulated as an optimization problem with a common decision variable x subject to linear inequality constraints:

$$\begin{aligned} \min_{x \in \mathbb{R}^d} \quad & F(x) = \sum_{i=1}^{N} x^T H_i x + q_i^T x \\ \text{such that} \quad & Ax \leq b, \end{aligned}$$

where $A \in \mathbb{R}^{m \times d}$, $b \in \mathbb{R}^m$, $q_i \in \mathbb{R}^d$ and matrices $H_i \in \mathbb{R}^{d \times d}$ are positive definite. Note that $F_i = x^T H_i x + q_i^T x$ is a private function, only known by agent i. The local function

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

F_i is convex and differentiable. Its gradient ∇F_i is L_i-Lipschitz continuous by setting $L_i = \|H_i^T H_i\|$. Let $L = \max_i\{L_i\}$. The corresponding penalized problem is given by

$$\min_{x\in\mathbb{R}^d} F_c(x) = \sum_{i=1}^{N}\left[x^T H_i x + q_i^T x + \frac{c}{N}\max\{0, Ax - b\}\right],$$

where c is the penalty parameter. In general, there are two ways to choose c such that it satisfies equation (3.6). One way is by heuristic, and the other one is by solving the unconstrained optimization problem defined in this equation. If c is determined in this way, we first require to find a feasible solution $\bar{x}$ satisfying $A\bar{x} < b$; then, for any given $\bar{\lambda} \geq 0$, the average consensus algorithm of Zhu and Martinez [31] is applied to solve equation (3.6). To simulate the time-varying weight matrix, we generate a pool of 20 weight matrices from connected random graphs and each weight matrix satisfies Assumption 3.3. Then the matrix $[\chi_{ij}^{(k)}]$ in step (3.8) is the product of k weight matrices randomly drawn from the above pool.

In order to carry out the comparison, we introduce the following distributed Lagrangian primal–dual subgradient (DLPDS) algorithm [31] for solving the saddle points of the corresponding Lagrangian function in equation (3.3) for $i \in V$ and $k \geq 1$:

$$\begin{cases} z_i^{(k)} = \sum_{j=1}^{N} W_{ij}^{(k-1)} x_j^{(k-1)}, \\ \mu_i^{(k)} = \sum_{j=1}^{N} W_{ij}^{(k-1)} \lambda_j^{(k-1)}, \\ x_i^{(k)} = z_i^{(k)} - \alpha^{(k)}[\nabla f_i(z_i^{(k)}) + \mu^{(k)T} S_g(z_i^{(k)})], \\ \lambda_i^{(k)} = [\mu_i^{(k)} + \alpha^{(k)} g(z_i^{(k)})]^+, \end{cases}$$

where $g(x) = Ax - b$, $\alpha^{(k)} = 1/k$ satisfies the step-size rule of Zhu and Martinez [31], $[\cdot]^+$ is a projection operator on $\mathbb{R}_+^p$ and $[W_{ij}^{(k-1)}]$ is the randomly chosen weight matrix satisfying Assumption 3.3.

For simplicity, we assume that H_i is a diagonal matrix with elements generated randomly in the interval $[1, 2]$, q_i is a vector with elements generated randomly in $[-1, 1]$, A is reduced to a vector in $\mathbb{R}^d$ with elements generated randomly in $[-1, 1]$, $b \geq 0$ and the initial points $x_i^{(0)}$ are generated randomly. In this numerical experiment, we take the penalty parameter $c = 5$ heuristically.

We report preliminary experimental results on the convergence behaviour of Algorithms DPGMC, proposed in this paper, and DLPDS, developed by Zhu and Martinez [31]. Figure 1 depicts the value of maximum error, $\max_{i\in V}[F(x_i^{(k)}) - F(x^*)]$, versus number of iterations with different nodes and dimensions for the first 500 iterations.

For all the four tested cases, Figure 1 shows that our proposed Algorithm 1 achieves faster convergence than DLPDS [31]. More specifically, from Figure 1(a), we can see that after 500 iterations, our proposed Algorithm 1 reaches an accuracy of

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

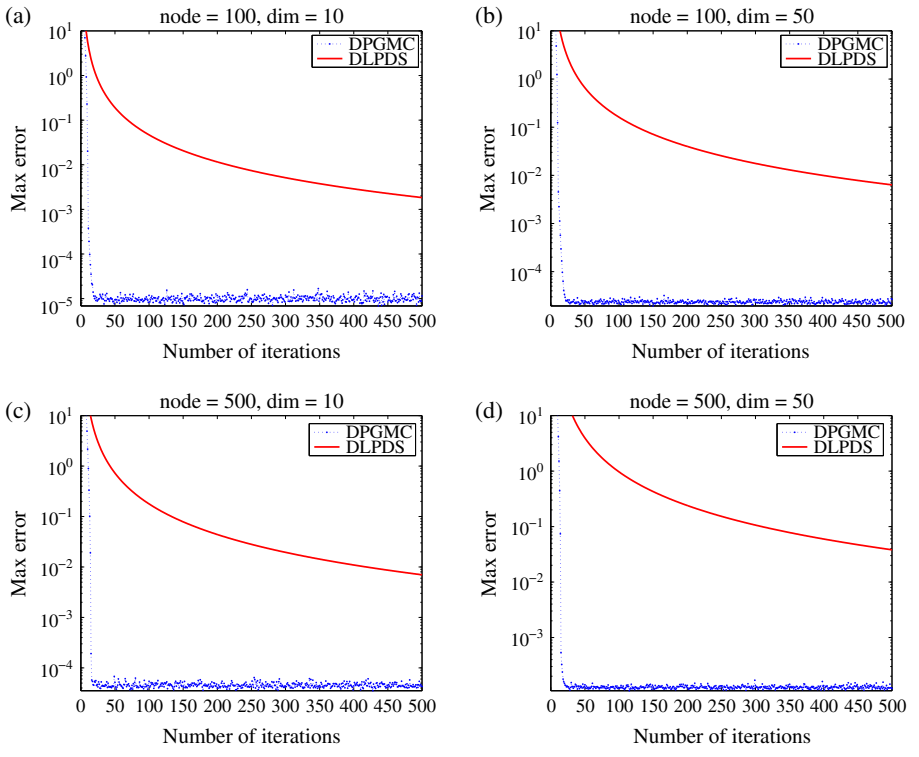

FIGURE 1. Maximum error versus number of iterations with different nodes and dimensions.

approximately 10^{-5}, while DLPDS only reaches an accuracy of approximately 10^{-2}. Furthermore, we observe that the value of maximum error obtained by DLPDS at 300 iterations is obtained by DPGMC at iteration 18. Similar results are also observed in the other three figures (b)–(d) in Figure 1. Thus, the performance of our proposed DPGMC is better than that of DLPDS. This is mainly caused by the slow convergence of subgradient-based methods. Comparing (a) with (d) in Figure 1, we can observe that the more the number of nodes and dimensions, the smaller is the accuracy for both algorithms.

6. Conclusion

In this paper, we have developed a distributed proximal-gradient algorithm with multi-step consensus for minimizing the sum of local convex functions, subject to global inequality constraints over a network. We have proved the convergence of the proposed algorithm with an explicit convergence rate, given in terms of the number of iterations, the network size and its topology. Compared to existing distributed primal–dual subgradient methods for solving distributed convex optimization under inequality constraints, our method is faster and simpler since no dual variable is involved. Also,

https://doi.org/10.1017/S1446181114000273 Published online by Cambridge University Press

simulation experiments show that our method performs better than the primal–dual subgradient methods.

Acknowledgements

We are grateful to the editor and the anonymous reviewers for their valuable suggestions which have helped us improve the paper. This research was partially supported by the Natural Science Foundation of Chongqing: cstc2013jjB00001, cstc2011jjA00010, cstc2013jjB0149 and cstc2013jcyjA00029, by CMEC under Grant KJ120616, by a Postgraduate Scholarship of Federation University Australia, by NSFC 11471062 and 11001288 and by the Key Project of the Chinese Ministry of Education (210179).

References

[1] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems", *SIAM J. Imaging Sci.* **2** (2009) 183–202; doi:10.1137/080716542.

[2] D. P. Bertsekas, "Necessary and sufficient conditions for a penalty method to be exact", *Math. Program.* **9** (1975) 87–99; doi:10.1007/bf01681332.

[3] D. P. Bertsekas, *Nonlinear programming* (Athena Scientific, Belmount, MA, 1999).

[4] D. P. Bertsekas, "Incremental proximal methods for large scale convex optimization", *Math. Program.* **129** (2011) 163–195; doi:10.1007/s1010701104720.

[5] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods* (Prentice Hall, Englewood Cliffs, NJ, 1989).

[6] A. I. Chen and A. Ozdaglar, "A fast distributed proximal-gradient method", in: *50th Annual Allerton Conference on Communication, Control and Computing, Allerton, New York, 2012* (IEEE, 2012) 601–608; doi:10.1109/allerton.2012.6483273.

[7] L. B. de Oliveira and E. Camponogara, "Multi-agent model predictive control of signaling split in urban traffic networks", *Transport. Res. Part C: Emerg. Tech.* **8** (2010) 120–139; doi:10.1016/j.trc.2009.04.022.

[8] J. C. Duchi, A. Agarwal and M. J. Wainwright, "Dual averaging for distributed optimization: convergence analysis and network scaling", *IEEE Trans. Automat. Control* **57** (2012) 592–606; doi:10.1109/tac.2011.2161027.

[9] R. Ebrahimian and R. Baldick, "State estimation distributed processing", *IEEE Trans. Power Syst.* **15** (2000) 1240–1246; doi:10.1109/59.898096.

[10] B. Johansson, T. Keviczky, M. Johansson and K. H. Johansson, "Subgradient methods and consensus algorithms for solving convex optimization problems", in: *Proceedings of the 47th IEEE Conference on Decision Control, Cancun, Mexico, 2008* (IEEE, 2008) 4185–4190; doi:10.1109/cdc.2008.4739339.

[11] J. Li, C. Wu, Z. Wu and Q. Long, "Gradient-free method for nonsmooth distributed optimization", *J. Global. Optim.* (2014) 1–16; doi:10.1007/s10898-014-0174-2.

[12] R. Loxton, Q. Lin and K. L. Teo, "Minimizing control variation in nonlinear optimal control", *Automatica* **49** (2013) 2652–2664; doi:10.1016/j.automatica.2013.05.027.

[13] C. Ma, X. Li, K. F. C. Yiu, Y. Yang and L. Zhang, "On an exact penalty function method for semi-infinite programming problems", *J. Ind. Manag. Optim.* **8** (2012) 705–726; doi:10.3934/jimo.2012.8.705.

[14] Z. Meng, Q. Hu and C. Dang, "A penalty function algorithm with objective parameters for nonlinear mathematical programming", *J. Ind. Manag. Optim.* **5** (2009) 585–601; doi:10.3934/jimo.2009.5.585.

[15] I. Necoara, V. Nedelcu and I. Dumitrache, "Parallel and distributed optimization methods for estimation and control in networks", *J. Process Control* **21** (2011) 756–766; doi:10.1016/j.jprocont.2010.12.010.
[16] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization", *SIAM J. Optim.* **12** (2001) 109–138; doi:10.1137/s1052623499362111.
[17] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization", *IEEE Trans. Automat. Control* **54** (2009) 48–61; doi:10.1109/tac.2008.2009515.
[18] A. Nedic and A. Ozdaglar, "Subgradient methods for saddle-point problems", *J. Optim. Theory Appl.* **142** (2009) 205–228; doi:10.1007/s1095700995227.
[19] A. Nedic, A. Ozdaglar and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks", *IEEE Trans. Automat. Control* **55** (2010) 922–938; doi:10.1109/tac.2010.2041686.
[20] Y. Nesterov, "Gradient methods for minimizing composite functions", *Math. Program.* **140** (2013) 125–161; doi:10.1007/s1010701206295.
[21] M. Schmidt, N. L. Roux and F. R. Bach, "Convergence rates of inexact proximal-gradient methods for convex optimization", *Adv. Neural Inf. Process. Syst.* (2011) 1458–1466; arXiv:1109.2415.
[22] Y. Shen, W. Zhang and B. He, "Relaxed augmented Lagrangian-based proximal point algorithms for convex optimization with linear constraints", *J. Ind. Manag. Optim.* **10** (2014) 743–759; doi:10.3934/jimo.2014.10.743.
[23] R. Srikant, *The mathematics of Internet congestion control* (Springer, Berlin, 2004).
[24] J. N. Tsitsiklis, "Problems in decentralized decision making and computation", Technical Report, DTIC Document, 1984.
[25] A. N. Venkat, "Distributed model predictive control: theory and applications", Ph. D. Thesis, CiteSeer, 2006.
[26] C. Wu, K. L. Teo and S. Y. Wu, "Minmax optimal control of linear systems with uncertainty and terminal state constraints", *Automatica* **49** (2013) 1809–1815; doi:10.1016/j.automatica.2013.02.052.
[27] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging", *Systems Control Lett.* **53** (2004) 65–78; doi:10.1016/j.sysconle.2004.02.022.
[28] L. Xiao, S. Boyd and S. J. Kim, "Distributed average consensus with least-mean-square deviation", *J. Parallel Distrib. Comput.* **67** (2007) 33–46; doi:10.1016/j.jpdc.2006.08.010.
[29] Y. Xiao, S. Y. Wu and B. He, "A proximal alternating direction method for l2-norm least squares problem in multi-task feature learning", *J. Ind. Manag. Optim.* **8** (2012) 1057–1069; doi:10.3934/jimo.2012.8.1057.
[30] C. Yu, K. L. Teo, L. Zhang and Y. Bai, "A new exact penalty function method for continuous inequality constrained optimization problems", *J. Ind. Manag. Optim.* **6** (2010) 895–910; doi:10.3934/jimo.2010.6.895.
[31] M. Zhu and S. Martinez, "On distributed convex optimization under inequality and equality constraints", *IEEE Trans. Automat. Control* **57** (2012) 151–164; doi:10.1109/tac.2011.2167817.