Adaptive Exact Penalty Design for Constrained Distributed Optimization

Hongbing Zhou, Xianlin Zeng, *Member, IEEE*, and Yiguang Hong, *Fellow, IEEE*

***Abstract*—This paper focuses on a distributed convex optimization problem with set constraints, where the local objective functions are convex but not necessarily differentiable. We employ an exact penalty method for the constrained optimization problem to avoid the projection of subgradients to convex sets, which may result in problems about algorithm trajectories caused by maybe nonconvex differential inclusions and quite high computational cost. To effectively find a suitable gain of the penalty function online, we propose an adaptive distributed algorithm with the help of the adaptive control idea in order to achieve an exact solution without any *a priori* computation or knowledge of the objective functions. By virtue of convex and nonsmooth analysis, we give a rigorous proof for the convergence of the proposed continuous-time algorithm.**

***Index Terms*—Adaptive algorithm, convex and nondifferentiable function, distributed optimization, exact penalty method.**

I. Introduction

In recent years, distributed optimization in multiagent networks has become increasingly important in various engineering areas. A well-known distributed optimization formulation has been given for the agents in a network to cooperatively optimize the sum of local objective functions, only known by their own agents, and many distributed algorithms, either discrete-time or continuous-time [1], [3], [4], have been proposed to cooperatively find an optimization solution. In particular, the continuous-time optimization design using the gradient flow of local functions and projection operators has been widely used in constrained optimization problems [3], [4]. Although some results have been obtained for differentiable objective functions, distributed nonsmooth optimization draws more and more attention [4]–[6], where the computation of projection from subgradients to constraint sets is often involved, and solutions to the differential inclusions of the proposed algorithms may not exist.

Manuscript received January 15, 2018; revised May 4, 2018 and November 30, 2018; accepted February 10, 2019. Date of publication March 4, 2019; date of current version October 30, 2019. This work was supported in part by the National Key Research and Development Program of China under Grant 2016YFB0901902, in part by the National Natural Science Foundation of China under Grant 61573344, Grant 61733018, Grant 61603378, and in part by the China Postdoctoral Science Foundation under Grant 2017M620020. Recommended by Associate Editor G. Gu. *(Corresponding author: Yiguang Hong.)*

H. Zhou is with the School of Mathematics and Statistics, Nanyang Normal University, Nanyang 473061, China (e-mail: zhouhongbing13@126.com).

X. Zeng is with the Key Laboratory of Intelligent Control and Decision of Complex Systems, School of Automation, Beijing Institute of Technology, Beijing 100081, China (e-mail: xianlin.zeng@bit.edu.cn).

Y. Hong is with the Key Laboratory of Systems and Control, Institute of Systems Science, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: yghong@iss.ac.cn).

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

Digital Object Identifier 10.1109/TAC.2019.2902612

The penalty method becomes popular after a detailed theoretical and computational study conducted in [13], for constrained optimization problems by changing them into a sequence of unconstrained problems with introducing penalty functions. There are two basic methods, namely, interior and exterior penalty methods. The interior penalty method is usually called the barrier function method, which is the main part of the well-known interior point method, while the exterior one focuses on optimizing the penalty function, where the nonnegative penalty term is only positive outside the constraint sets. The exact penalty method, as a special exterior penalty method, aims at optimizing the penalty function with a suitably selected gain in order to achieve the exact optimization solution of the original constrained problem. Few effective distributed designs with the exact penalty have been studied. For example, DeMiguel and Murray [7] introduced an l_1 exact penalty function to a modified collaborative optimization problem and analyzed the sensitivity and singularity of the problem. Then, Liang *et al.* [8] also used an l_1 exact penalty function to design a simplified distributed optimization algorithm with coupled constraints, while Zhou *et al.* [23] designed a simple distributed algorithm with some *a priori* knowledge of local objective functions.

Adaptive control is widely used to handle systems with uncertain parameters [9]–[11], and adaptive ideas were used in the distributed optimization design. For example, Lou *et al.* [2] employed an adaptive scheme to estimate the unbalanced weights of a multiagent digraph for the exact optimization solution, while Li *et al.* [16] updated the Lagrange multiplier for an optimization problem of twice differentiable objective functions without any constraints. Moreover, Towfic and Sayed [15] took a penalty method for constrained optimization problems with two adaptive algorithms to optimize the penalty function, where the objective function and the penalty function were assumed to be twice differentiable without set constraints.

This paper aims at providing an adaptive exact penalty method to solve a distributed constrained convex optimization problem. Here, an adaptive idea is first carried out on the penalty term in the distributed design. Based on the design idea to update the penalty gain in order to make sure that the penalty function is the exact one, we propose a distributed continuous-time algorithm for the distributed constrained optimization problem. Then, we strictly prove its convergence with help of nonsmooth analysis.

This paper is organized as follows. Section II gives necessary preliminaries, while Section III formulates our problem and proposes our algorithm. Then, Section IV presents the main results along with their proofs. Finally, Section V provides concluding remarks.

Notations: Denote $\mathbb{R}^n$ as the real Euclidean space with dimension n, $\mathbb{R}^N_{>0}$ as all the vectors in $\mathbb{R}^N$ with all elements be positive, and 1_N as the vector in $\mathbb{R}^N$ with all elements as 1. $A = (a_{ij})_{N\times n} \in \mathbb{R}^{N\times n}$ is an $N \times n$ real matrix, where a_{ij} is the ith row and jth column element of A. A^{T} denotes its transpose. With $\Omega \subset \mathbb{R}^n$, denote $\mathrm{int}(\Omega)$ as its interior, $bd(\Omega)$ as its boundary, and $N_\Omega(x) = \{z | z^{\mathrm{T}}(y - x) \le 0,\ y \in \Omega\}$ as its normal cone at x. For $x \in \mathbb{R}^d$, $\|x\| = \sqrt{x^{\mathrm{T}}x}$, $d(x,\Omega) = \inf\{\|y - x\||, y \in \Omega\}$, and $B(x;r) = \{y | \|y - x\| \le r\}$. Moreover, $\otimes$

denotes the Kronecker product and $\times$ denotes the Cartesian product. $\partial f(x)$ denotes the subdifferential of f at x, while $\partial_x \tilde{f}(x, c)$ denotes the subdifferential of $f(\cdot, c)$ at x. f is M-Lipschitz on $\mathbb{R}^n$ if f is Lipschitz continuous with a constant M, that is, $|f(x) - f(y)| \leq M\|x - y\|$ for $x, y \in \mathbb{R}^n$.

II. Preliminaries

In this section, we present some necessary preliminaries on graph theory, convex analysis, and differential inclusions.

A multiagent network can be described by an undirected graph $\mathcal{G} = (V, E)$, where $V = \{v_1, \ldots, v_N\}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges. $A = (a_{ij})_{N\times N} \in \mathbb{R}^{N\times N}$ is the adjacency matrix, satisfying $a_{ij} > 0$ if $(v_i, v_j) \in E$ and $a_{ij} = a_{ji}$. $\mathcal{G}$ is said to be connected if, for any $i \neq j$, there exist $i_1, \ldots, i_k$ such that $(v_i, v_{i_1}) \in E$, $(v_{i_k}, v_j) \in E$ and $(v_{i_s}, v_{i_{(s+1)}}) \in E$ for $s = 1, \ldots, k-1$. The Laplacian matrix is given by $L = D - A$, where $D = \text{diag}\{D_{11}, \ldots, D_{NN}\}$ and $D_{ii} = \sum_{j=1}^N a_{ij}$. It is well known that if $\mathcal{G}$ is connected, then L and $L \otimes I_n$ are positive semidefinite, and $Lx = 0$ if and only if $x_1 = \ldots = x_N$ with $x = [x_1, \ldots, x_N]^{\mathrm{T}} \in \mathbb{R}^N$. Moreover, $y^{\mathrm{T}}(L \otimes I_n)y = 0$ if and only if $(L \otimes I_n)y = 0$ with $y = [y_1^{\mathrm{T}}, \ldots, y_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{nN}$.

For a closed convex set Ω of $\mathbb{R}^n$, we give the following result, whose proof can be found in [22, Proposition 18.22] or [23, Lemma 4.4].

Lemma 2.1: For a closed convex set Ω, the distance function $d(x, \Omega)$ is a convex function on $\mathbb{R}^n$ with

$$\partial d(x, \Omega) = \begin{cases} \{0\}, \ x \in \text{int}(\Omega) \\ N_\Omega(x) \bigcap B(0; 1), \ x \in bd(\Omega) \\ \{\frac{x - P_\Omega(x)}{d(x, \Omega)}\}, \ x \notin \Omega. \end{cases} \tag{1}$$

Clearly, if $x \in \Omega$, then $\partial d(x, \Omega) \subset N_\Omega(x)$.

A continuous function $f(x) : \mathbb{R}^n \to \mathbb{R}$ is said to be *convex* if

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

for any $x, y \in \mathbb{R}^n$ and $\alpha \in (0, 1)$. $f(x)$ is *strongly convex* (μ-strongly convex) if there is $\mu > 0$ such that $f(x) - \frac{\mu}{2}\|x\|^2$ is convex.

For a convex function $f(x)$, its subdifferential at x is $\partial f(x) = \{g | f(y) \geq f(x) + g^{\mathrm{T}}(y - x) \forall y \in \mathbb{R}^n\}$. It is well known that $\partial f(x) \neq \emptyset$ and $g^{\mathrm{T}}(y - x) \leq f(y) - f(x)$ for any $g \in \partial f(x)$.

Next, we introduce a modified version of [20, Proposition 1.5.3], whose proof is omitted.

Lemma 2.2: Let Ω_i, $i = 1, \ldots, N$, be closed subsets of $\mathbb{R}^n$ with nonempty intersection, and let function f be M-Lipschitz continuous over $\mathbb{R}^n$. Let $\bar{a} \in \mathbb{R}^N_{>0}$ with $\bar{a}_i \geq M + \sum_{j=1}^{i-1} \bar{a}_j$, $i = 2, \ldots, N$. Then, for all $a \in \mathbb{R}^N$ satisfying $a_i \geq \bar{a}_i$, the set of minima of f over $\Omega_0 = \cap_{i=1}^N \Omega_i$ coincides with the set of minima of

$$f(x) + \sum_{i=1}^N a_i d(x, \Omega_i)$$

over $\mathbb{R}^n$.

A differential inclusion is given by

$$\dot{x}(t) \in \mathcal{F}(x(t)), x(0) = x_0, t \geq 0 \tag{2}$$

where $\mathcal{F}$ is a set-valued map from $\mathbb{R}^q$ to subsets of $\mathbb{R}^q$. A Caratheodory solution of (2) defined on $[0, \tau] \subset [0, \infty)$ is an absolutely continuous function $x : [0, \tau] \to \mathbb{R}^q$ such that (2) holds for almost all $t \in [0, \tau]$ with $\tau > 0$. The solution $t \mapsto x(t)$ to (2) is a *right maximal solution* if it cannot be extended forward in time. Suppose that all right maximal solutions to (2) exist on $[0, \infty)$. A set $\mathcal{M}$ is said to be *weakly invariant* (respectively, *strongly invariant*) with respect to (2) if, for every $x_0 \in \mathcal{M}$, $\mathcal{M}$ contains a maximal solution (respectively, all maximal solutions) of (2). For more details about differential inclusion, see [25].

An equilibrium of (2) is a point $x_e \in \mathbb{R}^q$ with $0_q \in \mathcal{F}(x_e)$. It is easy to see that x_e is an equilibrium of (2) if and only if the function $x(\cdot) = x_e$ is a solution of (2). Let $V : \mathbb{R}^q \to \mathbb{R}$ be a locally Lipschitz continuous function, and the *set-valued Lie derivative* [21] $\mathcal{L}_\mathcal{F} V : \mathbb{R}^q \to \mathfrak{B}(\mathbb{R})$ of V with respect to (2) is defined as $\mathcal{L}_\mathcal{F} V(x) \triangleq \{a \in \mathbb{R} :$ there exists $v \in \mathcal{F}(x)$ such that $p^{\mathrm{T}} v = a$ for all $p \in \partial V(x)\}$. $V(x)$ is said to be *regular* if the one-sided directional derivative $V'(x; d) := \lim_{t\downarrow 0} \frac{V(x+td)-V(x)}{t}$ exists and equals the generalized directional derivative $V^\circ(x; d) := \limsup_{y\to x, t\downarrow 0} \frac{V(y+td)-V(y)}{t}$ at any point x for any direction d (see [21, p. 39]), and it is known that any convex function is regular [21]. When $\mathcal{L}_\mathcal{F} V(x)$ is nonempty, denote $\max \mathcal{L}_\mathcal{F} V(x)$ as the largest element of $\mathcal{L}_\mathcal{F} V(x)$.

Then, we give a lemma on the Caratheodory solutions to (2) [24].

Lemma 2.3: There is a Caratheodory solution to (2) for any initial state if $\mathcal{F}$ takes nonempty compact convex values, and also, it is upper semicontinuous and locally bounded.

Next, we introduce a version of the invariance principle, based on nonsmooth regular functions.

Lemma 2.4: For (2), suppose that $\mathcal{F}$ is upper semicontinuous and locally bounded, and $\mathcal{F}(x)$ takes nonempty, compact, and convex values. Let $V : \mathbb{R}^q \to \mathbb{R}$ be a locally Lipschitz and regular function, $\mathcal{S} \subset \mathbb{R}^q$ be compact and strongly invariant for (2), $\phi(\cdot) \in \mathcal{S}$ be a Caratheodory solution of (2)

$$\mathcal{R} = \{x \in \mathbb{R}^q : 0 \in \mathcal{L}_\mathcal{F} V(x)\}$$

and $\mathcal{M}$ be the largest weakly invariant subset of $\overline{\mathcal{R}} \cap \mathcal{S}$, where $\overline{\mathcal{R}}$ is the closure of $\mathcal{R}$. If there exists $T = T(\phi(0)) \geq 0$ such that $\max \mathcal{L}_\mathcal{F} V(\phi(t)) \leq 0$ for all $t \geq T$, then $d(\phi(t), \mathcal{M}) \to 0$ as $t \to +\infty$.

Although this lemma is not exactly [24, Th. 2] since $T = 0$ in [24, Th. 2], the results are equivalent.

III. Problem Formulation and Algorithm

A. Problem Formulation

Consider a network of N agents described by a graph $\mathcal{G} = (V, E)$ and an optimization problem

$$\min_x f(x) = \sum_{i=1}^N f_i(x_i), x = (x_1^{\mathrm{T}}, \ldots, x_N^{\mathrm{T}})^{\mathrm{T}}$$
$$\text{s.t.} \ \ x_i \in \Omega_i, x_i = x_j, i, j \in \{1, \ldots, N\} \tag{3}$$

where $x_i \in \mathbb{R}^n$ is a variable, Ω_i is a closed and convex constraint set, and $f_i(x_i)$ is a local objective function for $i = 1, \ldots, N$. In this problem, information of f_i and Ω_i is only known to agent i. The goal is to propose a distributed algorithm for agents to cooperatively minimize the global function $f(x) = \sum_{i=1}^N f_i(x_i)$ via communications in the network.

The following assumptions are standard in the distributed optimization literature [1], [3].

Assumption 3.1: The communication graph is undirected and connected.

Assumption 3.2: Suppose problem (3) has a finite solution, and one of following conditions holds.

a) For each i, f_i is convex and Lipschitz continuous on $\mathbb{R}^n$.
b) For each i, f_i is strongly convex and $\text{int}(\bigcap_{i=1}^N \Omega_i) \neq \emptyset$.
c) For each i, $f_i(x)$ is a convex quadratic function and Ω_i is a convex compact set such that $\text{int}(\bigcap_{i=1}^N \Omega_i) \neq \emptyset$.

B. Distributed Algorithm

Then, we propose a distributed algorithm based on the exact penalty method for problem (3).

To illustrate the idea of our approach, we take case (a) of Assumption 3.2 as an example. Let us consider $\sum_{i=1}^N f_i(x_i) + c_i d(x_i, \Omega_i)$. Suppose that f_i is M_i-Lipschitz on $\mathbb{R}^n$, and if $c_i > \sum_{j=1}^N M_j + \sum_{j=1}^{i-1} c_j$ and $\bar{x} = (\tilde{x}^{\mathrm{T}}, \ldots, \tilde{x}^{\mathrm{T}})^{\mathrm{T}}$ is a minimum of $\sum_{i=1}^N f_i(x_i) + c_i d(x_i, \Omega_i)$ with $x_i = x_j$, then $\bar{x} \in \mathrm{argmin}_{x_i \in \Omega_i, x_i = x_j} f(x)$ according to Lemma 2.2. However, M_i is unknown in our case, and therefore, how to select c_i suitably to make a minimum of $\sum_{i=1}^N f_i(x_i) + c_i d(x_i, \Omega_i)$ with $x_i = x_j$ be a solution of $\mathrm{argmin}_{x_i \in \Omega_i, x_i = x_j} \sum_{i=1}^N f_i(x_i)$ is challenging.

Here, we propose a distributed adaptive algorithm by updating c as follows:

$$\begin{cases} \dot{x_i} & \in -(\partial f_i(x_i) + c_i \partial d(x_i, \Omega_i) + \sum_{j=1}^N a_{ij} \cdot \\ & \quad (\lambda_i - \lambda_j) + \sum_{j=1}^N a_{ij}(x_i - x_j)) \\ \dot{\lambda_i} & = \sum_{j=1}^N a_{ij}(x_i - x_j) \\ \dot{c_i} & = d(x_i, \Omega_i) \end{cases} \tag{4}$$

where $c_i(0) > 0$, $x_i(0)$, $\lambda_i(0) \in \mathbb{R}^n$ for $i = 1, \ldots, N$, x_i is to estimate the optimal solution for agent i, λ_i is to estimate the Lagrangian multiplier for agent i, and c_i is the adaptive penalty gain of agent i.

Algorithm (4) is a continuous-time optimization algorithm. Continuous-time algorithms, which can be implemented by electricity circuits, are increasingly popular in the field of systems and control. For example, continuous-time algorithms are widely used in applications such as economic dispatch of power systems (see [30] and [31]) and neural networks (see [32] and [33]).

Remark 3.1: If we apply the projection method here, we get $\dot{x_i} \in P_{\Omega_i}(x_i - \partial f_i(x_i) - \sum_{j=1}^N a_{ij} \cdot (\lambda_i - \lambda_j) - \alpha \sum_{j=1}^N a_{ij}(x_i - x_j)) - x_i$. However, it is not easy to check if there is a Caratheodory solution because the projection part may not be convex. For example, let $f(x_1, x_2) = |x_1| + |x_2|$ and $\Omega = \{(x_1, x_2) | x_2 \geq |x_1| + 1\}$. If $x_1 = x_2 = 0, \lambda_1 = \lambda_2$, then $\partial f(0,0) = [-1,1] \times [-1,1]$. However, $P_\Omega(-\partial f(0,0)) = \{(x_1, x_2) | x_2 = |x_1| + 1,\ x_1 \in [-0,5, 0.5]\}$ is not a convex set.

Denote $\tilde{f}_i(x_i, c_i) = f_i(x_i) + c_i d(x_i, \Omega_i)$, $\tilde{f}(x, c) = \sum_{i=1}^N (f_i(x_i) + c_i d(x_i, \Omega_i))$, and $\eta = (x^{\mathrm{T}}, \lambda^{\mathrm{T}}, c^{\mathrm{T}})^{\mathrm{T}}$, where $x = (x_1^{\mathrm{T}}, \ldots, x_N^{\mathrm{T}})^{\mathrm{T}}$, $\lambda = (\lambda_1^{\mathrm{T}}, \ldots, \lambda_N^{\mathrm{T}})^{\mathrm{T}}$, and $c = (c_1, \ldots, c_N)^{\mathrm{T}}$. Rewrite (4) in a compact form

$$\dot{\eta}(t) \in \mathcal{F}(\eta(t)) \tag{5}$$

where $\eta(0) = (x_0^{\mathrm{T}}, \lambda_0^{\mathrm{T}}, c_0^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{nN} \times \mathbb{R}^{nN} \times \mathbb{R}^N_{>0}$, and

$$\mathcal{F}(\eta) = \begin{bmatrix} -\partial_x \tilde{f}(x, c) - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}} \lambda \\ \tilde{\mathcal{L}} x \\ D(x, \bar{\Omega}) \end{bmatrix}$$

$\bar{\Omega} = \Omega_1 \times \cdots \times \Omega_N$, $D(x, \bar{\Omega}) = (d(x_1, \Omega_1), \ldots, d(x_N, \Omega_N))^{\mathrm{T}}$, $\tilde{\mathcal{L}} = L \otimes I_n$, and L is a Laplacian matrix of the communication graph.

Lemma 3.1: There is a Caratheodory solution to (5).

Its proof can be easily obtained by Lemma 2.3 and the fact that the set function $-\partial_x \tilde{f}(x, c) - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}} \lambda$ is compact convex valued, upper semicontinuous, and locally bounded for any (x, λ, c).

Note that $\partial_x \tilde{f}(x, c)$ in Algorithm (5) depends on $c = (c_1, \ldots, c_N)$, which makes it difficult to analyze, since c keeps changing in (5). We will show that the adaptive design guarantees the convergence of (5) in the following section.

Remark 3.2: Compared with distributed alternating direction method of multipliers (ADMM) algorithms, an advantage of Algorithm (5) is that it is applicable to problems with convex and continuous objective functions (which may be nonsmooth). Although distributed ADMMs can also deal with nonsmooth objective functions by using proximal operators (see [27]–[29]), objective functions need to have certain structures. On the other hand, a limitation of Algorithm (5) is that it cannot deal with noncontinuous convex objective functions. Note that distributed ADMMs only need lower semicontinuity of objective functions if the objective functions have certain structures.

IV. Main Results

In this section, we first consider a new problem before showing our main results

$$\min_x \tilde{f}(x, c) = \sum_{i=1}^N \tilde{f}_i(x_i, c_i)$$
$$\text{s.t. } x_i = x_j,\ i, j = 1, \ldots, N \tag{6}$$

where $\tilde{f}_i(x_i, c_i) = f_i(x_i) + c_i d(x_i, \Omega_i)$ and c is a constant vector in $\mathbb{R}^N$. The next result shows the relationship between problems (3) and (6).

Lemma 4.1: Under Assumption 3.2, there is a constant $c \in \mathbb{R}^N_{>0}$ such that $\mathrm{argmin}_{x \in \bar{\Omega},\ x_i = x_j} f(x) = \mathrm{argmin}_{x_i = x_j} \tilde{f}(x, c)$.

Proof: i) Suppose that Assumption 3.2(a) holds. Let f_i be M_i-Lipschitz. With Assumption 3.2, $\sum_{i=1}^N f_i(y)$ is $\sum_{i=1}^N M_i$-Lipschitz on $\mathbb{R}^n$. Let $\bar{a}$ be the same as in Lemma 2.2. Then, for $c \in \mathbb{R}^N$ with $c_i > \bar{a}_i$, $y \in \mathbb{R}^n \setminus \Omega_0$, $\min_{z \in \mathbb{R}^n \cap \Omega_0} \sum_{i=1}^N f_i(z) < \sum_{i=1}^N (f_i(y) + c_i d(y, \Omega_i))$ due to Lemma 2.2. Thus, $\mathrm{argmin}_{x_i = x_j,\ x_i \in \Omega_i} \tilde{f}(x, c) = \mathrm{argmin}_{x_i = x_j} \tilde{f}(x, c)$. The conclusion follows because $\tilde{f}(x, c) = f(x)$ on $\bar{\Omega}$ and $\tilde{f}(x, c) > f(x)$ otherwise.

ii) Suppose that Assumption 3.2(b) holds. Note that, if $s \in \mathrm{int}(\cap_{i=1}^N \Omega_i)$, denote $\bar{s} = (s^{\mathrm{T}}, \ldots, s^{\mathrm{T}})^{\mathrm{T}}$ and $\partial f(\bar{s}) = \partial_x \tilde{f}(\bar{s}, c)$ for any c because $\partial d(s, \Omega_i) = 0$ for $s \in \mathrm{int}(\Omega_i)$. Thus, if f_i is μ_i-strongly convex and $s \in \mathrm{int}(\cap_{i=1}^N \Omega_i)$ and $M = \sup\{\|g_i\| | g_i \in \partial f_i(s),\ i = 1, \ldots, N\}$, then, for any $c \in \mathbb{R}^N_{>0}$, $\mathrm{argmin}_{x \in \bar{\Omega},\ x_i = x_j} f(x)$ and $\mathrm{argmin}_{x_i = x_j} \tilde{f}(x, c)$ both stay in the bounded set $C_s = \{x | \|x_i - s\| \leq \frac{2M}{\mu},\ x_i = x_j\}$, where $\mu = \min\{\mu_1, \ldots, \mu_N\}$. It follows from the convexity of f_i that f_i is Lipschitz continuous on the compact set C_s. If f_i is M_i-Lipschitz, then $\sum_{i=1}^N f_i(y)$ is Lipschitz continuous with the Lipschitz constant $\sum_{i=1}^N M_i$. Using similar analysis to (i), we conclude that $\mathrm{argmin}_{x \in \bar{\Omega},\ x_i = x_j,\ x \in C_s} f(x) = \mathrm{argmin}_{x_i = x_j, x \in C_s} \tilde{f}(x, c)$. Therefore, we have

$$\begin{aligned} \mathrm{argmin}_{x \in \bar{\Omega},\ x_i = x_j} f(x) &= \mathrm{argmin}_{x \in \bar{\Omega},\ x_i = x_j,\ x \in C_s} f(x) \\ &= \mathrm{argmin}_{x_i = x_j, x \in C_s} \tilde{f}(x, c) \\ &= \mathrm{argmin}_{x_i = x_j} \tilde{f}(x, c). \end{aligned} \tag{7}$$

iii) Suppose that Assumption 3.2 (c) holds. Define $f_0(y) = f(1_N \otimes y)$ and $\Omega = \bigcap_{i=1}^N \Omega_i$, where $y \in \mathbb{R}^n$. One only needs to show $\mathrm{argmin}_{y \in \Omega} f_0(y) = \mathrm{argmin}_{y \in \mathbb{R}^n} [f_0(y) + \sum_{i=1}^N c_i d(y, \Omega_i)]$. Let $\hat{y} \in \mathrm{argmin}_{y \in \Omega} f_0(y)$. Since f_0 is a convex quadratic function and the distance function is invariant under orthogonal transformation $(d(Oy, O\Omega) = d(x, \Omega))$, we assume $f_0(y) = \sum_{j=1}^k a_j(y_j - b_j)^2 + \sum_{j=k+1}^n a_j y_j + a_{n+1}$, where $a_1, \ldots, a_k > 0$. Denote $\tilde{\Omega} = \{y | f_0(y) \leq f_0(\hat{y})\}$. Then, $\tilde{\Omega}$ is a convex set because of the convexity of f_0. Since Ω and the interior of $\tilde{\Omega}$ do not intersect, there is a support hyperplane separating Ω and $\tilde{\Omega}$. Denote $\sum_{j=k+1}^m a_j \hat{y}_j + a_{m+1} = \hat{p}$. Pick y such that $y \in \tilde{\Omega}$ and $y \notin \Omega$. Define $\bar{y} = [\bar{y}_1, \ldots, \bar{y}_N]^{\mathrm{T}} \in \mathbb{R}^n$ such that $\bar{y}_i = \hat{y}_i$ for $i = 1, \ldots, k$ and $\bar{y}_j = y_j$ for $j > k$. Denote $\sum_{j=k+1}^m a_j y_j + a_{m+1} = p$ and $\hat{\Omega} = \{(y_1, \ldots, y_N) | y_i = \hat{y}_i,\ i = 1, \ldots, k,\ \sum_{j=k+1}^m a_j y_j = \hat{p}\}$. Clearly, $\hat{\Omega}$ is a support hyperplane separating Ω and $\tilde{\Omega}$.

Then, we have $d(y,\Omega) \geq d(y,\hat{\Omega}) \geq d(\bar{y},\hat{\Omega}) = \gamma|p-\hat{p}|$, where $\gamma > 0$ is constant determined by $a_{k+1},\ldots,a_N$. For $c_0 > \frac{1}{\gamma}$, we have that $f_0(y) + c_0 d(y,\Omega) \geq \sum_{j=1}^k a_j(y_j-b_j)^2 + p + c_i\gamma|p-\hat{p}| \geq \sum_{i=1}^k a_i(y_i-b_i)^2 + \hat{p}$. By applying [18, Th. 1] repeatedly, there exist $c_1,\ldots,c_n$ such that $\sum_{i=1}^N c_i d(y,\Omega_i) \geq c_0 d(y,\Omega)$. Therefore, the minimum of $f_0(y) + \sum_{i=1}^N c_i d(y,\Omega_i)$ stays in the set $\{y \in \Omega | \sum_{i=1}^k a_i(y_i-b_i)^2 \leq \sum_{i=1}^k a_i(\hat{y}_i-b_i)^2\}$, under which f_0 is Lipschitz continuous. Hence, for sufficiently large c_i s, $\operatorname{argmin}_{y\in\Omega} f_0(y) = \operatorname{argmin} f_0(y) + \sum_{i=1}^N c_i d(y,\Omega_i)$.

Then, we provide some properties of solutions to Problem (6). ■

Lemma 4.2 (see[19, Th. 3.34]): Suppose that $\hat{x}$ is a solution of (6), $\tilde{f}(x,c)$ is continuous at a feasible point x_0, and Slater's condition is satisfied. Then, there is $\hat{\lambda} \in \mathbb{R}^{nN}$ such that

$$0 \in \partial_x \tilde{f}(\hat{x},c) + \tilde{\mathcal{L}}\hat{\lambda}. \tag{8}$$

Conversely, if (8) holds for some $(\hat{x},\hat{\lambda})$ and $\tilde{\mathcal{L}}\hat{x} = 0$, then $\hat{x}$ is a solution of problem (6).

Since there is no inequality constraints in problem (3) [or (6)], Slater's condition holds. With Lemmas 4.1 and 4.2, we build a relation between a solution to Problem (3) and an equilibrium of Algorithm (5) as follows.

Theorem 4.1: Under Assumptions 3.1 and 3.2, if $(x^*,\lambda^*,c^*) \in \mathbb{R}^{nN} \times \mathbb{R}^{nN} \times \mathbb{R}^N$ is an equilibrium of Algorithm (5), then x^* is a solution to Problem (3). Conversely, if $\hat{x}$ is a solution to Problem (3), then there exist $\hat{\lambda},\ \hat{c}$ such that $(\hat{x},\hat{\lambda},\hat{c})$ is an equilibrium of Algorithm (5).

Proof: Sufficiency: Let (x^*,λ^*,c^*) be an equilibrium of Algorithm (5). Then, $\tilde{\mathcal{L}}x^* = 0,\ x^* \in \bar{\Omega}$ and $0 \in (\partial_x \tilde{f}(x^*,c^*) + \tilde{\mathcal{L}}x^* + \tilde{\mathcal{L}}\lambda^*) = \partial_x \tilde{f}(x^*,c^*) + \tilde{\mathcal{L}}\lambda^*$. By Lemma 4.2, $x^* \in \operatorname{argmin}_{x_i=x_j} \tilde{f}(x,c^*)$. Because $\tilde{f}(x,c^*) = \sum_{i=1}^N f_i(x_i) + c_i^* d(x_i,\Omega_i) = f(x)$ for $x \in \bar{\Omega} = \prod_{i=1}^N \Omega_i$, $x^* \in \operatorname{argmin}_{x_i=x_j, x_i\in\Omega_i} f(x)$, which implies that x^* is a solution to Problem (3).

Necessity: Let $\hat{x}$ be a solution to Problem (3). Then, $\tilde{\mathcal{L}}\hat{x} = 0$ and $d(\hat{x}_i,\Omega_i) = 0$. Let $\hat{c}$ satisfy the condition of c in Lemma 4.1. By Lemma 4.1, $\hat{x}$ is also a solution to Problem (6) with $c = \hat{c}$. From Lemma 4.2, there exists $\hat{\lambda}$ such that $0 \in (\partial_x \tilde{f}(\hat{x},\hat{c}) + \tilde{\mathcal{L}}\hat{\lambda}) = (\partial_x \tilde{f}(\hat{x},\hat{c}) + \tilde{\mathcal{L}}\hat{x} + \tilde{\mathcal{L}}\hat{\lambda})$, which implies that $(\hat{x},\hat{\lambda},\ \hat{c})$ is an equilibrium of Algorithm (5).

Remark 4.1: With Assumption 3.2, there is a solution to Problem (3). By Theorem 4.1, there is an equilibrium of Algorithm (5). ■

Define functions $V_1(x,\lambda,c) = \frac{1}{2}(\|x-x^*\|^2 + \|\lambda-\lambda^*\|^2 + \|c-c^*\|^2)$ and $V_2(x,\lambda,c) = \tilde{f}(x,c) - \tilde{f}(x^*,c^*) + \frac{1}{2}x^{\mathrm{T}}\tilde{\mathcal{L}}x + (x-x^*)^{\mathrm{T}}\tilde{\mathcal{L}}\lambda + 1_N^{\mathrm{T}}(\overline{c}-c) + \frac{\|c^*\|_1}{2}$ for some $\overline{c} \in \mathbb{R}^N$, where (x^*,λ^*,c^*) is an equilibrium point of Algorithm (5). We provide a result for the properties of some Lyapunov-like functions.

Lemma 4.3: Suppose that Assumptions 3.1 and 3.2 hold. Then, we have the following.

i) The trajectory of V_1 along (5) satisfies

$$\max \mathcal{L}_{\mathcal{F}} V_1(x,\lambda,c) \leq -x^{\mathrm{T}}\tilde{\mathcal{L}}x \leq 0.$$

ii) There exists $\overline{c} \in \mathbb{R}^N$ such that $\bar{c}$ is an upper bound of c and the trajectory of V_2 along (5) satisfies that, for any $v_2 \in \mathcal{L}_{\mathcal{F}} V_2(x,\lambda,c)$

$$v_2 = -\|\tilde{v}_1\|^2 + \|\tilde{\mathcal{L}}x\|^2 + \sum_{i=1}^N d(x_i(t),\Omega_i)^2 - \sum_{i=1}^N d(x_i(t),\Omega_i)$$

for some $\tilde{v}_1 \in -\partial_x \tilde{f}(x,c) - \tilde{\mathcal{L}}x - \tilde{\mathcal{L}}\lambda$.

iii) With $0 < \alpha < \frac{1}{\lambda_{\max}(\tilde{\mathcal{L}})}$, $\beta = (1-\alpha\lambda_{\max}(\tilde{\mathcal{L}}))$, there exists $T(x_0,\lambda_0,c_0) > 0$ such that the trajectory of $V_1(x(t),\lambda(t),c(t)) + \alpha V_2(x(t),\lambda(t),c(t))$ along (5) satisfies $V_1(x(t),\lambda(t),c(t)) + \alpha V_2(x(t),\lambda(t),c(t)) \geq 0$, for $t \geq T(x_0,\lambda_0,c_0)$ and for any $v_1 \in \mathcal{L}_{\mathcal{F}} V_1(x,\lambda,c)$, $v_2 \in \mathcal{L}_{\mathcal{F}} V_2(x,\lambda,c)$, $t \geq T(x_0,\lambda_0,c_0)$

$$v_1 + \alpha v_2 \leq -\alpha\|\tilde{v}_1\|^2 - \beta x(t)^{\mathrm{T}}\tilde{\mathcal{L}}x(t) - 0.5\alpha\sum_{i=1}^N d(x_i(t),\Omega_i) \tag{9}$$

for some $\tilde{v}_1 \in -\partial_x \tilde{f}(x,c) - \tilde{\mathcal{L}}x - \tilde{\mathcal{L}}\lambda$.

Its proof is given in the Appendix.

Remark 4.2: By Lemma 4.3(i), we conclude that V_1 is bounded, which results in the boundedness of c. Thus, there exists $\bar{c} \in \mathbb{R}^N$ such that $\bar{c}_i > c_i(t)$ for all $t \geq 0$.

Then, it is time to introduce the convergence result.

Theorem 4.2: Under Assumptions 3.1 and 3.2, for any $(x_0,\lambda_0,c_0) \in \mathbb{R}^{nN} \times \mathbb{R}^{nN} \times \mathbb{R}^N_{>0}$, $(x(t),\lambda(t),c(t))$ converges to an equilibrium of Algorithm (5). Particularly, $x(t)$ converges to an optimal solution of Problem (3).

Proof: Let $V(x,\lambda,c) = V_1(x,\lambda,c) + \alpha V_2(x,\lambda,c)$, where V_1, V_2, and α are defined in Lemma 4.3. It follows from Lemma 4.3 that, for any $v \in \mathcal{L}_{\mathcal{F}} V(x,\lambda,c)$, $V(x,\lambda,c) \geq 0$ and

$$v \leq -\alpha\|\tilde{v}_1\|^2 - \beta x^{\mathrm{T}}\tilde{\mathcal{L}}x - 0.5\alpha\sum_{i=1}^N d(x_i,\Omega_i)$$

for $t \geq T(x_0,\lambda_0,c_0)$, where T and β are the same as given in Lemma 4.3. Thus, $\max \mathcal{L}_{\mathcal{F}} V(x,\lambda,c) \leq 0$ for $t \geq T(x_0,\lambda_0,c_0)$. By Lemma 2.4, $(x(t),\lambda(t),c(t))$ converges to $\mathcal{M}$, where $\mathcal{M}$ is the largest weakly invariant set of $\bar{\mathcal{R}} \cap S$, $\mathcal{R} = \{(x,\lambda,c) | 0 \in \mathcal{L}_{\mathcal{F}} V(x,\lambda,c)\}$ and $S = \{(x,\lambda,c) | V(x,\lambda,c) \leq V(x_0,\lambda_0,c_0)\}$. Let $(x,\lambda,c) \in \mathcal{M}$. By the definition of $\mathcal{M}$ and (9), (x,λ,c) satisfies $\|\tilde{v}_1\|^2 = 0,\ x^{\mathrm{T}}\tilde{\mathcal{L}}x = 0$ and $d(x_i,\Omega_i) = 0$, for some $\tilde{v}_1 \in -\partial_x \tilde{f}(x,c) - \tilde{\mathcal{L}}x - \tilde{\mathcal{L}}\lambda$. Then, $0 \in -\partial_x \tilde{f}(x,c) - \tilde{\mathcal{L}}x - \tilde{\mathcal{L}}\lambda,\ \tilde{\mathcal{L}}x = 0$, and $d(x_i,\Omega_i) = 0$. Therefore, (x,λ,c) is an equilibrium of Algorithm (5), which implies that any point in $\mathcal{M}$ is an equilibrium of Algorithm (5).

Let $\phi(t) = (x(t),\lambda(t),c(t))$ be a Caratheodory solution of (5). Since $\phi(\cdot)$ is bounded by $\max \mathcal{L}_{\mathcal{F}} V(x,\lambda,c) \leq 0$ for $t \geq T(x_0,\lambda_0,c_0)$, there exists $(\tilde{x},\tilde{\lambda},\tilde{c})$ and $\{t_k, k = 1,2,...\}$ such that $\phi(t_k) = (x(t_k),\lambda(t_k),c(t_k)) \to (\tilde{x},\tilde{\lambda},\tilde{c})$ as $k \to \infty$. Then, $(\tilde{x},\tilde{\lambda},\tilde{c}) \in \mathcal{M}$ since $\operatorname{dist}(\phi(t),\mathcal{M}) \to 0$. Thus, $(\tilde{x},\tilde{\lambda},\tilde{c})$ is an equilibrium of Algorithm (5). On the other hand, each equilibrium of Algorithm (5) is Lyapunov stable, since $V_1(x,\lambda,c) > 0$ for $(x,\lambda,c) \neq (x^*,\lambda^*,c^*)$ and $\max \mathcal{L}_{\mathcal{F}} V_1(x,\lambda,c) \leq 0$. Hence, for any $\epsilon > 0$, there is $\delta > 0$ such that $(x_0,\lambda_0,c_0) \in B((\tilde{x},\tilde{\lambda},\tilde{c}),\delta)$ implies $(x(t),\lambda(t),c(t)) \in B((\tilde{x},\tilde{\lambda},\tilde{c}),\epsilon)$ for all $t \geq 0$. Due to $\phi(t_k) = (x(t_k),\lambda(t_k),c(t_k)) \to (\tilde{x},\tilde{\lambda},\tilde{c})$, there exists $m > 0$ such that $\phi(t_m) = (x(t_m),\lambda(t_m),c(t_m)) \in B((\tilde{x},\tilde{\lambda},\tilde{c}),\delta)$, and then $(x(t),\lambda(t),c(t)) \in B((\tilde{x},\tilde{\lambda},\tilde{c}),\epsilon)$ for all $t \geq t_m$. Thus, $(x(t),\lambda(t),c(t))$ converges to an equilibrium of Algorithm (5). ■

Remark 4.3: Proving the convergence is not easy due to the nonsmoothness of $d(x,\Omega)$. We first carefully construct a Lyapunov-like function V_1. Then, we make full use of the form of $\partial d(x,\Omega)$ and the convexity of $d(x,\Omega)$ to get $\mathcal{L}_{\mathcal{F}} V_1 \leq 0$. However, the set $\{(x,\lambda,c) | \mathcal{L}_{\mathcal{F}} V_1(x,\lambda,c) = 0\}$ is not contained in the equilibrium set of (5), which makes it difficult to prove the convergence. Then, we construct another Lyapunov-like function V_2. Finally, we get the convergence proof with the help of nonsmooth analysis.

Remark 4.4: The combination of exact penalty and adaptive ideas can also be used to deal with more general problems. For example, the design may also be extended to some optimization problems with inequality or equality constraints, such as the economic dispatch problem discussed in [26].

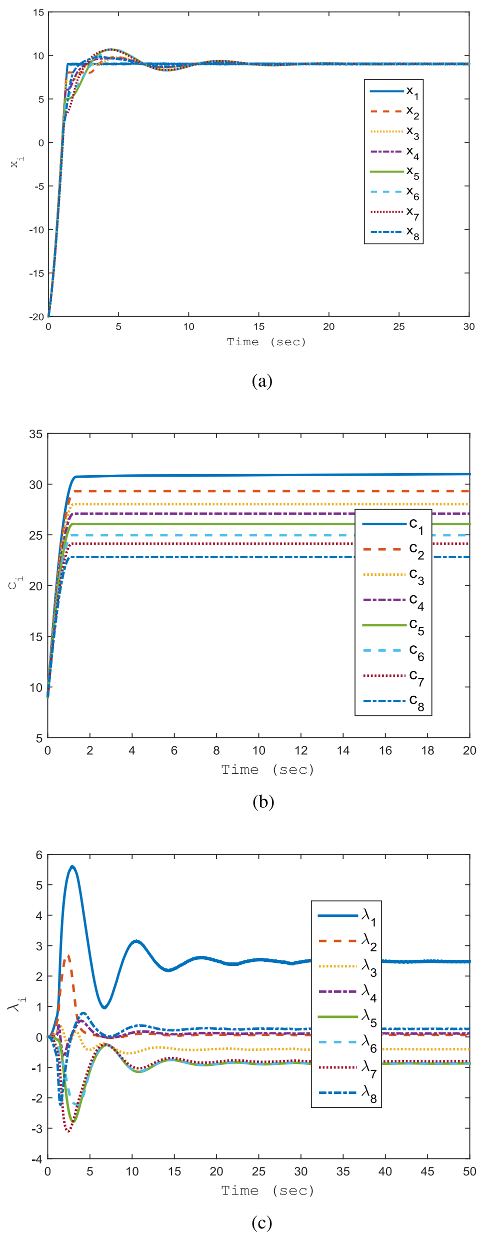

Fig. 1. Convergence property of Algorithm (5) for Example 4.1. (a) Trajectories of x_i's. (b) Trajectories of c_i's. (c) Trajectories of λ_i's.

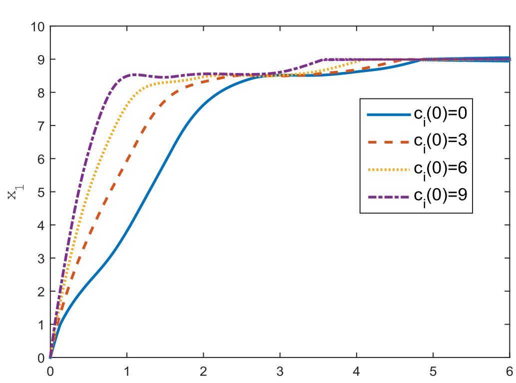

Fig. 2. Trajectories of x_1 for different choices of $c_i(0)$ for Example 4.1.

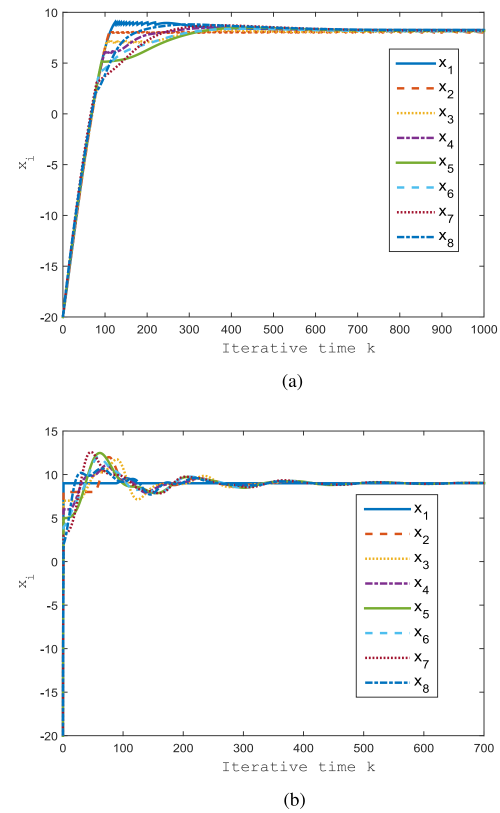

Fig. 3. Comparison of Algorithm (5) and PG-ADMM for Example 4.2. (a) Trajectories of x_i for algorithm (5). (b) Trajectories of x_i for distributed PG-ADMM.

Before the end of this section, we present some examples for Algorithm (5).

Example 4.1: Consider a network of eight agents and (3) with $x \in \mathbb{R}$ and $N = 8$. The local objective function for agent i is $f_i(x) = |x - i| + 1,\ \ i = 1, \ldots, 8$, and the local constraint for agent i is $\Omega_i = \{x \in \mathbb{R} | 10 - i \leq x \leq 10 + i\}$. The adjacency matrix of the communication graph $\mathcal{G}$ is given by

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}. \tag{10}$$

It can be verified that $\Omega_0 = \cap_{i=1}^{8} \Omega_i = [9, 11]$ and the optimal solution is $x_i = 9$, $i = 1, \ldots, 8$, which is on the boundary of $\bar{\Omega} = \prod_{i=1}^{8} \Omega_i$. The sampling time (step size) of this example is 0.01 s. Simulations are conducted in Figs. 1 and 2.

Fig. 1 shows the convergence of variables x_i, c_i, and λ_i, respectively. Also, x_i reaches consensus for $i = 1, \ldots, 8$, and moreover, $x = (x_1^{\mathrm{T}}, \ldots, x_N^{\mathrm{T}})^{\mathrm{T}}$ converges to the optimal point of Problem (3).

Fig. 2 shows the relationship between the performance of Algorithm (5) and the choice of $c_i(0)$. Specifically, fix $x_0 = [0, 0, 0, 0, 1, 1, 1, 1]$ and $\lambda_0 = [-0.6975, -0.1565, 0.7939, 0.7990, 1, 2, 3, 4]$. By changing $c_i(0)$ from 0 to 9, Fig. 2 shows that larger $c_i(0)$ results in faster numerical convergence performance.

Example 4.2: Consider an optimization problem with objective functions as $f_i(x) = \frac{1}{2}|x-i|^2 + 1,\ i = 1, \ldots, 8$, and all other parameters are the same as those of Example 4.1. We compare simulation results of Algorithm (5) and distributed proximal gradient (PG) ADMM in [28, Fig. 1].

Let sampling times (or step sizes) of both algorithms be 0.01 s. Define the initial condition of Algorithm (5) as $x_0 = [-20, -20, -20, -20, -20, -20, -20, -20]$, $\lambda_0 = [0, 0, 0, 0, 0, 0, 0, 0]$, and $c_0 = [0, 0, 0, 0, 0, 0, 0, 0]$. Let the initial condition of the PG-ADMM algorithm be $x_i^0 = -20$ and $p_i^0 = 0$ for all $i \in \{1, \ldots, n\}$. Since PG-ADMM is a discrete-time algorithm, we use iterative time k to show convergence performances. Simulation results of Algorithm (5) and the PG-ADMM algorithm are shown in Fig. 3. It shows that the PG-ADMM algorithm has a better numerical performance than that of Algorithm (5) for the same initial condition and step size.

V. Conclusion

In this paper, we focused on a class of constrained distributed optimization problems, where the local convex objective functions are nondifferentiable. To deal with the set constrains, we proposed a method by combining the exact penalty method and an adaptive idea. Then, we gave a distributed adaptive algorithm to solve the optimization problems by seeking a suitable penalty gain. Finally, we showed the convergence for the algorithm with nonsmooth analysis.

Appendix
Proof of Lemma 4.3

i) The trajectory of V_1 along (5) satisfies that $\mathcal{L}_{\mathcal{F}} V_1(x, \lambda, c) = (x - x^*)^{\mathrm{T}}(-\partial_x \tilde{f}(x, c) - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}}\lambda) + (\lambda - \lambda^*)^{\mathrm{T}} \tilde{\mathcal{L}} x + (c - c^*)^{\mathrm{T}} D(x, \bar{\Omega})$, where $\partial_x \tilde{f}(x, c) = \{g + Ch | g_i \in \partial f_i(x_i),\ h_i \in \partial d(x_i, \Omega_i),\ C = \mathrm{diag}\{c_1, \ldots, c_N\}\}$.

Let $g_i \in \partial f_i(x_i),\ h_i \in \partial d(x_i, \Omega_i)$. Denote $g = (g_1^{\mathrm{T}}, \ldots, g_N^{\mathrm{T}})^{\mathrm{T}}$, $h = (h_1^{\mathrm{T}}, \ldots, h_N^{\mathrm{T}})^{\mathrm{T}}$, and $H = (x - x^*)^{\mathrm{T}}(-g - Ch - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}}\lambda) + (\lambda - \lambda^*)^{\mathrm{T}} \tilde{\mathcal{L}} x + (c - c^*)^{\mathrm{T}} D(x, \bar{\Omega})$. Then, $\mathcal{L}_{\mathcal{F}} V_1(x, \lambda, c) = \{H | g_i \in \partial f_i(x_i),\ h_i \in \partial d(x_i, \Omega_i)\}$ and

$$\begin{aligned} H &= (x - x^*)^{\mathrm{T}}(-g - Ch - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}}\lambda) \\ &\quad + (\lambda - \lambda^*)^{\mathrm{T}} \tilde{\mathcal{L}} x + (c - c^*)^{\mathrm{T}} D(x, \bar{\Omega}) \\ &= (x - x^*)^{\mathrm{T}}(-g + g^* + C^* h^* + \tilde{\mathcal{L}}\lambda^* - Ch - \tilde{\mathcal{L}} x \\ &\quad - \tilde{\mathcal{L}}\lambda) + (\lambda - \lambda^*)^{\mathrm{T}} \tilde{\mathcal{L}} x + (c - c^*)^{\mathrm{T}} D(x, \bar{\Omega}) \\ &\le (x - x^*)^{\mathrm{T}}(C^* h^* - Ch) + (c - c^*)^{\mathrm{T}} D(x, \bar{\Omega}) \\ &\quad - x^{\mathrm{T}} \tilde{\mathcal{L}} x \\ &= \sum_{i=1}^{N} \Big((x_i - x_i^*)^{\mathrm{T}}(c_i^* h_i^* - c_i h_i) \\ &\quad + (c_i - c_i^*) d(x_i, \Omega_i) \Big) - x^{\mathrm{T}} \tilde{\mathcal{L}} x \end{aligned} \tag{11}$$

where $C^* = \mathrm{diag}\{c_1^*, \ldots, c_N^*\},\ g^* \in \partial f_i(x_i^*),\ h_i^* \in \partial d(x_i^*, \Omega_i)$; the second equality holds because x^* is the optimal point of $\{f(x) + c^* D(x, \bar{\Omega}) | x_i = x_j\}$ and Lemma 4.2. Moreover, the inequality holds because of the convexity of f_i and $\tilde{\mathcal{L}} x^* = 0$. Note that $\tilde{\mathcal{L}}$ is positive semidefinite under Assumption 3.1. If we can show $H_i \triangleq (x_i - x_i^*)^{\mathrm{T}}(c_i^* h_i^* - c_i h_i) + (c_i - c_i^*) d(x_i, \Omega_i) \le 0$ for any $h_i \in \partial d(x_i, \Omega_i)$ and each i, then $H \le \sum_{i=1}^{N} H_i - x^{\mathrm{T}} \tilde{\mathcal{L}} x \le 0$.

Then, we prove the conclusion in two cases: $x_i \in \Omega_i$ and $x_i \notin \Omega_i$.

1) $x_i \in \Omega_i$. In this case, $d(x_i, \Omega_i) = 0$ and $\partial d(x_i, \Omega_i) \subset N_{\Omega_i}(x_i)$. Note that $(x_i - x_i^*)^{\mathrm{T}} h_i^* \le 0$ and $-(x_i - x_i^*)^{\mathrm{T}} h_i \le 0$ because $h_i \in \partial d(x_i, \Omega_i),\ h_i^* \in \partial d(x_i^*, \Omega_i)$ and $\partial d(x_i^*, \Omega_i) \subset N_{\Omega_i}(x_i^*)$. Recall that $c_i,\ c_i^* > 0$ in Algorithm (5). Thus, $H_i \le 0$.

2) $x_i \notin \Omega_i$. In this case, $\partial d(x_i, \Omega_i) = \{\frac{x - P_{\Omega_i}(x_i)}{d(x_i, \Omega_i)}\}$, i.e. $h_i = \frac{x - P_{\Omega_i}(x_i)}{d(x_i, \Omega_i)}$. For $c_i \le c_i^*$, we have

$$\begin{aligned} H_i &= (x_i - x_i^*)^{\mathrm{T}}(c_i^* h_i^* - c_i h_i) + (c_i - c_i^*) d(x_i, \Omega_i) \\ &\le (x_i - x_i^*)^{\mathrm{T}}(c_i^* - c_i) h_i^* + (c_i - c_i^*) d(x_i, \Omega_i) \\ &\le (c_i^* - c_i) d(x_i, \Omega_i) + (c_i - c_i^*) d(x_i, \Omega_i) \\ &= 0 \end{aligned} \tag{12}$$

where the first inequality holds because $-(x_i - x_i^*)^{\mathrm{T}}(h_i - h_i^*) \le 0$ (due to the convexity of $d(x_i, \Omega_i)$) and the second inequality holds because $(x_i - x_i^*)^{\mathrm{T}} h_i^* \le d(x_i, \Omega_i) - d(x_i^*, \Omega_i) = d(x_i, \Omega_i)$ (due to the convexity of $d(x_i, \Omega_i)$) and $c_i \le c_i^*$.

For $c_i > c_i^*$, we have

$$\begin{aligned} H_i &= (x_i - x_i^*)^{\mathrm{T}}(c_i^* h_i^* - c_i h_i) + (c_i - c_i^*) d(x_i, \Omega_i) \\ &\le -(c_i - c_i^*)(x_i - x_i^*)^{\mathrm{T}} h_i + (c_i - c_i^*) d(x_i, \Omega_i) \\ &= -(c_i - c_i^*)(x_i - P_{\Omega_i}(x_i) + P_{\Omega_i}(x_i) - x_i^*)^{\mathrm{T}} \cdot \\ &\quad \frac{x_i - P_{\Omega_i}(x_i)}{d(x_i, \Omega_i)} + (c_i - c_i^*) d(x_i, \Omega_i) \\ &\le -(c_i - c_i^*)(x_i - P_{\Omega_i}(x_i))^{\mathrm{T}} \frac{x_i - P_{\Omega_i}(x_i)}{d(x_i, \Omega_i)} + \\ &\quad (c_i - c_i^*) d(x_i, \Omega_i) \\ &= 0 \end{aligned} \tag{13}$$

where the first inequality holds because $-(x_i - x_i^*)^{\mathrm{T}}(h_i - h_i^*) \le 0$ (due to the convexity of $d(x_i, \Omega_i)$) and the second inequality holds because $(P_{\Omega_i}(x_i) - x_i^*)^{\mathrm{T}}(x_i - P_{\Omega_i}(x_i)) \ge 0$ (a property of the projection operator) and $c_i > c_i^*$.

ii) By (i), the trajectory along (5) is bounded for any initial condition. Hence, there is $\bar{c} = \bar{c}(x_0, \lambda_0, c_0)$ with $\bar{c} \ge c(t)$ for all $t \ge 0$.

By definition, there exists $\tilde{v} \in \mathcal{F}(x, \lambda, c)$ such that $v_2 = p^{\mathrm{T}} \tilde{v}$ for all $p \in \partial V_2(x, \lambda, c)$. Denote $p = (p_1, p_2, p_3)$ and $\tilde{v} = (\tilde{v}_1, \tilde{v}_2, \tilde{v}_3)$, where $p_1 \in \partial_x V_2$, $p_2 \in \partial_\lambda V_2$ and $p_3 \in \partial_c V_2$, $\tilde{v}_1, \tilde{v}_2, \tilde{v}_3$ are the correspond components of $\tilde{v}$. Noticing that $\mathcal{F}(x, \lambda, c) = (-\partial_x V_2^{\mathrm{T}}, \partial_\lambda V_2^{\mathrm{T}}, \partial_c V_2^{\mathrm{T}})^{\mathrm{T}}$ by (5). Thus, by choosing $p_1 = -\tilde{v}_1, p_2 = \tilde{v}_2$, and $p_3 = \tilde{v}_3$, we have

$$\begin{aligned} v_2 &= p_1^{\mathrm{T}} \tilde{v}_1 + p_2^{\mathrm{T}} \tilde{v}_2 + p_3^{\mathrm{T}} \tilde{v}_3 \\ &= -\|\tilde{v}_1\|^2 + \|\tilde{\mathcal{L}} x\|^2 + \sum_{i=1}^{N} d(x_i(t), \Omega_i)^2 \\ &\quad - \sum_{i=1}^{N} d(x_i(t), \Omega_i). \end{aligned} \tag{14}$$

iii) Because $x^{\mathrm{T}} \tilde{\mathcal{L}} x \ge 0$ and $1_N^{\mathrm{T}}(\bar{c} - c) \ge 0$

$$\begin{aligned} V_2(x, \lambda, c) &\ge f(x) + c^{\mathrm{T}} D(x, \bar{\Omega}) - f(x^*) + (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} \lambda^* \\ &\quad + (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}}(\lambda - \lambda^*) + \frac{\|c^*\|_1}{2}. \end{aligned}$$

Moreover, we have

$$f(x) - f(x^*) + (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} \lambda^* \ge (g^* + \tilde{\mathcal{L}} \lambda^*)^{\mathrm{T}}(x - x^*).$$

By Lemma 4.2 and Theorem 4.1, $0 \in \partial f(x^*) + \sum_{i=1}^{N} c_i^* \partial d(x_i^*, \Omega_i) + \tilde{\mathcal{L}} \lambda^*$. Therefore, there exists $h_i^* \in \partial d(x_i^*, \Omega_i)$

such that

$$f(x) - f(x^*) + (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} \lambda^* \geq \sum_{i=1}^{N} -c_i^* h_i^{*\mathrm{T}} (x_i - x_i^*)$$
$$\geq \sum_{i=1}^{N} -c_i^* d(x_i, \Omega_i).$$

By (i), $c_i(t)$ is upper bounded. Hence, by $\dot{c}_i(t) = d(x_i, \Omega_i)$, $d(x_i, \Omega_i) \to 0$ for all i. Thus, there exists $T(x_0, \lambda_0, c_0) > 0$ such that

$$d(x_i, \Omega_i) \leq \frac{1}{2},\ t \geq T(x_0, \lambda_0, c_0). \tag{15}$$

Therefore, $f(x) - f(x^*) + (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} \lambda^* \geq -\frac{\|c^*\|_1}{2}$ for all $t \geq T(x_0, \lambda_0, c_0)$. In addition, $(x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} (\lambda - \lambda^*) \geq -\frac{\lambda_{\max}(\tilde{\mathcal{L}})}{2} (\|x - x^*\|^2 + \|\lambda - \lambda^*\|^2)$. Therefore, $V_1(x, \lambda, c) + \alpha V_2(x, \lambda, c) \geq V_1(x, \lambda, c) + \alpha (x - x^*)^{\mathrm{T}} \tilde{\mathcal{L}} (\lambda - \lambda^*) + (\frac{\|c^*\|_1}{2} - \frac{\|c^*\|_1}{2}) \geq \frac{1 - \alpha \lambda_{\max}(\tilde{\mathcal{L}})}{2} (\|x - x^*\|^2 + \|\lambda - \lambda^*\|^2) + \frac{1}{2} \|c - c^*\|^2 \geq 0$.

Based on (i) and (ii), we have that, for $v_1 \in \mathcal{L}_{\mathcal{F}} V_1(x)$ and $v_2 \in \mathcal{L}_{\mathcal{F}} V_2(x)$

$$v_1 + \alpha v_2 \leq -x^{\mathrm{T}} \tilde{\mathcal{L}} x - \alpha \|\tilde{v}_1\|^2 + \alpha \|\tilde{\mathcal{L}} x\|^2$$
$$+ \alpha \sum_{i=1}^{N} d(x_i(t), \Omega_i)^2 - \alpha \sum_{i=1}^{N} d(x_i(t), \Omega_i)$$

for some $\tilde{v}_1 \in -\partial_x \tilde{f}(x, c) - \tilde{\mathcal{L}} x - \tilde{\mathcal{L}} \lambda$. According to the results in linear algebra, $\|\tilde{\mathcal{L}} x\|^2 = x^{\mathrm{T}} \tilde{\mathcal{L}}^{\mathrm{T}} \tilde{\mathcal{L}} x \leq \lambda_{\max}(\tilde{\mathcal{L}}) x^{\mathrm{T}} \tilde{\mathcal{L}} x$. Let $\alpha < \frac{1}{\lambda_{\max}(\tilde{\mathcal{L}})}$, $\beta = (1 - \alpha \lambda_{\max}(\tilde{\mathcal{L}}))$. We get $-x^{\mathrm{T}} \tilde{\mathcal{L}} x + \alpha \|\tilde{\mathcal{L}} x\|^2 \leq -\beta x^{\mathrm{T}} \tilde{\mathcal{L}} x$. With T defined in (15), $d(x_i, \Omega_i) \leq \frac{1}{2}$, $d(x_i, \Omega_i)^2 - d(x_i, \Omega_i) \leq -\frac{1}{2} d(x_i, \Omega_i)$, $t \geq T(x_0, \lambda_0, c_0)$. Therefore, there exist $\alpha, \beta, T(x_0, \lambda_0, c_0) > 0$ such that $v_1 + \alpha v_2 \leq -\alpha \|\tilde{v}_1\|^2 - \beta x(t)^{\mathrm{T}} \tilde{\mathcal{L}} x(t) - 0.5 \alpha \sum_{i=1}^{N} d(x_i(t), \Omega_i)$ for all $t \geq T(x_0, \lambda_0, c_0)$.

References

[1] A. Nedic, A. Ozdaglar, and P. A. Parrilo, “Constrained consensus and optimization in multi-agent networks,” *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, Apr. 2010.

[2] Y. Lou, Y. Hong, L. Xie, G. Shi, and K. H. Johansson, “Nash equilibrium computation in subnetwork zero-sum games with switching communications,” *IEEE Trans. Autom. Control*, vol. 61, no. 10, pp. 2920–2935, Oct. 2016.

[3] S. Yang, Q. Liu, and J. Wang, “A multi-agent system with a proportional-integral protocol for distributed constrained optimization,” *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3461–3467, Jul. 2017.

[4] X. Zeng, P. Yi, and Y. Hong, “Distributed continuous-time algorithm for constrained convex optimizations via nonsmooth analysis approach,” *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5227–5233, Oct. 2017.

[5] B. Gharesifard and J. Cortes, “Distributed continuous-time convex optimization on weight-balanced digraphs,” *IEEE Trans. Autom. Control*, vol. 59, no. 3, pp. 781–786, Mar. 2014.

[6] J. Li *et al.*, “Gradient-free method for nonsmooth distributed optimization,” *J. Global Optim.*, vol. 61, no. 2, pp. 325–340, Feb. 2015.

[7] A. V. DeMiguel and W. Murray, “An analysis of collaborative optimization methods,” in *Proc. 8th Symp. Multidisciplinary Anal. Optim., Multidisciplinary Anal. Optim. Conf.*, Los Angeles, CA, USA, 2000, vol. 4720, 2000, pp. 34–46.

[8] S. Liang, X. Zeng, and Y. Hong, “Distributed nonsmooth optimization with coupled inequality constraints via modified Lagrangian function,” *IEEE Trans. Autom. Control*, vol. 63, no. 6, pp. 1753–1759, Jun. 2018.

[9] M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic, *Nonlinear and Adaptive Control Design*. New York, NY, USA: Wiley, 1995.

[10] Y. Song, X. Huang, and C. Wen, “Distributed continuous-time convex optimization on weight-balanced digraphs,” *IEEE Trans. Autom. Control*, vol. 61, no. 10, pp. 3227–3233, Oct. 2016.

[11] Y. Song, K. Zhao, and M. Krstic, “Adaptive control with exponential regulation in the absence of persistent excitation,” *IEEE Trans. Autom. Control*, vol. 62, no. 5, pp. 2589–2596, May 2017.

[12] D. Boukari and A. V. Fiacco, “Survey of penalty, exact-penalty and multiplier methods from 1968 to 1993,”*Optimization*, vol. 32, no. 4, pp. 301–334, 1995.

[13] A. V. Fiacco and G. P. McCormick, “Programming under nonlinear constraints by unconstrained minimization: A primal-dual method,” Research Analysis Corp., McLean, VA, USA, Tech. Paper RAC-TP-96, 1963.

[14] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, vol. 4, no. 4, pp. 373–395, Nov. 1984.

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

[16] Z. Li *et al.*, “Distributed adaptive convex optimization on directed graphs via continuous-time algorithms,” *IEEE Trans. Autom. Control*, vol. 63, no. 5, pp. 1434–1441, May 2018, doi: 10.1109/TAC.2017.2750103.

[17] A. Bacciotti and F. Ceragioli, “Stability and stabilization of discontinuous systems and nonsmooth Lyapunov functions,” *ESAIM: COCV*, vol. 4, pp. 361–376, Jun. 1999.

[18] A. Hoffmann, “The distance to the intersection of two convex sets expressed by the distances to each of them,” *Math. Nachrichten*, vol. 157, no. 1, pp. 361–376, Jun. 1992.

[19] A. P. Ruszczyski, *Nonlinear Optimization*. Princeton, NJ, USA: Princeton Univ. Press, 2006.

[20] D. P. Bertsekas and A. Scientific, *Convex Optimization Algorithms*. Belmont, MA, USA: Athena Scientific, 2015.

[21] F. H. Clarke, *Optimization and Nonsmooth Analysis*. New York, NY, USA: Wiley, 1983.

[22] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Cham, Switzerland: Springer, 2011.

[23] H. Zhou, X. Zeng, and Y. Hong, “An exact penalty method for constrained distributed optimization,” in *Proc. 36th Chin. Control Conf.*, Dalian, China, 2017, pp. 8827–8832.

[24] J. Cortes, “Discontinuous dynamical systems,” *IEEE Control Syst. Mag.*, vol. 28, no. 3, pp. 36–73, Jun. 2008.

[25] J. P. Aubin and A. Cellina, *Differential Inclusions*. Berlin, Germany: Springer-Verlag, 1984.

[26] T. Yang *et al.*, “A distributed algorithm for economic dispatch over time-varying directed networks with delays,” *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 5095–5106, Jun. 2017.

[27] G. Mateos, J. A. Bazerque, and G. B. Giannakis, “Distributed sparse linear regression,” *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5262–5276, Oct. 2010.

[28] N. S. Aybat, Z. Wang, T. Lin, and S. Ma, “Distributed linearized alternating direction method of multipliers for composite convex consensus optimization,” *IEEE Trans. Autom. Control*, vol. 63, no. 1, pp. 5–20, Jan. 2018.

[29] T. H. Chang, M. Hong, and X. Wang, “Multi-agent distributed optimization via inexact consensus ADMM,” *IEEE Trans. Signal Process.*, vol. 63, no. 2, pp. 482–497, Jan. 2015.

[30] P. Yi, Y. Hong, and F. Liu, “Initialization-free distributed algorithms for optimal resource allocation with feasibility constraints and application to economic dispatch of power systems,” *Automatica*, vol. 74, pp. 259–269, 2016.

[31] A. Cherukuri and J. Cortes, “Initialization-free distributed coordination for economic dispatch under varying loads and generator commitment,” *Automatica*, vol. 74, pp. 183–193, 2016.

[32] Q. Liu, S. Yang, and J. Wang, “A collective neurodynamic approach to distributed constrained optimization,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 8, pp. 1747–1758, Aug. 2017.

[33] Q. Liu and J. Wang, “A one-layer projection neural network for nonsmooth optimization subject to linear equalities and bound constraints,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 5, pp. 812–824, May 2013.