Proximal ADMM for Nonconvex and Nonsmooth Optimization

Yu Yang [a], Qing-Shan Jia [b], Zhanbo Xu [a], Xiaohong Guan [a,b], Costas J. Spanos [c]

[a] *School of Automation Science and Engineering, Xi'an Jiaotong University, Shaanxi, China.*

[b] *CFINS, Department of Automation, BNRist, Tsinghua University, Beijing, China.*

[c] *Electrical Engineering and Computer Sciences, University of California, Berkeley.*

Abstract

By enabling the nodes or agents to solve small-sized subproblems to achieve coordination, distributed algorithms are favored by many networked systems for efficient and scalable computation. While for convex problems, substantial distributed algorithms are available, the results for the more broad nonconvex counterparts are extremely lacking. This paper develops a distributed algorithm for a class of nonconvex and nonsmooth problems featured by i) a nonconvex objective formed by both separate and composite objective components regarding the decision components of interconnected agents, ii) local bounded convex constraints, and iii) coupled linear constraints. This problem is directly originated from smart buildings and is also broad in other domains. To provide a distributed algorithm with convergence guarantee, we revise the powerful tool of alternating direction method of multiplier (ADMM) and proposed a proximal ADMM. Specifically, noting that the main difficulty to establish the convergence for the nonconvex and nonsmooth optimization within the ADMM framework is to assume the boundness of dual updates, we propose to update the dual variables in a discounted manner. This leads to the establishment of a so-called sufficiently decreasing and lower bounded Lyapunov function, which is critical to establish the convergence. We prove that the method converges to some approximate stationary points. We besides showcase the efficacy and performance of the method by a numerical example and the concrete application to multi-zone heating, ventilation, and air-conditioning (HVAC) control in smart buildings.

Key words: distributed nonconvex and nonsmooth optimization, proximal ADMM, bounded Lagrangian multipliers, global convergence, smart buildings.

1 Introduction

By enabling the nodes or agents to solve small-sized subproblems to achieve coordination, distributed algorithms are favored by many networked systems to achieve efficient and scalable computation. While distributed algorithms for convex optimization have been studied extensively [1–3]), the results for the more broad nonconvex counterparts are extremely lacking. The direct extensions of distributed algorithms for convex problems to nonconvex counterparts is in general not applicable either due to the failure of convergence or the lack of theoretical convergence guarantee (see [4, 5] for some divergent examples). This paper focuses on developing a distributed algorithm for a class of nonconvex and nonsmooth problems in the form of

$$\min_{\mathbf{x}=\{\mathbf{x}_i\}_{i=1}^N} F(\mathbf{x}) = g(\mathbf{x}) + \sum_{i=1}^N f_i(\mathbf{x}_i) \qquad (\mathbf{P})$$

$$\text{s.t.} \sum_{i=1}^N \mathbf{A}_i \mathbf{x}_i = \mathbf{b}. \qquad (1a)$$

$$\mathbf{x}_i \in \mathbf{X}_i,\ i = 1, 2, \cdots, N. \qquad (1b)$$

⋆ This work is supported by the Republic of Singapore's National Research Foundation through a grant to the Berkeley Education Alliance for Research in Singapore (BEARS) for the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program. This work is also supported in part by National Natural Science Foundation of China (62192752, 62192750, 62125304, 62073182), 111 International Collaboration Project (BP2018006), and Tsinghua University Initiative Scientific Research Program. Yu Yang is the corresponding author.

Email addresses: yangyu21@xjtu.edu.cn (Yu Yang), jiaqs@tsinghua.edu.cn (Qing-Shan Jia), zhanbo.xu@xjtu.edu.cn (Zhanbo Xu), xhguan@xjtu.edu.cn (Xiaohong Guan), spanos@berkeley.edu (Costas J. Spanos).

where $i = 1, 2, \cdots, N$ denotes the computing nodes or agents, $\mathbf{x}_i \in \mathbf{R}^{n_i}$ is the local decision variables of agent i and $\mathbf{x} = \{\mathbf{x}_i\}_{i=1}^N \in \mathbf{R}^n$ with $n = \sum_{i=1}^N n_i$ is the stack of all decision components. We have $f_i : \mathbf{R}^{n_i} \to \mathbf{R}$ and $g : \mathbf{R}^n \to \mathbf{R}$ denote the separate and composite objective components, which are continuously differentiable but possibly nonconvex. As expressed by problem $(\mathbf{P})$, the agents are expected to optimize their local decision variables in a cooperative manner so as to achieve the optimal overall system performance measured by the objective $F(\mathbf{x}) = g(\mathbf{x}) + \sum_{i=1}^N f_i(\mathbf{x}_i)$ considering both their local bounded convex constraints $\mathbf{X}_i$ as well as the global coupled linear constraints (1a) encoded by $\mathbf{A}_i \in \mathbf{R}^{m \times n_i}$ and $b \in \mathbf{R}^m$. By defining $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_N) \in \mathbf{R}^{m \times n}$, the coupled linear constraints can be expressed by $\mathbf{Ax} = \mathbf{b}$. Note that the presence of the local constraints $\mathbf{X}_i$ and the nonconvex objective components f_i and g makes the problem nonconvex and nonsmooth, which represents the major challenge to develop distributed solution methods with convergence guarantee.

Problem $(\mathbf{P})$ is directly originated from smart buildings where smart devices are empowered to make local decisions while accounting for the interactions or the shared resource limits with the other devices in the proximity (see, for examples [6, 7]). Many other applications also fit into this formulation, including but not limited to smart sensing [8], electric vehicle charging management [3, 9], power system control [10], wireless communication control [11]. When the number of nodes is large, centralized methods usually suffer bottlenecks from the heavy computation, data storing and communication (see [6, 10, 11] and the references therein). Also, centralized methods may disrupt privacy as the complete information of all agents (e.g., the private local objectives) are required to reported to a central computing agent. As a result, distributed algorithms are usually preferred for privacy, computing efficiency, small data storage, and scaling properties.

When problem $(\mathbf{P})$ is convex, plentiful distributed solution methods are available. The methods can be distinguished by the presence of the composite objective component g and the number of decision blocks N. When g is null, we have the classic dual decomposition methods [12, 13], the well-known alternating direction method of multiplier (ADMM) for two decision blocks ($N = 2$) [14] and its extensions to multi-block settings ($N \geq 3$)[15–17]. While the classic ADMM and its variations propose to update the decision components in a sequential manner (usually called *Gauss-Seidel* decomposition), the works [2] and [18] have made some effort in developing parallel ADMM and its variations (usually called *Jacobian* decomposition). The above methods are generally limited to problems with separable objective functions (i.e., only f_i exist and $g = 0$). For the case where some composite objective component g also exists, linearized ADMM [19] and inexact linearized ADMM [20] have been further developed.

The above results are all for convex problems. Nevertheless, massive applications arising from the engineering systems and machine learning domains require to handle the type of problem $(\mathbf{P})$ with possibly nonconvex objectives f_i and g. The non-convexity may originate from the complex system performance metrics or the penalties imposed on the operation constraints. When the objective components f_i and g lack convexity (i.e., the monotonically non-decreasing property of gradients or subgradients is lost), developing distributed solution methods becomes a much more challenging problem, especially to establish the theoretical convergence guarantee. Though some fresh distributed solution methods for constrained nonconvex problems have been developed, they can not be applied to problem $(\mathbf{P})$ due to the nonsmooth structure caused by the local constraints X_i relevant to all decision blocks. This can be perceived from the following literature.

The existing works for constrained nonconvex optimization can be distinguished by `problem structures`, `main assumptions`, `decomposition scheme` (i.e., *Jacobian* or *Gaussian-Seidel*) and `convergence guarantee` as reported in Table 1. Overall, they can be uniformly expressed by the template of problem $(\mathbf{P})$ but are slightly different in the settings.

The first category (Type 1) is concerned with problem $(\mathbf{P})$ without any composite objective component g [18]. An accelerated distributed augmented Lagrangian (ADAL) method was proposed to handle the possibly nonconvex but continuously differentiable separable objectives f_i. This method follows the classic ADMM framework but introduces an interpolation procedure regarding the primal updates at each iteration, which reads as $\mathbf{A}_i\mathbf{x}_i^{k+1} = \mathbf{A}_i\mathbf{x}_i^k + \mathbf{T}\left(\mathbf{A}_i\hat{\mathbf{x}}_i^k - \mathbf{A}_i\mathbf{x}_i^k\right)$ (k the iteration and $\mathbf{T}$ is a weighted matrix). To our understanding, this can be interpreted as a means to slow down the primal update for enhancing the convergence in nonconvex settings. By assuming the existence of stationary points that satisfy the strong second-order optimality condition, this paper established the local convergence of the method. The notion of local convergence is that the convergence towards some local optima can be assured if starting with a point sufficiently close to that local optima.

The subsequent four categories (Type 2, 3, 4, 5) differ from the first one mainly in the presence of a last block encoded by $\mathbf{B}$. Note that [25] can be viewed as a special case of $\mathbf{B} = \mathbf{I}$, where $\mathbf{I}$ are identity matrices of suitable sizes. The last block is exceptional due to the unconstrained and Lipschitz differentiable property, which are critical to bound the dual updates to establish the convergence (see the references therein). That's why the last decision block is usually distinguished by some special notations (i.e., $\mathbf{y}$, $\mathbf{x}_0$). While the first category employs *Jacobian* scheme for primal update, these four categories follow the *Gauss-Seidel* paradigms using alternating optimization. Specially, the works [5] and [24] have made some effort to handle the possibly coupled

Table 1
Distributed constrained nonconvex optimization

# Type	Problem structures	Main assumptions	Methods	Types	Convergence	Papers
1	$\min_{(\mathbf{x}_i)_{i=1}^N} \sum_{i=1}^N f_i(\mathbf{x}_i)$ s.t. $\sum_{i=1}^N \mathbf{A}_i\mathbf{x}_i = \mathbf{b}$. $\mathbf{x}_i \in \mathbf{X}_i,\ i = 1, 2, \cdots, N$.	f_i continuously differentiable. Strong second-order optimality condition.	ADAL	Jacobian	Local convergence. Local optima.	[18]
2	$\min_{\mathbf{x}=(\mathbf{x}_i)_{i=0}^p, y} g(\mathbf{x}) + \sum_{i=0}^p f_i(\mathbf{x}_i) + h(\mathbf{y})$ s.t. $\sum_{i=0}^p \mathbf{A}_i\mathbf{x}_i + \mathbf{B}\mathbf{y} = 0$.	g and h Lipschitz continuous gradient. f_i weakly convex. $\mathrm{Im}(\mathbf{A}) \subseteq \mathrm{Im}(\mathbf{B})$.	ADMM	Gauss-Seidel	Global convergence. Stationary points.	[5, 21] [22, 23]
3	$\min_{\mathbf{x}=(\mathbf{x}_i)_{i=1}^N, y} g(\mathbf{x}, \mathbf{y}) + \sum_{i=1}^N f_i(\mathbf{x}_i) + h(\mathbf{y})$ s.t. $\sum_{i=1}^N \mathbf{A}_i\mathbf{x}_i + \mathbf{B}\mathbf{y} = 0$.	g and h Lipschitz continuous gradient. $\mathrm{Im}(\mathbf{A}) \subseteq \mathrm{Im}(\mathbf{B})$.	Linearized ADMM	Gauss-Seidel	Global convergence. Stationary points.	[24]
4	$\min_{(\mathbf{x}_k)_{k=0}^K} \sum_{k=1}^k g_k(\mathbf{x}_k) + h(\mathbf{x}_0)$ s.t. $\mathbf{x}_k = \mathbf{x}_0$. $\mathbf{x}_0 \in \mathbf{X}$.	g Lipschitz continuous gradient. h convex.	Flexible ADMM	Gauss-Seidel	Global convergence. Stationary points.	[25]
5	$\min_{(\mathbf{x}_k)_{k=0}^K} \sum_{k=1}^N g_x(\mathbf{x}_k) + \ell(\mathbf{x}_0)$ s.t. $\sum_{k=1}^K \mathbf{A}_k\mathbf{x}_k = \mathbf{x}_0$. $\mathbf{x}_k \in \mathbf{X}_k,\ k = 1, \cdots, N$.	ℓ Lipschitz continuous gradient. g nonconvex but smooth or convex but non-smooth.	Flexible ADMM	Gauss-Seidel	Global convergence. Stationary points.	[25]
6	$\min_{(\mathbf{x}_i)_{i=1}^N, \bar{\mathbf{x}}} \sum_{i=1}^N f_i(\mathbf{x}_i)$ s.t. $\sum_{i=1}^N \mathbf{A}_i\mathbf{x}_i + \mathbf{B}\bar{\mathbf{x}} = 0$. $\mathbf{x}_i \in \mathbf{X}_i, h_i(\mathbf{x}_i) = 0,$ $i = 1, \cdots, N$. $\bar{\mathbf{x}} \in \bar{\mathbf{X}}$.	f_i continuously differentiable. h_i non-linear (possibly nonconvex). $\mathbf{B}$ full column rank. $\mathbf{X}_i$ possibly nonconvex.	ALM + ADMM	Gauss-Seidel	Global convergence. Stationary points.	[26, 27]
7	$\min_{(\mathbf{x}_i)_{i=1}^N} g(\mathbf{x}) + \sum_{i=1}^N f_i(\mathbf{x}_i)$ s.t. $\sum_{i=1}^N \mathbf{A}_i\mathbf{x}_i = \mathbf{b}$. $\mathbf{x}_i \in \mathbf{X}_i,\ i = 1, 2, \cdots, N$.	f_i and g Lipschitz continuous gradient.	Proximal ADMM	Jacobian	Global convergence. Approximate stationary points.	This paper

Note: the set $\mathbf{X}_i$ and $\bar{\mathbf{X}}$ are bounded convex sets.

objective components g but via different ways. Specifically, [5] proposed to use alternate block updates and [24] employed the linearization technique. Particularly, [5, 25] build a general framework to establish the convergence for *Gauss-Seidel* ADMM towards local optima or stationary points in nonconvex settings, which comprises two key steps: 1) identifying a so-called sufficiently decreasing Lyapunov function, and 2) establishing the lower boundness property of the Lyapunov function. The sufficiently decreasing and lower boundness property of a proper Lyapunov function state that [5]

$$
\begin{aligned}
&T(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) - T(\mathbf{x}^k, \boldsymbol{\lambda}^k) \\
&\qquad \leq -a_{\mathbf{x}}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 - a_{\boldsymbol{\lambda}}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2. \\
&T(\mathbf{x}^k, \boldsymbol{\lambda}^k) > -\infty.
\end{aligned}
$$

where $T(\cdot, \cdot)$ is a general Lyapunov function, $\mathbf{x}$ and $\boldsymbol{\lambda}$ are primal and dual variables, $a_{\mathbf{x}}$ and $a_{\boldsymbol{\lambda}}$ are positive coefficients. The augmented Lagrangian (AL) function has been normally used as a Lyapunov function for establishing the convergence in nonconvex settings (see [5, 25] and the references therein). However, they depend on the following two necessary conditions on the last decision block encoded by $\mathbf{B}$ to bound the dual updates $\|\boldsymbol{\lambda}^{k+1}-\boldsymbol{\lambda}^{k}\|^2$ by the primal updates $\|\mathbf{x}^{k+1}-\mathbf{x}^{k}\|^2$ [5, 25].

a) $\mathbf{B}$ has full column rank and $\text{Im}(\mathbf{A}) \subseteq \text{Im}(\mathbf{B})$ ($\text{Im}(\cdot)$ represents the image of a matrix).

b) The last decision block is unconstrained and with Lipschitz differentiable objective.

Noted that the forth and fifth category originated from [25] are a special case of $\mathbf{B} = \mathbf{I}$, which certainly satisfy the necessary **condition** a).

Following the line of works, the sixth category (Type 6) studied the extension of ADMM to non-linearly constrained nonconvex problems [26, 27]. Since it is difficult (if not impossible) to directly handle the non-linear coupled constraints by the AL framework, [26] proposed to first convert the non-linearly constrained problems to linearly constrained problems by introducing duplicated decision variables for interconnected agents to decouple the non-linear constraints. This yields a linearly constrained nonconvex problems with non-linear local constraints. The work [26] argues that the direct extension of ADMM to the reformulated problem is problematic for the two necessary conditions **condition** a) and b) can not be satisfied simultaneously. To bypass the challenge, [26] proposed to introduce a block of slack variables working as the last block. To force the slack block to *zero*, this paper adopted a two-level solution method where the inner-level uses the classic ADMM to solve a relaxed problem associated with a penalty on the slack variables , and the outer-level gradually forces the slack variables towards *zero*.

As can be perceived from the literature, it is difficult (if not impossible) to develop a distributed solution method with convergence guarantee for ($\mathbf{P}$) due to the lack of a well-behaved last block satisfying **condition** a) and b). The work [18] provided a solution with local convergence guarantee but can not handle the probable composite objective components g. Though the idea of introducing slack variables proposed in [26] can provide a solution with global convergence guarantee but at the cost of heavy iteration complexity caused by the two-level structure. Despite these limitations, what we can learn from the literature is that the behaviors of dual variables is important to draw the convergence of ADMM for nonconvex problems.

This paper focuses on developing a distributed solution method for problem ($\mathbf{P}$) with theoretical convergence guarantee. Our main contributions are

- We propose a proximal ADMM by revising the dual update procedure of the classic ADMM into a discounted manner. This leads to the boundness of dual updates, which is critical to establish the convergence.
- We establish the global convergence of the method towards approximate stationary points by identifying a proper Lyapunov function which shows the required sufficiently decreasing and lower bounded property.
- We showcase the performance of the proposed distributed method with a numerical example and a concrete application arising from smart buildings, which demonstrate the method's effectiveness.

The reminder of this paper is organized as follows. In Section 2, we present the proximal ADMM. In Section 3, we study the convergence of the method. In Section 4, we showcase the method with a numerical example and the smart building application. In Section 5, we conclude this paper and discuss the future work.

2 Proximal ADMM

2.1 Notations

Throughout the paper, we will visit the following notations. We use the bold alphabets $\mathbf{x}, \mathbf{y}, \mathbf{a}, \mathbf{b}, \mathbf{c}$ and $\mathbf{A}, \mathbf{A}_i, \mathbf{Q}, \mathbf{M}$ to represent vectors and matrices. We use $\mathbf{I}_n$ or $\mathbf{I}$ as identity matrices of $n \times n$ or suitable size. We use the operator $:=$ to give definitions. We have $\mathbf{R}^n$ represent the n-dimensional real space and $(\mathbf{x}_i)_{i=1}^N := (\mathbf{x}_1^\top, \mathbf{x}_2^\top, \cdots, \mathbf{x}_N^\top)^\top$ is the stack of the subvector $\mathbf{x}_i \in \mathbf{R}^{n_i}$. We refer to $\|\cdot\|$ as the Euclidean norm of vector $\mathbf{x} \in \mathbf{R}^n$ without specification where $\|\mathbf{x}\| = \sqrt{\sum_{i=1}^n x_i^2}$, and $\langle \mathbf{x}, \mathbf{y} \rangle$ denote the dot product of vector $\mathbf{x}, \mathbf{y} \in \mathbf{R}^n$. We besides have $\|\mathbf{x}\|_{\mathbf{A}}^2 = \mathbf{x}^\top \mathbf{A} \mathbf{x}$. We use $\text{diag}(\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_N)$ to denote the diagonal matrix formed by the sub-matrices $\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_N$. We have the normal cone to a convex set $\mathbf{X} \subseteq \mathbf{R}^n$ at $\mathbf{x}^*$ defined by $N_{\mathbf{X}}(\mathbf{x}^*) := \{\nu \in \mathbf{R}^n | \langle \nu, \mathbf{x} - \mathbf{x}^* \rangle \le 0, \forall \mathbf{x} \in \mathbf{X}\}$. For $g : \mathbf{R}^n \to \mathbf{R}$, we denote $\nabla_i g(\mathbf{x}) = \partial g(\mathbf{x})/\partial x_i$ as the partial differential of g with respect to component x_i. We define $\text{dist}(\mathbf{x}, \mathbf{X}) = \min_{\mathbf{y} \in \mathbf{X}} \|\mathbf{x} - \mathbf{y}\|$ as the distance of vector $\mathbf{x} \in \mathbf{R}^n$ to the subset $\mathbf{X} \subseteq \mathbf{R}^n$.

2.2 Algorithm

In this part, we introduce the proximal ADMM for solving problem ($\mathbf{P}$). The proximal ADMM is a type of AL methods, depending on the AL technique to relax coupled constraints and then using the primal-dual scheme to update the variables. By associating the dual variables $\boldsymbol{\lambda} \in \mathbf{R}^m$ with the coupled constraints (1a), we have the AL function for problem ($\mathbf{P}$)

$$\mathbb{L}_\rho(\mathbf{x}, \boldsymbol{\lambda}) = F(\mathbf{x}) + \langle \boldsymbol{\lambda}, \mathbf{A}\mathbf{x} - \mathbf{b} \rangle + \frac{\rho}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \tag{2}$$

where $F(\mathbf{x}) = g(\mathbf{x}) + \sum_{i=1}^N f_i(\mathbf{x}_i)$ and ρ is the penalty parameter.

Following the standard AL methods, the proximal ADMM is mainly composed of `Primal update` and `Dual update` as shown in **Algorithm** 1. In `Primal update`, the primal variables $\mathbf{x} = (\mathbf{x}_i)_{i=1}^N$ are updated by minimizing the AL function over the local constraints $\mathbf{X} = \prod_{i=1}^N \mathbf{X}_i$ in a distributed manner. Particularly, to handle the composite objective component g, we linearize the composite term at each iteration k by $\langle \nabla g(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$. Note that the local objective terms f_i can also be linearized similarly and the proof of this paper still applies.

Algorithm 1 Proximal ADMM for problem (**P**)

```
1: Initialize: $\mathbf{x}^0$, $\boldsymbol{\lambda}^0$ and $\rho > 0$, $\tau \in [0, 1)$, and set $k \to 0$.
2: Repeat:
3:   Primal update:
       $\mathbf{x}_i^{k+1} = \arg\min_{\mathbf{x}_i \in \mathbf{X}_i} \left\{ \begin{array}{l} \langle \nabla g_i(\mathbf{x}^k), \mathbf{x}_i - \mathbf{x}_i^k \rangle \\ + f_i(\mathbf{x}_i) + \langle \boldsymbol{\lambda}^k, \mathbf{A}_i \mathbf{x}_i^k \rangle \\ + \rho/2 \|\mathbf{A}_i \mathbf{x}_i^k + \sum_{j \neq i} \mathbf{A}_j \mathbf{x}_j^k - \mathbf{b}\|^2 \\ + \beta/2 \|\mathbf{x}_i - \mathbf{x}_i^k\|_{\mathbf{B}_i}^2 \end{array} \right\}$   (5)
4:   Dual update:
       $\boldsymbol{\lambda}^{k+1} = (1 - \tau)\boldsymbol{\lambda}^k + \rho \left( \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b} \right)$   (6)
5: Until the stopping criterion (4) is reached.
```

To favor computation efficiency and scaling properties, we adopt the *Jacobian* scheme and empower the agents to update their decision components in parallel at each iteration with the preceding information of their interconnected agents. Particularly, to enhance convergence, a proximal term $\|\mathbf{x}_i - \mathbf{x}_i^{k+1}\|^2$ is imposed on the local objective of each agent (Step 3). This has been used in many *Jacobian* ADMMs both in convex settings [2, 28, 29] and in nonconvex settings [24, 30]. Note that the subproblems (5) are either convex or nonconvex optimization over the local constraints $\mathbf{X}_i$, depending on f_i. There are many first-order solvers available for solving those subproblems, such as the projected gradient method [31] and the proximal gradient method [32]. This paper focuses on developing the general decomposition framework for the distributed computation and therefore will not discuss those solution methods in detail. The major difference of the proximal ADMM from the existing AL methods is that we have modified the `Dual update` by imposing a discounting factor $(1 - \tau)$ ($\tau \in [0, 1)$) (Step 4). The idea and motivation behind are to update the dual variables by the constraints residual in a discounted manner so as to bound the dual variables in the iterative process, which has been identified as critical to draw the theoretical convergence. In this setting, the dual variables are the *discounted* running sum of the constraints residual, i.e.,

$$\begin{aligned} \boldsymbol{\lambda}^{k+1} &= (1-\tau)\boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\ &= (1-\tau)^2\boldsymbol{\lambda}^{k-1} + (1-\tau)\rho(\mathbf{A}\mathbf{x}^k - \mathbf{b}) \\ &\qquad + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\ &\cdots \\ &= (1-\tau)^{k+1}\boldsymbol{\lambda}^0 + \textstyle\sum_{\ell=0}^{k}(1-\tau)^{k-\ell}\rho(\mathbf{A}\mathbf{x}^{\ell+1} - \mathbf{b}). \end{aligned} \tag{3}$$

This differs from the classic AL methods where the dual variables are the running sum of the residual, i.e.,

$$\begin{aligned} \boldsymbol{\lambda}^{k+1} &= \boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\ &= \boldsymbol{\lambda}^{k-1} + \rho(\mathbf{A}\mathbf{x}^k - \mathbf{b}) + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \\ &\cdots \\ &= \boldsymbol{\lambda}^0 + \textstyle\sum_{\ell=0}^{k}\rho(\mathbf{A}\mathbf{x}^{\ell+1} - \mathbf{b}). \end{aligned}$$

From this perspective, the classic ADMM can be viewed as a special case of the proximal ADMM with $\tau = 0$. The `Primal update` and `Dual update` are alternated until the stopping criterion

$$\|T_c^{k+1} - T_c^k\| \leq \epsilon \tag{4}$$

is reached, where T_c^k is a Lynapunov function that can indicate the progress of distributed optimization, which will be discussed in details later. The parameter ϵ is a user-defined positive threshold.

3 Convergence Analysis

Before establishing the convergence of **Algorithm** 1, we first clarify the main assumptions.

3.1 Main assumptions

(A1) Function $f : \mathbf{R}^n \to \mathbf{R}$ and $g : \mathbf{R}^n \to \mathbf{R}$ have Lipschitz continuous gradient (i.e., Lipschitz differentiable) with modulus L_f and L_g over the set $\mathbf{X} = \mathbf{X}_1 \times \mathbf{X}_2 \times \cdots \times \mathbf{X}_N$, i.e., [22]

$$\begin{aligned} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| &\leq L_f\|\mathbf{x} - \mathbf{y}\|, \ \forall \mathbf{x}, \mathbf{y} \in \mathbf{X}. \\ \|\nabla g(\mathbf{x}) - \nabla g(\mathbf{y})\| &\leq L_g\|\mathbf{x} - \mathbf{y}\|, \ \forall \mathbf{x}, \mathbf{y} \in \mathbf{X}. \end{aligned}$$

(A2) Function $f : \mathbf{R}^n \to \mathbf{R}$ and $g : \mathbf{R}^n \to \mathbf{R}$ are lower bounded over the set $\mathbf{X} = \mathbf{X}_1 \times \mathbf{X}_2 \times \cdots \times \mathbf{X}_N$, i.e.,

$$\begin{aligned} f(\mathbf{x}) &> -\infty, \quad \forall \mathbf{x} \in \mathbf{X}. \\ g(\mathbf{x}) &> -\infty, \quad \forall \mathbf{x} \in \mathbf{X}. \end{aligned}$$

3.2 Main results

As discussed, there are two key steps to draw convergence for a distributed AL method in nonconvex settings: 1) identifying a so-called sufficiently decreasing Lyapunov function; and 2) establishing the lower boundness property of the Lyapunov function. To achieve the objective, we first draw the following two propositions.

Proposition 1 *For the sequences* $\{\mathbf{x}^k\}_{k \in \mathbf{K}}$ *and* $\{\boldsymbol{\lambda}^k\}_{k \in \mathbf{K}}$ *generated by* ***Algorithm*** *1, we have*

$$\begin{aligned} &\frac{1 - 2\tau^2}{2\rho}\left\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\right\|^2 + \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2 \\ &\quad + \frac{L_g}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + \frac{1}{2}\|\mathbf{w}^k\|_{\mathbf{Q}}^2 \\ &\leq \frac{1 - 2\tau^2}{2\rho}\left\|\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1}\right\|^2 + \frac{1}{2}\|\mathbf{x}^k - \mathbf{x}^{k-1}\|_{\mathbf{Q}}^2 \\ &\quad + \frac{L_g}{2}\|\mathbf{x}^k - \mathbf{x}^{k-1}\|^2 + \rho_F\left\|\mathbf{x}^{k+1} - \mathbf{x}^k\right\|^2 \\ &\quad - \tau(1 + \tau)/\rho\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2. \end{aligned}$$

where we have the iterations $\mathbf{K} := \{1, 2, \cdots, K\}$ *and*

$$\begin{aligned}
\mathbf{w}^k &: = (\mathbf{x}^{k+1} - \mathbf{x}^k) - (\mathbf{x}^k - \mathbf{x}^{k-1}) \\
G_{\mathbf{A}} &: = diag\Big(\mathbf{A}_1^\top \mathbf{A}_1, \cdots, \mathbf{A}_N^\top \mathbf{A}_N\Big) \\
G_{\mathbf{B}} &: = diag\Big(\mathbf{B}_1^\top \mathbf{B}_1, \cdots, \mathbf{B}_N^\top \mathbf{B}_N\Big) \\
\mathbf{Q} &: = \rho G_{\mathbf{A}} + \beta G_{\mathbf{B}} - \rho \mathbf{A}^\top \mathbf{A} \\
\rho_F &: = L_f + L_g.
\end{aligned}$$

Proof of Prop. 1: We defer the proof to **Appendix** A.

Let $\mathbb{L}_\rho^+(\mathbf{x}, \boldsymbol{\lambda}) := \mathbb{L}_\rho(\mathbf{x}, \boldsymbol{\lambda}) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}\|^2$ be the regularized AL function. We have the subsequent proposition to quantify the change of regularized AL function over the successive iterations.

Proposition 2 *For the sequences* $\{\mathbf{x}^k\}_{k\in\mathbf{K}}$ *and* $\{\boldsymbol{\lambda}^k\}_{k\in\mathbf{K}}$ *generated by* ***Algorithm*** *1, we have*

$$\begin{aligned}
&\mathbb{L}_\rho^+(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) - \mathbb{L}_\rho^+(\mathbf{x}^k, \boldsymbol{\lambda}^k) \\
&\le -\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2 + \frac{\rho_F}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \\
&\quad - \frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 + \frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2.
\end{aligned}$$

Proof of Prop. 2: We defer the proof to **Appendix** B.

In the literature, the AL function has been normally used as the Lyapunov function when the dual updates $\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2$ can be bounded by the primal updates $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2$ and the sufficiently decreasing property of the AL function can be established (see [5, 21–23] for examples). However, this is not the case for problem (**P**) due to the lack of a well-behaved last block (i.e., unconstrained and Lipschitz differentiable). This can be perceived from **Prop.** 2 which states that the (regularized) AL function is ascending by $\frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2$ and descending by $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2$. Indeed, the imposing dual discounted factor $1-\tau$ is to overcome the challenge of identifying a proper Lyapunov function. Specifically, based on **Prop.** 1, we note that the term $\frac{1-2\tau^2}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2$ is descending by $\frac{\tau(1+\tau)}{\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2$ and ascending by $\rho_F\left\|\mathbf{x}^{k+1} - \mathbf{x}^k\right\|^2$ over the iterations. This is exactly opposite to the descending and ascending properties of the regularized AL function $\mathbb{L}_\rho^+(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1})$ in **Prop.** 2. We therefore build the Lyapunov function

$$\begin{aligned}
&T_c(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}; \mathbf{x}^k, \boldsymbol{\lambda}^k) = \mathbb{L}_\rho^+(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) \\
&\quad + c\bigg(\frac{1-2\tau^2}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2 \\
&\quad + \frac{L_g}{2}\|\mathbf{x}^k - \mathbf{x}^{k-1}\|^2\bigg) \qquad (7)
\end{aligned}$$

where c is a constant parameter to be determined for providing the sufficiently decreasing and lower boundness property of the Lyapunov function.

Let $T_c^{k+1} := T_c(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}; \mathbf{x}^k, \boldsymbol{\lambda}^k)$ be the Lyapunov function at iteration k, we have the following proposition regarding the sufficiently decreasing property of the function.

Proposition 3 *For the sequences* $\{\mathbf{x}^k\}_{k\in\mathbf{K}}$ *and* $\{\boldsymbol{\lambda}^k\}_{k\in\mathbf{K}}$ *generated by* ***Algorithm*** *1, we have*

$$T_c^{k+1} - T_c^k \le -a_{\mathbf{x}}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 - a_{\boldsymbol{\lambda}}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 - \frac{c}{2}\|\mathbf{w}^k\|^2$$

where we have $\rho_F = L_f + L_g$ *and*

$$\begin{aligned}
a_{\mathbf{x}} &:= \frac{2\rho G_{\mathbf{A}} + 2\beta G_{\mathbf{B}} - \rho \mathbf{A}^\top \mathbf{A} - (2c+1)\rho_F \mathbf{I}_N}{2} \\
a_{\boldsymbol{\lambda}} &:= \frac{2c\tau(1+\tau) - (2-\tau)}{2\rho}.
\end{aligned}$$

Proof of Prop. 3: Based on **Prop.** 1 and **Prop.** 2, we have

$$\begin{aligned}
&T_c^{k+1} - T_c^k = -\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{Q}}^2 + \frac{\rho_F}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \\
&\quad - \frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 + \frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 \\
&\quad + c\Big(\rho_F \left\|\mathbf{x}^{k+1} - \mathbf{x}^k\right\|^2 - \tau(1+\tau)/\rho\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 \\
&\quad - 1/2\|\mathbf{w}^k\|_{\mathbf{Q}}^2\Big) \\
&\le -a_{\mathbf{x}}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 - a_{\boldsymbol{\lambda}}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 - \frac{c}{2}\|\mathbf{w}^k\|_{\mathbf{Q}}^2
\end{aligned}$$

where the inequality is obtained by rearranging the terms. We therefore close the proof.

Remark 1 ***Prop.*** *3 implies that we would have the sufficiently decreasing property hold by the constructed Lyapunov function* T_c^k *if we have* $a_{\mathbf{x}} > 0$, $a_{\boldsymbol{\lambda}} > 0$, $c \ge 0$ *and* $\mathbf{Q} \ge 0$. *Actually, this can be achieved by setting the tuples* (τ, ρ, β, $\mathbf{B}_i$, c) *properly for* ***Algorithm*** *1, which will be discussed shortly.*

As discussed, another key step to draw the convergence is to establish the lower boundness property of the Lyapunov function. To this end, we first prove the lower boundness property of the Lagrangian multipliers provided by the discounted dual update scheme.

Proposition 4 *Let* $\Delta^k := \|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|$ *be the constraints residual at iteration* k, $\Delta^{\max} := \max_{\mathbf{x}\in\mathbf{X}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|$ *denote the maximal constraints residual over the closed feasible set* $\mathbf{X}$, *and* ***Algorithm*** *1 start with any given initial dual variable* $\boldsymbol{\lambda}^0$, *we have* $\|\boldsymbol{\lambda}^k\|$ *is bounded, i.e.,*

$$\begin{aligned}
&\|\boldsymbol{\lambda}^k\| \le \|\boldsymbol{\lambda}^0\| + \tau^{-1}\rho\Delta^{\max} \\
or\ \ &\|\boldsymbol{\lambda}^k\|^2 \le 2\|\boldsymbol{\lambda}^0\|^0 + 2\tau^{-2}\rho^2(\Delta^{\max})^2.
\end{aligned} \qquad (8)$$

Proof of Prop. 4: Recall the dual update scheme in (3), we have

$$\begin{aligned}
\|\boldsymbol{\lambda}^k\| &= \|(1-\tau)^{k+1}\boldsymbol{\lambda}^0 + \textstyle\sum_{\ell=0}^k \rho(1-\tau)^{k-\ell}\Delta^{\ell+1}\| \\
&\le \|(1-\tau)^{k+1}\boldsymbol{\lambda}^0\| + \textstyle\sum_{\ell=0}^k \|\rho(1-\tau)^{k-\ell}\Delta^{\ell+1}\| \\
&\le \|(1-\tau)^{k+1}\boldsymbol{\lambda}^0\| + \rho\Delta^{\max}\frac{1-(1-\tau)^k}{\tau} \\
&\le \|\boldsymbol{\lambda}^0\| + \tau^{-1}\rho\Delta^{\max}
\end{aligned}$$

where the first inequality is based on the triangle inequality of norm, the second inequality infers from $\Delta^k \le$

$\Delta^{\max}$ for all iteration k, and the last inequality holds because of $\tau \in (0,1)$.

Further based on $(a+b)^2 \leq 2a^2 + 2b^2$, we directly have $\|\boldsymbol{\lambda}^k\|^2 \leq 2\|\boldsymbol{\lambda}^0\|2 + 2\tau^{-2}\rho^2(\Delta^{\max})^2$, we therefore complete the proof.

Based on **Prop.** 4, we are able to establish the lower boundness property of Lyapunov function as below.

Proposition 5 *For the sequences $\{\mathbf{x}^k\}_{k\in\mathbf{K}}$ and $\{\boldsymbol{\lambda}^k\}_{k\in\mathbf{K}}$ generated by **Algorithm** 1, we have*

$$T_c^{k+1} > -\infty, \forall k \in \mathbf{K}. \tag{9}$$

Proof of Prop. 5: By examining the terms of T_c^{k+1} defined in (7), we only require to establish the lower boundness property of $\mathbb{L}_\rho^+(\mathbf{x}^k, \boldsymbol{\lambda}^k) = \mathbb{L}_\rho(\mathbf{x}^k, \boldsymbol{\lambda}^k) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1}\|^2$ for the other terms are all non-negative. Based on **Prop.** 4, we directly have $-\frac{\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1}\|^2$ is lower bounded since $\|\boldsymbol{\lambda}^{k+1}\|^2$ is upper bounded. We therefore only need to prove that $\mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) = f(\mathbf{x}^{k+1}) + \langle \boldsymbol{\lambda}^{k+1}, \mathbf{A}\mathbf{x}^{k+1} - b\rangle + \rho/2 \left\|\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}\right\|^2$ is lower bounded. Note that we have $f(\mathbf{x}^{k+1}) > -\infty$ over the compact set $\mathbf{X}$ (see (A2)) and the last two terms are all positive. This infers that we only need to establish the lower boundness property for the second term $\langle \boldsymbol{\lambda}^{k+1}, \mathbf{A}\mathbf{x}^{k+1} - b\rangle$ to complete the proof. Based on the dual update (6), we have

$$\begin{aligned}
&\langle \boldsymbol{\lambda}^{k+1}, \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}\rangle = \left\langle \boldsymbol{\lambda}^{k+1}, \frac{\boldsymbol{\lambda}^{k+1} - (1-\tau)\boldsymbol{\lambda}^k}{\rho} \right\rangle \\
&= \left\langle \boldsymbol{\lambda}^{k+1}, \frac{1-\tau}{\rho}(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) + \frac{\tau}{\rho}\boldsymbol{\lambda}^{k+1} \right\rangle \\
&= \frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 + \frac{1-\tau}{\rho}\left\langle \boldsymbol{\lambda}^{k+1}, \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k \right\rangle \\
&= \frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 + \frac{1-\tau}{2\rho}\left(\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \|\boldsymbol{\lambda}^{k+1}\|^2 - \|\boldsymbol{\lambda}^k\|^2\right)
\end{aligned} \tag{10}$$

Since we have $\|\boldsymbol{\lambda}^k\|^2$ is upper bounded (see **Prop.** 4), we therefore conclude that $\mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1})$ is lower bounded for the other terms of (10) are all non-negative, we therefore complete the proof.

To present the main results regarding the convergence of **Algorithm** 1, we first give the definition on **Approximate stationary solution**.

Definition 1 ***(Approximate stationary solution)*** *For any given ϵ, we say a tuple $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ is an ϵ-stationary solution of problem $(\mathbf{P})$, if we have*

$$\begin{aligned}
&\text{dist}\big(\nabla f(\mathbf{x}^*) + g(\mathbf{x}^*) + \mathbf{A}^\top\boldsymbol{\lambda}^* + N_\mathbf{X}(\mathbf{x}^*), \mathbf{0}\big) \\
&\quad + \|\mathbf{A}\mathbf{x}^* - \mathbf{b}\| \leq \epsilon.
\end{aligned}$$

where $f(\mathbf{x}^*) = \sum_{i=1}^N f_i(\mathbf{x}_i^*)$.

In terms of the convergence of **Algorithm** 1 for problem $(\mathbf{P})$, we have the main results presented below.

Theorem 1 *For **Algorithm** 1 with the tuples (τ, ρ, β, $\mathbf{B}_i$, c) selected by*

$$\begin{aligned}
&\tau : \tau \in (0,1) \\
&c : c > \frac{2-\tau}{2\tau(1+\tau)} \\
&(\rho, \beta, \mathbf{B}_i): \\
&\begin{cases} 2\rho G_\mathbf{A} + 2\beta G_\mathbf{B} - \rho\mathbf{A}^\top\mathbf{A} \geq (2c+1)\rho_F \mathbf{I}_N \\ \mathbf{Q} := \rho G_\mathbf{A} + \beta G_\mathbf{B} - \rho\mathbf{A}^\top\mathbf{A} \geq 0 \end{cases}
\end{aligned} \tag{C1}$$

(a) *The generated sequence $\{\mathbf{x}^k\}_{k\in\mathbf{K}}$ and $\{\boldsymbol{\lambda}^k\}_{k\in\mathbf{K}}$ are bounded and convergent, i.e.,*

$$\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k \to 0, \quad \mathbf{x}^{k+1} - \mathbf{x}^k \to 0.$$

(b) *Suppose we have the limit tuples $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$, then $(\mathbf{x}^*, \hat{\boldsymbol{\lambda}}^*)$ with $\hat{\boldsymbol{\lambda}}^* = (1 + \tau\boldsymbol{\lambda}^*)$ is $\tau\rho^{-1}\|\boldsymbol{\lambda}^*\|$-stationary solution of problem $(\mathbf{P})$.*

Proof of Theorem 1: (a) Recall **Prop.** 3, we have

$$\begin{aligned}
\sum_{k=1}^K \left(T_c^k - T_c^{k+1}\right) &\geq a_\mathbf{x}\sum_{k=1}^K \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \\
&+ a_{\boldsymbol{\lambda}}\sum_{k=1}^K \|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{c}{2}\sum_{k=1}^K \|\mathbf{w}^k\|^2
\end{aligned}$$

By assuming $K \to \infty$, we have

$$\begin{aligned}
T_c^1 - \lim_{K\to\infty} T_c^{k+1} &\geq a_\mathbf{x}\sum_{k=1}^\infty \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 \\
&+ a_{\boldsymbol{\lambda}}\sum_{k=1}^\infty \|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{c}{2}\sum_{k=1}^\infty \|\mathbf{w}^k\|^2
\end{aligned}$$

Since we have $T_c^{k+1} > -\infty$ (see **Prop.** 5), we thus have

$$\infty \geq a_\mathbf{x}\sum_{k=1}^\infty \|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2 + a_{\boldsymbol{\lambda}}\sum_{k=1}^\infty \|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{c}{2}\sum_{k=1}^\infty \|\mathbf{w}^k\|^2.$$

We therefore conclude

$$\begin{aligned}
&\|\mathbf{x}^{k+1} - \mathbf{x}^k\| \to 0, \quad \|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\| \to 0, \\
&\|\mathbf{w}^k\| = \|(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) - (\boldsymbol{\lambda}^k - \boldsymbol{\lambda}^{k-1})\| \to 0.
\end{aligned}$$

(b) According to (a), we have the sequences $\{\mathbf{x}^k\}_{k\in\mathbf{K}}$ and $\{\boldsymbol{\lambda}^k\}_{k\in\mathbf{K}}$ converge to some limit point $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$, i.e., if $k \to \infty$, we have $\mathbf{x}^{k+1} \to \mathbf{x}^*$, $\boldsymbol{\lambda}^{k+1} \to \boldsymbol{\lambda}^*$ and $\mathbf{x}^{k+1} \to \mathbf{x}^k$ and $\boldsymbol{\lambda}^{k+1} \to \boldsymbol{\lambda}^k$.

Based on the dual update procedure (6), we have for the stationary tuples $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ that

$$\mathbf{A}\mathbf{x}^* - \mathbf{b} = \tau\rho^{-1}\boldsymbol{\lambda}^*. \tag{11}$$

Since we have $\hat{\boldsymbol{\lambda}}^k = \boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x}^k - \mathbf{b})$, we thus have $\hat{\boldsymbol{\lambda}}^k \to (1+\tau)\lambda^*$. Let $\hat{\boldsymbol{\lambda}}^* = (1+\tau)\boldsymbol{\lambda}^*$, we have $\hat{\boldsymbol{\lambda}}^k \to \hat{\boldsymbol{\lambda}}^*$.

Recall the first-order optimality condition (A.4) and assume $k \to \infty$ that the stationary point $(\mathbf{x}^\star, \boldsymbol{\lambda}^*)$ is reached, we would have

$$\langle \nabla f(\mathbf{x}^*) + \nabla g(\mathbf{x}^*) + \mathbf{A}^\top\hat{\boldsymbol{\lambda}}^*, \mathbf{x}^* - \mathbf{x}\rangle \leq 0, \forall \mathbf{x} \in \mathbf{X}.$$

This implies that there exist $\nu \in N_\mathbf{X}(\mathbf{x}^*)$ such that

$$\nabla f(\mathbf{x}^*) + \nabla g(\mathbf{x}^*) + \mathbf{A}^\top\hat{\boldsymbol{\lambda}}^* + N_\mathbf{X}(\mathbf{x}^*) \in 0.$$

We further have

$$\operatorname{dist}\big(\nabla f(\mathbf{x}^*)+\nabla g(\mathbf{x}^*)+\mathbf{A}^\top\hat{\boldsymbol{\lambda}}^* + N_{\mathbf{X}}(\mathbf{x}^*),0\big) = 0 \quad (12)$$

By combing (11) and (12), we therefore conclude

$$\begin{aligned}&\operatorname{dist}\big(\nabla f(\mathbf{x}^*) + \nabla g(\mathbf{x}^*) + \mathbf{A}^\top\hat{\boldsymbol{\lambda}}^* + N_{\mathbf{X}}(\mathbf{x}^*),0\big)\\ &+ \|\mathbf{A}\mathbf{x}^* - \mathbf{b}\| \leq \tau\rho^{-1}\|\boldsymbol{\lambda}^*\|,\end{aligned}$$

which closes the proof.

From **Theorem** 1, we note that if the convergent $\boldsymbol{\lambda}^*$ does not depend on τ and ρ, we could decrease τ or increase ρ to achieve any sub-optimality. If that is not the case, we give the following corollary to show that this still can be achieved by properly setting the initial point and parameters.

Corollary 1 *For any given $\epsilon > 0$, if **Algorithm** 1 starts with $\boldsymbol{\lambda}^0 = 0$ and $\mathbf{A}\mathbf{x}^0 = \mathbf{b}$, $\tau \in (0,1)$, and the penalty parameter ρ is selected that*

$$\begin{aligned}\rho \geq\ &\epsilon^{-1}\tau\big(4 + c(1-2\tau^2) + c/2\big)d_F + cL_g/2\|\mathbf{x}^0\|^2\\ &+ \epsilon^{-1}\tau c L_g/2\|\mathbf{x}^0\|^2 + \epsilon^{-1}\tau c\rho_F/4\ d_{\mathbf{x}},\end{aligned}$$

we have the limit tuples $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ are ϵ-stationary solution of problem $(\mathbf{P})$. *where we have $d_F = \max_{\mathbf{x}\in\mathbf{X}} f(\mathbf{x}) + g(\mathbf{x})$, $d_{\mathbf{x}} = \max_{\mathbf{x},\mathbf{y}\in\mathbf{X}} \|\mathbf{x}-\mathbf{y}\|^2$, and we assume $f(\mathbf{x}) \geq 0$ and $g(\mathbf{x}) \geq 0$ without losing any generality.*

Proof of Corollary 1: We only give the sketch of the proof and we defer the details to **Appendix** C. The proof is structured by two parts which include i) proving $\|\boldsymbol{\lambda}^*\|^2 \leq \rho\tau^{-1}T_c^0$, and ii) proving $T_c^0 \leq \big(4+c(1-2\tau^2)+c/2\big)d_F + cL_g/2\|\mathbf{x}^0\|^2 + c\rho_F/4\ d_{\mathbf{x}}$. Based on **Theorem** 1 and i), ii), we directly draw the conclusion.

4 Numerical Experiments

4.1 A numerical example

We first consider a numerical example with $N = 2$ agents given by

$$\begin{aligned}\min_{x_1,x_2}\quad & 0.1x_1^3 + 0.1x_2^3 + 0.1x_1x_2 \qquad (\mathbf{P1})\\ \text{s.t. } & x_1 + x_2 = 1\\ & -1 \leq x_1 \leq 1\\ & -1 \leq x_2 \leq 1\end{aligned}$$

For this example, we have $f_1(x_1) = 0.1x_1^3$, $f_2(x_2) = 0.1x_2^3$, and $g(x_1,x_2) = 0.1x_1x_2$. The Lipschitz continuous gradient modulus for f and g are $L_f = 0.6$ and $L_g = 0.2$. Besides, we have $\mathbf{A}_1 = 1$, $\mathbf{A}_2 = 1$, $\mathbf{A} = (1\ 1)$. The stationary point of the problem is $x_1^\star = 0.5, x_2^\star = 0.5$.

To our best knowledge, there is no distributed solution methods for solving problem (**P1**) with theoretical convergence guarantee. In the following, we apply the proposed proximal ADMM to solve this problem. We consider four different parameter settings for **Algorithm** 1:

S1) $\tau = 0.1$, $\rho = 10$, $\beta = 10$, $c = 8.7$
S2) $\tau = 0.1$, $\rho = 20$, $\beta = 20$, $c = 8.7$
S3) $\tau = 0.05$, $\rho = 5$, $\beta = 16$, $c = 18.6$
S4) $\tau = 0.05$, $\rho = 10$, $\beta = 16$, $c = 18.6$

Particularly, the other parameters, i.e., $\mathbf{B}_1 = \mathbf{B}_2 = 1$, $\tau = 0.1$, $x_1^0 = 0.2$, $x_2^0 = 0.8$ and $\lambda^0 = 0$ are kept the same for the four settings. Note that for S1 and S3, we have $\tau/\rho = 0.01$, and for S2 and S4, we have $\tau/\rho = 0.005$. We make these settings for comparision as the suboptimality of the method is closely related to the ratio of τ/ρ as stated in **Theorem** 1. In this part, we evaluate how the parameter settings will affect the convergence rate and the solution quality.

Before starting the algorithm, we can easily verify that the convergence condition (C1) stated in **Theorem** 1 are all satisfied by S1-S4. We use the `interior-point` method embedded in the `fmincon` solver of MATLAB to solve subproblems (5). We run **Algorithm** 1 sufficiently long (i.e., $K = 2000$ iterations and the Lyapunov function does not change apparently) for the settings S1-S4. We first examine the convergence of the method indicated by the Lyapunov function. Fig. 1 (a) shows the evolution of the Lyapunov function w.r.t. the iteration under the different settings S1-S4. From the results, we observe that for all the settings, the Lyapunov functions strictly decrease w.r.t. the iterations and finally stablize at some value that is close to the optima $f^\star = 0.1x_1^\star + 0.1x_2^\star + 0.1x_1^\star x_2^\star = 0.05$. By further examining the results, we note that a larger ratio of τ/ρ generally yields faster convergence rate as with S1 and S3 ($\tau/\rho = 0.01$) compared with S2 and S4 ($\tau/\rho = 0.005$). This is caused by the relatively small penalty factor ρ and proximal factor β required to satisfy the convergence condition (C1) as with S1 and S3 over S2 and S4. Note that the penalty facotor ρ and the proximal factor β can be interpreted as some means for slowing down the primal updates as they have an effect in penalizing the deviation from the current value $\mathbf{x}^k$. Oppositely, a smaller ratio of τ/ρ generally yields higher solution quality (i.e., smaller suboptimality gap) as with S2 and S4 ($\tau/\rho = 0.005$) compared with S1 and S3 ($\tau/\rho = 0.01$). This is in line with the results in **Theorem** 1.

To further examine the solution quality, we report the detailed results with the four settings (`Prox-ADMM-Sx`, x = 1, 2, 3, 4) and the centralized method (`centralized` based on the `interior-point` method embedded in the `fmincon` solver of MATLAB) in Table 2. Note that the convergent solution $\hat{x}_1$ and $\hat{x}_2$ with proximal ADMM under the four settings S1-S4 are quite close to the optimal solution $x_1^\star = 0.5$ and $x_2^\star = 0.5$ obtained with centralized method. More specifically, by measuring the sub-optimality by $\|\hat{\mathbf{x}} - \mathbf{x}^\star\|/\|\mathbf{x}^\star\|$ where $\hat{\mathbf{x}} = [\hat{x}_1, \hat{x}_2]$ and $\mathbf{x}^\star = [x_1^\star, x_2^\star]$ are the convergent and the optimal solution, we conclude that the sub-optimality of proximal ADMM is around 1.2E-3 with S1 and S3 ($\tau/\rho = 0.01$) and 5.9E-4 with S2 and S4 ($\tau/\rho = 0.005$). We conclude that we could achieve higher solution quality by imposing smaller ratio of τ/ρ but generally at the cost of slowing down the convergence rate as observed in Fig. 1 (a). This implies that a trade-off is necessary in terms of the solution quality and the convergence speed while configuring the algorithm (i.e., the ratio of τ/ρ) for specific applications. For this example, considering the trade-off of the sub-optimality and the convergence rate, we note that S4 would be a preferred option. We therefore display the convergence of the decision variables x_1 and x_2 in Fig. 1(b). Note that x_1 and x_2 gradually converge to the optimal solution $\mathbf{x}_1^\star = 0.5$ and $x_2^\star = 0.5$.

Table 2
Performance of proximal ADMM under the settings S1-S4 vs. Centralized method

Method	τ/ρ	$\hat{x}_1$	$\hat{x}_2$	Sub-optimality	Convergence Rate
`Centralized`	–	0.5	0.5	–	–
`Prox-ADMM-S1`	0.01	0.4994	0.4994	1.1E-3	No. 1
`Prox-ADMM-S2`	0.005	0.4997	0.4997	5.7E-4	No. 4
`Prox-ADMM-S3`	0.01	0.4994	0.4994	1.2E-3	No .2
`Prox-ADMM-S4`	0.005	0.4997	0.4997	5.9E-4	No. 3

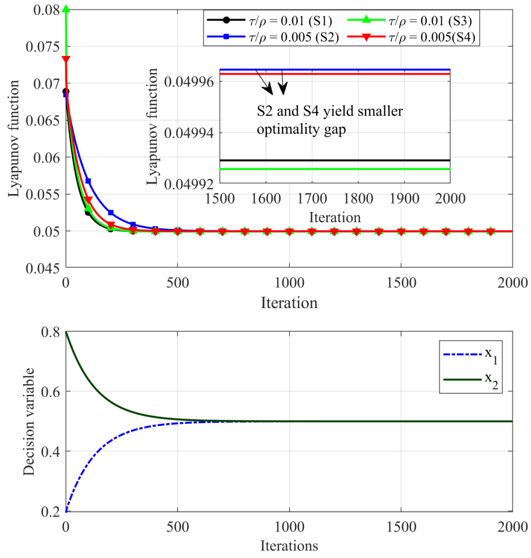

Fig. 1. (a) The evolution of primal variables x_1^k and x_2^k. (b) The evolution of the Lyapunov function T_c^k with S4.

4.2 Application: multi-zone HVAC control

To showcase the performance of proximal ADMM in application, we apply it to the multi-zone heating, ventilation, and air conditioning (HVAC) control arising from smart buildings. The goal is to optimize the HVAC operation to provide the comfortable temperature with minimal electricity bill. Due to the thermal capacity of buildings, the evolution of indoor temperature is a slow process affected both by the dynamic indoor occupancy (thermal loads) and the HVAC operation (cooling loads). The general solution is to design a model predictive controller for optimizing HVAC operation (i.e., zone mass flow and zone temperature trajectories) to minimize the overall electricity cost while respecting the comfortable temperature ranges based on the predicted information (i.e., indoor occupancy, outdoor temperature, electricity price, etc.). The general problem formulation is presented below.

$$\begin{aligned} \min_{\mathbf{m}^z,\mathbf{T}} \sum_t c_t\{ & c_p(1-d_r)\sum_i m_t^{zi}(T_t^o - T^c) \qquad (\mathbf{P2}) \\ & + c_p\eta d_r \sum_i m_t^{zi}(T_t^i - T^c) + \kappa_f\big(\textstyle\sum_i m_t^{zi}\big)^2\}\Delta_t \end{aligned}$$

$$\begin{aligned} \text{s.t.}\ \ T_{t+1}^i &= A_{ii}T_t^i + \textstyle\sum_{j\in N_i} A_{ij}T_t^j \\ &\quad + C_{ii}m_t^{zi}(T_t^i - T^c) + D_t^{ii},\ \forall i,t. && (13a) \\ T_{\min}^i &\le T_t^i \le T_{\max}^i, \quad \forall i,t. && (13b) \\ m_{\min}^{zi} &\le m_t^{zi} \le m_{\max}^{zi},\ \forall i,t. && (13c) \\ \textstyle\sum_i m_t^{zi} &\le \overline{m},\ \forall t. && (13d) \end{aligned}$$

where i and t are zone and time indices, $\mathbf{T} = (T_t^i)_{\forall i,t}$ and $\mathbf{m}^z = (m_t^{zi})_{\forall i,t}$ are zone temperature and zone mass flow rates, which are decision variables. The other notations are parameters. The main task is to optimize the zone mass flow rates to provide the temperature trajectories within the comfortable ranges $[T_{\min}^i, T_{\max}^i]$ with the minimal electricity cost measured by the objective. The problem is subject to the constraints covering zone thermal dynamics (13a), comfortable temperature margins (13b), zone mass flow rate limits (13c), and total zone mass flow rate limits (13d).

For the multi-zone HVAC control, centralized strategies are generally not suitable due to the computation and communication overheads, and distributed methods have been regarded as desirable solutions. However, the non-convexity makes it challenging to develop a distributed mechanism which can enable zone-level computation while still achieving the coordination among the zones to minimize the overall cost. This section demonstrates that the proposed method can work as an effective distributed solution. Before we show the results, we first restate the problem in the standard format:

$$\begin{aligned} \min_{\mathbf{m}^z,\mathbf{T}} \sum_t c_t\{ & c_p(1-d_r)\sum_i m_t^{zi}(T_t^o - T^c) \qquad (\mathbf{P3}) \\ & + c_p\eta d_r \sum_i m_t^{zi}(T_t^{ii} - T^c) + \kappa_f\big(\textstyle\sum_i m_t^{zi}\big)^2\}\Delta_t \\ & + M\textstyle\sum_i\sum_t \big(T_{t+1}^{ii} - A_{ii}T_t^{ii} - \sum_{j\in N_i} A_{ij}T_t^{ij} \\ & \quad - C_{ii}m_t^{zi}(T_t^{ii} - T^c) - D_t^{ii}\big)^2 \end{aligned}$$

$$\begin{aligned} \text{s.t.}\ \ T_t^{ij} &= \overline{T}_t^j,\quad \forall i,j,t. && (14a) \\ T_{\min}^i &\le T_t^{ii} \le T_{\max}^i, \quad \forall i,t. && (14b) \\ T_{\min}^i &\le \overline{T}_t^j \le T_{\max}^i, \quad \forall i,t. && (14c) \\ m_{\min}^{zi} &\le m_t^{zi} \le m_{\max}^{zi}, \quad \forall i,t. && (14d) \\ \textstyle\sum_i m_t^{zi} &\le \overline{m},\ \forall t. && (14e) \end{aligned}$$

where we have augmented the decision component for each zone to involve the copy of temperature for its neighboring zones, i.e., $\mathbf{T}^i := \{T_t^{ij}\}_{j\in\mathbf{N}_i,t}$. Besides, to drive the consistence of zone temperature, we introduce a block of consensus variable $\overline{\mathbf{T}} = \{\overline{T}_t^j\}_{j,t}$. Considering the challenging to handle the hard non-linear constraints (13a), we employ the penalty method and penalize the violations of constraints with quadratic terms. In this regard, problem (**P3**) fits into the template (**P**). Particularly, we have $N+1$ computing agents, where agents 1 to N correspond to the zones with the augmented decision variable $\mathbf{x}_i = (\{T_t^{ij}\}_{j \in N_i, t}, \{m_t^{zi}\}_t)$, and agent 0 control the consensus decision variable $\overline{\mathbf{T}} := \{\overline{T}_j\}_{j \in N}$. Constraints (14a) and (14e) represents the coupled linear constraints which can be expressed in the compact form $\mathbf{Ax} = \mathbf{b}$ if necessary. The other constraints comprise the local bounded convex constraints for the agents.

We consider a case study with $N = 10$ zones and the predicted horizon $T = 48$ time slots (a whole day with a sampling interval of 30 mins). We set the lower and upper comfortable temperature bounds as $T_{\text{min}}^i = 24°\text{C}$ and $T_{\text{max}}^i = 26°\text{C}$. The specifications for HVAC system can refer to [6, 7]. We apply the proposed proximal ADMM to solve this problem in a distributed manner. The algorithm configurations are $\rho = 2.0$, $\tau = 0.1$, $\beta = 3.0$, $\mathbf{B}_i = \mathbf{I}$ (suitable sizes), and $c = 8.7$. We first examine the convergence of the algorithm measured by the `Lyapunov function` and the norm of (coupled) `constraints residual`. We run the algorithm suitably long when both the residual and Lyapunov function do not change apparently ($K = 200$ iterations for this example). We visualize the `Lyapunov function` and `constraints residual` in Fig. 2. Note that the `Lyapunov function` strictly declines along the iterations, which is consistent with our theoretical analysis. Besides, the `constraints residual` almost strictly decreasing toward *zero* along the iterations as well. We find that the overall norm of the `constraints residual` at the end of iterations is about 0.38, which is quite small considering the problem scale $T \cdot N = 480$. This justifies the convergence property of proximal ADMM for the smart building application.

We next evaluate the solution quality measured by the HVAC electricity cost and human comfort. We randomly pick 3 zones (zone 1, zone 3, and zone 7) and display the predicted `zone occupancy` (inputs), the zone mass flow rates (`zone MFR`, control variables) , and the zone temperature (`zone temp.`, control variables) over the 48 simulated time slots in Fig. 3. Note that the variations of `zone MFR` are almost consistent with the `zone occupancy`. This is reasonable as the `zone occupancy` determines the thermal loads which need to be balanced by the zone mass flows. We see that the `zone temp.` are all maintained within the comfortable range $[24, 26]°\text{C}$. This infers the satisfaction of `human comfort`. To further evaluate the solution quality and computation efficiency, we compare the proximal ADMM (`Prox-ADMM`) with centralized method (`Centralized`). Specifically, we use the `interior-point` embedded in the `fmincon` solver of MATLAB to solve both the subproblems (5) and the centralized problem. For the `Centralized`, we run the algorithm without considering the running time with the objective to approach the best possible solution. We compare the two methods in three folds, i.e., `electricity cost`, the norm of `constraints residual`, and `computation time` as reported in Table 3. We see that `electricity cost` with `Prox-ADMM` is about 160.20 (s\$) versus 153.12 (s\$) yield by `Centralized`. This infers that the sub-optimality of `Prox-ADMM` in terms of the objective is about 5.0%. Particularly, we observe a marginal `constraints residual` (0.38) for `Prox-ADMM` , which is caused by the discounting factor τ. However, the `Prox-ADMM` obviously outperforms the `Centralized` in computation efficiency. The average `computing time` for each zone is about 50 min with `Prox-ADMM` (parallel computation) while the `Centralized` takes more than 10 h. Note that we have picked $T = 48$ time slots (a whole day) as the predicted horizon, the computing time could be largely sharpened in practice with a much smaller prediction horizon, say $T = 10$ time slots (5h). This is to our expectations as the `Prox-ADMM` empowers the agents (zone controller of smart buildings) to solve small subproblems in parallel instead of relying on a central agent solving the overall heavy problem as with the `Centralized`.

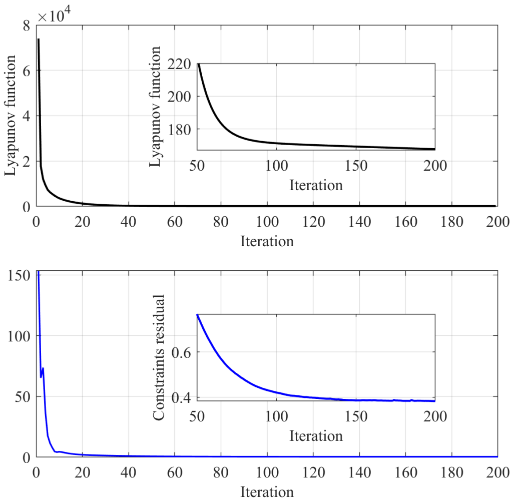

Fig. 2. (a) The evolution of Lyapunov function. (b) The evolution of the norm of constraints residual.

Table 3
Prox-ADMM vs. Centralized for HVAC control in smart buildings ($N = 10$ zones)

Method	Electricity cost (s\$)	Human comfort	Constraints residual	Computing time
`Centralized`	153.12	Y	0	$\geq$ 10h
`Prox-ADMM`	160.54	Y	0.38	50 min

5 Conclusion and Future Work

This paper focused on developing a distributed algorithm for a class of structured nonconvex and nonsmooth problems with convergence guarantee. The problems are

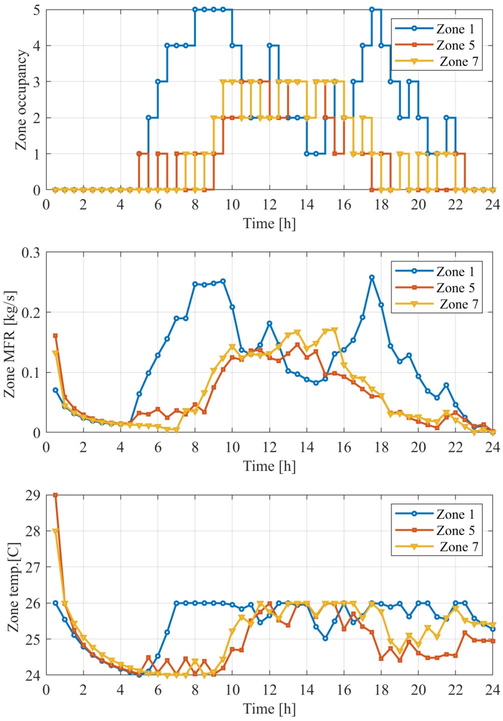

Fig. 3. (a) Zone occupancy. (b) Zone mass flow rate (Zone MFR). (c) Zone temperature (Zone temp.).

featured by i) a possibly nonconvex objective composed of both separate and composite objective components, ii) local bounded convex constraints, and iii) global coupled linear constraints. This class of problems are broad in application but lack distributed solutions with convergence guarantee. We employed the powerful alternating direction method of multiplier (ADMM) tool for constrained optimization but faced the challenges to establish the convergence. Noting that the underlying obstacle is to assume the boundness of dual updates, we revised the classic ADMM and proposed to update the dual variables in a distributed manner. This leads to a proximal ADMM with the convergence guarantee towards the approximate stationary points of the problem. We demonstrated the convergence and solution quality of the distributed method by a numerical example and a concrete application to the multi-zone heating, ventilation, and air-condition (HVAC) control arising from smart buildings. This paper has relied on the discounted dual update scheme to establish the convergence of distributed AL method in nonconvex and nonsmooth settings, some interesting future work along this line includes studying whether the discounted dual update scheme could speed up or enhance the convergence of the existing distributed AL methods for convex problems.

References

[1] W. Shi, Q. Ling, K. Yuan, G. Wu, and W. Yin, "On the linear convergence of the ADMM in decentralized consensus optimization," *IEEE Transactions on Signal Processing*, vol. 62, no. 7, pp. 1750–1761, 2014.

[2] W. Deng, M.-J. Lai, Z. Peng, and W. Yin, "Parallel multi-block ADMM with O (1/k) convergence," *Journal of Scientific Computing*, vol. 71, no. 2, pp. 712–736, 2017.

[3] A. Falsone, I. Notarnicola, G. Notarstefano, and M. Prandini, "Tracking-ADMM for distributed constraint-coupled optimization," *Automatica*, vol. 117, p. 108962, 2020.

[4] B. Houska, J. Frasch, and M. Diehl, "An augmented lagrangian based algorithm for distributed nonconvex optimization," *SIAM Journal on Optimization*, vol. 26, no. 2, pp. 1101–1127, 2016.

[5] Y. Wang, W. Yin, and J. Zeng, "Global convergence of ADMM in nonconvex nonsmooth optimization," *Journal of Scientific Computing*, vol. 78, no. 1, pp. 29–63, 2019.

[6] Y. Yang, G. Hu, and C. J. Spanos, "HVAC energy cost optimization for a multizone building via a decentralized approach," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1950–1960, 2020.

[7] Y. Yang, S. Srinivasan, G. Hu, and C. J. Spanos, "Distributed Control of Multizone HVAC Systems Considering Indoor Air Quality," *IEEE Transactions on Control Systems Technology*, 2021.

[8] J. A. Ansere, G. Han, L. Liu, Y. Peng, and M. Kamal, "Optimal resource allocation in energy-efficient internet-of-things networks with imperfect CSI," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 5401–5411, 2020.

[9] L. Zhang, V. Kekatos, and G. B. Giannakis, "Scalable electric vehicle charging protocols," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1451–1462, 2016.

[10] M. K. Arpanahi, M. H. Golshan, and P. Siano, "A Comprehensive and Efficient Decentralized Framework for Coordinated Multiperiod Economic Dispatch of Transmission and Distribution Systems," *IEEE Systems Journal*, 2020.

[11] S. Hashempour, A. A. Suratgar, and A. Afshar, "Distributed Nonconvex Optimization for Energy Efficiency in Mobile Ad Hoc Networks," *IEEE Systems Journal*, 2021.

[12] I. Necoara and V. Nedelcu, "On linear convergence of a distributed dual gradient algorithm for linearly constrained separable convex problems," *Automatica*, vol. 55, pp. 209–216, 2015.

[13] A. Falsone, K. Margellos, S. Garatti, and M. Prandini, "Dual decomposition for multi-agent distributed optimization with coupling constraints," *Automatica*, vol. 84, pp. 149–158, 2017.

[14] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[15] T.-Y. Lin, S.-Q. Ma, and S.-Z. Zhang, "On the sublinear convergence rate of multi-block ADMM,"

Journal of the Operations Research Society of China, vol. 3, no. 3, pp. 251–274, 2015.

[16] X. Cai, D. Han, and X. Yuan, "On the convergence of the direct extension of ADMM for three-block separable convex minimization models with one strongly convex function," *Computational Optimization and Applications*, vol. 66, no. 1, pp. 39–73, 2017.

[17] J. Bai, J. Li, F. Xu, and H. Zhang, "Generalized symmetric ADMM for separable convex optimization," *Computational optimization and applications*, vol. 70, no. 1, pp. 129–170, 2018.

[18] N. Chatzipanagiotis and M. M. Zavlanos, "On the convergence of a distributed augmented lagrangian method for nonconvex optimization," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4405–4420, 2017.

[19] N. S. Aybat, Z. Wang, T. Lin, and S. Ma, "Distributed linearized alternating direction method of multipliers for composite convex consensus optimization," *IEEE Transactions on Automatic Control*, vol. 63, no. 1, pp. 5–20, 2017.

[20] J. Bai, W. W. Hager, and H. Zhang, "An inexact accelerated stochastic ADMM for separable convex optimization," *Computational Optimization and Applications*, pp. 1–40, 2022.

[21] L. Yang, T. K. Pong, and X. Chen, "Alternating direction method of multipliers for a class of nonconvex and nonsmooth problems with applications to background/foreground extraction," *SIAM Journal on Imaging Sciences*, vol. 10, no. 1, pp. 74–110, 2017.

[22] K. Guo, D. Han, and T.-T. Wu, "Convergence of alternating direction method for minimizing sum of two nonconvex functions with linear constraints," *International Journal of Computer Mathematics*, vol. 94, no. 8, pp. 1653–1669, 2017.

[23] G. Li and T. K. Pong, "Global convergence of splitting methods for nonconvex composite optimization," *SIAM Journal on Optimization*, vol. 25, no. 4, pp. 2434–2460, 2015.

[24] Q. Liu, X. Shen, and Y. Gu, "Linearized ADMM for nonconvex nonsmooth optimization with convergence analysis," *IEEE Access*, vol. 7, pp. 76131–76144, 2019.

[25] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 337–364, 2016.

[26] K. Sun and X. A. Sun, "A two-level distributed algorithm for general constrained non-convex optimization with global convergence," *arXiv preprint arXiv:1902.07654*, 2019.

[27] K. Sun and X. A. Sun, "A two-level ADMM algorithm for AC OPF with convergence guarantees," *IEEE Transactions on Power Systems*, 2021.

[28] X. Li, G. Feng, and L. Xie, "Distributed proximal algorithms for multiagent optimization with coupled inequality constraints," *IEEE Transactions on Automatic Control*, vol. 66, no. 3, pp. 1223–1230, 2020.

[29] T.-H. Chang, M. Hong, and X. Wang, "Multi-agent distributed optimization via inexact consensus ADMM," *IEEE Transactions on Signal Processing*, vol. 63, no. 2, pp. 482–497, 2014.

[30] S. Lu, J. D. Lee, M. Razaviyayn, and M. Hong, "Linearized ADMM converges to second-order stationary points for non-convex problems," *IEEE Transactions on Signal Processing*, vol. 69, pp. 4859–4874, 2021.

[31] P. Jain and P. Kar, "Non-convex optimization for machine learning," 2017.

[32] H. Li and Z. Lin, "Accelerated proximal gradient methods for nonconvex programming," *Advances in neural information processing systems*, vol. 28, 2015.

A Proof of Proposition 1

Prop. 1 is established based on the first-order optimality condition of subproblems (5) and the Lipschitz continuous gradient property of f and g.

We first establish the following equality and notation.

$$\begin{aligned}
&\mathbf{A}_i\mathbf{x}_i^{k+1} + \textstyle\sum_{j\neq i}\mathbf{A}_j\mathbf{x}_j^k - \mathbf{b} && \text{(A.1)}\\
&\quad = \mathbf{A}\mathbf{x}^k - \mathbf{b} + \mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k).\\
&\quad = \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b} + \mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k+1}) + \mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k).\\
&\hat{\boldsymbol{\lambda}}^k := \boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}). && \text{(A.2)}
\end{aligned}$$

For subproblems (5), the first-order optimality condition states there exists $\nu_i^{k+1} \in N_{\mathbf{X}_i}(\mathbf{x}_i^{k+1})$ such that

$$\begin{aligned}
0 = \;& \nabla f_i(\mathbf{x}_i^{k+1}) + \nabla g_i(\mathbf{x}^k) + \mathbf{A}_i^\top \boldsymbol{\lambda}^k\\
& + \rho\mathbf{A}_i^\top(\mathbf{A}_i\mathbf{x}_i^{k+1} + \textstyle\sum_{j\neq i}\mathbf{A}_j\mathbf{x}_j^k - \mathbf{b})\\
& + \beta\mathbf{B}_i^\top\mathbf{B}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k) + \nu_i^{k+1}\\
= \;& \nabla f_i(\mathbf{x}_i^{k+1}) + \nabla g_i(\mathbf{x}^k) + \mathbf{A}_i^\top\big(\boldsymbol{\lambda}^k + \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b})\big)\\
& + \rho\mathbf{A}_i^\top\mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k+1}) + \rho\mathbf{A}_i^\top\mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k)\\
& + \beta\mathbf{B}_i^\top\mathbf{B}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k) + \nu_i^{k+1} \qquad \text{by (A.1)}\\
= \;& \nabla f_i(\mathbf{x}_i^{k+1}) + \nabla g_i(\mathbf{x}^k) + \mathbf{A}_i^\top\hat{\boldsymbol{\lambda}}^k + \rho\mathbf{A}_i^\top\mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k+1})\\
& + \rho\mathbf{A}_i^\top\mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k)\\
& + \beta\mathbf{B}_i^\top\mathbf{B}_i\big(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k\big) + \nu_i^{k+1} \qquad \text{by (A.2).}
\end{aligned}$$

Multiplying by $(\mathbf{x}_i^{k+1} - \mathbf{x}_i)$ in both sides, we have

$$\begin{aligned}
&\langle\nabla f_i(\mathbf{x}_i^{k+1}), \mathbf{x}_i^{k+1} - \mathbf{x}_i\rangle + \langle\nabla_i g(\mathbf{x}^k), \mathbf{x}_i^{k+1} - \mathbf{x}_i\rangle\\
&+ \langle\hat{\boldsymbol{\lambda}}^k, \mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i)\rangle\\
&+ \rho\langle\mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k+1}), \mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i)\rangle\\
&+ \rho\langle\mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k)\mathbf{A}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i)\rangle && \text{(A.3)}\\
&+ \beta\langle\mathbf{B}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i^k), \mathbf{B}_i(\mathbf{x}_i^{k+1} - \mathbf{x}_i)\rangle\\
=& -\langle\nu_i^{k+1}, \mathbf{x}_i^{k+1} - \mathbf{x}_i\rangle \leq 0, \quad \forall\mathbf{x}_i \in \mathbf{X}_i.
\end{aligned}$$

Summing up (A.3) over i, we have $\forall\mathbf{x}_i \in \mathbf{X}_i$,

$$
\begin{aligned}
&\langle \nabla f(\mathbf{x}^{k+1}), \mathbf{x}^{k+1} - \mathbf{x}\rangle + \langle \nabla g(\mathbf{x}^{k}), \mathbf{x}^{k+1} - \mathbf{x}\rangle \\
&\quad + \langle \hat{\boldsymbol{\lambda}}^{k}, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x})\rangle + (\mathbf{x}^{k+1} - \mathbf{x})\rho\mathbf{A}^{\top}\mathbf{A}(\mathbf{x}^{k} - \mathbf{x}^{k+1}) \\
&\quad + \textstyle\sum_i (\mathbf{x}_i^{k+1} - \mathbf{x}_i)^{\top}(\rho\mathbf{A}_i^{\top}\mathbf{A}_i + \beta\mathbf{B}_i^{\top}\mathbf{B}_i)(\mathbf{x}_i^{k+1} - \mathbf{x}_i^{k}) \le 0.
\end{aligned}
$$

Plugging in $\mathbf{Q} := \rho G_{\mathbf{A}} + \beta G_{\mathbf{B}} - \rho\mathbf{A}^{\top}\mathbf{A}$, we have

$$
\begin{aligned}
&\langle \nabla f(\mathbf{x}^{k+1}), \mathbf{x}^{k+1} - \mathbf{x}\rangle + \langle \nabla g(\mathbf{x}^{k}), \mathbf{x}^{k+1} - \mathbf{x}\rangle \\
&\quad + \langle \hat{\boldsymbol{\lambda}}^{k}, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x})\rangle \\
&\quad + (\mathbf{x}^{k+1} - \mathbf{x})^{\top}\mathbf{Q}(\mathbf{x}^{k+1} - \mathbf{x}^{k}) \le 0, \quad \forall \mathbf{x} \in \mathbf{X}.
\end{aligned} \tag{A.4}
$$

By induction, we have for iteration $k-1$ that

$$
\begin{aligned}
&\Big\langle \nabla f(\mathbf{x}^{k}), \mathbf{x}^{k} - \mathbf{x}\Big\rangle + \langle \nabla g(\mathbf{x}^{k-1}), \mathbf{x}^{k} - \mathbf{x}\rangle \\
&\quad + \Big\langle \hat{\boldsymbol{\lambda}}^{k-1}, \mathbf{A}(\mathbf{x}^{k} - \mathbf{x})\Big\rangle \\
&\qquad + (\mathbf{x}^{k} - \mathbf{x})^{\top}\mathbf{Q}(\mathbf{x}^{k} - \mathbf{x}^{k-1}) \le 0, \quad \forall \mathbf{x} \in \mathbf{X}.
\end{aligned} \tag{A.5}
$$

By setting $\mathbf{x} := \mathbf{x}^{k}$ and $\mathbf{x} := \mathbf{x}^{k+1}$ with (A.4) and (A.5), we have

$$
\begin{aligned}
&\langle \nabla f(\mathbf{x}^{k+1}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle + \langle \nabla g(\mathbf{x}^{k}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \\
&\quad + \langle \hat{\boldsymbol{\lambda}}^{k}, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^{k})\rangle \\
&\quad + (\mathbf{x}^{k+1} - \mathbf{x}^{k})^{\top}\mathbf{Q}(\mathbf{x}^{k+1} - \mathbf{x}^{k}) \le 0.
\end{aligned} \tag{A.6}
$$

$$
\begin{aligned}
&\langle \nabla f(\mathbf{x}^{k}), \mathbf{x}^{k} - \mathbf{x}^{k+1}\rangle + \langle \nabla g(\mathbf{x}^{k-1}), \mathbf{x}^{k} - \mathbf{x}^{k+1}\rangle \\
&\quad + \langle \hat{\boldsymbol{\lambda}}^{k-1}, \mathbf{A}(\mathbf{x}^{k} - \mathbf{x}^{k+1})\rangle \\
&\quad + (\mathbf{x}^{k} - \mathbf{x}^{k+1})^{\top}\mathbf{Q}(\mathbf{x}^{k} - \mathbf{x}^{k-1}) \le 0.
\end{aligned} \tag{A.7}
$$

Summing up (A.6) and (A.7) and plugging in $\mathbf{w}^{k} := (\mathbf{x}^{k+1} - \mathbf{x}^{k}) - (\mathbf{x}^{k} - \mathbf{x}^{k-1})$, we have

$$
\begin{aligned}
&\langle \nabla f(\mathbf{x}^{k+1}) - \partial f(\mathbf{x}^{k}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \\
&\quad + \langle \nabla g(\mathbf{x}^{k}) - \nabla g(\mathbf{x}^{k-1}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \\
&\quad + \langle \hat{\boldsymbol{\lambda}}^{k} - \hat{\boldsymbol{\lambda}}^{k-1}, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^{k})\rangle \\
&\quad + (\mathbf{x}^{k+1} - \mathbf{x}^{k})^{\top}\mathbf{Q}\mathbf{w}^{k} \le 0.
\end{aligned} \tag{A.8}
$$

Based on the Lipschitz continuous gradient property of f over the compact set $\mathbf{x} \in \mathbf{X}$, we have

$$
\langle \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^{k}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \ge -L_f\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|^2. \tag{A.9}
$$

We also have

$$
\begin{aligned}
&\langle \nabla g(\mathbf{x}^{k}) - \nabla g(\mathbf{x}^{k-1}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \\
&= \langle \frac{\nabla g(\mathbf{x}^{k}) - \nabla g(\mathbf{x}^{k-1})}{\sqrt{L_g}}, \sqrt{L_g}(\mathbf{x}^{k+1} - \mathbf{x}^{k})\rangle \\
&\ge -\frac{1}{2L_g}\|\nabla g(\mathbf{x}^{k}) - \nabla g(\mathbf{x}^{k-1})\|^2 - \frac{L_g}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|^2 \\
&\ge -\frac{L_g}{2}\|\mathbf{x}^{k} - \mathbf{x}^{k-1}\|^2 - \frac{L_g}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|^2
\end{aligned}
$$

where the last equality is based on the Lipschitz continuous gradient property of g.

Besides, we have

$$
\begin{aligned}
&\langle \hat{\boldsymbol{\lambda}}^{k} - \hat{\boldsymbol{\lambda}}^{k-1}, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^{k})\rangle \\
&= \Big\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k} + \tau(\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}), \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^{k})\Big\rangle \\
&= \Big\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k} + \tau(\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}), \\
&\qquad \frac{\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}}{\rho} - \frac{(1-\tau)}{\rho}(\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1})\Big\rangle \\
&= \frac{\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2}{\rho} - \frac{(1-2\tau)}{\rho}\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}, \boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\rangle \\
&\quad - \frac{\tau(1-\tau)}{\rho}\|\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\|^2 \\
&\ge \frac{\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2}{\rho} - \frac{1-2\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2 \\
&\quad - \frac{1-2\tau}{2\rho}\|\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\|^2 - \frac{\tau(1-\tau)}{\rho}\|\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\|^2 \\
&= \frac{1-2\tau^2}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2 - \frac{1-2\tau^2}{2\rho}\|\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\|^2 \\
&\quad + \tau(\tau+1)/\rho\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2
\end{aligned} \tag{A.10}
$$

where the inequality is based on $\langle \mathbf{a}, \mathbf{b}\rangle \le \frac{1}{2}(\|\mathbf{a}\|^2 + \|\mathbf{b}\|^2)$.

Based on the inequality $\mathbf{b}^{\top}\mathbf{M}(\mathbf{b} - \mathbf{c}) = \frac{1}{2}(\|\mathbf{b} - \mathbf{c}\|_{\mathbf{M}}^2 + \|\mathbf{b}\|_{\mathbf{M}}^2 - \|\mathbf{c}\|_{\mathbf{M}}^2)$, and by setting $\mathbf{M} = \mathbf{Q}$, $\mathbf{b} = \mathbf{x}^{k+1} - \mathbf{x}^{k}$, and $\mathbf{c} = \mathbf{x}^{k} - \mathbf{x}^{k-1}$, we have

$$
\begin{aligned}
(\mathbf{x}^{k+1} - \mathbf{x}^{k})^{\top}\mathbf{Q}\mathbf{w}^{k} = &\frac{1}{2}(\|\mathbf{w}^{k}\|_{\mathbf{Q}}^2 + \|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|_{\mathbf{Q}}^2 \\
&- \|\mathbf{x}^{k} - \mathbf{x}^{k-1}\|_{\mathbf{Q}}^2).
\end{aligned} \tag{A.11}
$$

Plugging (A.9), (A.10), (A.11) into (A.8), we have

$$
\begin{aligned}
&\frac{1-2\tau^2}{2\rho}\left\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\right\|^2 + \frac{1}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|_{\mathbf{Q}}^2 \\
&\quad + \frac{L_g}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\|^2 + \frac{1}{2}\|\mathbf{w}^{k}\|_{\mathbf{Q}}^2 \\
&\le \frac{1-2\tau^2}{2\rho}\left\|\boldsymbol{\lambda}^{k} - \boldsymbol{\lambda}^{k-1}\right\|^2 + \frac{1}{2}\|\mathbf{x}^{k} - \mathbf{x}^{k-1}\|_{\mathbf{Q}}^2 \\
&\quad + \frac{L_g}{2}\|\mathbf{x}^{k} - \mathbf{x}^{k-1}\|^2 + (L_g + L_f)\left\|\mathbf{x}^{k+1} - \mathbf{x}^{k}\right\|^2 \\
&\quad - \tau(1+\tau)/\rho\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^{k}\|^2.
\end{aligned}
$$

We therefore complete the proof.

B Proof of Proposition 2

Before starting the proof, we first establish the following inequalities to be used. Based on the Lipschitz continuous gradient property of $f : \mathbf{R}^n \to \mathbf{R}$ over $\mathbf{x} \in \mathbf{X}$ (see (A1)), we have [22]

$$
\begin{aligned}
f(\mathbf{x}^{k+1}) - f(\mathbf{x}^{k}) \le\ &\langle \nabla f(\mathbf{x}^{k+1}), \mathbf{x}^{k+1} - \mathbf{x}^{k}\rangle \\
&+ L_f/2\|\mathbf{x}^{k} - \mathbf{x}^{k+1}\|^2.
\end{aligned} \tag{B.1}
$$

Similarly, for $g : \mathbf{R}^n \to \mathbf{R}$ with Lipschitz continuous gradient over $\mathbf{x} \in \mathbf{X}$ (see (A1)), we have [22]

$$\begin{aligned} g(\mathbf{x}^{k+1}) - g(\mathbf{x}^k) \le \;& \langle \nabla g(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle \\ & + L_g/2\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2. \end{aligned} \tag{B.2}$$

Besides, we have

$$\begin{aligned} & \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}\|^2 - \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 \\ = \;& \frac{\rho}{2}\left\langle \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k), \mathbf{A}\mathbf{x}^{k+1} + \mathbf{A}\mathbf{x}^k - 2\mathbf{b} \right\rangle \\ = \;& \frac{\rho}{2}\left\langle \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k), \mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k+1}) + 2(\mathbf{A}\mathbf{x}^k - \mathbf{b}) \right\rangle \\ = & -\frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 + \left\langle \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k), \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \right\rangle. \end{aligned} \tag{B.3}$$

We next quantify the decrease of $\mathbb{L}_\rho(\mathbf{x}, \boldsymbol{\lambda})$ with respect to (w.r.t.) the primal updates. We have

$$\begin{aligned} & \mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^k) - \mathbb{L}_\rho(\mathbf{x}^k, \boldsymbol{\lambda}^k) \\ = \;& f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k) + g(\mathbf{x}^{k+1}) - g(\mathbf{x}^k) + \left\langle \boldsymbol{\lambda}^k, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k) \right\rangle \\ & + \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}\|^2 - \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^k - \mathbf{b}\|^2 \\ \le \;& \langle \nabla f(\mathbf{x}^{k+1}) + \nabla g(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \rho_F/2\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \\ & + \left\langle \boldsymbol{\lambda}^k, \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k) \right\rangle - \frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 \\ & + \left\langle \mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k), \rho(\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}) \right\rangle \text{ by (B.1), (B.2), (B.3)} \\ = \;& \langle \nabla f(\mathbf{x}^{k+1}) + \nabla g(\mathbf{x}^k) + \mathbf{A}^\top \hat{\boldsymbol{\lambda}}^k, \mathbf{x}^{k+1} - \mathbf{x}^k \rangle \\ & + \rho_F/2\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 - \rho/2\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 \text{ by (A.2)} \\ \le & -\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2_{\mathbf{Q}} + \rho_F/2\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \\ & -\rho/2\ \|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 \text{ by (A.6).} \end{aligned} \tag{B.4}$$

We next quantify the change of $\mathbb{L}_\rho(\mathbf{x}, \boldsymbol{\lambda})$ w.r.t. dual update. We have

$$\begin{aligned} & \mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) - \mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^k) \\ = & \left\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k, \mathbf{A}\mathbf{x}^{k+1} - \mathbf{b} \right\rangle \\ = & \left\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k, \frac{\boldsymbol{\lambda}^{k+1} - (1-\tau)\boldsymbol{\lambda}^k}{\rho} \right\rangle \\ = & \left\langle \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k, \frac{1-\tau}{\rho}(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) + \frac{\tau}{\rho}\boldsymbol{\lambda}^{k+1} \right\rangle \\ = & \frac{(1-\tau)}{\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{\tau}{2\rho}\Big(\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 \\ & + \|\boldsymbol{\lambda}^{k+1}\|^2 - \|\boldsymbol{\lambda}^k\|^2\Big) \\ = & \frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^k\|^2. \end{aligned} \tag{B.5}$$

Combining (B.4) and (B.5), we have

$$\begin{aligned} & \mathbb{L}_\rho(\mathbf{x}^{k+1}, \boldsymbol{\lambda}^{k+1}) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 - \big(\mathbb{L}_\rho(\mathbf{x}^k, \boldsymbol{\lambda}^k) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^k\|^2\big) \\ \le & -\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2_{\mathbf{Q}} + \frac{\rho_F}{2}\|\mathbf{x}^k - \mathbf{x}^{k+1}\|^2 \\ & - \frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^{k+1} - \mathbf{x}^k)\|^2 + \frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2. \end{aligned}$$

We therefore close the proof.

C Proof of Corollary 1

i) Prove $\|\boldsymbol{\lambda}^*\|^2 \le \rho\tau^{-1}\mathbf{T}_c^0$: Based on the sufficiently decreasing property of $\mathbf{T}_c^{k+1}$ (see **Prop.** 3), we have

$$\mathbf{T}_c^{k+1} \le \mathbf{T}_c^0 \tag{C.1}$$

Recalling the definition of the Lyapunov function in (7) and invoking (10), we have

$$\begin{aligned} \mathbf{T}_c^{k+1} = \;& f(\mathbf{x}^{k+1}) + g(\mathbf{x}^{k+1}) + \frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 \\ & + \frac{1-\tau}{2\rho}(\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 + \|\boldsymbol{\lambda}^{k+1}\|^2 - \|\boldsymbol{\lambda}^k\|^2) \\ & + \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^{k+1} - \mathbf{b}\|^2 + c\Big(\frac{1-2\tau^2}{2\rho}\|\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k\|^2 \\ & + 1/2\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2_{\mathbf{Q}} + L_g/2\|\mathbf{x}^k - \mathbf{x}^{k-1}\|^2\Big) \end{aligned} \tag{C.2}$$

We have $f \ge 0$ and $g \ge 0$ over $\mathbf{X}$. By combing (C.1) and (C.2), we have (the other terms are all non-negative)

$$\frac{1-\tau}{2\rho}(\|\boldsymbol{\lambda}^{k+1}\|^2 - \|\boldsymbol{\lambda}^k\|^2) + \frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 \le \mathbf{T}_c^0 \tag{C.3}$$

We next prove $\frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 \le \mathbf{T}_c^0$ by induction. For $k = 0$, we can properly pick the initial point to satisfy the inequality. For iteration k, we assume $\frac{\tau}{\rho}\|\boldsymbol{\lambda}^k\|^2 \le \mathbf{T}_c^0$. We consider two possible cases for iteration $k+1$, i.e., if $\|\boldsymbol{\lambda}^{k+1}\|^2 \le \|\boldsymbol{\lambda}^k\|^2$, we straightforwardly have $\frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 \le \frac{\tau}{\rho}\|\boldsymbol{\lambda}^k\|^2 \le \mathbf{T}_c^0$, and else if $\|\boldsymbol{\lambda}^{k+1}\|^2 \ge \|\boldsymbol{\lambda}^k\|^2$, we also have $\frac{\tau}{\rho}\|\boldsymbol{\lambda}^{k+1}\|^2 \le \mathbf{T}_c^0$ by (C.3). We therefore conclude $\|\boldsymbol{\lambda}^*\|^2 \le \rho\tau^{-1}\mathbf{T}_c^0$.

ii) Prove $\mathbf{T}_c^0 \le \big(4 + c(1 - 2\tau^2) + c/2\big)d_F + cL_g/2\|\mathbf{x}^0\|^2 + c\rho_F/4\ d_{\mathbf{x}}$: Invoking **Prop.** 2 and set $k = 0$, we have

$$\begin{aligned} & \mathbb{L}_\rho(\mathbf{x}^1, \boldsymbol{\lambda}^1) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^1\|^2 \le \mathbb{L}_\rho(\mathbf{x}^0, \boldsymbol{\lambda}^0) - \frac{\tau}{2\rho}\|\boldsymbol{\lambda}^0\|^2 \\ & - \|\mathbf{x}^1 - \mathbf{x}^0\|^2_{\mathbf{Q}} + \frac{\rho_F}{2}\|\mathbf{x}^1 - \mathbf{x}^0\|^2 - \frac{\rho}{2}\|\mathbf{A}(\mathbf{x}^1 - \mathbf{x}^0)\|^2 \\ & + \frac{2-\tau}{2\rho}\|\boldsymbol{\lambda}^1 - \boldsymbol{\lambda}^0\|^2. \end{aligned}$$

By invoking (10), $\boldsymbol{\lambda}^0 = 0$, $\mathbf{A}\mathbf{x}^0 = b$, $\mathbf{Q} := \rho G_{\mathbf{A}} + \beta G_{\mathbf{B}} - \rho\mathbf{A}^\top\mathbf{A}$ and set $\boldsymbol{\lambda}^{-1} = 0$, we have

$$\begin{aligned} & \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^1 - \mathbf{b}\|^2 + \frac{2\mathbf{Q} + \rho\mathbf{A}^\top\mathbf{A} - \rho_f\mathbf{I}_N}{2}\|\mathbf{x}^1 - \mathbf{x}^0\|^2 \\ & \le f(\mathbf{x}^0) + g(\mathbf{x}^0) - f(\mathbf{x}^1) - g(\mathbf{x}^1) \end{aligned}$$

Since we have $f(\mathbf{x}) \geq 0$ and $f(\mathbf{x}) \geq 0$ over the $\mathbf{X}$, we have (the term $\frac{\rho \mathbf{A}^\top \mathbf{A}}{2}\|\mathbf{x}^1 - \mathbf{x}^0\|^2$ is non-negative)

$$\begin{aligned} &\frac{\rho}{2}\|\mathbf{A}\mathbf{x}^1 - \mathbf{b}\|^2 \leq d_F. && \text{(C.4)} \\ &\frac{2\mathbf{Q} - \rho_F \mathbf{I}_N}{2}\|\mathbf{x}^1 - \mathbf{x}^0\|^2 \leq d_F \\ \Rightarrow\ &\|\mathbf{x}^1 - \mathbf{x}^0\|_{\mathbf{Q}}^2 \leq d_F + \rho_F/2 d_{\mathbf{x}}. && \text{(C.5)} \end{aligned}$$

where the last inequality is by $d_{\mathbf{x}} := \max_{\mathbf{x},\mathbf{y}} \|\mathbf{x} - \mathbf{y}\|^2$.

Further, based on the dual update, we have

$$\frac{1}{2\rho}\|\boldsymbol{\lambda}^1\|^2 = \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^1 - \mathbf{b}\|^2 \leq d_F \qquad \text{(C.6)}$$

Further, we bound $\mathbf{T}_c^0$ and we have

$$\begin{aligned} \mathbf{T}_c^0 &= f(\mathbf{x}^1) + g(\mathbf{x}^1) + \frac{2 + c(1 - 2\tau^2)}{2\rho}\|\boldsymbol{\lambda}^1\|^2 + \frac{\rho}{2}\|\mathbf{A}\mathbf{x}^1 - \mathbf{b}\|^2 \\ &\quad + \frac{c}{2}\|\mathbf{x}^1 - \mathbf{x}^0\|_{\mathbf{Q}}^2 + \frac{cL_g}{2}\|\mathbf{x}^0\|^2 \ \text{ by (C.2) and } \boldsymbol{\lambda}^0 = 0 \\ &\leq\ d_F + (2 + c(1 - 2\tau^2))d_F + d_F \\ &\qquad + \frac{c}{2}d_F + \frac{c\rho_F}{4}d_{\mathbf{x}} + \frac{cL_g}{2}\|\mathbf{x}^0\|^2 \ \text{ by (C.4), (C.5), (C.6)} \\ &=\ \big(4 + c(1 - 2\tau^2) + c/2\big)d_F + cL_g/2\|\mathbf{x}^0\|^2 + c\rho_F/4\ d_{\mathbf{x}} \end{aligned}$$

Based on **Theorem** 1 and i), ii), we therefore have

$$\begin{aligned} &\operatorname{dist}\big(\nabla f(\mathbf{x}^*) + \nabla g(\mathbf{x}^*) + \mathbf{A}^\top \boldsymbol{\lambda}^* + N_{\mathbf{X}}(\mathbf{x}^*), 0\big) \\ &\qquad + \|\mathbf{A}\mathbf{x}^* - \mathbf{b}\|^2 \leq \epsilon, \end{aligned}$$

which thus closes the proof.