Implementing the Alternating Direction Method of Multipliers for Big Datasets

Yue, Hangrui; Yang, Qingzhi; Wang, Xiangfeng; Yuan, Xiaoming

Published in:
SIAM Journal on Scientific Computing

DOI:
10.1137/17M1146567

Published: 01/01/2018

Document Version:
Publisher's PDF, also known as Version of record

SIAM J. SCI. COMPUT.
Vol. 40, No. 5, pp. A3121–A3156

IMPLEMENTING THE ALTERNATING DIRECTION METHOD OF MULTIPLIERS FOR BIG DATASETS: A CASE STUDY OF LEAST ABSOLUTE SHRINKAGE AND SELECTION OPERATOR*

HANGRUI YUE†, QINGZHI YANG†, XIANGFENG WANG‡, AND XIAOMING YUAN§

Abstract. The alternating direction method of multipliers (ADMM) has been extensively used in a wide variety of different applications. When large datasets with high-dimensional variables are considered, subproblems arising from the ADMM must be inexactly solved even though they may theoretically have closed-form solutions. Such a scenario immediately poses mathematical ambiguities such as how these subproblems should be accurately solved and whether the convergence can still be guaranteed. Although the ADMM is well known, it seems that these topics should be deeply investigated. In this paper, we study the mathematics of how to implement the ADMM for a large dataset scenarios. More specifically, we attempt to focus on the convex programming case where there is a quadratic function with extremely high-dimensional variables in the objective function of the model; thereby there is a huge-scale system for linear equations needing to be solved at each iteration of the ADMM. It is revealed that there is no need, indeed it is impossible, to exactly solve this linear system, and we attempt to propose an adjustable inexactness criterion to automatically and inexactly solve this linear system. We further attempt to identify the safe-guard number for the internally nested iterations that can sufficiently ensure this inexactness criterion if the linear system would be solved by a standard numerical linear algebra solver. The convergence, together with the worst-case convergence rate measured by the iteration complexity, is rigorously established for the ADMM with inexactly solved subproblems. Some numerical experiments for large datasets of the least absolute shrinkage and selection operator containing millions of variables are reported to show the efficiency of the mentioned inaccurate implementation of the ADMM.

Key words. convex programming, alternating direction method of multipliers, high dimension, big data, LASSO, distributed LASSO, convergence

AMS subject classifications. 65K10, 90C06, 90C25, 47H05

DOI. 10.1137/17M1146567

1. Introduction.

In this study, we discuss how to implement alternating direction method of multipliers (ADMM) for large datasets in the convex programming context and present a rigorous mathematical analysis for its convergence. The ADMM was originally proposed in [8, 21] for nonlinear elliptic equations, and it has been extensively used across a broad range of applications in various areas, such as image processing, statistical learning, computer vision, wireless communication networks, and so on. It becomes a benchmark first-order solver for various convex minimization

*Submitted to the journal's Methods and Algorithms for Scientific Computing section September 6, 2017; accepted for publication (in revised form) June 15, 2018; published electronically September 25, 2018.

http://www.siam.org/journals/sisc/40-5/M114656.html

Funding: The first author was supported by NSFC grant 11671217 and by the Ph.D. Candidate Research Innovation Foundation of Nankai University. The second author was supported by NSFC grants 11271206 and 11671217. The third author was supported by NSFC grant 11501210 and 61672231. The fourth author was supported by the General Research Fund from Hong Kong Research Grants Council: 12302318.

†School of Mathematical Sciences and LPMC, Nankai University, Tianjin, China (yuehangrui@gmail.com, qz-yang@nankai.edu.cn).

‡Shanghai Key Lab for Trustworthy Computing, School of Computer Science and Software Engineering, East China Normal University, Shanghai, China (xfwang@sei.ecnu.edu.cn).

§Corresponding author. Department of Mathematics, The University of Hong Kong, Hong Kong, China (xmyuan@hku.hk).

models with separable objective functions and is being extensively explored in the nonconvex or multi-block contexts. We refer to [5, 13, 20] for some review papers on the ADMM.

Let us focus on the separable convex programming problems with linear constraints and an objective function in the form of two functions without coupled variables:

$$\begin{aligned} \min_{x\in\mathbb{R}^n, y\in\mathbb{R}^m} \quad & \underbrace{\frac{1}{2}\|Qx-q\|_2^2}_{f(x)} + g(y) \\ \text{such that (s.t.)} \quad & Ax+By=b, \end{aligned} \tag{1.1}$$

where $Q\in\mathbb{R}^{p\times n}$, $q\in\mathbb{R}^p$, $g:\mathbb{R}^m\to\mathbb{R}$ is a general convex (not necessarily smooth) closed function, $A\in\mathbb{R}^{\ell\times n}$, $B\in\mathbb{R}^{\ell\times m}$, and $b\in\mathbb{R}^\ell$. Instead of considering a generic convex function $f(x)$ in (1.1), we concentrate only on the quadratic case because our emphasis, as it will be delineated, is on the implementation of ADMM when the x-subproblem at each iteration is a system of linear equations. We particularly assume $p \ll n$, so that the model (1.1) captures the fundamental model of the least absolute shrinkage and selection operator (LASSO):

$$\min_{x\in\mathbb{R}^n} \ \frac{1}{2}\|Qx-q\|_2^2 + \tau\|x\|_1. \tag{1.2}$$

Note that the LASSO model (1.2) can be easily reformulated as the special case of (1.1) with $A=I_{n\times n}$, $B=-I_{n\times n}$, $b=0$, $g(y)=\tau\|y\|_1$; see (1.6) for details. For the LASSO problem (1.2), the first term, in (1.2) is a data fidelity term, and the term $\|x\|_1$ prompts the sparsest solution of the underdetermined system of linear equations $Qx=q$ with $p\ll n$. The LASSO model (1.2) was initiated in [43], and it is fundamentally applied to several fields, such as compressive sensing [17], statistical learning [23], medical image processing [33], radar signal recovery [3], robust feature selection in machine learning [38], etc. More sophisticated applications of the LASSO model (1.2) also include distributed optimization problems arising in multiagent network models such as those approaches previously proposed in [1, 4, 9, 10, 37, 51]. The scheme proposed in this study is applicable to more general models (1.1), but with an exclusive emphasis on the LASSO model (1.2) because of its importance. Throughout, we assume that the matrix A in (1.1) is of full column rank, the inverse of $A^\top A$ can be easily computed, and the solution set of (1.1) is nonempty to avoid triviality.

Let us define the augmented Lagrangian function of (1.1) as

$$\mathcal{L}_\beta(x,y,\lambda) = f(x)+g(y)-\lambda^\top(Ax+By-b)+\frac{\beta}{2}\|Ax+By-b\|_2^2, \tag{1.3}$$

where $\lambda\in\mathbb{R}^\ell$ is the Lagrange multiplier and $\beta>0$ represents a penalty parameter. Then, the iterative scheme of ADMM for (1.1) reads as

$$x^{k+1} := \arg\min_{x\in\mathbb{R}^n} \mathcal{L}_\beta\left(x, y^k, \lambda^k\right), \tag{1.4a}$$

$$y^{k+1} := \arg\min_{y\in\mathbb{R}^m} \mathcal{L}_\beta\left(x^{k+1}, y, \lambda^k\right), \tag{1.4b}$$

$$\lambda^{k+1} := \lambda^k - \beta\left(Ax^{k+1}+By^{k+1}-b\right). \tag{1.4c}$$

It can be clearly seen that subproblems in the ADMM (1.4) are generally easier than the original problem (1.1). Indeed, it is commented in [5] that "a simple algorithm

derived from basic ADMM will often offer performance that is at least comparable to very specialized algorithms (even in the serial setting), and in most cases, the simple ADMM will be efficient enough to be useful." To simplify the discussion, we assume that β is fixed in the proposed theoretical analysis even though it does require us to discuss how to dynamically adjust it for numerical implementation and analysis.

It is worth mentioning that for some special cases, (1.4a) and (1.4b) may be easy enough to have closed-form solutions, and thus no internal iterations are involved when implementing (1.4); refer to (1.9) and (1.10) when the LASSO model (1.2) is taken into account. This feature indeed is the main reason that the ADMM algorithm is extensively used in the mentioned areas. Meanwhile, various inexact versions of the ADMM have been reported in the literature for different settings. For instance, the proximal version of ADMM in [11, 24, 26] adds appropriate quadratic terms in order to regularize the subproblems, thereby alleviating these subproblems. In particular, some interesting cases are the linearized version of ADMM (e.g., [25, 46, 48]) and the preconditioned version of ADMM (e.g., [6, 7]). For the case where $f(x)$ in (1.1) is quadratic, as elaborated in [6, 7], by choosing appropriate preconditioners, the preconditioned version of ADMM realizes inaccurately solving the resulting subproblems by various iterative schemes (e.g., the damped Jacobi, symmetric Gauss–Seidel, and symmetric successive over-relaxation). Generically, (1.4a) and (1.4b) should be iteratively solved, and only approximate solutions can be pursued by internal iterations. Hence, the ADMM (1.4) should be generally implemented as the following with internally nested iterations:

$$\begin{cases} x^{k+1} \approx \arg\min\limits_{x\in\mathbb{R}^n} \mathcal{L}_\beta\left(x, y^k, \lambda^k\right), \\ y^{k+1} \approx \arg\min\limits_{y\in\mathbb{R}^m} \mathcal{L}_\beta\left(x^{k+1}, y, \lambda^k\right), \\ \lambda^{k+1} = \lambda^k - \beta\left(Ax^{k+1} + By^{k+1} - b\right). \end{cases} \tag{1.5}$$

It is mathematically important to define well the inexactness criterion in (1.5) and study the rigorous convergence with the x- and y-subproblems solved inexactly. It should be mentioned that although the convergence of ADMM has been well studied in both earlier literature (e.g., [12, 16, 18, 19, 22, 27]) and recent published papers [28, 29, 36], these results are valid only for the exact version (1.4), in which both (1.4a) and (1.4b) are assumed to be exactly solved. Hence, the convergence of (1.5) with internally nested iterations should be analyzed from scratch. When the generic case (1.5) is considered, a general rule to guarantee the convergence of (1.5) is that the accuracy of an inexactness criterion should keep increasing as iterations go on. Thus, how to efficiently specify an inexactness criterion and accuracy in (1.5), which is a significant issue for numerical implementation, can be discussed using a specific scenario for the generic model (1.1). In this paper, we concentrate on the latter case, and scholars may refer to [12, 13, 14, 24, 39, 50] for the former case. (Note that these works require summable conditions on the sequence of accurate constants represented in terms of either absolute or relative errors, demonstrating that the sequence of accurate constants to solve the subproblems should be converged to 0 and specified in advance.)

There are different choices for solving the LASSO model (1.2); the ADMM (1.4) is a competitive one; see [5]. Let us delineate the detail of the application of the ADMM (1.4) to the LASSO model (1.2) and use this application to properly illustrate the proposed idea to deal with the mathematical issues arising from implementation of the ADMM for some big-data scenarios. The first step is introducing an auxiliary

variable $y \in \mathbb{R}^n$, in which the LASSO model (1.2) is explicitly rewritten in the form of (1.1):

$$\min_{x\in\mathbb{R}^n,\ y\in\mathbb{R}^n} \quad \frac{1}{2}\|Qx-q\|_2^2+\tau\|y\|_1 \qquad \text{s.t.} \qquad x=y, \tag{1.6}$$

and the ADMM (1.4) can be accordingly specified as

$$\begin{cases} x^{k+1}=\arg\min\limits_{x\in\mathbb{R}^n}\left\{\frac{1}{2}\|Qx-q\|_2^2+\frac{\beta}{2}\left\|x-y^k-\frac{\lambda^k}{\beta}\right\|_2^2\right\},\\ y^{k+1}=\arg\min\limits_{y\in\mathbb{R}^n}\left\{\tau\|y\|_1+\frac{\beta}{2}\left\|y-x^{k+1}+\frac{\lambda^k}{\beta}\right\|_2^2\right\},\\ \lambda^{k+1}=\lambda^k-\beta\left(x^{k+1}-y^{k+1}\right). \end{cases} \tag{1.7}$$

Methodologically, the implementation of (1.7) seems to be easy. The x-subproblem in (1.7) can be expressed as

$$\left(Q^\top Q+\beta I\right)x=Q^\top q+\beta\left(y^k+\frac{\lambda^k}{\beta}\right), \tag{1.8}$$

and its solution is analytically given by

$$x^{k+1}=\left(Q^\top Q+\beta I\right)^{-1}\left[Q^\top q+\beta\left(y^k+\frac{\lambda^k}{\beta}\right)\right] \tag{1.9}$$

while the solution of the y-subproblem is given by

$$y^{k+1}=\mathcal{S}\left(x^{k+1}-\frac{\lambda^k}{\beta},\frac{\tau}{\beta}\right), \tag{1.10}$$

where $\mathcal{S}(x,a)$ denotes the shrinkage operator (see [43]), i.e.,

$$\mathcal{S}(x,a):=x-\max\{\min\{x,a\},-a\}.$$

Despite the closed-form solution given in (1.9), the big-data scenario is taken into account for the LASSO model (1.2) with high-dimensional variables; i.e., both p and n could be huge. For such a big-data scenario, it becomes impossible or extremely expensive to exactly solve the linear system (1.8) by either direct or iterative methods. Indeed, even though standard numerical linear algebra solvers (e.g., conjugate gradient (CG) and preconditioned conjugate gradient (PCG) algorithms) guarantee finding the accurate solution of (1.8) after n iterations, it should not be attempted to execute all n iterations when n is huge. We thus just need to iteratively solve the system (1.8) by less than n iterations. Some questions immediately arise: how accurate should an iterate be if a specific solver is applied to solve the linear system (1.8)? and for implementation purposes, how many iterations should be executed to solve the linear system (1.8)? For example, according to the authors' website,[1] (see also [5, section 8.2.1] it is suggested to solve the linear system (1.8) by using the least squares with QR factorization (LSQR) algorithm (see [40]) with the fixed accuracy of 10^{-6}. It becomes interesting to ask if the convergence of ADMM can still be guaranteed when all the x-subproblems are inaccurately solved with a fixed accuracy. On the other hand, it seems to be puzzling to fix which level of accuracy in advance, because neither

[1] http://stanford.edu/~boyd/papers/admm/lasso/lasso.html.

very high nor very low accuracy is appropriate for generating satisfactory numerical performance. Indeed, if the accuracy is fixed, then it requires us to tune the level of accuracy in advance, and the "optimal" level of accuracy may vary among different specific applications of the generic model (1.1). Also, there is no clear justification to testify pursuing an exact solution for the x-subproblem, or an approximate solution with high-accuracy, is necessary at the earlier stage of the iteration. All these questions urge us to find an inexactness criterion that can dynamically and automatically adjust the accuracy so as to solve the linear system (1.8) inexactly for the big-data scenario of the LASSO model (1.2) and to rigorously prove the convergence of the inexact version of ADMM with this inexact criterion.

The main findings achieved in this study are as follows: (1) proposing an adjustable inexactness criterion for inexactly solving (1.4a), thereby proposing an inexact version of the ADMM for the model (1.1); (2) rigorously proving the convergence of the inexact version of ADMM, as well as estimating its convergence rate in terms of the iteration complexity; and (3) specifying the safe-guard iteration numbers when numerical linear algebra solvers are applied to solve (1.4a). These results theoretically guarantee the convergence and practically ensure the efficiency required for inexact implementation of the ADMM with an implementable inexactness criterion. Based on numerical results, when a standard numerical linear algebra solver is chosen for solving (1.4a), only a few CG steps are typically enough to meet the defined inexactness criterion. In other words, it is neither necessary nor possible to more accurately solve the involved linear system. This property significantly reduces computations to solve (1.4a) and provides efficient applications of the ADMM to big-data scenarios of the model (1.1).

The resulting analysis is indeed complicated; we thus only consider the simple case where the x-subproblem in (1.4) is inaccurately solved and the second one is assumed involving a closed-form solution. This essentially demonstrates that we may assume that the function $g(y)$ in the generic model (1.1) is relatively easy (e.g., the ℓ_1 penalty or some other popular penalty terms), so that the subproblem (1.4b) can be easily solved. This simplification realizes a clear execution of the proposed idea and analysis. As mentioned earlier, discussing the implementation of the ADMM (1.4) for the big-data scenario of the LASSO model (1.2) is still our emphasis.

The remaining part of this paper is organized as follows. The inexactness criterion and corresponding inexact version of ADMM are presented in section 2. Then, we prove the convergence of the inexact version of ADMM in section 3 and establish its worst-case convergence rate in section 4. In section 5, we discuss how to specify the safe-guard iteration numbers for the internally nested iterations when some standard numerical linear algebra solvers are used to solve (1.4a). In section 6, we test some big datasets of the LASSO model (1.2) and report the numerical results. Finally, some conclusions are drawn in section 7.

2. Algorithm. In this section we specify an inexactness criterion for (1.4a) and present an inexact version of ADMM for the model (1.1) with special consideration of the big-data scenario where p and n are both assumed to be huge.

Let us first check the x-subproblem (1.4a). Obviously, it can be rewritten as

$$\begin{aligned} x^{k+1} &= \arg\min_{x\in\mathbb{R}^n} \mathcal{L}_\beta\left(x, y^k, \lambda^k\right) \\ &= \arg\min_{x\in\mathbb{R}^n} \left\{ \frac{1}{2}\left\|Qx-q\right\|_2^2 + \frac{\beta}{2}\left\| Ax + By^k - b - \frac{\lambda^k}{\beta}\right\|_2^2 \right\}, \end{aligned} \tag{2.1}$$

and thus x^{k+1} is given by the system of linear equations

$$Hx^{k+1} = h^k \tag{2.2}$$

with

$$H := \left(Q^\top Q + \beta A^\top A\right) \quad \text{and} \quad h^k := Q^\top q - \beta A^\top \left(By^k - b - \frac{\lambda^k}{\beta}\right). \tag{2.3}$$

More explicitly, we have

$$x^{k+1} = H^{-1}h^k = \left(Q^\top Q + \beta A^\top A\right)^{-1} \left(Q^\top q - \beta A^\top \left(By^k - b - \frac{\lambda^k}{\beta}\right)\right). \tag{2.4}$$

Recall that our interest is the big-data scenario where n is huge. It is thus not preferable to directly solve the linear system (2.2) with the matrix H in dimension of $n \times n$.

Alternatively, as mentioned in [5, section 4.2.4], we can calculate x^{k+1} via the following process:

$$\hat{H}\eta^{k+1} = Q\left(\beta A^\top A\right)^{-1} h^k, \tag{2.5a}$$

$$x^{k+1} = \left(\beta A^\top A\right)^{-1} \left(h^k - Q^\top \eta^{k+1}\right) \tag{2.5b}$$

with

$$\hat{H} := \left(Q\left(\beta A^\top A\right)^{-1} Q^\top + I\right). \tag{2.6}$$

To see the reason, we have

$$\begin{aligned} x^{k+1} &= \left(\beta A^\top A\right)^{-1} \left(h^k - Q^\top \hat{H}^{-1} Q \left(\beta A^\top A\right)^{-1} h^k\right) \\ &= \left[\left(\beta A^\top A\right)^{-1} - \left(\beta A^\top A\right)^{-1} Q^\top (Q\left(\beta A^\top A\right)^{-1} Q^\top + I)^{-1} Q \left(\beta A^\top A\right)^{-1}\right] h^k \\ &= \left(Q^\top Q + \beta A^\top A\right)^{-1} h^k = H^{-1}h^k, \end{aligned}$$

where the last equality comes from the Woodbury matrix identity; see [47]. For the case with $p \ll n$, the dimension of $\hat{H} \in \mathbb{R}^{p\times p}$ in (2.6) is much smaller than $H \in \mathbb{R}^{n\times n}$ in (2.2). Hence, we prefer the procedure (2.5) to (2.4) for solving the linear system (2.2) when $p \ll n$.

To perform (2.5), we need to compute the inverse of the matrix $A^\top A \in \mathbb{R}^{n\times n}$. For many applications, the matrix A may be simple, such as an identity matrix, while the matrix Q in the objective function of the model (1.1) may be general enough to lack any special structure. Thus it is generally easier to compute $(A^\top A)^{-1}$ than H^{-1} despite the same dimensionality. The LASSO model (1.2) with A an identity matrix is such an application. One more example is the consensus problem in [5, section 7] with wide applications in wireless communication area, which can also be reformulated as (1.1) with A an identity matrix.

For big-data scenarios of the model (1.1), even p could still be huge, and hence it is not preferable to solve (2.5a) exactly by a direct method or inexactly up to a very high accuracy by an iterative method. Thus, we need to further consider how to solve the linear system (2.5a) inexactly. For this purpose, we need to specify an inexactness criterion. Obviously, the residual of the linear system (2.5a) is

$$e_k(\eta) := Q\left(\beta A^\top A\right)^{-1} h^k - \hat{H}\eta. \tag{2.7}$$

At the $(k+1)$th iteration, we suggest finding an approximate solution of the linear system (2.5a), η^{k+1}, such that

$$\left\|e_k\left(\eta^{k+1}\right)\right\|_2 \leq \sigma \left\|e_k\left(\eta^k\right)\right\|_2, \tag{2.8}$$

where σ is an arbitrary constant satisfying

$$0 < \sigma < \frac{\sqrt{2\beta}}{\sqrt{2\beta} + \left\|Q\left(A^\top A\right)^{-1} A^\top\right\|_2} \in (0,1). \tag{2.9}$$

The parameter σ measures the relative error of $\|e_k(\eta^k)\|$, and it plays the role of controlling the accuracy of solving the linear system (2.5a) inexactly. It is worth noting that σ is fixed as a constant in (2.8); hence the inexactness criterion (2.8) significantly differs from the ones mentioned in the literature which need a sequence of accuracy constants and require summable conditions on the sequence. Moreover, specifying the sequence of accuracy constants a priori (which must be done manually) is very challenging, and inappropriate values may easily deteriorate the numerical performance of the inexact versions of the ADMM mentioned in the literature. The inexactness criterion (2.8), however, becomes fully automatic for numerical implementation by just choosing a constant according to (2.9). Obviously, larger values of σ imply looser criteria and thus less computation for solving the linear system (2.5a) inexactly; indeed in our numerical experiments, we choose values very close to the upper bound given in (2.9).

Standard numerical linear algebra solvers such as the Jacobi, Gauss–Seidel, successive over-relaxation (SOR), CG, and PCG methods can all be used to achieve the inexactness criterion (2.8) for the internal iterations (i.e., step 3 in the algorithm below). Accordingly, an inexactness version of ADMM (1.4) for a big-data scenario of the model (1.1) with internally nested iterations for solving the x-subproblem at each iteration in the sense of the inexactness criterion (2.8) can be presented as below in Algorithm 1.

Algorithm 1 Inexact ADMM (**InADMM**) for big-data scenarios of (1.1)

1: **Require** $\left(\eta^0, y^0, \lambda^0\right) \in \mathbb{R}^p \times \mathbb{R}^m \times \mathbb{R}^\ell$, $\beta > 0$, σ satisfying (2.9);
2: $k = 0$;
3: **while** not converged **do**
4: Find η^{k+1} such that (2.8) is satisfied where h^k and $e_k(\eta)$ are defined in (2.3) and (2.7);
5: $x^{k+1} := \left(\beta A^\top A\right)^{-1}\left(h^k - Q^\top \eta^{k+1}\right)$;
6: $y^{k+1} := \arg\min\limits_{y\in\mathbb{R}^m} \left\{g(y) - \left(\lambda^k\right)^\top \left(Ax^{k+1} + By - b\right) + \frac{\beta}{2}\left\|Ax^{k+1} + By - b\right\|_2^2\right\}$;
7: $\lambda^{k+1} := \lambda^k - \beta\left(Ax^{k+1} + By^{k+1} - b\right)$.
8: $k \leftarrow k+1$;
9: **end while**

3. Convergence analysis. In this section, we prove the convergence of the proposed inexact ADMM, InADMM (1). As mentioned, though the convergence of the original ADMM in the ideal exact form of (1.4) has been well studied in the literature, these existing results cannot be directly extended to the proposed InADMM (1) because of the specific inexact steps for calculating x^{k+1}. We hence present the complete analysis for the convergence of InADMM (1). We start from some known preliminaries.

3.1. Preliminaries. Let $\Omega := \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^\ell$. To present our analysis in more compact notation, we denote vectors $w \in \Omega$ and $u \in \mathbb{R}^m \times \mathbb{R}^\ell$, the matrix $\mathcal{M} \in \mathbb{R}^{(m+\ell)\times(m+\ell)}$, and the function $F(w)$ as the following:

(3.1)
$$w = \begin{pmatrix} x \\ y \\ \lambda \end{pmatrix},\ u = \begin{pmatrix} y \\ \lambda \end{pmatrix},\ \mathcal{M} = \begin{pmatrix} \beta B^\top B & 0 \\ 0 & \frac{1}{\beta} I_{\ell\times\ell} \end{pmatrix},\ F(w) = \begin{pmatrix} Q^\top (Qx - q) - A^\top \lambda \\ -B^\top \lambda \\ Ax + By - b \end{pmatrix}.$$

Note that the matrix $\mathcal{M}$ is not necessarily positive definite because the matrix B is not assumed to be of full column rank in (1.1). In what follows, the semi-norm $\|u\|_{\mathcal{M}}$ denotes the number $\sqrt{u^\top \mathcal{M} u}$.

As initiated in [28], the problem (1.1) can be stated as the variational inequality of finding $w^\star = (x^\star, y^\star, \lambda^\star) \in \Omega$ such that

$$\mathrm{VI}(\Omega, F) : g(y) - g(y^\star) + (w - w^\star)^\top F(w^\star) \geq 0\ \forall w \in \Omega. \tag{3.2}$$

The solution set of the variational inequality (3.2) is denoted by $\Omega^\star$. As analyzed in [15, 28], the solution set $\Omega^\star$ has the characterization shown in the following theorem. We skip the proof, which can be found in [28] as well.

THEOREM 3.1. *Let $\Omega^\star$ be the solution set of the variational inequality* (3.2)*. Then, we have*

$$\Omega^\star = \bigcap_{w\in\Omega} \left\{ \hat{w} \in \Omega : g(y) - g(\hat{y}) + (w - \hat{w})^\top F(w) \geq 0 \right\}. \tag{3.3}$$

3.2. Optimality. To derive the convergence of the proposed InADMM (1), it is necessary to discern the difference of its iterate from a solution point, or the optimality of each iterate. More specifically, we need to quantify how accurately the x-subproblem in the ideal (exact) form of the ADMM (1.4) is approached by the inexact step (i.e., step 3) of the proposed InADMM (1).

For this purpose, with the x^{k+1} generated by the InADMM (1), we have

$$\begin{aligned} \nabla_x \mathcal{L}_\beta \left(x^{k+1}, y^k, \lambda^k\right) &= \left(Q^\top Q + \beta A^\top A\right) x^{k+1} - h^k \\ &= \left(Q^\top Q + \beta A^\top A\right) \left(\beta A^\top A\right)^{-1} \left(h^k - Q^\top \eta^{k+1}\right) - h^k \\ &= Q^\top \left[Q \left(\beta A^\top A\right)^{-1} h^k - \left(Q \left(\beta A^\top A\right)^{-1} Q^\top + I \right) \eta^{k+1} \right] \\ &= Q^\top e_k \left(\eta^{k+1}\right). \end{aligned}$$

For the exact version of ADMM (1.4) where x is required to be solved exactly, it obviously means $\nabla_x \mathcal{L}_\beta \left(x, y^k, \lambda^k\right) = 0$. Therefore, we can quantitatively regard $\|Q^\top e_k \left(\eta^{k+1}\right)\|_2$ as the difference of the inexact solution of the x-subproblem generated by the InADMM (1) from the exact solution generated by the exact version of the ADMM (1.4), in the sense of the residual of the partial gradient of the augmented Lagrangian function. Recall that the y- and λ-subproblems are assumed to be solved exactly in the InADMM (1). Hence, for the iterate $w^{k+1} = (x^{k+1}, y^{k+1}, \lambda^{k+1})$ generated by the InADMM (1), its optimality can be expressed as the following:

(3.4)
$$\begin{cases} \nabla_x \mathcal{L}_\beta \left(x^{k+1}, y^k, \lambda^k\right) = Q^\top e_k(\eta^{k+1}), \\ g(y) - g\left(y^{k+1}\right) + \left(y - y^{k+1}\right)^\top \left(-B^\top \lambda^k + \beta B^\top \left(Ax^{k+1} + By^{k+1} - b\right)\right) \geq 0\ \forall y \in \mathbb{R}^m, \\ \lambda^{k+1} = \lambda^k - \beta \left(Ax^{k+1} + By^{k+1} - b\right). \end{cases}$$

We reiterate that if $\|e_k(\eta^{k+1})\|_2 = 0$, then the InADMM (1) reduces to the exact version of ADMM (1.4) with known convergence. But we focus on the big-data scenarios where it is not possible to pursue $\|e_k(\eta^{k+1})\|_2 = 0$, and only an approximate solution subject to the inexactness criterion (2.8) is realized by internally nested iterations.

To prove the convergence of the sequence generated by the InADMM (1) whose optimality is given in (3.4), it is also crucial to analyze how the residual $\|e_k(\eta^{k+1})\|_2$ evolves according to the iterations. Recall that the y- and λ-subproblems are assumed to be solved exactly in the InADMM (1). We thus know that

$$B^\top \lambda^{k-1} \in \partial g(y^{k-1}), \quad B^\top \lambda^k \in \partial g(y^k),$$

and thus

$$\left(y^{k-1} - y^k\right)^\top B^\top \left(\lambda^{k-1} - \lambda^k\right) \geq 0. \tag{3.5}$$

Hence, it is easily derived that

$$\begin{aligned} \left\|\lambda^{k-1} - \lambda^k - \beta B\left(y^{k-1} - y^k\right)\right\|_2^2 &\leq \left\|\lambda^{k-1} - \lambda^k\right\|_2^2 + \beta^2 \left\|B\left(y^{k-1} - y^k\right)\right\|_2^2 \\ &= \beta \left\|u^{k-1} - u^k\right\|_{\mathcal{M}}^2, \end{aligned} \tag{3.6}$$

where $\mathcal{M}$ is defined in (3.1). As a result, we have

$$\begin{aligned} &\left\|e_k(\eta^{k+1})\right\|_2 \\ &\leq \sigma \left\|e_k(\eta^k)\right\|_2 = \sigma \left\|Q\left(\beta A^\top A\right)^{-1} h^k - \hat{H}\eta^k\right\|_2 \\ &\leq \sigma \left\|Q\left(\beta A^\top A\right)^{-1} h^{k-1} - \hat{H}\eta^k\right\|_2 + \sigma \left\|Q\left(\beta A^\top A\right)^{-1}\left(h^k - h^{k-1}\right)\right\|_2 \\ &\overset{h^{k-1}, h^k}{\leq} \sigma \left\|e_{k-1}(\eta^k)\right\|_2 + \sigma \left\|Q\left(\beta A^\top A\right)^{-1} A^\top\right\|_2 \cdot \left\|\beta B\left(y^{k-1} - y^k\right) - \left(\lambda^{k-1} - \lambda^k\right)\right\|_2 \\ &\overset{(3.6)}{\leq} \sigma \left\|e_{k-1}(\eta^k)\right\|_2 + \frac{\sigma}{\sqrt{\beta}} \left\|Q(A^\top A)^{-1} A^\top\right\|_2 \cdot \left\|u^{k-1} - u^k\right\|_{\mathcal{M}}. \end{aligned} \tag{3.7}$$

Therefore, for two consecutive iterates generated by the InADMM (1), it follows from (3.7) that their residuals arising from solving the x-subproblems inexactly are related precisely by

$$\left\|e_k(\eta^{k+1})\right\|_2 \leq \sigma \left\|e_{k-1}(\eta^k)\right\|_2 + \sigma\gamma \left\|u^{k-1} - u^k\right\|_{\mathcal{M}} \tag{3.8}$$

with

$$\gamma = \frac{\left\|Q\left(A^\top A\right)^{-1} A^\top\right\|_2}{\sqrt{\beta}}. \tag{3.9}$$

This relationship will be often used in the coming analysis.

Moreover, recall that the parameter σ controlling the accuracy in (2.8) is restricted by the condition (2.9). Hence, it follows from the definition of γ in (3.9) that

$$0 < \frac{\gamma^2\sigma^2}{2(1-\sigma)^2} = \left(\frac{\sigma}{2(1-\sigma)}\right)\left(\frac{\gamma^2\sigma}{1-\sigma}\right) < 1,$$

and obviously there exists a $\mu > 0$ such that

$$0 < \frac{\mu}{2}\frac{\sigma}{1-\sigma} < 1 \quad \text{and} \quad 0 < \frac{\gamma^2}{\mu}\frac{\sigma}{1-\sigma} < 1. \tag{3.10}$$

3.3. Convergence. Now we prove the convergence of the sequence generated by the InADMM (1). To simplify the notation, let us introduce an auxiliary variable $\bar{w}^k$ as

$$\bar{w}^k = \begin{pmatrix} \bar{x}^k \\ \bar{y}^k \\ \bar{\lambda}^k \end{pmatrix} = \begin{pmatrix} x^{k+1} \\ y^{k+1} \\ \lambda^k - \beta(Ax^{k+1} + By^k - b) \end{pmatrix}. \tag{3.11}$$

This notation is only for the simplification of notation in our analysis; it is not required to be computed for implementing the InADMM (1).

Recall the variational inequality reformulation (3.2) of the model (1.1). First of all, we analyze how different the point $\bar{w}^k$ defined in (3.11) is from a solution point of (3.2) and how to quantify this difference by iterates generated by the InADMM (1).

PROPOSITION 3.2. *Let $\{w^k\}$ be the sequence generated by the InADMM* (1)*; let $\bar{w}^k$ be defined in* (3.11) *and $\mathcal{M}$ in* (3.1)*. Then, for all $w \in \Omega$, it holds that*

$$\begin{aligned} &g\left(\bar{y}^k\right) - g(y) + \left(\bar{w}^k - w\right)^\top F\left(\bar{w}^k\right) \\ &\quad \le \frac{1}{2}\left(\left\|u^k - u\right\|_{\mathcal{M}}^2 - \left\|u^{k+1} - u\right\|_{\mathcal{M}}^2 - \left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2\right) \\ &\qquad + \left(x^{k+1} - x\right)^\top \nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right). \end{aligned} \tag{3.12}$$

Proof. First we rewrite $\nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right)$ as

$$\begin{aligned} \nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) &= Q^\top\left(Qx^{k+1} - q\right) - A^\top\left(\lambda^k - \beta\left(Ax^{k+1} + By^k - b\right)\right) \\ &= Q^\top\left(Qx^{k+1} - q\right) - A^\top\bar{\lambda}^k. \end{aligned}$$

Then, combining it with the optimality condition with respect to y, for all $y \in \mathbb{R}^m$, we have

$$g(y) - g\left(y^{k+1}\right) + \left(y - y^{k+1}\right)\left[-B^\top\lambda^k + \beta B^\top\left(Ax^{k+1} + By^{k+1} - b\right)\right] \ge 0, \tag{3.13}$$

with which we obtain, for all $w \in \Omega$, that

$$\begin{aligned} &g(y) - g\left(\bar{y}^k\right) + \left(w - \bar{w}^k\right)^\top F\left(\bar{w}^k\right) \\ &= \left(x - x^{k+1}\right)^\top\left(Q^\top\left(Qx^{k+1} - q\right) - A^\top\bar{\lambda}^k\right) + g(y) - g\left(y^{k+1}\right) + \left(y - y^{k+1}\right)^\top\left(-B^\top\bar{\lambda}^k\right) \\ &\quad + \left(\lambda - \bar{\lambda}^k\right)\left(Ax^{k+1} + By^{k+1} - b\right) \\ &= \left(x - x^{k+1}\right)^\top\nabla_x\mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) + \left(y - y^{k+1}\right)\left[-B^\top\lambda^k + \beta B^\top\left(Ax^{k+1} + By^{k+1} - b\right)\right] \\ &\quad + g(y) - g\left(y^{k+1}\right) + \beta\left(y - y^{k+1}\right)B^\top B\left(y^k - y^{k+1}\right) + \frac{1}{\beta}\left(\lambda - \bar{\lambda}^k\right)^\top\left(\lambda^k - \lambda^{k+1}\right) \\ &\overset{(3.13)}{\ge} \left(x - x^{k+1}\right)^\top\nabla_x\mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) + \beta\left(y - y^{k+1}\right)B^\top B\left(y^k - y^{k+1}\right) \\ &\quad + \frac{1}{\beta}\left(\lambda - \lambda^{k+1}\right)^\top\left(\lambda^k - \lambda^{k+1}\right) + \frac{1}{\beta}\left(\lambda^{k+1} - \bar{\lambda}^k\right)^\top\left(\lambda^k - \lambda^{k+1}\right). \end{aligned}$$

Moreover, notice the elementary equation

$$(a-c)^\top (b-c) = \frac{1}{2}\left(\|a-c\|_2^2 - \|a-b\|_2^2 + \|b-c\|_2^2\right). \tag{3.14}$$

Thus, for all $w \in \Omega$, we have

$$\begin{aligned}
&g(y) - g\left(\bar{y}^k\right) + \left(w - \bar{w}^k\right)^\top F\left(\bar{w}^k\right) \\
&\overset{(3.14)}{\geq} \left(x - x^{k+1}\right)^\top \nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) \\
&\quad + \frac{\beta}{2}\left(\left\|y - y^{k+1}\right\|_{B^\top B}^2 - \left\|y - y^k\right\|_{B^\top B}^2 + \left\|y^k - y^{k+1}\right\|_{B^\top B}^2\right) \\
&\quad + \frac{1}{2\beta}\left(\left\|\lambda - \lambda^{k+1}\right\|_2^2 - \left\|\lambda - \lambda^k\right\|_2^2 + \left\|\lambda^k - \lambda^{k+1}\right\|_2^2\right) + \left(y^k - y^{k+1}\right)^\top B^\top\left(\lambda^k - \lambda^{k+1}\right) \\
&\overset{(3.5)}{\geq} \left(x - x^{k+1}\right)^\top \nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) + \frac{\beta}{2}\left(\left\|B\left(y - y^{k+1}\right)\right\|_2^2 - \left\|B\left(y - y^k\right)\right\|_2^2\right) \\
&\quad + \frac{1}{2\beta}\left(\left\|\lambda - \lambda^{k+1}\right\|_2^2 - \left\|\lambda - \lambda^k\right\|_2^2\right) + \frac{1}{2}\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2.
\end{aligned} \tag{3.15}$$

∎

Using the notation of $\mathcal{M}$ in (3.1), (3.15) can be rewritten as (3.12), and the proof is complete.

The difference between the inequality (3.12) and the variational inequality reformulation (3.2) reflects the difference of the point $\bar{w}^k$ from a solution point $w^\star$. For the right-hand side of (3.12), the first three terms are quadratic and they are easy to manipulate over different indicators by algebraic operations, but it is not that explicit how the last crossing term can be controlled towards the eventual goal of proving the convergence of the sequence $\{w^k\}$. We thus look into this term particularly and show that the sum of these crossing terms over K iterations can be bounded by some quadratic terms as well. This result is summarized in the following proposition.

PROPOSITION 3.3. *Let $\{w^k\}$ be the sequence generated by InADMM* (1)*. For all $x \in \mathbb{R}^n$, $K > 1$, and μ satisfying* (3.10)*, it holds that*

$$\begin{aligned}
&\sum_{k=1}^{K}\left(x^{k+1} - x\right)^\top \nabla_x \mathcal{L}_\beta\left(x^{k+1}, y^k, \lambda^k\right) \\
&\quad \leq \frac{\sigma}{1-\sigma}\left\{\frac{\mu}{2}\sum_{k=1}^{K}\left\|Q\left(x^{k+1} - x\right)\right\|_2^2\right. \\
&\qquad \left. + \frac{1}{2\mu}\left[\sum_{k=1}^{K-1}\gamma^2\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2 + \left(\left\|e_0\left(\eta^1\right)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2\right]\right\}.
\end{aligned} \tag{3.16}$$

Proof. Recall the result (3.8). By mathematical induction, for all $k \geq 1$, we have

$$\left\|e_k(\eta^{k+1})\right\|_2 \leq \sum_{i=0}^{k-1}\sigma^{k-i}\gamma\left\|u^i - u^{i+1}\right\|_{\mathcal{M}} + \sigma^k\left\|e_0(\eta^1)\right\|_2. \tag{3.17}$$

Then, combining it with the optimality condition with respect to the x-subproblem at each iteration, we have that, for all $x \in \mathbb{R}^n$ and $K > 1$, the following inequalities hold:

$$\begin{aligned}
&\sum_{k=1}^{K}\left(x^{k+1}-x\right)^{\top}\nabla_x\mathcal{L}_\beta\left(x^{k+1},y^k,\lambda^k\right)=\sum_{k=1}^{K}\left(x^{k+1}-x\right)^{\top}Q^{\top}e_k(\eta^{k+1})\\
\leq\ &\sum_{k=1}^{K}\left\|Q\left(x^{k+1}-x\right)\right\|_2\cdot\left\|e_k(\eta^{k+1})\right\|_2\\
\overset{(3.17)}{\leq}\ &\sum_{k=1}^{K}\sum_{i=0}^{k-1}\sigma^{k-i}\gamma\left\|Q\left(x^{k+1}-x\right)\right\|_2\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sum_{k=1}^{K}\sigma^k\left\|Q\left(x^{k+1}-x\right)\right\|_2\cdot\left\|e_0(\eta^1)\right\|_2\\
=\ &\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sigma^{k-i}\gamma\left\|Q\left(x^{k+1}-x\right)\right\|_2\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sum_{k=1}^{K}\sigma^k\left\|Q\left(x^{k+1}-x\right)\right\|_2\left\|e_0(\eta^1)\right\|_2\\
&\quad+\sum_{k=1}^{K}\sigma^k\gamma\left\|Q\left(x^{k+1}-x\right)\right\|_2\left\|u^0-u^1\right\|_{\mathcal{M}}\\
\leq\ &\frac{\mu}{2}\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sigma^{k-i}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2+\frac{1}{2\mu}\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sigma^{k-i}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2\\
&\quad+\frac{\mu}{2}\sum_{k=1}^{K}\sigma^k\left\|Q\left(x^{k+1}-x\right)\right\|_2^2+\frac{1}{2\mu}\sum_{k=1}^{K}\sigma^k\left[\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right]^2\\
=\ &\frac{\mu}{2}\sum_{k=1}^{K}\sum_{i=0}^{k-1}\sigma^{k-i}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2+\frac{1}{2\mu}\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sigma^{k-i}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2\\
&\quad+\frac{1}{2\mu}\sum_{k=1}^{K}\sigma^k\left[\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right]^2.
\end{aligned}$$

Furthermore, for all $x \in \mathbb{R}^n$ and $K > 1$, we have

$$\sum_{k=1}^{K}\sum_{i=0}^{k-1}\sigma^{k-i}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2\overset{\sigma\in(0,1)}{=}\sum_{k=1}^{K}\frac{\sigma-\sigma^{k+1}}{1-\sigma}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2 \tag{3.18}$$

and

$$\begin{aligned}
\sum_{k=2}^{K}\sum_{i=1}^{k-1}\sigma^{k-i}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2&=\sum_{i=1}^{K-1}\sum_{k=i+1}^{K}\sigma^{k-i}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2\\
&\overset{\sigma\in(0,1)}{=}\sum_{i=1}^{K-1}\frac{\sigma-\sigma^{K-i+1}}{1-\sigma}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2.
\end{aligned} \tag{3.19}$$

Combining the above equalities and inequalities, we obtain

$$\begin{aligned}
&\sum_{k=1}^{K}\left(x^{k+1}-x\right)^{\top}\nabla_x\mathcal{L}_\beta\left(x^{k+1},y^k,\lambda^k\right)\\
\overset{(3.18)(3.19)}{=}\ &\frac{\mu}{2}\sum_{k=1}^{K}\frac{\sigma-\sigma^{k+1}}{1-\sigma}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2+\frac{1}{2\mu}\sum_{i=1}^{K-1}\frac{\sigma-\sigma^{K-i+1}}{1-\sigma}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2\\
&\quad+\frac{1}{2\mu}\frac{\sigma-\sigma^{K+1}}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2
\end{aligned}$$

$$\leq \frac{\mu}{2}\sum_{k=1}^{K}\frac{\sigma}{1-\sigma}\left\|Q\left(x^{k+1}-x\right)\right\|_2^2+\frac{1}{2\mu}\sum_{i=1}^{K-1}\frac{\sigma}{1-\sigma}\gamma^2\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2$$
$$+\frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2,$$

which implies the conclusion (3.16). The proof is complete. ∎

The convergence of the proposed InADMM (1) is established in the following theorem.

THEOREM 3.4. *Let $\{w^k\}$ be the sequence generated by the InADMM* (1)*. Then, we have the following assertions:*

(1) $\left\|e_k(\eta^{k+1})\right\|_2 \overset{k\to\infty}{\longrightarrow} 0, \quad \left\|B\left(y^k-y^{k+1}\right)\right\|_2 \overset{k\to\infty}{\longrightarrow} 0$;

(2) $\left\|Ax^{k+1}+By^{k+1}-b\right\|_2 \overset{k\to\infty}{\longrightarrow} 0$; $f\left(x^k\right)+g\left(y^k\right) \overset{k\to\infty}{\longrightarrow} f\left(x^\star\right)+g\left(y^\star\right)$ *for any given* $w^\star\in\Omega^\star$.

Proof. First, recall the definition of $F(w)$ in (3.1). We have that

(3.20)
$$\left(w-\bar{w}^k\right)^\top\left(F(w)-F\left(\bar{w}^k\right)\right)=\left(x-x^{k+1}\right)^\top Q^\top\left(Qx-Qx^{k+1}\right)=\left\|Q\left(x-x^{k+1}\right)\right\|_2^2.$$

Then, using the results (3.12) and (3.16) established in Propositions 3.2 and 3.3, respectively, we obtain

$$\begin{aligned}
&\sum_{k=1}^{K}\left\{g\left(\bar{y}^k\right)-g(y)+\left(\bar{w}^k-w\right)^\top F(w)\right\}\\
=\;&\sum_{k=1}^{K}\left\{g\left(\bar{y}^k\right)-g(y)+\left(\bar{w}^k-w\right)^\top F\left(\bar{w}^k\right)+\left(\bar{w}^k-w\right)^\top\left[F(w)-F\left(\bar{w}^k\right)\right]\right\}\\
\overset{(3.12)}{\leq}\;&\frac{1}{2}\left(\left\|u^1-u\right\|_{\mathcal{M}}^2-\left\|u^{K+1}-u\right\|_{\mathcal{M}}^2\right)+\sum_{k=1}^{K}\left\{\left(x^{k+1}-x\right)^\top\nabla_x\mathcal{L}_\beta\left(x^{k+1},y^k,\lambda^k\right)\right.\\
&\left.-\left(w-\bar{w}^k\right)^\top\left[F(w)-F\left(\bar{w}^k\right)\right]\right\}-\sum_{k=1}^{K}\frac{1}{2}\left\|u^k-u^{k+1}\right\|_{\mathcal{M}}^2\\
\overset{(3.16)(3.20)}{\leq}\;&\frac{1}{2}\left(\left\|u^1-u\right\|_{\mathcal{M}}^2-\left\|u^{K+1}-u\right\|_{\mathcal{M}}^2\right)+\sum_{k=1}^{K}\left(\frac{\mu}{2}\frac{\sigma}{1-\sigma}-1\right)\left\|Q\left(x^{k+1}-x\right)\right\|_2^2\\
&+\sum_{k=1}^{K-1}\frac{1}{2}\left(\frac{\sigma}{1-\sigma}\frac{\gamma^2}{\mu}-1\right)\left\|u^k-u^{k+1}\right\|_{\mathcal{M}}^2-\frac{1}{2}\left\|u^K-u^{K+1}\right\|_{\mathcal{M}}^2\\
&+\frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2. \qquad (3.21)
\end{aligned}$$

For any given $w^\star\in\Omega^\star$, we have

$$g\left(\bar{y}^k\right)-g\left(y^\star\right)+\left(\bar{w}^k-w^\star\right)^\top F\left(w^\star\right)\geq 0\;\forall k.$$

Setting $w=w^\star$ in (3.21), together with the above property, for any $K>1$, we have

$$\begin{aligned}
&\sum_{k=1}^{K}\left(1-\frac{\mu}{2}\frac{\sigma}{1-\sigma}\right)\left\|Q\left(x^{k+1}-x^\star\right)\right\|_2^2+\sum_{k=1}^{K-1}\left(\frac{1}{2}-\frac{\gamma^2}{2\mu}\frac{\sigma}{1-\sigma}\right)\left\|u^k-u^{k+1}\right\|_{\mathcal{M}}^2\\
&\leq\frac{1}{2}\left\|u^1-u^\star\right\|_{\mathcal{M}}^2+\frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2-\frac{1}{2}\left\|u^{K+1}-u^\star\right\|_{\mathcal{M}}^2\\
&-\frac{1}{2}\left\|u^K-u^{K+1}\right\|_{\mathcal{M}}^2. \qquad (3.22)
\end{aligned}$$

Using (3.10), we conclude that

(3.23)
$$\left\|Q\left(x^{k+1}-x^{\star}\right)\right\|_2 \overset{k\to\infty}{\longrightarrow} 0 \quad \left\|u^k-u^{k+1}\right\|_{\mathcal{M}} \overset{k\to\infty}{\longrightarrow} 0, \quad \text{and} \quad \left\|u^{K+1}-u^{\star}\right\|_{\mathcal{M}}<\infty.$$

Furthermore, for any $\varepsilon>0$, there exists a k_0 such that for all $k\geq k_0$, we have

$$\left\|u^k-u^{k+1}\right\|_{\mathcal{M}}\leq\varepsilon \quad \text{and} \quad \sigma^k\leq\varepsilon.$$

For all $k>2k_0$, it follows from (3.17) that

$$\begin{aligned}
\left\|e_k(\eta^{k+1})\right\|_2 &\leq \sum_{i=0}^{k-1}\sigma^{k-i}\gamma\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sigma^k\left\|e_0(\eta^1)\right\|_2\\
&=\sum_{i=0}^{k_0-1}\sigma^{k-i}\gamma\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sum_{k_0}^{k-1}\sigma^{k-i}\gamma\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sigma^k\left\|e_0(\eta^1)\right\|_2\\
&\leq\left(\max_{0\leq i\leq k_0-1}\left\{\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\right\}\gamma\sum_{i=0}^{k_0-1}\sigma^{k-k_0-i}\right)\cdot\sigma^{k_0}+\sigma^k\left\|e_0(\eta^1)\right\|_2\\
&+\left(\sum_{k_0}^{k-1}\sigma^{k-i}\gamma\right)\cdot\max_{k_0\leq i\leq k-1}\left\{\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\right\}\\
&\leq\left[\left(\max_{0\leq i\leq k_0-1}\left\{\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\right\}\gamma\sum_{i=0}^{k_0-1}\sigma^{k-k_0-i}\right)+\left(\sum_{k_0}^{k-1}\sigma^{k-i}\gamma\right)+\left\|e_0(\eta^1)\right\|_2\right]\cdot\varepsilon,
\end{aligned}$$

which implies that

$$\left\|e_k(\eta^{k+1})\right\|_2 \overset{k\to\infty}{\longrightarrow} 0. \tag{3.24}$$

Moreover, note that $\left\|B\left(y^k-y^{k+1}\right)\right\|_2 \overset{k\to\infty}{\longrightarrow} 0$ can be obtained by the fact $\left\|u^k-u^{k+1}\right\|_{\mathcal{M}} \overset{k\to\infty}{\longrightarrow} 0$. The first assertion is proved.

Now we prove the second assertion. For the first part, $\left\|Ax^{k+1}+By^{k+1}-b\right\|_2 \overset{k\to\infty}{\longrightarrow} 0$, it follows immediately from the facts $\left\|Ax^{k+1}+By^{k+1}-b\right\|_2=\frac{1}{\beta}\left\|\lambda^k-\lambda^{k+1}\right\|_2$ and $\left\|u^k-u^{k+1}\right\|_{\mathcal{M}} \overset{k\to\infty}{\longrightarrow} 0$. Note that the optimality conditions of the y-subproblem at the $(k+1)$th iteration and a solution point $y^{\star}$ can be, respectively, written as

$$\begin{cases} g(y)-g\left(y^{k+1}\right)+\left(y-y^{k+1}\right)^{\top}\left(-B^{\top}\lambda^{k+1}\right)\geq 0,\\ g(y)-g\left(y^{\star}\right)+\left(y-y^{\star}\right)^{\top}\left(-B^{\top}\lambda^{\star}\right)\geq 0. \end{cases}$$

Accordingly, taking $y=y^{\star}$ and $y=y^{k+1}$, respectively, in the above inequalities, we have

$$\left(y^{k+1}-y^{\star}\right)^{\top}B^{\top}\lambda^{\star}\leq g\left(y^{k+1}\right)-g\left(y^{\star}\right)\leq\left(y^{k+1}-y^{\star}\right)^{\top}B^{\top}\lambda^{k+1}. \tag{3.25}$$

The same technique can also be applied to the x-subproblem and a solution point $x^{\star}$. Additionally, using the convexity of f, we have

$$\begin{aligned}
\left(x^{k+1}-x^{\star}\right)^{\top}A^{\top}\lambda^{\star} &\leq f\left(x^{k+1}\right)-f\left(x^{\star}\right)\\
&\leq\left(x^{k+1}-x^{\star}\right)^{\top}Q^{\top}\left(Qx^{k+1}-q\right)\\
&=\left(x^{k+1}-x^{\star}\right)^{\top}\left[A^{\top}\left(\lambda^k-\beta\left(Ax^{k+1}+By^k-b\right)\right)+Q^{\top}e_k\left(\eta^{k+1}\right)\right].
\end{aligned} \tag{3.26}$$

Then, summarizing (3.25) and (3.26), we obtain

$$\begin{aligned}
&\frac{1}{\beta}\left(\lambda^k-\lambda^{k+1}\right)^\top\lambda^\star\\
&\quad=\left(x^{k+1}-x^\star\right)^\top A^\top\lambda^\star+\left(y^{k+1}-y^\star\right)^\top B^\top\lambda^\star\\
&\quad\le\left[f\left(x^{k+1}\right)+g\left(y^{k+1}\right)\right]-\left[g\left(y^\star\right)+f\left(x^\star\right)\right]\\
&\quad\le\left(y^{k+1}-y^\star\right)^\top B^\top\lambda^{k+1}\\
&\qquad+\left(x^{k+1}-x^\star\right)^\top\left[A^\top\left(\lambda^k-\beta\left(Ax^{k+1}+By^k-b\right)\right)+Q^\top e_k\left(\eta^{k+1}\right)\right]\\
&\quad\le\frac{1}{\beta}\left(\lambda^k-\lambda^{k+1}\right)^\top\lambda^{k+1}\\
&\qquad+\beta\left(x^{k+1}-x^\star\right)^\top A^\top B\left(y^{k+1}-y^k\right)+\left(x^{k+1}-x^\star\right)^\top Q^\top e_k\left(\eta^{k+1}\right).
\end{aligned}\tag{3.27}$$

Since

$$\left\|e_k(\eta^{k+1})\right\|_2\to 0,\ \left\|Qx^{k+1}-Qx^\star\right\|_2\to 0,\ \left\|u^k-u^{k+1}\right\|_{\mathcal{M}}\to 0,\ \left\|u^{K+1}-u^\star\right\|_{\mathcal{M}}<\infty,$$

and

$$A\left(x^{k+1}-x^\star\right)=\frac{1}{\beta}\left(\lambda^k-\lambda^{k+1}\right)-B\left(y^{k+1}-y^\star\right)\quad\Rightarrow\quad\left\|A\left(x^{k+1}-x^\star\right)\right\|_2^2<\infty,$$

both the left- and right-hand sides of (3.27) converge to zero. As a result, we have

$$f\left(x^{k+1}\right)+g\left(y^{k+1}\right)\overset{k\to\infty}{\longrightarrow}f\left(x^\star\right)+g\left(y^\star\right),$$

which is the second assertion of this theorem. The proof is complete. ∎

3.4. More discussions. In Theorem 3.4, the convergence of the proposed InADMM (1) is established in the sense that the constraint violation converges to zero and the objective function values converge to the optimal value. The key is that our inexactness criterion (2.8) ensures that the error arising from solving the x-subproblems inexactly vanishes as the iteration goes on, i.e., $\left\|e_k(\eta^{k+1})\right\|_2\overset{k\to\infty}{\longrightarrow}0$ as proved in Theorem 3.4. Hence, the proposed InADMM (1) inherits the convergence of the exact version of ADMM (1.4), even though its x-subproblems are solved inexactly. With stronger assumptions on the model (1.1), stronger convergence results can be derived. Such a topic has its own interests, and it has been widely studied in the ADMM literature. Discussing such extensions is beyond the scope of this paper; hence we just briefly mention some such extensions in this subsection and focus on establishing the convergence of the InADMM (1) in terms of iterates under additional assumptions.

THEOREM 3.5. *Let $\left\{w^k\right\}$ be the sequence generated by the InADMM* (1)*. If B is further assumed to be of full column rank in* (1.1)*, then the convergence of InADMM* (1) *holds in the sense of its iterates, i.e.,*

$$\left\{x^k\right\}\overset{k\to\infty}{\longrightarrow}x^\infty,\quad\left\{y^k\right\}\overset{k\to\infty}{\longrightarrow}y^\infty,\quad\left\{\lambda^k\right\}\overset{k\to\infty}{\longrightarrow}\lambda^\infty,$$

where $w^\infty\in\Omega^\star$.

Proof. Recall the proof of Theorem 3.4. If B is of full column rank, then the matrix $\mathcal{M}$ defined in (3.1) is positive definite. Thus, it follows from (3.23) that $\left\{u^k\right\}_{i=1}^\infty$ is bounded. Hence, there exists a subsequence $\left\{u^{k_i}\right\}_{i=1}^\infty$ converging to $u^\infty=$

$(y^\infty, \lambda^\infty)$. Let us define $x^\infty := \left(A^\top A\right)^{-1} A^\top (b - By^\infty)$. Specifying step 5 of the InADMM (1) for the kth iteration, we have

$$Ax^k = \frac{1}{\beta}\left(\lambda^{k-1} - \lambda^k\right) - \left(By^k - b\right),$$

which can be equivalently written as

$$x^k = \left(A^\top A\right)^{-1}\left[A^\top\left(\frac{1}{\beta}\left(\lambda^{k-1} - \lambda^k\right) - \left(By^k - b\right)\right)\right], \tag{3.28}$$

because A is assumed to be of full column rank. According to the second assertion in Theorem 3.4, we have $\left\|\lambda^{k_i+1} - \lambda^{k_i}\right\|_2 \overset{i\to\infty}{\longrightarrow} 0$, together with the convergence of $\left\{y^{k_i}\right\}$ to y^∞, we know that (3.28) implies

$$x^{k_i} \overset{i\to\infty}{\longrightarrow} \left(A^\top A\right)^{-1} A^\top (b - By^\infty) = x^\infty.$$

Note that the optimality conditions (3.4) can be rewritten as

$$\begin{cases} \left(x - x^{k+1}\right)^\top \left[Q^\top\left(Qx^{k+1} - q\right) - A^\top\left(\lambda^k - \beta\left(Ax^{k+1} + By^k - b\right)\right) - Q^\top e_k(\eta^{k+1})\right] \\ \quad \geq 0, \\ g(y) - g\left(y^{k+1}\right) + \left(y - y^{k+1}\right)^\top\left(-B^\top\lambda^k + \beta B^\top\left(Ax^{k+1} + By^{k+1} - b\right)\right) \geq 0, \\ \left(\lambda - \lambda^{k+1}\right)^\top\left(Ax^{k+1} + By^{k+1} - b + \dfrac{1}{\beta}\left(\lambda^{k+1} - \lambda^k\right)\right) \geq 0 \end{cases}$$

for all $w \in \Omega$. Using the notation in (3.1), for the kth iterate, we have

$$g(y) - g\left(y^k\right) + \left(w - w^k\right)^\top F(w^k) + \left(w - w^k\right)^\top \begin{pmatrix} \beta A^\top B\left(y^{k-1} - y^k\right) - Q^\top e_{k-1}(\eta^k) \\ 0 \\ \frac{1}{\beta}\left(\lambda^k - \lambda^{k-1}\right) \end{pmatrix} \geq 0 \tag{3.29}$$

for all $w \in \Omega$. Recall the results in Theorem 3.4. If we consider the subsequence $\{w^{k_i}\}$ converging to $w^\infty := (x^\infty, y^\infty, \lambda^\infty)$ and taking the limit in (3.29) over k_i, we obtain

$$g(y) - g\left(y^\infty\right) + \left(w - w^\infty\right)^\top F(w^\infty) \geq 0 \quad \forall w \in \Omega,$$

which implies $w^\infty \in \Omega^\star$. Recall the convergence of all the following sequences:

$$\left\{u^{k_i} = \left(y^{k_i}, \lambda^{k_i}\right)\right\}, \quad \left\{\left\|e_k(\eta^{k+1})\right\|_2\right\}, \quad \text{and} \quad \left\{\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}\right\}.$$

As a result, for any $\varepsilon > 0$, there exists a k_ℓ such that

$$\left\|u^{k_\ell} - u^\infty\right\|_{\mathcal{M}} < \varepsilon, \quad \left\|e_{k_\ell - 1}(\eta^{k_\ell})\right\|_2 < \varepsilon, \quad \text{and} \quad \left\|u^{k_\ell - 1} - u^{k_\ell}\right\|_{\mathcal{M}} < \varepsilon.$$

Moreover, similar to the derivation from (3.16) to (3.22), considering $\{w^k\}$ from k_ℓ to k instead of from 1 to k, we have for all $k > k_\ell + 1$

$$\begin{aligned} \left\|u^k - u^\infty\right\|_{\mathcal{M}}^2 &\leq \left\|u^{k_\ell} - u^\infty\right\|_{\mathcal{M}}^2 + \frac{1}{\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_{k_\ell - 1}\left(\eta^{k_\ell}\right)\right\|_2 + \gamma\left\|u^{k_\ell - 1} - u^{k_\ell}\right\|_{\mathcal{M}}\right)^2 \\ &\leq \varepsilon^2 + \frac{1}{\mu}\frac{\sigma}{1-\sigma}\left(\varepsilon + \gamma\varepsilon\right)^2, \end{aligned} \tag{3.30}$$

which directly implies $y^k \overset{k\to\infty}{\longrightarrow} y^\infty$ and $\lambda^k \overset{k\to\infty}{\longrightarrow} \lambda^\infty$. Recalling the definition of x^k in (3.28) and the result $y^k \overset{k\to\infty}{\longrightarrow} y^\infty$, we further obtain the convergence of $\left\{x^k\right\}$ to x^∞. This completes the proof. ∎

Theorems 3.4 and 3.5 show that the proposed InADMM (1) completely inherits the known convergence results of the original exact version of ADMM (1.4). That is, we prove the convergence of the InADMM (1) in the sense of the constraint violation and optimal objective function value without the full-column-rank assumption of B and its convergence in the sense of iterates with this assumption. Indeed, without the full-column-rank assumption of B, the convergence result in Theorem 3.4 can be alternatively enhanced if the function $g(y)$ is assumed to be level bounded. We summarize this extension in the following corollary.

COROLLARY 3.6. *Let $\{w^k\}$ be the sequence generated by the InADMM* (1)*. If $g(y)$ is further assumed to be level bounded in* (1.1)*, then the convergence of InADMM* (1) *is partially in the sense of iterates, i.e.,*

$$\{x^k\} \overset{k\to\infty}{\longrightarrow} x^\infty, \quad \{By^k\} \overset{k\to\infty}{\longrightarrow} By^\infty, \quad \{\lambda^k\} \overset{k\to\infty}{\longrightarrow} \lambda^\infty,$$

where $w^\infty = (x^\infty, y^\infty, \lambda^\infty) \in \Omega^\star$.

Proof. The proof is analogous to that of Theorem 3.5. First, recall the results in Theorem 3.4:

$$f\left(x^k\right) + g\left(y^k\right) \overset{k\to\infty}{\longrightarrow} f\left(x^\star\right) + g\left(y^\star\right). \tag{3.31}$$

Then, it follows from (3.23) that $Qx^k \overset{k\to\infty}{\longrightarrow} Qx^\star$, and thus $f\left(x^k\right) \overset{k\to\infty}{\longrightarrow} f\left(x^\star\right)$. Together with (3.31), we have $g\left(y^k\right) \overset{k\to\infty}{\longrightarrow} g\left(y^\star\right)$. If $g(y)$ is assumed to be level bounded, so is $\{y^k\}_{i=1}^\infty$ (see [42, Chapter 1, Definition 1.8]). As a result, the sequence $\{u^k\}$ is bounded, and the inequality (3.30) still holds. Since B is not necessarily of full column rank, from (3.30) we only conclude that $By^k \overset{k\to\infty}{\longrightarrow} By^\infty$ and $\lambda^k \overset{k\to\infty}{\longrightarrow} \lambda^\infty$. The convergence of $\{x^k\}$ can be trivially derived by a proof similar to that of Theorem 3.5. The proof is complete. ∎

Remark 3.7. For many concrete applications of the model (1.1), $g(y)$ is proper and level bounded. For examples, consider $g(y) = \|y\|_1$ in the LASSO model (1.6), the indicator function of a bounded closed set, the nuclear norm function (see, e.g., [41]), and any strongly convex function.

Remark 3.8. For our exclusive emphasis, the LASSO model (1.2) and its variant in distributed optimization (see (6.3) to be studied in section 6.2), the function $g(y)$ is level bounded and the matrix B is of full column rank. Therefore, the convergence in the sense of iterates established in Theorem 3.5 is guaranteed for the sequence generated by the InADMM (1).

4. Worst-case convergence rate. In [28, 29, 36], the $\mathcal{O}\left(\frac{1}{t}\right)$ worst-case convergence rate measured by the iteration complexity has been established for the exact version of ADMM (1.4) in both the ergodic and nonergodic senses, where t is the iteration counter. In this section, we extend similar analysis to the proposed InADMM (1). Despite of the similar roadmap in proofs, because of the consideration of the inexact solution for the subproblem (2.1), the analysis for InADMM (1) turns out to be technically more complicated.

4.1. Ergodic convergence rate. We first prove the $\mathcal{O}\left(\frac{1}{t}\right)$ worst-case convergence rate in the ergodic sense for the InADMM (1). The proof for the exact version of ADMM (1.4) is referred to [28, 36].

THEOREM 4.1. *Let $\{w^k\}$ be the sequence generated by the proposed InADMM (1); and $\bar{w}^k$ be defined in (3.11). For any integer $t > 1$, we further define*

$$\hat{w}_t = \frac{1}{t}\sum_{k=1}^{t} \bar{w}^k. \tag{4.1}$$

Then, for all $w \in \Omega$, it holds that

$$\begin{aligned} &g(\hat{y}_t) - g(y) + (\hat{w}_t - w)^\top F(w) \\ &\le \frac{1}{t}\left[\frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2 + \frac{1}{2}\left\|u^0 - u\right\|_{\mathcal{M}}^2\right] = \mathcal{O}\left(\frac{1}{t}\right), \end{aligned} \tag{4.2}$$

where $\mathcal{M}$ is defined in (3.1), γ is defined in (3.9), and μ satisfies (3.10).

Proof. Recall the inequality (3.21). We thus have

$$\begin{aligned} &\sum_{k=1}^{t}\left\{g\left(\bar{y}^k\right) - g(y) + \left(\bar{w}^k - w\right)^\top F(w)\right\} \\ &\le \frac{1}{2}\left(\left\|u^1 - u\right\|_{\mathcal{M}}^2 - \left\|u^{t+1} - u\right\|_{\mathcal{M}}^2\right) + \sum_{k=1}^{t}\left(\frac{\mu}{2}\frac{\sigma}{1-\sigma} - 1\right)\left\|Q\left(x^{k+1} - x\right)\right\|_2^2 \\ &+ \sum_{k=1}^{t-1}\frac{1}{2}\left(\frac{\sigma}{1-\sigma}\frac{\gamma^2}{\mu} - 1\right)\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2 - \frac{1}{2}\left\|u^t - u^{t+1}\right\|_{\mathcal{M}}^2 \\ &+ \frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2. \end{aligned}$$

Then, using (3.10), for all $w \in \Omega$, we have

$$\begin{aligned} &g(\hat{y}_t) - g(y) + (\hat{w}_t - w)^\top F(w) \\ &\overset{\text{Convexity}}{\le} \frac{1}{t}\sum_{k=1}^{t}\left\{g\left(\bar{y}^k\right) - g(y) + \left(\bar{w}^k - w\right)^\top F(w)\right\} \\ &\le \frac{1}{t}\left[\frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2 + \frac{1}{2}\left\|u^1 - u\right\|_{\mathcal{M}}^2\right], \end{aligned}$$

which completes the proof. ∎

Theorem 4.1 shows that after t iterations of the InADMM (1), we can find an approximate solution of the variational inequality (3.2) with an accuracy of $\mathcal{O}\left(\frac{1}{t}\right)$. This approximate solution is given in (4.1), and it is the average of all the points $\bar{w}^k$ which can be computed by all the known iterates generated by the InADMM (1). Hence, this is an $\mathcal{O}\left(\frac{1}{t}\right)$ worst-case convergence rate in the ergodic sense for the proposed InADMM (1).

4.2. Nonergodic convergence rate. Then we prove the $\mathcal{O}\left(\frac{1}{t}\right)$ worst-case convergence rate in a nonergodic sense. Note that the proof for the exact version of ADMM (1.4) is referred to [29].

To estimate the worst-case convergence rate in a nonergodic sense, we first need to clarify a criterion to precisely measure the accuracy of an iterate. Recall the optimality condition of an iterate generated by the InADMM (1) is given in (3.4). Then, it is easy to derive that for the iterate $(x^{k+1}, y^{k+1}, \lambda^{k+1})$ generated by the InADMM (1), for all $w \in \Omega$, it holds that

$$g(y) - g(y^{k+1}) + \left(w - w^{k+1}\right)^\top \left\{F(w^{k+1}) + \begin{pmatrix} \beta A^\top B\left(y^k - y^{k+1}\right) - Q^\top e_k(\eta^{k+1}) \\ 0 \\ \frac{1}{\beta}\left(\lambda^{k+1} - \lambda^k\right) \end{pmatrix}\right\} \ge 0.$$

Recall the variational inequality reformulation (3.2) and the notation in (3.1). It is clear that $(x^{k+1}, y^{k+1}, \lambda^{k+1})$ is a solution point of (3.2) if and only if $\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2 = 0$ and $\left\|e_k(\eta^{k+1})\right\|_2^2 = 0$. Hence, it is reasonable to measure the accuracy of the iterate $(x^{k+1}, y^{k+1}, \lambda^{k+1})$ by $\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2$ and $\left\|e_k(\eta^{k+1})\right\|_2^2$. Our purpose is thus to prove that after t iterations of the InADMM (1), both $\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2$ and $\left\|e_k(\eta^{k+1})\right\|_2^2$ can be bounded by upper bounds in order of $\mathcal{O}\left(\frac{1}{t}\right)$.

THEOREM 4.2. *Let $\{w^k\}$ be the sequence generated by the InADMM* (1)*. Then, for any integer $t > 1$, we have*

(4.3)
$$\min_{1\le k\le t}\left\{\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2\right\} \le \frac{1}{t}\left[\frac{1}{\nu}\left\|u^1 - u^\star\right\|_{\mathcal{M}}^2 + \frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2\right]$$

and

$$\begin{aligned}&\min_{1\le k\le t}\left\{\left\|e_k(\eta^{k+1})\right\|_2^2\right\}\\ &\quad\le \frac{1}{t}\left\{2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2\cdot\left[\frac{1}{\nu}\left\|u^1 - u^\star\right\|_{\mathcal{M}}^2 + \frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2\right]\right\}\end{aligned}$$

(4.4)
$$+\frac{1}{t^2}\left[2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2\right],$$

where $w^\star \in \Omega^\star$, γ is defined as (3.9)*, μ satisfies* (3.10)*, and $\nu = 1 - \frac{\gamma^2}{\mu}\frac{\sigma}{1-\sigma} > 0$.*

Proof. Recall the inequality (3.22), and choose $w^\star \in \Omega^\star$. We obtain

$$\begin{aligned}&\sum_{k=1}^{t+1}\left(1 - \frac{\mu}{2}\frac{\sigma}{1-\sigma}\right)\left\|Q\left(x^{k+1} - x^\star\right)\right\|_2^2 + \sum_{k=1}^{t}\left(\frac{1}{2} - \frac{\gamma^2}{2\mu}\frac{\sigma}{1-\sigma}\right)\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2\\ &\le \frac{1}{2}\left\|u^1 - u^\star\right\|_{\mathcal{M}}^2 + \frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2 - \frac{1}{2}\left\|u^{t+2} - u^\star\right\|_{\mathcal{M}}^2,\end{aligned}$$

which implies

$$\begin{aligned}&\sum_{k=1}^{t}\left(\frac{1}{2} - \frac{\gamma^2}{2\mu}\frac{\sigma}{1-\sigma}\right)\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2\\ &\le \frac{1}{2}\left\|u^1 - u^\star\right\|_{\mathcal{M}}^2 + \frac{1}{2\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2.\end{aligned} \tag{4.5}$$

Consequently, we have

$$\begin{aligned}&\min_{1\le k\le t}\left\{\left\|u^k - u^{k+1}\right\|_{\mathcal{M}}^2\right\}\\ &\le \frac{1}{t}\left[\frac{1}{\nu}\left\|u^1 - u^\star\right\|_{\mathcal{M}}^2 + \frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2 + \gamma\left\|u^0 - u^1\right\|_{\mathcal{M}}\right)^2\right],\end{aligned}$$

and the assertion (4.3) is proved. Note that ν is positive because of (3.10).

Then, it follows from the inequality (3.17) that

$$\left\|e_k(\eta^{k+1})\right\|_2 \le \sum_{i=0}^{k-1}\sigma^{k-i}\gamma\left\|u^i - u^{i+1}\right\|_{\mathcal{M}} + \sigma^k\left\|e_0(\eta^1)\right\|_2.$$

Summarizing the above inequality from $k=1$ to $k=t$, we obtain

$$\sum_{k=1}^{t}\left\|e_k(\eta^{k+1})\right\|_2 \le \sum_{k=1}^{t}\left\{\sum_{i=0}^{k-1}\sigma^{k-i}\gamma\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\sigma^k\left\|e_0(\eta^1)\right\|_2\right\}. \tag{4.6}$$

In addition, we have

$$\begin{aligned}&\sum_{k=1}^{t}\sum_{i=0}^{k-1}\sigma^{k-i}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}=\sum_{i=0}^{t-1}\sum_{k=i+1}^{t}\sigma^{k-i}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\\&=\sum_{i=0}^{t-1}\frac{\sigma-\sigma^{t-i+1}}{1-\sigma}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\le\sum_{i=0}^{t-1}\frac{\sigma}{1-\sigma}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\end{aligned}$$

and

$$\sum_{k=1}^{t}\sigma^k\left\|e_0(\eta^1)\right\|_2\le\frac{\sigma-\sigma^{t+1}}{1-\sigma}\left\|e_0(\eta^1)\right\|_2\le\frac{\sigma}{1-\sigma}\left\|e_0(\eta^1)\right\|_2.$$

Then, by simple calculation, we have

$$\begin{aligned}&\left(\sum_{k=1}^{t}\left\|e_k(\eta^{k+1})\right\|_2\right)^2\\&\overset{(4.6)}{\le}\left(\sum_{i=0}^{t-1}\frac{\sigma}{1-\sigma}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}+\frac{\sigma}{1-\sigma}\left\|e_0(\eta^1)\right\|_2\right)^2\\&\le 2\left(\sum_{i=1}^{t-1}\frac{\sigma}{1-\sigma}\gamma\cdot\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\right)^2+2\left(\frac{\sigma}{1-\sigma}\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)\right)^2\\&\le 2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2\cdot\left(\sum_{i=1}^{t}\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}\right)^2+2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\\&\le 2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2 t\cdot\left(\sum_{i=1}^{t}\left\|u^i-u^{i+1}\right\|_{\mathcal{M}}^2\right)+2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\\&\overset{(4.5)}{\le} 2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2 t\cdot\left[\frac{1}{\nu}\left\|u^1-u^\star\right\|_{\mathcal{M}}^2+\frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\right]\\&\quad+2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2.\end{aligned}$$

Further we get

$$\begin{aligned}&\left(t\cdot\min_{1\le k\le t}\left\|e_k\left(\eta^{k+1}\right)\right\|_2\right)^2\le\left(\sum_{k=1}^{t}\left\|e_k(\eta^{k+1})\right\|_2\right)^2\\&\le 2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2 t\cdot\left[\frac{1}{\nu}\left\|u^1-u^\star\right\|_{\mathcal{M}}^2+\frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\right]\\&+2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2,\end{aligned}$$

which implies

$$\begin{aligned}
&\min_{1\le k\le t}\left\{\left\|e_k(\eta^{k+1})\right\|_2^2\right\}\\
&\le \frac{1}{t}\left\{2\left(\frac{\sigma}{1-\sigma}\gamma\right)^2\cdot\left[\frac{1}{\nu}\left\|u^1-u^\star\right\|_{\mathcal{M}}^2+\frac{1}{\nu\mu}\frac{\sigma}{1-\sigma}\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\right]\right\}\\
&\quad+\frac{1}{t^2}\left[2\left(\frac{\sigma}{1-\sigma}\right)^2\cdot\left(\left\|e_0(\eta^1)\right\|_2+\gamma\left\|u^0-u^1\right\|_{\mathcal{M}}\right)^2\right].
\end{aligned}$$

The proof is complete. ∎

Remark 4.3. Note that the numbers in the right-hand sides of both (4.3) and (4.4) are order of $\mathcal{O}\left(\frac{1}{t}\right)$. Hence, Theorem 4.2 offers the $\mathcal{O}\left(\frac{1}{t}\right)$ worst-case convergence rate in a nonergodic sense for the proposed InADMM (1).

In this section, we show that the proposed InADMM (1) also inherits the known worst-case convergence rates established in [28, 29, 36] for the original exact version of ADMM (1.4).

5. Safe-guard numbers for internally nested iterations. In this section, we discuss how to ensure the inexactness criterion (2.8) when a standard numerical linear algebra solver is applied to iteratively solve the linear system (2.5a).

Recall that (2.5a) is

$$\hat{H}\eta^{k+1}-Q\left(\beta A^\top A\right)^{-1}h^k=0. \tag{5.1}$$

We aim at finding the safe-guard iteration number, denoted by $n_{\max}(\sigma)$, for a specific solver if it is applied to iteratively solve the linear system (5.1), so as to meet the inexactness criterion (2.8). Hence, the implementation of InADMM (1) is automatic without any ambiguity and user-friendly. Note that n is the counter for the internally nested iterations, and the counter k for the outer-loop iterations is fixed in this section.

Let η_n^k denote the nth internal iterate generated by a solver for the linear system (5.1), and set $\eta_0^k=\eta^k$ as the initial iterate. The matrix $\hat{H}$ defined in (2.6) is positive definite because of the full-column-rank assumption of A. So we use the matrix norm $\|\eta\|_{\hat{H}}:=\sqrt{\eta^\top\hat{H}\eta}$. Let us denote the spectrum radius of $\hat{H}$ by $\rho(\hat{H})$ and its condition number by $\kappa(\hat{H})$.

5.1. CG and PCG. First, we analyze the safe-guard number for the CG. According to, e.g., [44, Lecture 38, Theorem 38.5], the error at the nth iterate generated by the CG satisfies

$$\left\|\eta_n^k-\eta_{exact}^k\right\|_{\hat{H}}\le 2c^n\left\|\eta_0^k-\eta_{exact}^k\right\|_{\hat{H}}, \tag{5.2}$$

where η_{exact}^k denotes the exact solution of the linear system (5.1), and the constant c is defined as

$$c=\frac{\sqrt{\kappa(\hat{H})}-1}{\sqrt{\kappa(\hat{H})}+1}<1.$$

Therefore, for the nth iterate η_n^k generated by the CG, it satisfies that

$$\begin{aligned}\left\|e_k\left(\eta_n^k\right)\right\|_2 &= \left\|\hat{H}\eta_n^k - Q\left(\beta A^\top A\right)^{-1} h^k\right\|_2 \\ &\le \sqrt{\rho(\hat{H})}\cdot\left\|\eta_n^k - \eta_{exact}^k\right\|_{\hat{H}} \overset{(5.2)}{\le} 2\sqrt{\rho(\hat{H})}\cdot c^n\left\|\eta_0^k - \eta_{exact}^k\right\|_{\hat{H}} \\ &\le 2\sqrt{\kappa(\hat{H})}\cdot c^n\left\|\hat{H}\left(\eta_0^k - \eta_{exact}^k\right)\right\|_2 \\ &= 2\sqrt{\kappa(\hat{H})}\cdot c^n\left\|\hat{H}\eta_0^k - Q\left(\beta A^\top A\right)^{-1} h^k\right\|_2, \\ &= 2\sqrt{\kappa(\hat{H})}\cdot c^n\left\|e_k\left(\eta_0^k\right)\right\|_2 = 2\sqrt{\kappa(\hat{H})}\cdot c^n\left\|e_k\left(\eta^k\right)\right\|_2,\end{aligned}$$

where $e_k(\eta)$ is defined in (2.7), and note that $\eta_0^k = \eta^k$. Obviously, to guarantee the inexactness criterion (2.8) with a given σ, it suffices to hold

$$2\sqrt{\kappa(\hat{H})}\cdot c^{n(\sigma)} \le \sigma.$$

In other words, to apply the CG for iteratively solving (5.1), the inexactness criterion (2.8) is guaranteed by the safe-guard iteration number

$$n_{\max}(\sigma) := \left\lceil \log_c\left(\sigma/\left(2\sqrt{\kappa(\hat{H})}\right)\right)\right\rceil. \tag{5.3}$$

In addition, if the linear system (5.1) is ill conditioned, i.e., $\kappa(\hat{H})$ is large, we consider using the PCG for the linear system (5.1). For this case, the preconditioned surrogate of (5.1) is solved instead:

$$P^{-1}\left(\hat{H}\eta^{k+1} - Q\left(\beta A^\top A\right)^{-1} h^k\right) = 0,$$

in which P is the preconditioner. For this case, the safe-guard iteration number for the PCG is given by (5.3) but with $c = \frac{\sqrt{\kappa(P^{-1}\hat{H})}-1}{\sqrt{\kappa(P^{-1}\hat{H})}+1}$.

5.2. Jacobian, Gauss–Seidel, and SOR. For other solvers such as the Jacobian, Gauss–Seidel, and SOR methods, similar analysis can be conducted for finding the safe-guard numbers. More specifically, we decompose the matrix $\hat{H}$ as

$$\hat{H} := D - L^\top - L,$$

where D is a diagonal matrix and L is a lower triangular matrix. Let us consider a conceptual and general iterative scheme

$$\eta_{n+1}^k = T\eta_n^k + S^{-1}h^k, \tag{5.4}$$

where S and T satisfy $S - ST = \hat{H}$ with $\rho(T) < 1$ to ensure the convergence. Then, the Jacobian, Gauss–Seidel, and SOR methods can all be specified by the general scheme (5.4) as follows:

$$\left\{\begin{array}{cl} \textbf{Jacobi}: & T = D^{-1}\left(L^\top + L\right), \quad S = D, \\ \textbf{Gauss–Seidel}: & T = (D-L)^{-1}L^\top, \quad S = D - L, \\ \textbf{SOR}: & T = (D - wL)^{-1}\left[(1-w)D + wL^\top\right], \quad S = \frac{D}{w} - L. \end{array}\right. \tag{5.5}$$

Our analysis is thus valid for any scheme that can be recovered by the general scheme (5.4), though the three mentioned ones are still our main purpose.

According to [45, Chapter 3.2], for the nth iteration generated by the scheme (5.4) for (5.1), η_n^k, it holds that

$$\begin{aligned}
&\left\|e_k\left(\eta_n^k\right)\right\|_2 = \left\|\hat{H}\eta_n^k - Q\left(\beta A^\top A\right)^{-1} h^k\right\|_2 = \left\|\hat{H}\left(\eta_n^k - \eta_{exact}^k\right)\right\|_2 \\
&= \left\|\hat{H}T^n\left(\eta_0^k - \eta_{exact}^k\right)\right\|_2 \\
&\le \left\|\hat{H}T^n\hat{H}^{-1}\right\|_2 \cdot \left\|\hat{H}\left(\eta_0^k - \eta_{exact}^k\right)\right\|_2 = \left\|\hat{H}T^n\hat{H}^{-1}\right\|_2 \cdot \left\|\hat{H}\eta_0^k - Q\left(\beta A^\top A\right)^{-1} h^k\right\|_2 \\
&= \left\|\hat{H}T^n\hat{H}^{-1}\right\|_2 \cdot \left\|e_k\left(\eta_0^k\right)\right\|_2 = \left\|\hat{H}T^n\hat{H}^{-1}\right\|_2 \cdot \left\|e_k\left(\eta^k\right)\right\|_2.
\end{aligned}$$

Therefore, to ensure the inexactness criterion (2.8) at η_n^k, it suffices to guarantee $\|\hat{H}T^n\hat{H}^{-1}\|_2 \le \sigma$. Since $\rho(T) < 1$ is required to ensure the convergence of the general scheme (5.4), immediately we know that $\|\hat{H}T^n\hat{H}^{-1}\|_2 \overset{n\to\infty}{\longrightarrow} 0$. Hence, $\|\hat{H}T^n\hat{H}^{-1}\|_2$ must be smaller than the given positive scalar σ for a sufficiently large n and the safe-guard number $n_{\max}(\sigma)$ can be discerned accordingly on the cost of estimating $\|\hat{H}T^n\hat{H}^{-1}\|_2$. Below we give some more meticulous analysis for estimating the safe-guard numbers for (5.4).

Let us recall the average rate of convergence and asymptotic rate of convergence; see, e.g., [45, Chapter 3, Theorem 3.4]. Then we have

$$\begin{aligned}
\lim_{n\to\infty} \frac{-\ln\left\|\hat{H}T^n\hat{H}^{-1}\right\|_2}{n} &= \lim_{n\to\infty} \frac{-\ln\left\|(\hat{H}T\hat{H}^{-1})^n\right\|_2}{n} \\
&= -\ln\rho(\hat{H}T\hat{H}^{-1}) := R_\infty(\hat{H}T\hat{H}^{-1}) \\
&= -\ln\rho(T). \qquad (5.6)
\end{aligned}$$

Hence, instead of estimating $\|\hat{H}T^n\hat{H}^{-1}\|_2$ which is usually computationally expensive, we can replace it by $[\rho(T)]^n$ and accordingly estimate the safe-guard number via $[\rho(T)]^n < \sigma$. That is, the safe-guard number for (5.4) can be well estimated by the number $\lceil \log_{\rho(T)}\sigma \rceil$. For some specific cases of the general scheme (5.4), this estimate can be precise. For example, if $S = aI$ with $a > \rho(\hat{H})$ in (5.4), then we have $\|\hat{H}T^n\hat{H}^{-1}\|_2 = \rho^n(T)$, and thus the safe-guard number is precisely given by $n_{\max}(\sigma) = \lceil \log_{\rho(T)}\sigma \rceil$. Also, if it is known that the condition $\|\hat{H}T\hat{H}^{-1}\|_2 < 1$ is satisfied, then we have

$$\left\|\hat{H}T^n\hat{H}^{-1}\right\|_2 = \left\|(\hat{H}T\hat{H}^{-1})^n\right\|_2 \le \left\|\hat{H}T\hat{H}^{-1}\right\|_2^n,$$

with which the safe-guard number is precisely given by $n_{\max}(\sigma) = \left\lceil \log_{\left\|\hat{H}T\hat{H}^{-1}\right\|_2}\sigma \right\rceil$.

As we shall show in section 6, usually these safe-guard numbers are very small, in one or two digits, although the dimension of the involved linear system is huge. Hence, the internally nested iterations to meet the inexactness criterion (2.8) via iteratively solving the linear system (5.1) can be very efficient. This feature guarantees the efficiency of the proposed InADMM (1) for big-data scenarios of (1.1) with huge-dimensional variables. Last, we reiterate that the safe-guard numbers are sufficient to guarantee the inexact criterion (2.8), and they are still over-estimated; we refer to numerical results in section 6. To implement the proposed InADMM (1), the inexactness criterion (2.8) automatically guarantees the safe level of accuracy, and there is no need to follow these safe-guard numbers to execute the internally nested iterations.

6. Numerical experiments. In this section we test some big datasets of the benchmark LASSO model (1.2) and its variant arising in distributed optimization (see (6.3)) and numerically show the efficiency of the proposed InADMM (1). Our theoretical assertions, especially the necessity of solving the involved system of linear equations inexactly subject to the proposed inexactness criterion for big datasets, are numerically verified. All the numerical experiments were implemented on a laptop with Intel(R) Core(TM) i5-6300U CPU@ 2.40GHz 2.50GHz and 8.00 GB memory. All the codes were written in MATLAB 2016a.

6.1. LASSO. Recall that the benchmark LASSO model (1.2) can be reformulated as (1.6) and the iterative scheme of the exact version of ADMM reads as (1.7). To apply the proposed InADMM (1), the resulting y- and λ-subproblems are the same as (1.7) except that the x-subproblem (1.8) is solved inexactly by (2.5a) subject to the inexactness criterion (2.8)–(2.9).

6.1.1. Synthetic dataset. We first generate some synthetic datasets for the benchmark LASSO model (1.2) with gradually increasing dimensionality.

The matrix Q is generated by the Matlab script *sprandn(p,n,d)* where p and n are the dimensions and d is the density of nonzero entries. We report five cases as shown in Table 1. Then, a vector $x_0 \in \mathbb{R}^n$ is generated by *sprandn(n,*1, $\frac{100}{n}$*)*, and the vector $q \in \mathbb{R}^p$ is calculated by $Qx_0 + 0.1\varepsilon$ with $\varepsilon \in \mathbb{R}^p$ a standard normally distributed random noise vector. Following [5], the parameter τ is set to be $0.1\|Q^\top q\|_\infty$ for effectively identifying nonzero entries.

To illustrate the necessity of solving the x-subproblem (1.8) inexactly, we also test the case where these subproblems are solved exactly by direct methods when the dimension of the linear system (1.8) is not that high. This is also the reason why we purposively generate some small- or medium-size synthetic datasets so that the x-subproblem can be solved exactly by direct methods, and thus the exact and inexact versions of ADMM can be compared. We particularly test Cholesky factorization and the LSQR method as in [5].[2] The corresponding iterative schemes are denoted by ADMM$_{Cholesky}$ and ADMM$_{LSQR}$, respectively. That is, ADMM$_{Cholesky}$ means the Cholesky factorization $\hat{H} = L^\top L$ is executed, and then the solution of (2.5a) is given by $\eta^{k+1} = \frac{1}{\beta} L^{-1}\left(L^{-\top} Q h^k\right)$. As in [5], the stopping criterion for implementing ADMM$_{Cholesky}$ and ADMM$_{LSQR}$ is

$$\left\|x^k - y^k\right\|_2 < \sqrt{n} \times \epsilon^{abs} + \epsilon^{rel} \times \max\left\{\left\|x^k\right\|_2, \left\|y^k\right\|_2\right\},$$

$$\left\|\beta\left(y^k - y^{k-1}\right)\right\|_2 < \sqrt{n} \times \epsilon^{abs} + \epsilon^{rel} \times \left\|\lambda^k\right\|_2,$$

where $\epsilon^{abs} = 10^{-4}$ and $\epsilon^{rel} = 10^{-3}$. Certainly, if too huge-dimensional cases are considered, it is hard to apply these direct methods to solve the x-subproblem (1.8) exactly.

Note that when the LASSO model (1.2) is considered, we have $A^\top A = I_{n\times n}$, and hence the linear system (2.5a) reduces to

$$\hat{H}\eta = \frac{1}{\beta} Q h^k \tag{6.1}$$

with $\hat{H}$ and h^k given in (2.6) and (2.3), respectively.

[2] http://stanford.edu/~boyd/papers/admm/.

Meanwhile, to show the superiority of the proposed automatically adjustable inexactness criterion (2.8), we particularly compare it with the cases where the linear system (6.1) is solved inexactly but subject to a priori fixed levels of accuracy:

$$\frac{\left\| \frac{1}{\beta} Qh^k - \hat{H}\eta \right\|_2}{\left\| \frac{1}{\beta} Qh^k \right\|_2} \leq 10^{-t},$$

where the integer t denotes an accuracy level. We shall test various values of $t = 2, 4, 6, 8, 10$, representing low to high fixed levels of accuracy.

For simplicity, let us just focus on implementing the CG for the linear system (6.1). According to section 5, the safe-guard iteration number for the CG is given by (5.3). In practice, if it is computationally expensive to compute $\kappa(\hat{H})$ exactly, then instead of (5.3), we can also compute an upper bound of κ as $\kappa_u := \frac{\|Q\|_2^2}{\beta} + 1$ and accordingly $c_u := \frac{\sqrt{\kappa_u}-1}{\sqrt{\kappa_u}+1}$, which is also an upper bound of c, and estimate the safe-guard iteration number $n_{\max}(\sigma)$ theoretically given in (5.3) via

$$\left\lceil \log_{c_u} \left(\sigma / \left(2\sqrt{\kappa_u} \right) \right) \right\rceil. \tag{6.2}$$

The initial iterate (η^0, y^0, λ^0) is set to be zeros, $\beta = 0.05\|Q^\top q\|_\infty$, σ is chosen as

$$\sigma = \frac{0.99}{1 + \frac{\|Q\|_2}{\sqrt{2\beta}}} < \frac{\sqrt{2\beta}}{\sqrt{2\beta} + \|Q\|_2},$$

so as to satisfy the condition (2.9), and $\|Q\|_2$ is obtained by the state-of-art power iteration in [35]. For comparison, all the inexact versions of ADMM, including InADMM (1) and the cases where the x-subproblem (1.8) is solved up to fixed accuracy levels, terminate when the objective function values are better than those obtained by $\text{ADMM}_{Cholesky}$ and ADMM_{LSQR}. That is,

$$\begin{aligned} &\frac{1}{2} \left\| Qy^k - q \right\|_2^2 + \tau \left\| y^k \right\|_1 \\ &\quad < \min \left\{ \text{Objective of } \text{ADMM}_{Cholesky}, \text{Objective of } \text{ADMM}_{LSQR} \right\}, \end{aligned}$$

which is reasonable if we recall that the LASSO model (1.2) is unconstrained. For all the methods under comparison, the initial iterate for executing the $(k+1)$th's internally nested iteration is taken as the kth outer-loop iterate η^k.

We list the parameters defining the synthetic datasets in Table 1, and the corresponding values of τ, β, σ and the safe-guard numbers of the CG. We report some numerical results in Table 2 for the synthetic datasets. In this table, "Iteration" means the overall outer iteration number, "Mean/Max CG" are the mean and maximal iteration numbers of the CG for solving the linear systems among all outer iterations, "Time" is the computing time in seconds, "Obj" is objective function value when the stopping criterion is satisfied, and "$n_{\max}(\sigma)$" is the safe-guard iteration number of the CG computed by (6.2) to guarantee the inexactness criterion (2.8) when the InADMM (1) is implemented. We use "$\sim$" for the case where the iteration number exceeds the maximum outer-loop iteration number (which is set as 500 in our code) or where it is inapplicable.

In Table 2, we see that both $ADMM_{Cholesky}$ and $ADMM_{LSQR}$ are generally much slower than inexact versions of ADMM which solve this linear system inexactly,

TABLE 1
Values of τ, β, σ, and $n(\sigma)$ for synthetic datasets.

(p, n, d)	τ	β	σ	$n_{\max}(\sigma)$
$(10^4, 1.5*10^4, 50\%)$	1067.321	533.6606	0.1889	13
$(10^4, 1.5*10^4, 20\%)$	540.2915	270.1457	0.1960	12
$(2.5*10^4, 5*10^4, 1\%)$	73.1531	36.5766	0.1818	14
$(10^5, 1.5*10^5, 0.1\%)$	25.3148	12.6574	0.1816	15
$(10^5, 10^6, 0.01\%)$	4.0682	2.0341	0.1227	26

TABLE 2
Comparison between InADMM (1)*, $ADMM_{Cholesky}$, $ADMM_{LSQR}$, and $ADMM_{1e-t}$.*

(p, n, d)	Algorithm	Iter	Mean/max CG	Time	Obj
$(10^4, 1.5*10^4, 50\%)$	$\text{ADMM}_{Cholesky}$	87	$\sim$	58.4204	6.0922e+04
	ADMM_{LSQR}	87	$\sim$	222.6887	6.0922e+04
	ADMM_{1e-10}	87	20.5172/29	459.7782	6.0922e+04
	ADMM_{1e-8}	87	14.1034/23	342.5131	6.0922e+04
	ADMM_{1e-6}	87	7.5862/16	193.0946	6.0922e+04
	ADMM_{1e-4}	82	2.6098/10	86.5464	6.0922e+04
	ADMM_{1e-2}	> 500	$\sim$	$\sim$	$\sim$
	InADMM	85	1.2471/2	69.3687	6.0922e+04
$(10^4, 1.5*10^4, 20\%)$	$\text{ADMM}_{Cholesky}$	80	$\sim$	57.1158	3.6288e+04
	ADMM_{LSQR}	80	$\sim$	96.9766	3.6288e+04
	ADMM_{1e-10}	80	19.9500/29	198.0422	3.6288e+04
	ADMM_{1e-8}	80	13.8125/23	145.9727	3.6288e+04
	ADMM_{1e-6}	80	7.4000/16	90.8541	3.6288e+04
	ADMM_{1e-4}	75	2.5733/10	40.5114	3.6288e+04
	ADMM_{1e-2}	> 500	$\sim$	$\sim$	$\sim$
	InADMM	79	1.1899/2	31.5043	3.6288e+04
$(2.5*10^4, 5*10^4, 1\%)$	$\text{ADMM}_{Cholesky}$	83	$\sim$	4581.3164	4.6552e+03
	ADMM_{LSQR}	83	$\sim$	65.1848	4.6552e+03
	ADMM_{1e-10}	83	17.9277/25	121.7677	4.6552e+03
	ADMM_{1e-8}	83	12.6627/20	88.4123	4.6552e+03
	ADMM_{1e-6}	83	7.0723/14	53.7587	4.6552e+03
	ADMM_{1e-4}	81	2.6049/9	25.6452	4.6552e+03
	ADMM_{1e-2}	> 500	$\sim$	$\sim$	$\sim$
	InADMM	83	1.2892/2	22.1035	4.6552e+03
$(10^5, 1.5*10^5, 0.1\%)$	$\text{ADMM}_{Cholesky}$	$\sim$	$\sim$	> 5000	$\sim$
	ADMM_{LSQR}	70	$\sim$	115.3126	1.4368e+03
	ADMM_{1e-10}	70	22.0429/30	239.8781	1.4368e+03
	ADMM_{1e-8}	70	15.4714/24	171.0421	1.4368e+03
	ADMM_{1e-6}	71	8.4789/17	103.4054	1.4368e+03
	ADMM_{1e-4}	69	2.9565/10	47.2624	1.4368e+03
	ADMM_{1e-2}	> 500	$\sim$	$\sim$	$\sim$
	InADMM	71	1.3239/3	36.5243	1.4368e+03
$(10^5, 10^6, 0.01\%)$	$\text{ADMM}_{Cholesky}$	$\sim$	$\sim$	> 5000	$\sim$
	ADMM_{LSQR}	134	$\sim$	164.9873	289.2456
	ADMM_{1e-10}	134	10.179/14	197.8944	289.2456
	ADMM_{1e-8}	134	7.7239/11	151.0366	289.2456
	ADMM_{1e-6}	135	4.8370/9	104.9057	289.2456
	ADMM_{1e-4}	137	2.1679/5	71.1910	289.2456
	ADMM_{1e-2}	> 500	$\sim$	$\sim$	$\sim$
	InADMM	141	1.1418/2	59.3988	289.2430

especially if the dimension of the dataset is larger. In our experiments, the time for Cholesky decomposition exceeds 5000 seconds for the latter two cases, and thus their comparisons are not included in Table 2. Results in this table show that generally it is necessary to solve the linear system (6.1) inexactly when the dimension is high. Also,

TABLE 3
Numerical comparison on RCV1 dataset.

Algorithm	Iter	Mean/Max CG.	Time	Obj
ADMM_{LSQR}	21	$\sim/\sim$	2.5185	7.0631e+03
ADMM_{1e-10}	21	16.9524/20	4.0077	7.0631e+03
ADMM_{1e-8}	21	13.6190/17	3.2112	7.0631e+03
ADMM_{1e-6}	21	9.2857/12	2.3173	7.0631e+03
ADMM_{1e-4}	21	5.1429/8	1.4650	7.0631e+03
ADMM_{1e-2}	> 500	$\sim/\sim$	$\sim$	$\sim$
InADMM	22	2.8182/3	1.1996	7.0631e+03

it is not efficient to control the accuracy of the inexact solutions of (6.1) by either too low or too high accuracy, and there is no evidence in fixing level of the accuracy. With the automatically adjustable inexactness criterion (2.8), the proposed InADMM (1) works well for all the tested cases, and it automatically avoids the difficulty caused by extremely low or high accuracy for the linear system (6.1). It is conclusive that solving the linear system (6.1) up to a too high accuracy does not help at all in accelerating the convergence of the ADMM; meanwhile it is easy to imagine that solving it with a too low accuracy returns low-quality output and hence ruins the convergence. Our experiments show that the accuracy of 10^{-4} turns out to be good for the generated datasets, but the point is that there is no clue for choosing the accuracy level a priori. As theoretically analyzed, despite the high dimension of the linear system (6.1), only a few CG iterations are needed to satisfy the inexactness criterion (2.8) and hence to guarantee the convergence of InADMM (1). This feature significantly helps save computation for big-data cases of the LASSO model (1.2). We notice that $n_{\max}(\sigma)$ for InADMM (1) is just a theoretical and overestimated upper bound to guarantee (2.8); practically the iteration numbers that are really executed by the CG for these datasets to ensure (2.8) are just at most 2 or 3. This essentially explains the efficiency of the InADMM (1) shown in Table 2.

6.1.2. Real dataset. We also test two popular real datasets: "RCV1"[3] in [32] and "news20"[4] in [31] for the LASSO model (1.2). Implementation details of various versions of the ADMM are the same as those stated in the last subsection, unless otherwise specified. The $\text{ADMM}_{Cholesky}$ is not compared in this subsection, because the dimensions of the these two datasets are too large and it is too time-consuming to execute the Cholesky factorization.

For RCV1, in terms of the LASSO model (1.2), the dimension of the data matrix Q is 20242×47236. We set $\tau = 0.1\|Q^\top q\|_\infty = 26.4381$, $\beta = 0.05\|Q^\top q\|_\infty = 13.2190$, and $\sigma = 0.1934$ with $n_{\max}(\sigma) = 13$. The matrix Q is sparse, and thus generally all the tested versions of the ADMM are quite fast. We report some numerical results in Table 3, from which the efficiency of InADMM (1) is clearly shown. For this dataset, it is empirically observed that the accuracy of 10^{-4} is good for the internally nested iterations. Also, for this dataset, the InADMM (1) requires only three CG steps to meet the inexactness criterion (2.8), and hence it is very efficient.

We further test the news20 dataset. In terms of the LASSO model (1.2), the dimension of the data matrix Q for this dataset is $19,996 \times 1,355,191$. Note that Q is also spare for this dataset. We take $\tau = 0.1\|Q^\top q\|_\infty = 12.3306$, $\beta = 0.05\|Q^\top q\|_\infty = 6.1653$, and $\sigma = 0.0921$ with $n_{\max}(\sigma) = 35$. Some numerical results are reported in

[3] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html♯rcv1.binary.
[4] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html♯news20.

TABLE 4
Numerical comparison on news20 dataset.

Algorithm	Iteration	Mean/max CG	Time	Obj
ADMM_{LSQR}	18	$\sim/\sim$	20.8350	7.3341e+03
ADMM_{1e-10}	19	18.1579/22	28.1866	7.3339e+03
ADMM_{1e-8}	19	13.947/17	22.3447	7.3339e+03
ADMM_{1e-6}	19	19.1579/13	16.8641	7.3339e+03
ADMM_{1e-4}	18	5.7778/9	10.4031	7.3339e+03
ADMM_{1e-2}	110	0.7364/4	20.7784	7.3338e+03
InADMM	19	3.2632/4	8.1695	7.3340e+03

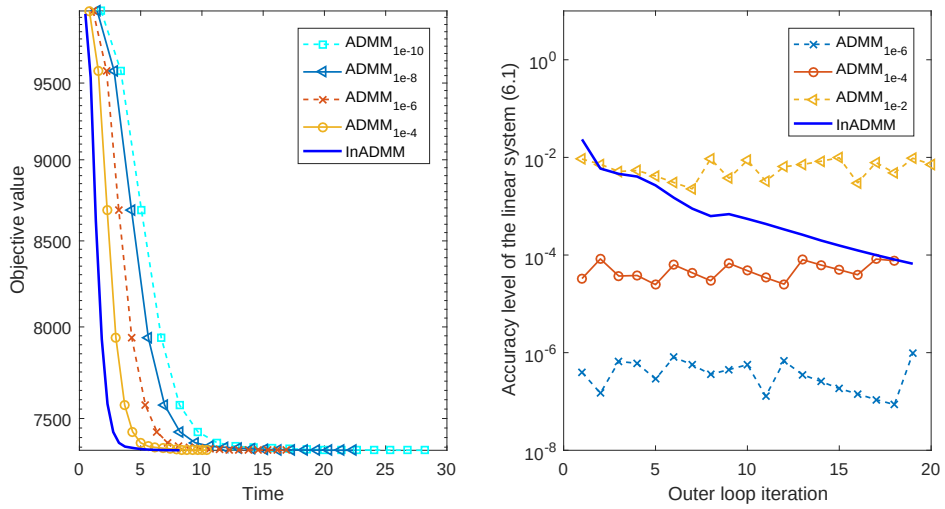

FIG. 1. *Objective value with respect to computing time and accuracy level of the linear system* (6.1) *with respect to the outer loop iteration on news20 dataset.*

Table 4. For this dataset, the accuracy of 10^{-4} is also good for the internally nested iterations, and the InADMM (1) requires four CG steps to meet the inexactness criterion (2.8). We further plot the evolutions of objective function values with respect to the computing time and error of subproblems with respect to the outer-loop iterations in Figure 1. The curves in this figure show that InADMM (1) achieves the near-optimal objective function value faster, and the proposed inexactness criterion automatically generates the good choice of accuracy of about 10^{-4} for solving the involved linear system, and it automatically avoids too high or too low accuracy.

6.2. Distributed LASSO. The generic LASSO model (1.2) also accounts for various distributed optimization models arising in multiagent networks, and hence it can be specified for concrete applications in this area. In a multiagent network, the agents seek to collaborate to accomplish certain tasks. For example, distributed database servers, may cooperate for parameter learning in order to fully exploit the data collected from individual servers, or a computation task may be executed by collaborative microprocessors with individual memories and storage spaces. We refer to [1, 4, 9, 10, 37, 51] for a few examples. In this subsection, we test some big datasets

arising in such a distributed optimization problem and numerically show the efficiency of the proposed InADMM (1).

6.2.1. Model and specification of the application of InADMM (1). Some distributed optimization problems can be modelled as the LASSO model (1.2) with the specific sum form

$$\min_x \sum_{i=1}^{N} \frac{1}{2} \|Q_i x - q_i\|_2^2 + \tau \|x\|_1, \tag{6.3}$$

where $x \in \mathbb{R}^n$ is the common decision variable, $\frac{1}{2}\|Q_i x - q_i\|_2^2$ is the cost function associated with agent i, Q_i is some data matrix (not necessarily of full column rank), and $\|x\|_1$ reflects the sparsity character of x (see, e.g., [2]). Note that the penalty term $\|x\|_1$ can be replaced by more general ones such as the structured group sparsity regularization (see, e.g., [49]), but we do not discuss these more complicated cases in this paper. In the setting of distributed optimization, it is commonly assumed that each agent i only has knowledge about the local information Q_i and q_i. The challenge is to obtain, for each agent in the system, the optimal $x^\star$ of (6.3) using only local information and messages exchanged with neighbors; see [10, 37, 51].

Clearly, we can reformulate (6.3) as

(6.4)

$$\min_{x\in\mathbb{R}^n} \ \frac{1}{2}\sum_{i=1}^{N} \|Q_i x - q_i\|_2^2 + \tau\|x\|_1 \quad \Leftrightarrow \quad \begin{array}{ll} \min\limits_{\{x_i\in\mathbb{R}^n\},x\in\mathbb{R}^n} & \frac{1}{2}\sum\limits_{i=1}^{N} \|Q_i x_i - q_i\|_2^2 + \tau\|x\|_1 \\ \text{s.t.} & x_i = x, \end{array}$$

so that we can apply various ADMM schemes. Note that the model (6.4) corresponds to the model (1.1) with the matrix A being a $(nN)\times(nN)$-dimensional identity matrix (of full column rank) and $Q = \operatorname{diag}\{Q_1, \ldots, Q_N\}$.

For the distributed LASSO model (6.3), the x-subproblem (2.5a) can be specified as N smaller linear systems with each corresponding to an agent or server of a distributed network. That is, we can calculate x_i^{k+1} via the following process:

$$\hat{H}_i \eta_i^{k+1} = \frac{1}{\beta} Q_i h_i^k, \tag{6.5a}$$

$$x_i^{k+1} = \frac{1}{\beta} Q_i \left(h_i^k - Q_i^\top \eta_i^{k+1}\right) \tag{6.5b}$$

with

$$\hat{H}_i = \frac{1}{\beta} Q_i^\top Q_i + I \quad \text{and} \quad h_i^k = Q_i^\top q_i + \beta x^k + \lambda_i^k. \tag{6.6}$$

In (6.6), λ_i denotes the Lagrange multiplier associated with the constraint $x_i = x$. For big-data scenarios of the distributed LASSO model (6.3), the individual matrix Q_i may be still of huge dimension though it may have some special structures such as the sparsity. Therefore, as the inexactness criterion (2.8), we should consider solving the linear system (6.5a) inexactly for $i = 1, \ldots, N$. Let us define the residual of the linear system (6.5a) as

$$e_k^i(\eta_i) := \frac{1}{\beta} Q_i h_i^k - \hat{H}_i \eta_i, \quad i = 1, \ldots, N.$$

Moreover, we choose σ to satisfy

$$0 < \sigma < \frac{1}{1 + \frac{\max\{\|Q_i\|_2\}}{\sqrt{2\beta}}} = \frac{1}{1 + \frac{\|Q\|_2}{\sqrt{2\beta}}} \in (0, 1), \tag{6.7}$$

where the equation holds because $Q = \mathrm{diag}\{Q_1, \ldots, Q_N\}$. Then, we suggest solving the linear system (6.5a) inexactly subject to the inexactness criterion

$$\left\|e_k^i\left(\eta_i^{k+1}\right)\right\|_2 \leq \sigma \cdot \left\|e_k^i\left(\eta_i^k\right)\right\|_2, \quad i = 1, \ldots, N.$$

Note that if we take all the distributed matrices Q_i and vectors q_i into consideration for $i = 1, 2, \ldots, N$, it holds that

$$\begin{aligned}\left\|e_k\left(\eta^{k+1}\right)\right\|_2 &= \left\|\hat{H}\eta^{k+1} - \frac{1}{\beta}Qh^k\right\|_2 = \sqrt{\sum_{i=1}^{N}\left\|\hat{H}_i\eta_i^{k+1} - \frac{1}{\beta}Q_ih_i^k\right\|_2^2} \\ &\leq \sqrt{\sigma^2\sum_{i=1}^{m}\left\|\hat{H}_i\eta_i^k - \frac{1}{\beta}Q_ih_i^k\right\|_2^2} \leq \sigma \cdot \left\|\hat{H}\eta^k - \frac{1}{\beta}Qh^k\right\|_2 = \sigma \cdot \left\|e_k\left(\eta^k\right)\right\|_2. \end{aligned} \tag{6.8}$$

Hence, (6.8) is a specification of (2.8) when the general LASSO model (1.2) is specified as the distributed LASSO model (6.3).

To show the necessity and efficiency of the automatically adjustable inexactness criterion (6.8), as in section 6.1, we also compare it with the case where the linear system (6.5a) is solved either exactly or up to a priori fixed accuracy levels. More specifically, we compare ADMM$_{LSQR}$ which means the linear system (6.5a) is solved exactly by the LSQR and ADMM$_{1e-t}$ which means the linear system (6.5a) is solved subject to the inexactness criterion with a fixed accuracy level:

$$\frac{\left\|\hat{H}_i\eta_i - \frac{1}{\beta}Q_ih_i^k\right\|_2}{\left\|\frac{1}{\beta}Q_ih_i^k\right\|_2} \leq 10^{-t}.$$

We test the cases where $t = 2, 4, 6, 8, 10$. Also, the InADMM (1) and ADMM$_{1e-t}$ are terminated only when the generated objective function values are better than those found by ADMM$_{LSQR}$. Note that the extremely high dimensionality of this dataset prevents us from executing the Cholesky decomposition, and thus we do not compare the case where the linear system (6.5a) is solved directly by the Cholesky decomposition.

6.2.2. Numerical results. We test two real big datasets: "url"[5] in [34] and "avazu-app"[6] in [30]. For both of the datasets, their dimensions are much higher than those of the RCV1 and news20 datasets.

The url dataset contains 121 days of a directory for malicious URL (spam, phishing, exploits, and so on) detection and the total dataset size is about 470 MB. The data sample has 3,231,961 features and $2,396,130$ data samples for the total 121 days. In this experiment, because of our limited computation infrastructure, we only consider the cases of the first 10/15/20 days from the whole data and treat each day's dataset as a subsystem. As a result, the dataset dimension of each subsystem is about $15,000 \times 2,396,130$.

[5] http://www.sysnet.ucsd.edu/projects/url/.

[6] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html♯avazu.

TABLE 5
Values of τ, β, and σ for url dataset.

N	τ	β	σ
N=10	766.6	383.3	0.0198
N=15	775.6	387.8	0.0198
N=20	775.6	387.8	0.0198

TABLE 6
Numerical comparison on url dataset.

N	Algorithm	Iteration	Mean/Max CG	Time	Obj
$N = 10$	ADMM_{LSQR}	31	$\sim/\sim$	774.7665	5.9316e+3
	ADMM_{1e-10}	31	35.8226/52	594.7653	5.9316e+3
	ADMM_{1e-8}	32	22.5344/37	398.8079	5.9318e+3
	ADMM_{1e-6}	30	10.3400/23	200.5246	5.9138e+3
	ADMM_{1e-4}	> 500	$\sim/\sim$	$\sim$	$\sim$
	ADMM_{1e-2}	> 500	$\sim/\sim$	$\sim$	$\sim$
	InADMM	30	7.4267/14	165.1480	5.9268e+3
$N = 15$	ADMM_{LSQR}	40	$\sim/\sim$	1430.7386	7.8744e+3
	ADMM_{1e-10}	40	33.8533/57	1087.9498	7.8744e+3
	ADMM_{1e-8}	40	21.1567/40	716.4767	7.8744e+3
	ADMM_{1e-6}	39	9.2137/23	361.1047	7.8730e+3
	ADMM_{1e-4}	> 500	$\sim/\sim$	$\sim$	$\sim$
	ADMM_{1e-2}	> 500	$\sim/\sim$	$\sim$	$\sim$
	InADMM	36	7.9537/17	313.6074	7.8723e+3
$N = 20$	ADMM_{LSQR}	34	$\sim/\sim$	1655.6177	9.5673e+3
	ADMM_{1e-10}	34	34.4029/57	1258.1451	9.5673e+3
	ADMM_{1e-8}	34	21.5441/40	889.1597	9.5673e+3
	ADMM_{1e-6}	32	9.8594/23	422.1671	9.5671e+3
	ADMM_{1e-4}	> 500	$\sim/\sim$	$\sim$	$\sim$
	ADMM_{1e-2}	> 500	$\sim/\sim$	$\sim$	$\sim$
	InADMM	37	8.0527/24	441.9357	9.5650e+3

For the url dataset, the values of τ, β, and τ are calculated by the formulas $\tau = \max\left\{0.1\|Q_i^\top q_i\|_\infty\right\}_{i=1}^N$, $\beta = 0.05\max\left\{\|Q_i^\top q_i\|_\infty\right\}_{i=1}^N$, and $\sigma = \min\{\sigma_i = \frac{0.99}{1+\frac{\|Q_i\|_2}{\sqrt{2\beta}}}\}$, respectively. We consider the distributed scenarios of $N = 10, 15, 20$, and correspondingly the values of these constants are listed in Table 5.

Some results for the url dataset are reported in Table 6. As observed previously, a too-high accuracy level for the internally nested iterations such as ADMM_{1e-10} and ADMM_{1e-8} slows down the convergence, while a too-low accuracy level such as ADMM_{1e-2} and ADMM_{1e-4} does not guarantee the convergence. For this dataset, it turns out that 10^{-6} is appropriate for the internally nested iterations, and if this accuracy happens to be found, the resulting numerical performance is very competitive. But again there is no clue at all to discern this appropriate level of accuracy in advance. On the contrary, the proposed inexactness criterion (6.8) can automatically find this appropriate level of accuracy. To see the efficiency of InADMM (1) more clearly, in Figure 2 we plot the evolution of the objective function values with respect to the computing time and the evolution of the mean of inner-loop iteration numbers with respect to the outer-loop iterations. We see that generally the CG steps for the internally nested iterations to ensure the convergence of the InADMM (1) are quite stable. In Figure 3, we also plot the evolutions of the primal and dual residuals of the model (6.4) with respect to the outer-loop iterations.

The avazu-app dataset was used in a competition on click-through rate prediction jointly hosted by Avazu and Kaggle in 2014. The participants were asked to learn a

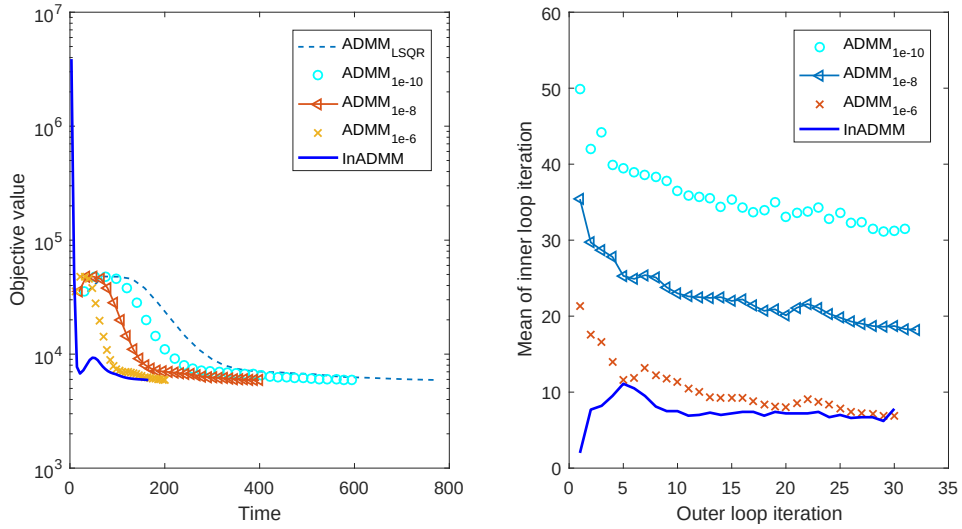

FIG. 2. *Objective value with respect to the computing time and mean of inner loop iteration number with respect to the outer loop iteration number on url dataset ($N = 10$).*

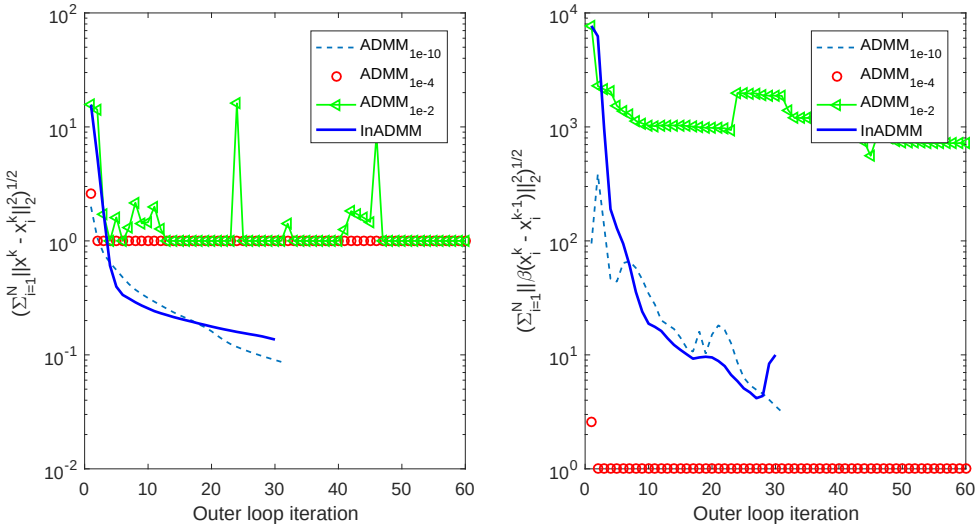

FIG. 3. *Norms of primal residual and dual residual with respect to the outer loop iteration number on url dataset ($N = 10$).*

model from the first 10 days of advertising log and predict the click probability for the impressions on the 11th day. This dataset contains 1,000,000 features and 14,596,137 samples; the total dataset size is about 394 MB. We split this dataset into 29 groups (hence, $N = 29$ in (6.3)); the first 28 groups have $500,000$ samples, and the last one has $596,137$ samples. Accordingly, the dataset dimension of each subsystem is $500,000 \times 1,000,000$ for the first 28 groups and $596,137 \times 1,000,000$ for the last one. Values of the parameters τ, β, and σ are computed by the same formulas as those for

TABLE 7
Numerical comparison on avazu-app dataset.

Algorithm	Iteration	Mean/Max CG	Time	Obj
ADMM_{LSQR}	36	$\sim/\sim$	1446.5865	7.2533e+05
ADMM_{1e-10}	37	18.6048/26	2887.6285	7.2533e+05
ADMM_{1e-8}	37	13.4921/21	2178.0520	7.2533e+05
ADMM_{1e-6}	36	8.2299/16	1425.0370	7.2533e+05
ADMM_{1e-4}	30	3.4747/12	633.4167	7.2533e+05
ADMM_{1e-2}	> 500	$\sim/\sim$	$\sim$	$\sim$
InADMM	38	3.1343/8	809.2827	7.2533e+05

the url dataset and they are $\tau = 2219.0, \beta = 1109.5, \sigma = 0.0825$, respectively. Other implementation details are the same as those mentioned in Section 6.1.

Some numerical results for testing the avazu-app dataset are reported in Table 7. Similar conclusions as those for the previous experiments can be derived. In particular, for this dateset, it seems that 10^{-4}, instead of 10^{-6} for the url dataset, is appropriate for the internally nested iterations. Because of the extremely high dimensionality of the variables, slightly increasing the accuracy for the internally nested iterations results in significantly additional computation and thus slows down the overall speed very much. For such a big dataset, it is more evident to use the proposed inexactness criterion (6.8) when implementing the ADMM, rather than a trial-and-error procedure of seeking an appropriate level of accuracy.

7. Conclusions. In this paper, we discussed how to effectively implement ADMM for large datasets in the convex programming context, with an emphasis on the problem of LASSO. It was revealed that the system of linear equations arising at each iteration of the ADMM should be inexactly solved, in which an adjustable inexactness criterion was proposed. This inexactness criterion automatically avoided very high or very low accuracy required for solving the subproblems. We also attempted to specify the safe-guard iteration numbers for several standard numerical linear algebra solvers when they are utilized for the subproblems, thereby realizing the inexact implementation of the ADMM with an internally nested iterative procedure, which should be fully automatic. It is noteworthy that existing convergence results for the exact version of the ADMM are not applicable. Hence, the convergence was proved and the worst-case convergence rate measured by the iteration complexity was established for the proposed inexact version of the ADMM with an internally nested iterative procedure. Some large datasets containing millions of variables were tested to numerically show the efficiency of the inexact version of the ADMM. These results showed that only a few steps for the internally nested iterations are often required to effectively guarantee the convergence of the inexact version of ADMM, in spite of the high dimensionality of their variables. Hence, the inexact implementation of ADMM for large datasets can be significantly accelerated; if subproblems could be inexactly solved in the presence of an appropriate inexactness criterion, accordingly, the convergence can be rigorously guaranteed.

REFERENCES

[1] G. ANDREWS, *Foundations of Multithreaded, Parallel, and Distributed Programming*, Addison-Wesley, Boston, MA, 2000.

[2] F. BACH, R. JENATTON, J. MAIRAL, AND G. OBOZINSKI, *Optimization with sparsity-inducing penalties*, Found. Trends Mach. Learn., 4 (2012), pp. 1–106, https://doi.org/10.1561/2200000015.

[3] R. Baraniuk and P. Steeghs, *Compressive radar imaging*, in Proceedings of the 2007 IEEE Radar Conference, IEEE, 2007, https://doi.org/10.1109/RADAR.2007.374203.
[4] R. Bekkerman, M. Bilenko, and J. Langford, *Scaling up Machine Learning: Parallel and Distributed Approaches*, Cambridge University Press, Cambridge, UK, 2011.
[5] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Mach. Learn., 3 (2011), pp. 1–122, https://doi.org/10.1561/2200000016.
[6] K. Bredies and H. Sun, *Preconditioned Douglas–Rachford splitting methods for convex-concave saddle-point problems*, SIAM J. Numer. Anal., 53 (2015), pp. 421–444, https://doi.org/10.1137/140965028.
[7] K. Bredies and H. Sun, *A proximal point analysis of the preconditioned alternating direction method of multipliers*, J. Optim. Theory Appl., 173 (2017), pp. 878–907, https://doi.org/10.1007/s10957-017-1112-5.
[8] T. Chan and R. Glowinski, *Finite Element Approximation and Iterative Solution of a Class of Mildly Non-Linear Elliptic Equations*, Report STAN-CS-78-674, Computer Science Department, Stanford University Stanford, 1978, https://pdfs.semanticscholar.org/6ed0/e4127c1beaf5f86ad768b72868184badeae7.pdf.
[9] T.-H. Chang, M. Hong, and X. Wang, *Multi-agent distributed optimization via inexact consensus ADMM*, IEEE Trans. Signal Process., 63 (2015), pp. 482–497, https://doi.org/10.1109/tsp.2014.2367458.
[10] J. Chen and A. Sayed, *Diffusion adaptation strategies for distributed optimization and learning over networks*, IEEE Trans. Signal Process., 60 (2012), pp. 4289–4305, https://doi.org/10.1109/tsp.2012.2198470.
[11] W. Deng and W. Yin, *On the global and linear convergence of the generalized alternating direction method of multipliers*, J. Sci. Comput., 66 (2015), pp. 889–916, https://doi.org/10.1007/s10915-015-0048-x.
[12] J. Eckstein and D. Bertsekas, *On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators*, Math. Program., 55 (1992), pp. 293–318, https://doi.org/10.1007/bf01581204.
[13] J. Eckstein and W. Yao, *Understanding the convergence of the alternating direction method of multipliers: Theoretical and computational perspectives*, Pac. J. Optim., 11 (2015), pp. 619–644.
[14] J. Eckstein and W. Yao, *Relative-error approximate versions of Douglas–Rachford splitting and special cases of the ADMM*, Math. Program., (2017), https://doi.org/10.1007/s10107-017-1160-5.
[15] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, New York, 2003, https://doi.org/10.1007/b97543.
[16] M. Fortin and R. Glowinski, *Chapter 1 augmented Lagrangian methods in quadratic programming*, in Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems, Elsevier, New York, 1983, pp. 1–46, https://doi.org/10.1016/s0168-2024(08)70026-2.
[17] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Applied and Numerical Harmonic Analysis, Springer, New York, 2013.
[18] D. Gabay, *Chapter IX. Applications of the method of multipliers to variational inequalities*, in Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems, Elsevier, New York, 1983, pp. 299–331, https://doi.org/10.1016/s0168-2024(08)70034-1.
[19] D. Gabay and B. Mercier, *A dual algorithm for the solution of nonlinear variational problems via finite element approximation*, Comput. Math. Appl., 2 (1976), pp. 17–40, https://doi.org/10.1016/0898-1221(76)90003-1.
[20] R. Glowinski, *On alternating direction methods of multipliers: A historical perspective*, in Computational Methods in Applied Sciences, Springer, Dordrecht, the Netherlands, 2014, pp. 59–82, https://doi.org/10.1007/978-94-017-9054-3_4.
[21] R. Glowinski and A. Marrocco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires*, RAIRA Anal. Number., 9 (1975), pp. 41–76, https://doi.org/10.1051/m2an/197509r200411.
[22] R. Glowinski and P. Tallec, *Augmented Lagrangian and Operator-splitting Methods in Nonlinear Mechanics*, Stud. Appl. Math., SIAM, Philadelphia, 1989.
[23] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer Series in Statistics, Springer, New York, 2013.

[24] B. He, L.-Z. Liao, D. Han, and H. Yang, *A new inexact alternating directions method for monotone variational inequalities*, Math. Program., 92 (2002), pp. 103–118, https://doi.org/10.1007/s101070100280.
[25] B. He, F. Ma, and X. Yuan, *Optimal Linearized Alternating Direction Method of Multipliers for Convex Programming*, e-print, http://www.optimization-online.org/DB_HTML/2017/09/6228.html.
[26] B. He, H.-K. Xu, and X. Yuan, *On the proximal jacobian decomposition of ALM for multiple-block separable convex minimization problems and its relationship to ADMM*, SIAM J. Sci. Comput., 66 (2015), pp. 1204–1217, https://doi.org/10.1007/s10915-015-0060-1.
[27] B. He and H. Yang, *Some convergence properties of a method of multipliers for linearly constrained monotone variational inequalities*, Oper. Res. Lett., 23 (1998), pp. 151–161, https://doi.org/10.1016/s0167-6377(98)00044-3.
[28] B. He and X. Yuan, *On the $O(1/n)$ convergence rate of the Douglas–Rachford alternating direction method*, SIAM J. Numer. Anal., 50 (2012), pp. 700–709, https://doi.org/10.1137/110836936.
[29] B. He and X. Yuan, *On non-ergodic convergence rate of Douglas–Rachford alternating direction method of multipliers*, Numer. Math., 130 (2015), pp. 567–577, https://doi.org/10.1007/s00211-014-0673-6.
[30] Y. Juan, Y. Zhuang, W.-S. Chin, and C.-J. Lin, *Field-aware factorization machines for CTR prediction*, in Proceedings of the 10th ACM Conference on Recommender Systems, ACM Press, Providence, RI, 2016, https://doi.org/10.1145/2959100.2959134.
[31] S. Keerthi and D. DeCoste, *A modified finite Newton method for fast solution of large scale linear svms*, J. Mach. Learn. Res., 6 (2005), pp. 341–361, http://www.jmlr.org/papers/volume6/keerthi05a/keerthi05a.pdf.
[32] D. Lewis, Y. Yang, T. G. Rose, and F. Li, *RCV1: A new benchmark collection for text categorization research*, J. Mach. Learn. Res., 5 (2004), pp. 361–397, http://www.jmlr.org/papers/volume5/lewis04a/lewis04a.pdf.
[33] M. Lustig, D. Donoho, and J. Pauly, *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magnetic Resonance Med., 58 (2007), pp. 1182–1195, https://doi.org/10.1002/mrm.21391.
[34] J. Ma, L. Saul, S. Savage, and G. Voelker, *Identifying suspicious URLs: An application of large-scale online learning*, in Proceedings of the 26th annual international conference on machine learning, ACM, 2009, pp. 681–688, https://doi.org/10.1145/1553374.1553462.
[35] R. Mises and H. Pollaczek-Geiringer, *Praktische verfahren der gleichungsauflsung*, ZAMM Z. Angew. Math. Mech., 9 (1929), pp. 58–77, https://doi.org/10.1002/zamm.19290090105.
[36] R. D. C. Monteiro and B. F. Svaiter, *Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers*, SIAM J. Optim., 23 (2013), pp. 475–507, https://doi.org/10.1137/110849468.
[37] A. Nedic, A. Ozdaglar, and P. Parrilo, *Constrained consensus and optimization in multi-agent networks*, IEEE Trans. Automat. Control, 55 (2010), pp. 922–938, https://doi.org/10.1109/tac.2010.2041686.
[38] J. Neumann, C. Schnrr, and G. Steidl, *Combined SVM-based feature selection and classification*, Mach. Learn., 61 (2005), pp. 129–150, https://doi.org/10.1007/s10994-005-1505-9.
[39] M. K. Ng, F. Wang, and X. Yuan, *Inexact alternating direction methods for image recovery*, SIAM J. Sci. Comput., 33 (2011), pp. 1643–1668, https://doi.org/10.1137/100807697.
[40] C. Paige and M. Saunders, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43–71, https://doi.org/10.1145/355984.355989.
[41] B. Recht, M. Fazel, and P. A. Parrilo, *Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization*, SIAM Rev., 52 (2010), pp. 471–501, https://doi.org/10.1137/070697835.
[42] R. Rockafellar, M. Wets, and R. Wets, *Variational Analysis*, Springer, Berlin, 2009.
[43] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J. R. Stat. Soc. Ser. B. Stat. Methodol. (1996), pp. 267–288, http://www.jstor.org/stable/2346178.
[44] L. Trefethen and D. Bau, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.
[45] R. Varga, *Matrix Iterative Analysis*, Springer, Berlin, 2009.
[46] X. Wang and X. Yuan, *The linearized alternating direction method of multipliers for Dantzig selector*, SIAM J. Sci. Comput., 34 (2012), pp. A2792–A2811, https://doi.org/10.1137/110833543.
[47] M. Woodbury, *Inverting Modified Matrices*, Memorandum Report 42, Statistical Research Group, Princeton University, Princeton, NJ, 1950.

[48] J. Yang and X. Yuan, *Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization*, Math. Comp., 82 (2013), pp. 301–329, https://doi.org/10.1090/s0025-5718-2012-02598-1.

[49] M. Yuan and Y. Lin, *Model selection and estimation in regression with grouped variables*, J. R. Stat. Soc. Ser. B. Stat. Methodol., 68 (2006), pp. 49–67, https://doi.org/10.1111/j.1467-9868.2005.00532.x.

[50] X. Yuan, *The improvement with relative errors of He et al.'s inexact alternating direction method for monotone variational inequalities*, Math. Comput. Model., 42 (2005), pp. 1225–1236, https://doi.org/10.1016/j.mcm.2005.04.007.

[51] M. Zhu and S. Martinez, *On distributed convex optimization under inequality and equality constraints*, IEEE Trans. Automat. Control, 57 (2012), pp. 151–164, https://doi.org/10.1109/tac.2011.2167817.