IMA Journal of Numerical Analysis (2015) **35**, 394–426
doi:10.1093/imanum/drt060
Advance Access publication on January 3, 2014

A splitting method for separable convex programming

Bingsheng He
International Centre of Management Science and Engineering, School of Management and Engineering, and Department of Mathematics, Nanjing University, Nanjing, China
hebma@nju.edu.cn

Min Tao
Department of Mathematics, Nanjing University, Nanjing 210093, China
taom@njupt.edu.cn

and

Xiaoming Yuan*
Department of Mathematics, Hong Kong Baptist University, Hong Kong, China
*Corresponding author: xmyuan@hkbu.edu.hk

[Received on 15 April 2011; revised on 1 November 2013]

We propose a splitting method for solving a separable convex minimization problem with linear constraints, where the objective function is expressed as the sum of m individual functions without coupled variables. Treating the functions in the objective separately, the new method belongs to the category of operator splitting methods. We show the global convergence and estimate a worst-case convergence rate for the new method, and then illustrate its numerical efficiency by some applications.

Keywords: convex programming; separable structure; operator splitting methods; image processing.

1. Introduction

We consider a separable convex minimization problem with linear constraints and its objective function is expressed as the sum of m individual functions without coupled variables:

$$\min\left\{\sum_{i=1}^{m}\theta_i(x_i)\;\middle|\;\sum_{i=1}^{m}A_ix_i=b;\ x_i\in\mathcal{X}_i,\ i=1,2,\ldots,m\right\},\tag{1.1}$$

where $\theta_i:\Re^{n_i}\rightarrow\Re$ $(i=1,2,\ldots,m)$ are closed proper convex functions (not necessarily smooth); $A_i\in\Re^{l\times n_i}$ $(i=1,2,\ldots,m)$; $\mathcal{X}_i\subseteq\Re^{n_i}$ $(i=1,2,\ldots,m)$ are nonempty closed convex sets; $b\in\Re^l$ and $\sum_{i=1}^m n_i=n$. Throughout, the solution set of (1.1) is assumed to be nonempty and A_i's $(i=1,\ldots,m)$ are assumed to be full column-rank.

In the literature, operator splitting methods for the special case of (1.1) with $m=2$ have been well studied, and the most popular method perhaps is the alternating direction method of multipliers (ADMM) proposed originally in Glowinski & Marrocco (1975) (see also Gabay & Mercier, 1976). More specifically, for solving the special case of (1.1) with $m=2$

$$\min\{\theta_1(x_1)+\theta_2(x_2)\,|\,A_1x_1+A_2x_2=b,\ x_1\in\mathcal{X}_1,\ x_2\in\mathcal{X}_2\},\tag{1.2}$$

the iterative scheme of ADMM is

$$\begin{cases} x_1^{k+1} = \text{Argmin}\{\theta_1(x_1) - (\lambda^k)^{\mathrm{T}}(A_1x_1) + \frac{1}{2}\|A_1x_1 + A_2x_2^k - b\|_H^2 \mid x \in \mathcal{X}_1\}; \\ x_2^{k+1} = \text{Argmin}\{\theta_2(x_2) - (\lambda^k)^{\mathrm{T}}(A_2x_2) + \frac{1}{2}\|A_1x_1^{k+1} + A_2x_2 - b\|_H^2 \mid x_2 \in \mathcal{X}_2\}; \\ \lambda^{k+1} = \lambda^k - H(A_1x_1^{k+1} + A_2x_2^{k+1} - b), \end{cases} \tag{1.3}$$

where λ^k is the Lagrange multiplier and $H \in \Re^{l \times l}$ is a positive definite matrix playing the role of a penalty parameter. In practice, we can simply take H to be a diagonal matrix. We refer the reader to Boyd *et al.* (2010) and references therein for the history of ADMM and its impressive applications exploited recently. The scheme (1.3) shows that the idea of ADMM is to split the augmented Lagrangian function of (1.2) in the Gauss–Seidel fashion, and thus to minimize the variables x_1 and x_2 separately in alternating order. This splitting strategy makes it possible to exploit θ_i's properties individually, and the resulting subproblems of ADMM are often simple enough to have closed-form solutions or can be solved efficiently up to high precisions for many applications.

In addition to the special case with $m = 2$, we are interested in the general case of (1.1) with $m \geqslant 3$ (see Kiwiel *et al.*, 1999; Tibshirani *et al.*, 2005; Setzer *et al.*, 2010; Tao & Yuan, 2011 for some applications), i.e., the objective of (1.1) consists of more than two individual functions, and we want to develop a splitting method analogous to ADMM such that these functions can be treated separately. An immediate idea for this purpose is to extend the scheme (1.3) directly, resulting in an ADMM-like scheme

$$\begin{cases} x_i^{k+1} = \text{Argmin}\{\theta_i(x_i) - (\lambda^k)^{\mathrm{T}} p_i(x_i) + \frac{1}{2}\|p_i(x_i)\|_H^2 \mid x_i \in \mathcal{X}_i\}, \quad i = 1, 2, \ldots, m; \\ \lambda^{k+1} = \lambda^k - H\left(\displaystyle\sum_{j=1}^{m} A_j x_j^{k+1} - b\right), \end{cases} \tag{1.4}$$

where

$$p_i(x_i) = \sum_{j=1}^{i-1} A_j x_j^{k+1} + A_i x_i + \sum_{j=i+1}^{m} A_j x_j^k - b, \quad i = 1, \ldots, m.$$

The convergence of this direct extension (1.4), however, is not clear yet, even though its numerical efficiency has been verified empirically by some recent applications (see, e.g., Peng *et al.*; Tao & Yuan, 2011). This lack of convergence has recently inspired some ADMM-based efforts in the prediction-correction fashion, whose main idea is to generate a new iterate by correcting the output of (1.4) with some correction steps; see, e.g., Han *et al.* and He *et al.* Our purpose in this paper is to develop a splitting method for (1.1) with proved convergence while without any correction step; and meanwhile, its decomposed subproblems are no more difficult to solve than those in (1.4).

The rest of the paper is organized as follows. In Section 2, we provide some preliminary results which are useful for further discussion. In Section 3, we present the new method followed by some remarks. Some theoretical properties useful for further analysis are proved in Section 4. We then analyse the convergence of the new method in Section 5 and further discuss it under weaker assumptions in Section 6. After that, we analyse the convergence rate for the new method in Section 7. In Section 8, we report some numerical results to verify the efficiency of the new method. Finally, some conclusions are made in Section 9.

2. Preliminaries

In this section, we summarize some basic definitions and related properties that will be used in later analysis.

2.1 *Variational characterization*

Let $\mathcal{W} := \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_m \times \Re^l$. By deriving its optimality condition, it is easy to see that (1.1) is equivalent to finding $w^* = (x_1^*, x_2^*, \ldots, x_m^*, \lambda^*) \in \mathcal{W}$ such that

$$\begin{cases} \theta_1(x_1) - \theta_1(x_1^*) + (x_1 - x_1^*)^{\mathrm{T}}(-A_1^{\mathrm{T}}\lambda^*) \geqslant 0, \\ \theta_2(x_2) - \theta_2(x_2^*) + (x_2 - x_2^*)^{\mathrm{T}}(-A_2^{\mathrm{T}}\lambda^*) \geqslant 0, \\ \qquad\vdots \\ \theta_m(x_m) - \theta_m(x_m^*) + (x_m - x_m^*)^{\mathrm{T}}(-A_m^{\mathrm{T}}\lambda^*) \geqslant 0, \\ \qquad \sum_{i=1}^m A_i x_i^* - b = 0, \end{cases} \quad \forall w = (x_1, x_2, \ldots, x_m, \lambda) \in \mathcal{W}, \tag{2.1}$$

or, in a more compact form

$$\mathrm{VI}(\mathcal{W}, F, \theta) \quad \theta(x) - \theta(x^*) + (w - w^*)^{\mathrm{T}} F(w^*) \geqslant 0 \quad \forall w \in \mathcal{W}, \tag{2.2a}$$

where

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix}, \quad \theta(x) = \sum_{i=1}^m \theta_i(x_i), \quad w = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \\ \lambda \end{pmatrix} \quad \text{and} \quad F(w) = \begin{pmatrix} -A_1^{\mathrm{T}}\lambda \\ -A_2^{\mathrm{T}}\lambda \\ \vdots \\ -A_m^{\mathrm{T}}\lambda \\ \sum_{i=1}^m A_i x_i - b \end{pmatrix}. \tag{2.2b}$$

Note that x collects all the primal variables in (1.1) and it is a subvector of w. Obviously, we have the following lemma regarding $F(w)$ defined above. We omit its proof since it is trivial.

LEMMA 2.1 The mapping $F(w)$ defined in (2.2b) satisfies

$$(w' - w)^{\mathrm{T}}(F(w') - F(w)) = 0 \quad \forall w', w \in \Re^{n+l}. \tag{2.3}$$

Under the nonempty assumption on the solution set of (1.1), the solution set of $\mathrm{VI}(\mathcal{W}, F, \theta)$, which is denoted by $\mathcal{W}^*$ from now on, is also nonempty and convex (see Facchinei & Pang, 2003, Theorem 2.3.5). Moreover, the following theorem provides a characterization on $\mathcal{W}^*$, and it is inspired by Facchinei & Pang (2003, Theorem 2.3.5). Since its proof is almost the same as that of He & Yuan (2012, Theorem 2.1), we omit it.

THEOREM 2.2 The solution set of $\mathrm{VI}(\mathcal{W}, F, \theta)$ is convex and it can be characterized as

$$\mathcal{W}^* = \bigcap_{w \in \mathcal{W}} \{\tilde{w} \in \mathcal{W} : \theta(x) - \theta(\tilde{x}) + (w - \tilde{w})^{\mathrm{T}} F(w) \geqslant 0\}. \tag{2.4}$$

The identities summarized in the following lemma are useful in our analysis. We omit their proofs which are very elementary.

LEMMA 2.3 Let $U \in \Re^{n\times n}$ be symmetric and positive definite. Then we have

$$(a-b)^{\mathrm{T}}U(c-d) = \tfrac{1}{2}(\|a-d\|_U^2 - \|a-c\|_U^2) + \tfrac{1}{2}(\|c-b\|_U^2 - \|d-b\|_U^2) \quad \forall a,b,c,d \in \Re^n; \tag{2.5}$$

and

$$\|a\|_U^2 - \|b\|_U^2 = 2a^{\mathrm{T}}U(a-b) - \|a-b\|_U^2 \quad \forall a,b \in \Re^n. \tag{2.6}$$

2.2 *Some notations*

Then, we define some matrices which will simplify our notations significantly in later analysis. For $m \geqslant 3$, let a block diagonal matrix be defined as

$$G = \mathrm{diag}\{\mu A_2^{\mathrm{T}}HA_2, \mu A_3^{\mathrm{T}}HA_3, \ldots, \mu A_m^{\mathrm{T}}HA_m, H^{-1}\} \tag{2.7}$$

and two more matrices be defined as

$$M = \begin{pmatrix} \mu & 0 & \cdots & 0 & 1 \\ 0 & \mu & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & 1 \\ 0 & \cdots & 0 & \mu & 1 \\ 1 & \cdots & 1 & 1 & 1 \end{pmatrix}_{m\times m} \quad \text{and} \quad N = \begin{pmatrix} I & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & I & 0 \\ -HA_2 & \cdots & -HA_m & I \end{pmatrix}_{(\sum_{i=2}^m n_i + l)\times(\sum_{i=2}^m n_i + l)}, \tag{2.8}$$

where $\mu > 0$ is a positive constant and $H \in \Re^{l\times l}$ is a positive definite matrix. Note that the matrix G defined in (2.7) is positive definite under the assumptions that A_i's $(i = 2, \ldots, m)$ are of full column-rank and H is positive definite.

Moreover, we introduce some useful notations for the convenience of further analysis. Revisiting the iterative schemes of ADMM (1.3), it is easy to observe that $(x_2^{k+1}, \lambda^{k+1})$ is a function of (x_2^k, λ^k), and the variable x_1^k is not a part of the iteration. Thus, x_1 is called an intermediate variable in Boyd *et al.* (2010). For our new method to be proposed, x_1 is again such an intermediate variable. We thus introduce the notations $v = (x_2, \ldots, x_m, \lambda)$ and $\mathcal{V} = \mathcal{X}_2 \times \cdots \times \mathcal{X}_m \times \Re^l$ to differentiate the variables which are truly involved in the iteration from the intermediate variable. Accordingly, $v^k := (x_2^k, \ldots, x_m^k, \lambda^k)$ and

$$\mathcal{V}^* = \{(x_2^*, \ldots, x_m^*, \lambda^*) \mid (x_1^*, x_2^*, \ldots, x_m^*, \lambda^*) \in \mathcal{W}^*\}.$$

Finally, we summarize some facts regarding the matrices defined in (2.7) and (2.8) in a lemma.

LEMMA 2.4 Let the matrices G and N be defined in (2.7) and (2.8), respectively. We have

$$v^{\mathrm{T}}GNv = \tfrac{1}{2}v^{\mathrm{T}}(GN + N^{\mathrm{T}}G)v \quad \forall v \in \mathcal{V}; \tag{2.9}$$

and the matrix defined as

$$GN + N^{\mathrm{T}}G - N^{\mathrm{T}}GN$$

is positive semidefinte if $\mu \geqslant m-1$.

Proof. The assertion (2.9) is trivial. We thus omit the proof. For the second assertion, we note that

$$\begin{aligned} GN + N^{\mathrm{T}}G - N^{\mathrm{T}}GN &= G - (I - N^{\mathrm{T}})G(I - N) \\ &= \operatorname{diag}\{\mu A_2^{\mathrm{T}}HA_2, \mu A_3^{\mathrm{T}}HA_3, \ldots, \mu A_m^{\mathrm{T}}HA_m, H^{-1}\} \\ &\quad - \begin{pmatrix} A_2^{\mathrm{T}} \\ \vdots \\ A_m^{\mathrm{T}} \\ 0 \end{pmatrix} H(A_2, \ldots, A_m, 0). \end{aligned}$$

Thus, its positive semidefiniteness is an immediate conclusion if $\mu \geqslant m - 1$. □

3. The new method

In this section, we present a new splitting method for solving (1.1) with $m \geqslant 3$ and give some remarks.

Let $H \in \Re^{l\times l}$ be a positive definite matrix and $\mu > m - 1$ be a constant. In particular, we can simply take $H = \operatorname{diag}\{\beta_1, \ldots, \beta_l\}$, where β_i's are positive constants.

Algorithm: A new splitting method for solving (1.1)

Step 0. Choose an initial iterate $v^0 := (x_2^0, \ldots, x_m^0, \lambda^0) \in \mathcal{X}_2 \times \cdots \mathcal{X}_m \times \Re^l$ arbitrarily and generate the new iterate via the following scheme.

Step 1. Find x_1^{k+1} such that

$$x_1^{k+1} = \operatorname{Argmin}\left\{\theta_1(x_1) - (\lambda^k)^{\mathrm{T}}A_1x_1 + \frac{1}{2}\left\|A_1x_1 + \sum_{i=2}^{m} A_ix_i^k - b\right\|_H^2 \,\middle|\, x_1 \in \mathcal{X}_1\right\}. \tag{3.1}$$

Step 2. Update the Lagrange multiplier with x_1^{k+1}:

$$\tilde{\lambda}^k = \lambda^k - H\left(A_1x_1^{k+1} + \sum_{i=2}^{m} A_ix_i^k - b\right). \tag{3.2}$$

Step 3. Find x_i^{k+1} $(i = 2, \ldots, m)$ (if possible, simultaneously) such that

$$x_i^{k+1} = \operatorname{Argmin}\{\theta_i(x_i) - (\tilde{\lambda}^k)^{\mathrm{T}}A_ix_i + \frac{\mu}{2}\|A_i(x_i - x_i^k)\|_H^2 \mid x_i \in \mathcal{X}_i\}. \tag{3.3}$$

Step 4. Update the Lagrange multiplier with x_i^{k+1} $(i = 1, 2, \ldots, m)$:

$$\lambda^{k+1} = \lambda^k - H\left(\sum_{i=1}^{m} A_ix_i^{k+1} - b\right). \tag{3.4}$$

Below we give some remarks relevant to the proposed method.

Remark 3.1 The proposed method is related to some pre-existing methods. For example, for the case $m = 1$, the proposed method with $H = \beta \cdot I_l$, where I_l denotes the identity matrix in $\Re^{l\times l}$ is exactly the classical augmented Lagrangian method (Hestenes, 1969; Powell, 1969). Moreover, for the case $m = 2$, it is easy to verify that the proposed method approaches the ADMM (1.3) as $\mu \to 1$. To see why, first note that (3.1) is the same as the x_1-subproblem in (1.3). Then, for this special case, since (3.2) and (3.3) are specified as

$$\tilde{\lambda}^k = \lambda^k - H(A_1 x_1^{k+1} + A_2 x_2^k - b)$$

and

$$x_2^{k+1} = \operatorname{Argmin}\{\theta_2(x_2) - (\tilde{\lambda}^k)^{\mathrm{T}}(A_2 x_2) + \tfrac{1}{2}\|A_2(x_2 - x_2^k)\|_H^2 \mid x_2 \in \mathcal{X}_2\},$$

respectively, by combining these two facts we have

$$x_2^{k+1} = \operatorname{Argmin}\{\theta_2(x_2) - (\lambda^k)^{\mathrm{T}}(A_2 x_2) + \tfrac{1}{2}\|A_1 x_1^{k+1} + A_2 x_2 - b\|_H^2 \mid x_2 \in \mathcal{X}_2\},$$

which is exactly the x_2-subproblem in (1.3). In addition, the so-named variant alternating splitting augmented Lagrangian method (VASALM) in Tao & Yuan (2011) is also a special case of the proposed method with matrix variables when $m = 3$.

Remark 3.2 We require $\mu > m - 1$ in the proposed method, and it is essentially for the purpose of ensuring

$$\|v^k - v^{k+1}\|_G^2 + 2(\lambda^k - \lambda^{k+1})^{\mathrm{T}}\left(\sum_{i=2}^m A_i(x_i^k - x_i^{k+1})\right) \geqslant c_0\|v^k - v^{k+1}\|_G^2, \tag{3.5}$$

where c_0 is a certain positive constant and G is defined in (2.7); see Lemma 5.2 in Section 5.

4. Some properties

In this section, we prove some theoretical properties for the sequence generated by the proposed method, which are useful for later analysis of establishing the convergence and estimating a convergence rate.

The following lemma follows directly from the first-order optimality conditions of the subproblems in the proposed method.

Lemma 4.1 Let $\{w^k\}$ be generated by the proposed method. Then, we have $x_1^{k+1} \in \mathcal{X}_1$ such that

$$\theta_1(x_1) - \theta_1(x_1^{k+1}) + (x_1 - x_1^{k+1})^{\mathrm{T}}(-A_1^{\mathrm{T}}\tilde{\lambda}^k) \geqslant 0 \quad \forall x_1 \in \mathcal{X}_1; \tag{4.1}$$

and $x_i^{k+1} \in \mathcal{X}_i$ $(i = 2, \ldots, m)$ such that

$$\theta_i(x_i) - \theta_i(x_i^{k+1}) + (x_i - x_i^{k+1})^{\mathrm{T}}\{-A_i^{\mathrm{T}}\tilde{\lambda}^k + \mu A_i^{\mathrm{T}} H A_i(x_i^{k+1} - x_i^k)\} \geqslant 0 \quad \forall x_i \in \mathcal{X}_i. \tag{4.2}$$

Proof. According to the optimality condition of the x_1-subproblem (3.1), we have $x_1^{k+1} \in \mathcal{X}_1$ such that

$$\theta_1(x_1) - \theta_1(x_1^{k+1}) + (x_1 - x_1^{k+1})^{\mathrm{T}}\left\{-A_1^{\mathrm{T}}\left(\lambda^k - H\left(A_1 x_1^{k+1} + \sum_{i=2}^m A_i x_i^k - b\right)\right)\right\} \geqslant 0 \quad \forall x_1 \in \mathcal{X}_1.$$

Substituting (3.2) into the last inequality, we obtain the assertion (4.1). The second assertion (4.2) follows from the optimality condition of the x_i-subproblem (3.3) directly. □

Proof of the following lemma is trivial, and we omit it.

LEMMA 4.2 For the iterate generated by the proposed method, we have

$$\lambda^{k+1} - \tilde{\lambda}^k = H\left(\sum_{i=2}^{m} A_i(x_i^k - x_i^{k+1})\right) \tag{4.3}$$

and

$$\sum_{i=1}^{m} A_i x_i^{k+1} - b = H^{-1}(\lambda^k - \lambda^{k+1}). \tag{4.4}$$

For convenience of further analysis, we use the sequence $\{w^k\}$ generated by the proposed method to construct an auxiliary sequence as

$$\tilde{w}^k = \begin{pmatrix} \tilde{x}_1^k \\ \tilde{x}_2^k \\ \vdots \\ \tilde{x}_m^k \\ \tilde{\lambda}^k \end{pmatrix} = \begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \\ \vdots \\ x_m^{k+1} \\ \lambda^k - H\left(A_1 x_1^{k+1} + \sum_{i=2}^{m} A_i x_i^k - b\right) \end{pmatrix}. \tag{4.5}$$

Consequently, the notation $\tilde{v}^k = (\tilde{x}_2^k, \ldots, \tilde{x}_m^k, \tilde{\lambda}^k)$ is clear. The notations $\tilde{w}^k$ and $\tilde{v}^k$ are only used in our theoretical analysis. First, we can establish a useful identity by using the notation of $\tilde{v}^k$.

LEMMA 4.3 Let $\{w^k\}$ be generated by the proposed method and $\{\tilde{w}^k\}$ be given in (4.5). Then we have

$$\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2 = \mu \sum_{i=2}^{m} \|A_i(x_i^k - \tilde{x}_i^k)\|_H^2 + \|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2 - \left\|\sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k)\right\|_H^2, \tag{4.6}$$

where G is defined in (2.7).

Proof. First, recall $\tilde{x}_i^k = x_i^{k+1}$ $(i = 2, \ldots, m)$. We thus have

$$\|\tilde{v}^k - v^{k+1}\|_G = \|\tilde{\lambda}^k - \lambda^{k+1}\|_{H^{-1}}^2 \overset{(4.3)}{=} \left\|H \sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k)\right\|_{H^{-1}}^2 = \left\|\sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k)\right\|_H^2$$

and

$$\|\tilde{v}^k - v^k\|_G^2 = \mu \sum_{i=2}^{m} \|A_i(x_i^k - \tilde{x}_i^k)\|_H^2 + \|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2.$$

From the above two equations, the assertion (4.6) follows directly. □

Recall the characterization of $\mathcal{W}^*$ in (2.4). The following lemma reflects the discrepancy of $\tilde{w}^k$ from a solution point in $\mathcal{W}^*$.

LEMMA 4.4 Let $\{w^k\}$ be generated by the proposed method and $\{\tilde{w}^k\}$ be given in (4.5). We have

$$\tilde{w}^k \in \mathcal{W}, \quad \theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}} F(\tilde{w}^k) + (v - \tilde{v}^k)^{\mathrm{T}} GN(\tilde{v}^k - v^k) \geqslant 0 \quad \forall w \in \mathcal{W}, \tag{4.7}$$

where G and N are defined in (2.7) and (2.8), respectively.

Proof. First, it follows from (3.2) and $x_1^{k+1} = \tilde{x}_1^k$ that

$$(\lambda - \tilde{\lambda}^k)^{\mathrm{T}} \left\{ (A_1\tilde{x}_1^k + \cdots + A_m\tilde{x}_m^k - b) - \sum_{i=2}^m A_i(\tilde{x}_i^k - x_i^k) + H^{-1}(\tilde{\lambda}^k - \lambda^k) \right\} \geqslant 0 \quad \forall \lambda \in \Re^l.$$

Rewriting the inequalities (4.1–4.2) together and substituting x_i^{k+1} with $\tilde{x}_i^k$ $(i = 1, \ldots, m)$, we have

$$\begin{cases} \theta_1(x_1) - \theta(\tilde{x}_1) + (x_1 - \tilde{x}_1^k)^{\mathrm{T}}\{-A_1^{\mathrm{T}}\tilde{\lambda}^k\} \geqslant 0, \\ \theta_2(x_2) - \theta_2(\tilde{x}_2) + (x_2 - \tilde{x}_2^k)^{\mathrm{T}}\{-A_2^{\mathrm{T}}\tilde{\lambda}^k + \mu A_2^{\mathrm{T}} H A_2(\tilde{x}_2^k - x_2^k)\} \geqslant 0, \\ \quad \cdots \quad \cdots \\ \theta(x_m) - \theta(\tilde{x}_m) + (x_m - \tilde{x}_m^k)^{\mathrm{T}}\{-A_m^{\mathrm{T}}\tilde{\lambda}^k + \mu A_m^{\mathrm{T}} H A_m(\tilde{x}_m^k - x_m^k)\} \geqslant 0, \\ (\lambda - \tilde{\lambda}^k)^{\mathrm{T}} \left\{ (A_1\tilde{x}_1^k + \cdots + A_m\tilde{x}_m^k - b) - \sum_{i=2}^m A_i(\tilde{x}_i^k - x_i^k) + H^{-1}(\tilde{\lambda}^k - \lambda^k) \right\} \geqslant 0, \end{cases} \quad \forall w \in \mathcal{W}. \tag{4.8}$$

Adding all these inequalities together and using the definitions of F (2.2b), G (2.7) and N (2.8), the assertion (4.7) follows immediately. □

Based on (2.4), Lemma 4.4 thus indicates that the proximity of $\tilde{w}^k$ to a solution point in $\mathcal{W}^*$ is measured by the term $(v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k)$. Hence, we are interested in estimating this term more precisely. In particular, we express this cross term by the difference of some quadratic terms in the following lemma. Preceding the proof, we notice the relationship

$$v^{k+1} = v^k - N(v^k - \tilde{v}^k), \tag{4.9}$$

which is obvious based on the definitions of N in (2.8) and $\tilde{w}^k$ in (4.5).

LEMMA 4.5 Let $\{w^k\}$ be generated by the proposed method and $\{\tilde{w}^k\}$ be given in (4.5). Then we have

$$\begin{aligned} &(v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k) + \tfrac{1}{2}(\|v - v^k\|_G^2 - \|v - v^{k+1}\|_G^2) \\ &\quad = \tfrac{1}{2}(\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2) \quad \forall v \in \mathcal{V}, \end{aligned} \tag{4.10}$$

where G is defined in (2.7).

Proof. By using the relationship in (4.9), it follows that

$$(v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k) = (v - \tilde{v}^k)^{\mathrm{T}} G(v^k - v^{k+1}).$$

Because of Lemma 2.3, we have the identity

$$(v - \tilde{v}^k)^{\mathrm{T}} G(v^k - v^{k+1}) = \tfrac{1}{2}(\|v - v^{k+1}\|_G^2 - \|v - v^k\|_G^2) + \tfrac{1}{2}(\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2).$$

Adding the above two equations and rearranging them, we obtain the assertion (4.10) and the proof is complete. □

Lemmas 4.4 and 4.5 enable us to derive some very useful conclusions for establishing both the convergence and a convergence rate for the proposed method.

THEOREM 4.6 Let $\{w^k\}$ be generated by the proposed method and $\{\tilde{w}^k\}$ be given in (4.5). Then, we have

$$\theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}} F(\tilde{w}^k) + \tfrac{1}{2}(\|v - v^k\|_G^2 - \|v - v^{k+1}\|_G^2) \geqslant 0 \quad \forall w \in \mathcal{W} \tag{4.11}$$

and

$$\theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}} F(w) + \tfrac{1}{2}(\|v - v^k\|_G^2 - \|v - v^{k+1}\|_G^2) \geqslant 0 \quad \forall w \in \mathcal{W}. \tag{4.12}$$

Proof. First, it follows from (4.7) that

$$\theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}} F(\tilde{w}^k) \geqslant (v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k) \quad \forall w \in \mathcal{W}. \tag{4.13}$$

On the other hand, by using the Cauchy–Schwarz inequality and $\mu > m - 1$, we have

$$\mu \sum_{i=2}^{m} \|A_i(x_i^k - \tilde{x}_i^k)\|_H^2 \geqslant (m-1) \cdot \sum_{i=2}^{m} \|A_i(x_i^k - \tilde{x}_i^k)\|_H^2 \geqslant \left\| \sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k) \right\|_H^2 .$$

Substituting into the right-hand side of (4.6), we obtain

$$\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2 \geqslant \|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2,$$

and consequently it follows from (4.10) that

$$(v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k) + \tfrac{1}{2}(\|v - v^k\|_G^2 - \|v - v^{k+1}\|_G^2) \geqslant 0 \quad \forall v \in \mathcal{V}. \tag{4.14}$$

Adding (4.13) and (4.14), the assertion (4.11) is proved. For the second assertion, because of (2.3) in Lemma 2.1, we have

$$(w - \tilde{w}^k)^{\mathrm{T}} F(w) = (w - \tilde{w}^k)^{\mathrm{T}} F(\tilde{w}^k) \quad \forall w \in \mathcal{W}.$$

Adding (4.11) and the above inequality, the assertion (4.12) follows immediately and the theorem is proved. □

5. Convergence

In this section, we establish the convergence for the proposed method. First of all, we have to prove that the condition $\mu > m - 1$ suffices to guarantee (3.5). For this purpose, we first prove a lemma.

LEMMA 5.1 For $m \geqslant 2$, let the $m \times m$ symmetric matrix M be defined in (2.8). Then, it has $(m-2)$ multiple eigenvalues equivalent to μ, i.e.,

$$\nu_1 = \nu_2 = \cdots = \nu_{m-2} = \mu,$$

and the other two eigenvalues are given by

$$\nu_{m-1} = \frac{(\mu+1) + \sqrt{(\mu+1)^2 + 4((m-1)-\mu)}}{2}$$

and

$$\nu_m = \frac{(\mu+1) - \sqrt{(\mu+1)^2 + 4((m-1)-\mu)}}{2}.$$

Proof. Let $e \in \Re^{m-1}$ be the vector whose all elements are 1. Thus, we can rewrite M (2.8) into

$$M = \begin{pmatrix} \mu I_{m-1} & e \\ e^{\mathrm{T}} & 1 \end{pmatrix}.$$

Without loss of generality, we assume that the eigenvectors of M have the forms

$$z = \begin{pmatrix} y \\ 0 \end{pmatrix} \quad \text{or} \quad z = \begin{pmatrix} y \\ 1 \end{pmatrix},$$

where $y \in \Re^{m-1}$. For the first case, we have

$$\begin{cases} \mu y = \nu y, \\ e^{\mathrm{T}} y = 0. \end{cases} \tag{5.1}$$

It is clear that the following vectors in $\Re^{m-1}$:

$$y^1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix}, \quad y^2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \quad \cdots \quad y^{m-2} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ -1 \end{pmatrix}$$

are linearly independent and they satisfy (5.1) with $\nu = \mu$. Thus,

$$z^i = \begin{pmatrix} y^i \\ 0 \end{pmatrix}, \quad i = 1, \ldots, m-2,$$

are eigenvectors of M and the related eigenvalues are

$$\nu_1 = \nu_2 = \cdots = \nu_{m-2} = \mu.$$

For the second case, $z^{\mathrm{T}} = (y^{\mathrm{T}}, 1)$, we have

$$\begin{cases} \mu y + e = \nu y, \\ e^{\mathrm{T}} y + 1 = \nu. \end{cases} \tag{5.2}$$

It follows from (5.2) that

$$(\nu - \mu)(\nu - 1) - (m - 1) = 0,$$

and thus the assertion is proved. □

LEMMA 5.2 The condition (3.5) is ensured if $\mu > m - 1$.

Proof. First, we notice the following equations:

$$\|v^k - v^{k+1}\|_G^2 + 2(\lambda^k - \lambda^{k+1})^{\mathrm{T}} \left(\sum_{i=2}^{m} A_i (x_i^k - x_i^{k+1}) \right)$$

$$= \begin{pmatrix} H^{1/2} A_2 (x_2^k - x_2^{k+1}) \\ H^{1/2} A_3 (x_3^k - x_3^{k+1}) \\ \vdots \\ H^{1/2} A_m (x_m^k - x_m^{k+1}) \\ H^{-1/2} (\lambda^k - \lambda^{k+1}) \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \mu I_l & 0 & \cdots & 0 & I_l \\ 0 & \mu I_l & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & I_l \\ 0 & \cdots & 0 & \mu I_l & I_l \\ I_l & \cdots & I_l & I_l & I_l \end{pmatrix} \begin{pmatrix} H^{1/2} A_2 (x_2^k - x_2^{k+1}) \\ H^{1/2} A_3 (x_3^k - x_3^{k+1}) \\ \vdots \\ H^{1/2} A_m (x_m^k - x_m^{k+1}) \\ H^{-1/2} (\lambda^k - \lambda^{k+1}) \end{pmatrix}$$

and

$$\|v^k - v^{k+1}\|_G^2 = \begin{pmatrix} H^{1/2} A_2 (x_2^k - x_2^{k+1}) \\ H^{1/2} A_3 (x_3^k - x_3^{k+1}) \\ \vdots \\ H^{1/2} A_m (x_m^k - x_m^{k+1}) \\ H^{-1/2} (\lambda^k - \lambda^{k+1}) \end{pmatrix}^{\mathrm{T}} \cdot \operatorname{diag}\{\mu I_l, \ldots, \mu I_l, I_l\} \cdot \begin{pmatrix} H^{1/2} A_2 (x_2^k - x_2^{k+1}) \\ H^{1/2} A_3 (x_3^k - x_3^{k+1}) \\ \vdots \\ H^{1/2} A_m (x_m^k - x_m^{k+1}) \\ H^{-1/2} (\lambda^k - \lambda^{k+1}) \end{pmatrix}.$$

Therefore, the requirement (3.5) is satisfied when the $ml \times ml$ matrix

$$\tilde{M} = \begin{pmatrix} \mu I_l & 0 & \cdots & 0 & I_l \\ 0 & \mu I_l & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & I_l \\ 0 & \cdots & 0 & \mu I_l & I_l \\ I_l & \cdots & I_l & I_l & I_l \end{pmatrix}_{ml \times ml} \tag{5.3}$$

is positive definite.

Note that the matrix $\tilde{M}$ has the same largest (respectively, smallest) eigenvalues as the $m \times m$ symmetric matrix M defined in (2.8). For $m \geqslant 2$, it is easy to verify that

$$\nu_m = \frac{(\mu+1) - \sqrt{(\mu+1)^2 + 4((m-1)-\mu)}}{2} \tag{5.4}$$

is the smallest eigenvalue of M. In addition, $\nu_m > 0$ if and only if $\mu > (m-1)$. Therefore, $\mu > m-1$ is sufficient to guarantee the requirement (3.5) with the constant $c_0 := \nu_m/\rho(G) > 0$, where $\rho(G)$ denotes the spectrum radius of G. □

Now, we start to prove the convergence of the proposed method. The proof follows the standard framework of contraction-type methods in Blum & Oettli (1975). With Lemmas 4.4 and 4.5, we can show that the sequence $\{v^k\}$ generated by the proposed methods is contractive with respect to $\mathcal{V}^*$ under the G-norm.

THEOREM 5.3 Let w^{k+1} be generated by the proposed method and c_0 be the constant satisfying (3.5). Then we have

$$\|v^{k+1} - v^*\|_G^2 \leqslant \|v^k - v^*\|_G^2 - c_0\|v^k - v^{k+1}\|_G^2 \quad \forall v^* \in \mathcal{V}^*, \tag{5.5}$$

where G, $v^k = (x_2^k, \dots, x_m^k, \lambda^k)$ and $\mathcal{V}^*$ are defined in Section 2.2.

Proof. Note (4.10) is true for any $v \in \mathcal{V}$. Let $w^* \in \mathcal{W}^*$. Setting $v = v^*$, we obtain

$$\|v^k - v^*\|_G^2 - \|v^{k+1} - v^*\|_G^2 = \|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2 + 2(\tilde{v}^k - v^*)^{\mathrm{T}} GN(v^k - \tilde{v}^k). \tag{5.6}$$

On the other hand, setting $w = w^*$ in (4.7), we obtain

$$(\tilde{v}^k - v^*)^{\mathrm{T}} GN(v^k - \tilde{v}^k) \geqslant \theta(\tilde{x}^k) - \theta(x^*) + (\tilde{w}^k - w^*)^{\mathrm{T}} F(\tilde{w}^k). \tag{5.7}$$

Since $w^* \in \mathcal{W}^*$, according to (2.2a), we have

$$\theta(\tilde{x}^k) - \theta(x^*) + (\tilde{w}^k - w^*)^{\mathrm{T}} F(w^*) \geqslant 0.$$

Recall (2.3) in Lemma 2.1. We thus conclude that the right-hand side of (5.7) is non-negative. Therefore, (5.6) and (5.7) imply that

$$\|v^k - v^*\|_G^2 - \|v^{k+1} - v^*\|_G^2 \geqslant \|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2. \tag{5.8}$$

Note that $\tilde{x}_i^k = x_i^{k+1}$ $(i = 2, \dots, m)$ and by using (4.6), we obtain

$$\begin{aligned} &\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2 \\ &\quad = \mu \sum_{i=2}^m \|A_i(x_i^k - \tilde{x}_i^k)\|_H^2 + \|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2 - \left\| \sum_{i=2}^m A_i(x_i^k - \tilde{x}_i^k) \right\|_H^2 \end{aligned}$$

$$= \left(\mu \sum_{i=2}^{m} \|A_i(x_i^k - x_i^{k+1})\|_H^2 + \|\lambda^k - \lambda^{k+1}\|_{H^{-1}}^2 \right)$$
$$+ \left(\|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2 - \|\lambda^k - \lambda^{k+1}\|_{H^{-1}}^2 - \left\| \sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k) \right\|_H^2 \right)$$
$$= \|v^k - v^{k+1}\|_G^2 + \left(\|\lambda^k - \tilde{\lambda}^k\|_{H^{-1}}^2 - \|\lambda^k - \lambda^{k+1}\|_{H^{-1}}^2 - \left\| \sum_{i=2}^{m} A_i(x_i^k - \tilde{x}_i^k) \right\|_H^2 \right). \tag{5.9}$$

By using (4.3), we have

$$\lambda^k - \tilde{\lambda}^k = (\lambda^k - \lambda^{k+1}) + H\left(\sum_{i=2}^{m} A_i(x_i^k - x_i^{k+1}) \right).$$

Substituting in (5.9), we obtain

$$\|\tilde{v}^k - v^k\|_G^2 - \|\tilde{v}^k - v^{k+1}\|_G^2 = \|v^k - v^{k+1}\|_G^2 + 2(\lambda^k - \lambda^{k+1})^{\mathrm{T}} \left(\sum_{i=2}^{m} A_i(x_i^k - x_i^{k+1}) \right).$$

Consequently with (5.8), we obtain

$$\|v^k - v^*\|_G^2 - \|v^{k+1} - v^*\|_G^2 \geqslant \|v^k - v^{k+1}\|_G^2 + 2(\lambda^k - \lambda^{k+1})^{\mathrm{T}} \left(\sum_{i=2}^{m} A_i(x_i^k - x_i^{k+1}) \right). \tag{5.10}$$

Because $\mu > m - 1$, the condition (3.5) holds (see Lemma 5.2). The assertion (5.5) follows immediately from (5.10) and (3.5). □

Now, we are at the stage to prove the convergence of the proposed method.

THEOREM 5.4 The sequence $\{w^k\}$ generated by the proposed method converges to a point in $\mathcal{W}^*$.

Proof. The proof consists of the following three claims:

(1) The sequence $\{w^k\}$ is bounded, thus it has at least one cluster point.

(2) Any cluster point of $\{w^k\}$ is a solution point of $\mathrm{VI}(\mathcal{W}, F, \theta)$.

(3) The sequence $\{w^k\}$ has only one cluster point.

We first complete the first claim. The boundedness of $\{v^k\}$ is obvious based on (5.5) and the full column-rank assumption on A_i's $(i = 2, 3, \ldots, m)$. Thus, the rest is to prove the boundedness of $\{x_1^k\}$. It follows from (5.5) that

$$c_0 \cdot \sum_{k=0}^{\infty} \|v^k - v^{k+1}\|_G^2 \leqslant \|v^0 - v^*\|_G^2 \quad \forall v^* \in \mathcal{V}^*,$$

which implies that

$$\lim_{k\to\infty} \|v^k - v^{k+1}\|_G = 0.$$

Recall the definition of G in (2.7). Since H is positive definite, we have that

$$\lim_{k\to\infty} \|A_i(x_i^k - x_i^{k+1})\| = 0, \quad i = 2,\dots,m; \quad \text{and} \quad \lim_{k\to\infty} \|\lambda^k - \lambda^{k+1}\| = 0. \tag{5.11}$$

That is, we have shown that the sequence $\{\lambda^k - \lambda^{k+1}\}$ is also bounded. On the other hand, it follows from (4.4) that

$$A_1x_1^{k+1} = H^{-1}(\lambda^k - \lambda^{k+1}) + b - \sum_{i=2}^m A_ix_i^{k+1}.$$

Since A_1 is assumed to be full column-rank, we have

$$x_1^{k+1} = (A_1^{\mathrm{T}}A_1)^{-1}A_1^{\mathrm{T}}\left\{H^{-1}(\lambda^k - \lambda^{k+1}) + b - \sum_{i=2}^m A_ix_i^{k+1}\right\}, \tag{5.12}$$

from which we have

$$\|x_1^{k+1}\| \leqslant \|(A_1^{\mathrm{T}}A_1)^{-1}\| \cdot \|A_1^{\mathrm{T}}\| \cdot \left\{\|H^{-1}\| \cdot \|\lambda^k - \lambda^{k+1}\| + \|b\| + \sum_{i=2}^m \|A_i\| \cdot \|x_i^{k+1}\|\right\}.$$

Recall the boundedness of $\{\lambda^k - \lambda^{k+1}\}$ and $\{v^k\}$. Hence, the boundedness of $\{x_1^k\}$ is ensured by the boundedness of $\{\lambda^k - \lambda^{k+1}\}$ and $\{v^k\}$. We thus have that the sequence $\{w^k\}$ has at least one cluster point, and the first claim is proved.

Let $w^\infty = (x_1^\infty, v^\infty)$ be a cluster point of the sequence $\{w^k\}$ and $\{w^{k_j}\}$ be the subsequence converging to w^∞. It follows from Theorem 4.6 (by using (4.11)) that

$$\theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}}F(\tilde{w}^k) + \tfrac{1}{2}(\|v - v^k\|_G^2 - \|v - v^{k+1}\|_G^2) \geqslant 0 \quad \forall w \in \mathcal{W}, \tag{5.13}$$

where $F(\cdot)$ is given in (2.2b). Recall (4.3). We conclude that $\lambda^{k+1} - \tilde{\lambda}^k \to 0$. According to the definition of x and $\tilde{w}^k$ in (2.2b), (4.5), we have

$$\tilde{x}^k = x^{k+1} \quad \text{and} \quad \tilde{w}^k - w^{k+1} \to 0, \quad x^k - x^{k+1} \to 0, \; w^k - w^{k+1} \to 0.$$

Then, it follows from (5.13) that

$$\theta(x) - \theta(x^{k_j}) + (w - w^{k_j})^{\mathrm{T}}F(w^{k_j}) + \tfrac{1}{2}(\|v - v^{k_j}\|_G^2 - \|v - v^{k_j+1}\|_G^2) \geqslant 0 \quad \forall w \in \mathcal{W}. \tag{5.14}$$

Taking the limit over j in (5.14), considering the continuity of a convex function in its domain, and combining $\lim_{k\to\infty} \|v^k - v^{k+1}\| = 0$, we conclude that

$$\theta(x) - \theta(x^\infty) + (w - w^\infty)^{\mathrm{T}}F(w^\infty) \geqslant 0 \quad \forall w \in \mathcal{W}.$$

According to Theorem 2.2, w^∞ is a solution point of VI$(\mathcal{W}, F, \theta)$. Thus, the second claim is proved.

Finally, we prove the third claim. In fact, (5.5) implies that the sequence $\{v^k\}$ has the only cluster point $\{v^\infty\}$. In other words, $x_i \to x_i^\infty$ $(i = 2,\dots,m)$. Recall that $\lambda^k - \lambda^{k+1} \to 0$ (see (5.6)). Thus, it follows from (5.12) that $x_1^k \to x_1^\infty := (A_1^{\mathrm{T}}A_1)^{-1}A_1^{\mathrm{T}}[b - \sum_{i=2}^m A_ix_i^\infty]$. Overall, we have shown that the sequence $\{w^k\}$ converges to w^∞, which is a point in $\mathcal{W}^*$. The proof is completed. □

6. Further discussion of convergence

In Section 5, we establish the convergence for the proposed method under the assumption that $A_i (i = 1, \ldots, m)$ are all full column-rank. As we have shown in Theorem 5.3, this assumption is to ensure the boundedness of the sequence $\{w^k\}$. If we remove this assumption, the boundedness of $\{w^k\}$ cannot be ensured. But some weaker convergence analogous to that in Zhang *et al.* (2010a,b) can be established for the proposed method, without the full column rank assumption on A_i's. More specifically, the conclusion (5.5) can be written as

$$\sum_{i=2}^{m} \mu \|A_i x_i^k - A_i x_i^*\|_H^2 + \|\lambda^k - \lambda^*\|_{H^{-1}}^2 \leqslant \sum_{i=2}^{m} \mu \|A_i x_i^k - A_i x_i^*\|_H^2 + \|\lambda^k - \lambda^*\|_{H^{-1}}^2 - c_0 \|v^k - v^{k+1}\|_G^2 \quad \forall v^* \in \mathcal{V}^*,$$

from which we can obtain the boundedness of $\{A_i x_i^k\}$ $(i = 2, \ldots, m)$ and $\{\lambda^k\}$. Thus, the following weaker convergence is immediately derived.

(1) There exists a subsequence $\{w^{k_j}\}$ such that

$$\lim_{j\to\infty} A_i x_i^{k_j} = A_i x_i^*, \quad i = 2, \ldots, m \quad \text{and} \quad \lim_{j\to\infty} \lambda^{k_j} = \lambda^*,$$

where $(x_2^*, \ldots, x_m^*, \lambda^*)$ is a certain point in $\mathcal{V}^*$.

(2) Any limit point of $\{w^k\}$ is a point in $\mathcal{W}^*$.

Taking a closer look at the iterative scheme of the proposed method, we can easily find that implementation of the proposed algorithm only requires the tuple $\{(A_2 x_2^k, \ldots, A_m x_m^k, \lambda^k)\}$. Hence, it is still useful to investigate the convergence for $\{(A_2 x_2^k, \ldots, A_m x_m^k, \lambda^k)\}$ as in Zhang *et al.* (2010b).

7. Convergence rate

In this section, we analyse the worst-case convergence rate for the proposed method. More specifically, we show in both an ergodic and a nonergodic sense that after t iterations at most, the proposed method can find an approximate solution of $\mathrm{VI}(\mathcal{W}, F, \theta)$ whose accuracy is $O(1/t)$.

We first show a worst-case $O(1/t)$ convergence rate in an ergodic sense. The proof follows the analytic framework in He (2011) for a class of projection and contraction methods and in He & Yuan (2012) for the ADMM (1.3).

THEOREM 7.1 Let $\{w^k\}$ be the sequence generated by the proposed method and $\{\tilde{w}^k\}$ be given by (4.5). For any integer number $t > 0$, let

$$\tilde{w}_t := \frac{1}{t+1} \sum_{k=0}^{t} \tilde{w}^k. \tag{7.1}$$

Then, we have $\tilde{w}_t \in \mathcal{W}$ and

$$\theta(\tilde{x}_t) - \theta(x) + (\tilde{w}_t - w)^{\mathrm{T}} F(w) \leqslant \frac{1}{2(t+1)} \|v - v^0\|_G^2 \quad \forall w \in \mathcal{W}, \tag{7.2}$$

i.e., $\tilde{w}_t$ is an approximate solution point of $\mathrm{VI}(\mathcal{W}, F, \theta)$ with the accuracy of $O(1/t)$.

Proof. First, because $\tilde{x}^k = x^{k+1}$, it holds that $\tilde{w}^k \in \mathcal{W}$ for all $k \geqslant 0$. Thus, together with convexity of $\mathcal{X}_i$ $(i = 1, \ldots, m)$, the definition in (7.1) implies that $\tilde{w}_t \in \mathcal{W}$. Secondly, adding the inequality (4.12) over $k = 0, 1, \ldots, t$, we obtain

$$(t+1)\theta(x) - \sum_{k=0}^{t} \theta(\tilde{x}^k) + \left((t+1)w - \sum_{k=0}^{t} \tilde{w}^k \right)^{\mathrm{T}} F(w) + \frac{1}{2}\|v - v^0\|_G^2 \geqslant 0 \quad \forall w \in \mathcal{W}.$$

Combining the notation of $\tilde{w}_t$, it can be written as

$$\frac{1}{t+1} \sum_{k=0}^{t} \theta(\tilde{x}^k) - \theta(x) + (\tilde{w}_t - w)^{\mathrm{T}} F(w) \leqslant \frac{1}{2(t+1)} \|v - v^0\|_G^2 \quad \forall w \in \mathcal{W}. \tag{7.3}$$

Since $\theta(x)$ is convex and

$$\tilde{x}_t = \frac{1}{t+1} \sum_{k=0}^{t} \tilde{x}^k,$$

we have $\theta(\tilde{x}_t) \leqslant (1/(t+1)) \sum_{k=0}^{t} \theta(\tilde{x}^k)$. Substituting it in inequality (7.3), the assertion of this theorem follows directly. □

Therefore, for any given compact set $\mathcal{D} \subset \Omega$, let $\tilde{d} := \sup\{\|v - v^0\|_G^2 \mid w \in \mathcal{D}\}$. Then, after t iterations of the proposed method, the point $\tilde{w}_t$ defined in (7.1) satisfies

$$\sup_{w \in \mathcal{D}} \{\theta(\tilde{x}_t) - \theta(x) + (\tilde{w}_t - w)^{\mathrm{T}} F(w)\} \leqslant \frac{\tilde{d}}{2(t+1)},$$

which means that $\tilde{w}_t$ is an approximate solution of VI$(\mathcal{W}, F, \theta)$ with the accuracy $O(1/t)$. That is, a worst-case $O(1/t)$ convergence rate is established in an ergodic sense for the proposed method.

Now, we provide another approach to establish the same convergence rate, but in a nonergodic sense. For this purpose, recall we have emphasized in Lemma 4.4 that the term $(v - \tilde{v}^k)^{\mathrm{T}} GN(v^k - \tilde{v}^k)$ measures the accuracy of current iterate to a solution point in $\mathcal{W}^*$. Since we have $N(\tilde{v}^k - v^k) = v^{k+1} - v^k$ (see (4.9)), the assertion (4.7) in Lemma 4.4 can be rewritten as

$$\tilde{w}^k \in \mathcal{W}, \quad \theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^{\mathrm{T}} F(\tilde{w}^k) + (v - \tilde{v}^k)^{\mathrm{T}} G(v^{k+1} - v^k) \geqslant 0 \quad \forall w \in \mathcal{W},$$

which means that w^{k+1} is a solution point in $\mathcal{W}^*$ if $\|v^k - v^{k+1}\|_G^2 = 0$ (according to Theorem 2.2). Thus, we can view $\|v^k - v^{k+1}\|_G^2$ as a residual or an error bound to measure the accuracy of w^{k+1}, with the interest in estimating the convergence rate of the proposed method in terms of the reduction of $\|v^k - v^{k+1}\|_G^2$. More specifically, we show that after t iterations, the iterate generated by the proposed method ensures that $\|v^k - v^{k+1}\|_G^2 \leqslant \epsilon$, where $\epsilon = O(1/t)$, i.e., a worst-case $O(1/t)$ convergence rate is established in a nonergodic sense.

To establish the nonergodic worst-case $O(1/t)$ convergence rate for the proposed method, we first show a lemma.

LEMMA 7.2 The sequence $\{\|v^k - v^{k+1}\|_G\}$ generated by the proposed method is monotonically non-increasing, i.e.,

$$\|v^{k+1} - v^{k+2}\|_G^2 \leqslant \|v^k - v^{k+1}\|_G^2 \quad \forall k \geqslant 1. \tag{7.4}$$

Proof. Since (4.9), we just need to prove the following statement equivalent to (7.4):

$$\|N(v^k - \tilde{v}^k)\|_G^2 - \|N(v^{k+1} - \tilde{v}^{k+1})\|_G^2 \geqslant 0. \tag{7.5}$$

First, applying the assertion (4.7) to the $(k+1)$th iteration, we have

$$\theta(x) - \theta(\tilde{x}^{k+1}) + (w - \tilde{w}^{k+1})^{\mathrm{T}} F(\tilde{w}^{k+1}) + (v - \tilde{v}^{k+1})^{\mathrm{T}} GN(\tilde{v}^{k+1} - v^{k+1}) \geqslant 0 \quad \forall w \in \mathcal{W}. \tag{7.6}$$

Then, setting $w = \tilde{w}^{k+1}$ in (4.7) and $w = \tilde{w}^k$ in (7.6), respectively, adding the resulting inequalities and using (2.1) in Lemma 2.1, we derive the assertion

$$(\tilde{v}^k - \tilde{v}^{k+1})^{\mathrm{T}} GN\{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\} \geqslant 0. \tag{7.7}$$

Recall the assertion (2.9) in Lemma 2.4. Thus, adding the term

$$\{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\}^{\mathrm{T}} GN\{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\}$$

to both sides of (7.7) and using the identity (2.9), we obtain

$$\begin{aligned} &(v^k - v^{k+1})^{\mathrm{T}} GN\{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\} \\ &\quad \geqslant \{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\}^{\mathrm{T}} \frac{(GN + N^{\mathrm{T}}G)}{2} \{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\}. \end{aligned}$$

Substituting (4.9) into the last inequality, we obtain

$$(v^k - \tilde{v}^k)^{\mathrm{T}} N^{\mathrm{T}} GN\{(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\} \geqslant \tfrac{1}{2}\|(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\|_{(GN+N^{\mathrm{T}}G)}^2. \tag{7.8}$$

Moreover, setting $a = N(v^k - \tilde{v}^k)$ and $b = N(v^{k+1} - \tilde{v}^{k+1})$ in the identity (2.6) and using the inequality (7.8), we obtain

$$\begin{aligned} \|N(v^k - \tilde{v}^k)\|_G^2 - \|N(v^{k+1} - \tilde{v}^{k+1})\|_G^2 &\geqslant \|(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\|_{(GN+N^{\mathrm{T}}G)}^2 \\ &\quad - \|N(v^k - \tilde{v}^k) - N(v^{k+1} - \tilde{v}^{k+1})\|_G^2 \\ &= \|(v^k - \tilde{v}^k) - (v^{k+1} - \tilde{v}^{k+1})\|_{(GN+N^{\mathrm{T}}G-N^{\mathrm{T}}GN)}^2. \end{aligned} \tag{7.9}$$

Since the matrix $GN + N^{\mathrm{T}}G - N^{\mathrm{T}}GN$ is positive semidefinite (see Lemma 2.4), the right-hand side of (7.9) is non-negative. The inequality (7.5) thus holds and the lemma is proved. □

Now, we are ready to establish a nonergodic worst-case $O(1/t)$ convergence rate for the proposed method, mainly based on conclusions proved in Theorem 5.3 and Lemma 7.2.

THEOREM 7.3 Let $\{v^t\}$ be the sequence generated by the proposed method. Then, we have

$$\|v^t - v^{t+1}\|_G^2 \leqslant \frac{1}{(t+1)c_0}\|v^0 - v^*\|_G^2 \quad \forall v^* \in \mathcal{V}^*, \tag{7.10}$$

where G is defined in (2.7) and c_0 is the constant specified in Lemma 5.2.

Proof. First, it follows from (5.5) that

$$c_0 \sum_{p=0}^{\infty} \|v^k - v^{k+1}\|_G^2 \leqslant \|v^0 - v^*\|_G^2 \quad \forall k \in \mathcal{N},\ \forall v^* \in \mathcal{V}^*. \tag{7.11}$$

Since Lemma 7.2 shows that the sequence $\{\|v^k - v^{k+1}\|_G^2\}$ is monotonically nonincreasing, we have

$$(t+1)\|v^t - v^{t+1}\|_G^2 \leqslant \sum_{k=0}^{t} \|v^k - v^{k+1}\|_G^2, \tag{7.12}$$

which implies the assertion (5.5) immediately. □

Recall $\mathcal{W}^*$ is convex and closed under our assumptions (see Facchinei & Pang, 2003, Theorem 2.3.5). Let

$$d := \inf\{\|v^0 - v^*\|_G^2 \mid v^* \in \mathcal{V}^*\}.$$

For any given $\epsilon > 0$, Theorem 7.3 indicates that the proposed method requires at most $\lfloor d/c_0\epsilon \rfloor$ iterations to ensure that $\|v^k - v^{k+1}\|_G^2 \leqslant \epsilon$. A nonergodic worst-case $O(1/t)$ convergence rate is thus established for the proposed method.

8. Numerical results

In this section, we illustrate the efficiency of the proposed algorithm by some numerical experiments. Since we released a preprint of this paper on Optimization Online in June 2010, the algorithm proposed in this paper has been used by other authors to solve some applications such as some non-negative matrix factorization and dimensionality reduction problems in Esser *et al.* (2012), and gene regulatory network identification problems in Liang *et al.* (2012).[1] Here, we further illustrate the efficiency of the proposed algorithm by applying it to solve the robust principal component analysis (RPCA) model and the image inpainting problem. We coded the proposed algorithm by MATLAB 7.12 (R2011a). All experiments were implemented on a ThinkPad notebook with an Intel Core i5-2140M CPU at 2.3 GHz and 4 GB of memory.

8.1 *The RPCA model*

In Candés *et al.* (2011), the RPCA model was proposed

$$\begin{aligned} \min_{A,E} \quad & \|A\|_* + \tau\|E\|_1 \\ \text{s.t.} \quad & A + E = C, \end{aligned} \tag{8.1}$$

where $C \in \Re^{l\times n}$ is a given matrix (data); $\|\cdot\|_*$ is the nuclear norm which is defined as the sum of all singular values, and it is to induce the low-rank feature in the component A; $\|\cdot\|_1$ denotes the sum of absolute values of all entries (an extension of the l_1 norm for vectors), and it is to induce sparsity in the component E; and $\tau > 0$ is a constant balancing the low-rank and sparsity. To handle the case where

[1] The old title of this paper is ‘A splitting method for separate convex programming with linking linear constraints’.

only incomplete entries of C are observable and there is Gaussian noise in observation, the RPCA model with incomplete and noisy observations was then proposed in Tao & Yuan (2011):

$$\begin{aligned} \min_{A,E} \quad & \|A\|_* + \tau\|E\|_1 \\ \text{s.t.} \quad & \|P_\Omega(C - A - E)\|_F \leqslant \delta, \end{aligned} \tag{8.2}$$

where Ω is a subset of the index set of entries $\{1, 2, \ldots, l\} \times \{1, 2, \ldots, n\}$ which denotes the observable entries $\{C_{ij}, (i,j) \in \Omega\}$; the operator $P_\Omega : \mathcal{R}^{l\times n} \to \mathcal{R}^{l\times n}$ summarizes the incomplete observation information, and it is the orthogonal projection onto the span of matrices vanishing outside of Ω so that the ijth entry of $P_\Omega(X)$ is X_{ij} if $(i,j) \in \Omega$ and zero otherwise; $\|\cdot\|_F$ is the standard Frobenius norm; and $\delta > 0$ is the magnitude of Gaussian noise corrupting the observed data. We refer the reader to Tao & Yuan (2011) for more details of the model (8.2).

To see how the model (1.1) under our consideration captures (8.2), let $\mathbf{B}{:=}\{Z \in \Re^{l\times n} \mid \|P_\Omega(Z)\|_F \leqslant \delta\}$. Then, (8.2) can be reformulated as

$$\begin{aligned} \min_{A,E,Z} \quad & \|A\|_* + \tau\|E\|_1 \\ \text{s.t.} \quad & A + E + Z = P_\Omega(C), \\ & Z \in \mathbf{B}, \end{aligned} \tag{8.3}$$

which is a concrete application of (1.1) with $m = 3$, except that the vector variables and coefficients in (1.1) are replaced by matrix variables and linear operators in matrix spaces, respectively. Although we focus on the case of (1.1) with vector variables in our previous theoretical analysis, the proposed method and theoretical analysis can be trivially extended to the case with matrix variables. More specifically, (8.3) can be explained as a special case of (1.1) with the specification

$$(x_1, x_2, x_3) = (A, E, Z) \in \Re^{l\times n} \times \Re^{l\times n} \times \mathbf{B};$$
$$\theta_1(A) = \|A\|_*, \quad \theta_2(E) = \tau\|E\|_1, \quad \theta_3(Z) = 0;$$

all linear operators in the linear constraints are identity operators and $b = P_\Omega(C)$.

In this section, we apply the proposed algorithm to solve (8.2) in two different scenarios: synthetic simulation and the background extraction problem from surveillance video with missing and noisy data. As we have mentioned, the motivation of proposing the new method is to retain the advantage of the extended ADMM scheme (1.4), which can yield simple subproblems because of the possibility of exploiting θ_i's properties separately; and meantime, to avoid any correction step to ensure convergence. To illustrate the efficiency of the proposed algorithm (HTY for short), we thus compare it with the extended ADMM scheme (1.4) (ADMM for short), and the parallel splitting augmented Lagrangian method in He (2009) (PSALM for short) which requires a correction step at each iteration. We refer the reader to Tao & Yuan (2011) for elaboration on the resulting subproblems of HTY and ADMM. Note that the resulting subproblems of these two methods are of equal difficulty. In addition, the resulting subproblems at the prediction step of PSALM are similar to those of ADMM. We thus omit them.

8.1.1 *Synthetic simulations.* Let us first test the model (8.2) with synthetic dataset, where the solution is known. As in Tao & Yuan (2011), the low-rank component A^* is generated by $A^* = LR^{\mathrm{T}}$, where L and R are independent $l \times r$ and $n \times r$ matrices, respectively. Entries of L and R are independently and identically distributed (i.i.d.) Gaussian random variables with zero means and unit variance. Hence, the rank of A^* is r. The index of observed entries, i.e., Ω, is determined at random. The support $\Gamma \subset \Omega$ of the impulsive noise E^* (sparse but large) is chosen uniformly at random, and the nonzero entries of E^* are i.i.d. uniformly in the interval $[-500, 500]$. Then, the matrix C is given by $C = A^* + E^*$. Let sr, spr and rr represent the ratios of sample (observed) entries (i.e., $|\Omega|/mn$), the number of nonzero entries of E (i.e., $\|E\|_0/mn$) and the rank of A^* (i.e., r/m), respectively. The parameter τ in (8.2) is fixed as $\tau = 1/\sqrt{l}$, and the parameter $\delta = 0$.

For all the implemented methods and all the tested scenarios, we set $H = \beta I$, and the value of β is determined simply by

$$\beta = \begin{cases} 0.08 \dfrac{|\Omega|}{\|P_\Omega(C)\|_1} & \text{if } \mathtt{spr} = 0.05; \\ 0.15 \dfrac{|\Omega|}{\|P_\Omega(C)\|_1} & \text{if } \mathtt{spr} = 0.1, \end{cases} \tag{8.4}$$

and the initial iterate is $(A^0, E^0, Z^0) = (0, 0, 0)$. Since (8.2) is a special case of (1.1) with $m = 3$, $\mu > 2$ is required to ensure the convergence of HTY. We thus set $\mu = 2.01$ when implementing HTY. For the parameter γ required by the correction step of PSALM, we set it as $\gamma = 0.8$.

As in Tao & Yuan (2011), we use the stopping criterion in terms of the relative errors of the recovered low-rank and sparse components

$$\text{RelChg} := \frac{\|(A^{k+1}, E^{k+1}) - (A^k, E^k)\|_F}{\|(A^k, E^k)\|_F + 1} \leqslant \text{Tol} \tag{8.5}$$

to terminate all tested methods, where Tol > 0 is a tolerance. We set Tol $= 1e-5$ in our experiments.

Moreover, in our experiments, we executed the singular value decomposition (SVD) by implementing the package of PROPACK in Larsen (1998) to compute those singular values that are larger than a particular threshold and their corresponding singular vectors in A-subproblems (see details in Tao & Yuan, 2011). We denote by $(\hat{A}, \hat{E})$ the iterate when the stopping criterion (8.5) is achieved.

We tested the cases where $\mathtt{sr} = 0.8$, $l = n = 500, 1000$, and some different choices of rr and spr as given in Table 1. For the tested methods, HTY, ADMM and PSALM, we report the relative error of the recovered sparse component (ErrsSP $:= \|\hat{E} - E^*\|_F / \|E^*\|_F$), the relative error of the recovered low-rank component (ErrsLR $:= \|\hat{A} - A^*\|_F / \|A^*\|_F$), the computing time in seconds ('Time(s)') and the number of SVD required by A-subproblems ('#SVD'). We observed that when the stopping criterion (8.5) is satisfied, the tested methods achieve the same objective function value for each tested scenario. The objective function values ('obj') are also reported in Table 1. According to the data in Table 1, we see that to achieve the same level of recovery, ADMM is the fastest; but HTY (with proved convergence) is numerically competitive to ADMM (without proved convergence). We believe the reason why PSALM is slower than ADMM and HTY is because its correction step at each iteration ruin the low-rank characteristic.

To see the comparison clearly, for the particular case where $l = n = 500$, $\mathtt{spr} = 0.05$, $\mathtt{rr} = 0.05$ and $\mathtt{sr} = 0.8$, we visualize in Fig. 1 the respective evolutions of the recovered rank, the relative error ErrsLR and ErrsSP with respect to iterations.

8.1.2 *Background extraction from surveillance video with missing and noise data.* We then investigate an application of (8.2): extracting background from surveillance video with missing and noisy data.

TABLE 1 *Recovery results of HTY, ADMM and PSALM for* (8.2) *with* $\texttt{sr} = 80\%$ *and* $\delta = 0$

			$\frac{\Vert\hat{E}-E^*\Vert_F}{\Vert E^*\Vert_F}$			$\frac{\Vert\hat{A}-A^*\Vert_F}{\Vert A^*\Vert_F}$			#SVD			Time (s)			obj (e+5)
$l=n$	rr	spr	HTY	ADMM	PSALM	HTY	ADMM	PSALM	HTY	ADMM	PSALM	HTY	ADMM	PSALM	HTY/ADMM/PSALM
500	0.05	0.05	2.64e−5	2.60e−5	4.68e−5	2.33e−4	1.33e−4	3.30e−4	37	24	55	19.0	16.2	63.1	1.516
		0.1	2.16e−5	2.86e−5	3.79e−5	3.06e−4	2.27e−4	3.96e−4	38	21	76	16.05	10.09	82.07	2.918
	0.1	0.05	3.93e−5	3.89e−5	4.14e−5	2.70e−4	2.10e−4	2.50e−4	43	24	70	16.40	13.17	93.68	1.638
		0.1	2.91e−5	3.68e−5	6.00e−5	3.20e−4	3.17e−4	5.86e−4	46	26	89	27.49	19.41	129.53	3.040
1000	0.05	0.05	1.03e−5	1.44e−5	3.09e−5	1.60e−4	6.04e−5	1.85e−4	45	23	64	56.60	58.59	586.36	4.460
		0.1	1.17e−5	1.60e−5	2.57e−5	2.08e−4	1.06e−4	2.51e−4	47	22	86	91.37	65.82	886.41	8.420
	0.1	0.05	1.49e−5	1.64e−5	3.58e−5	1.41e−4	6.73e−5	1.83e−4	56	29	78	148.00	112.84	601.06	4.942
		0.1	1.91e−5	1.91e−5	3.27e−5	2.33e−4	1.17e−4	2.52e−4	59	31	107	253.70	193.00	1176.34	8.903

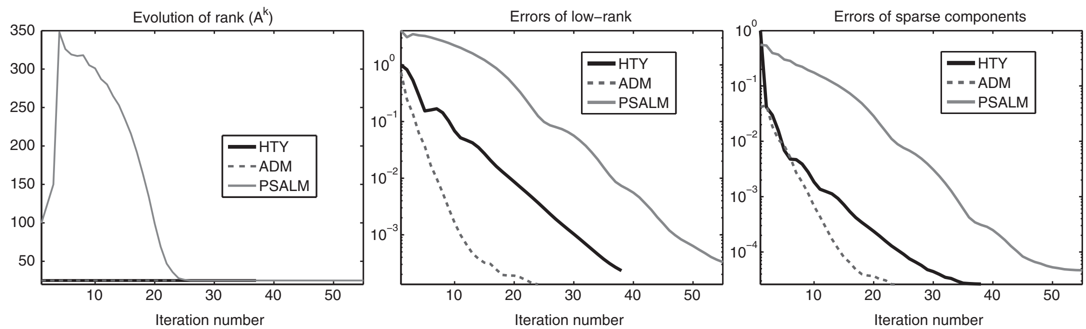

FIG. 1. Evolution of recovered rank (left), *ErrsLR* (middle) and *ErrSP* for HTY, ADMM and PSALM.

To understand this concrete application, we first provide some preliminary background of this application and refer the reader to, e.g., Candés *et al.* (2011), for more details. More specifically, video consists of a sequence of frames, and mathematically it is a natural candidate for low-rank modelling due to high correlation between frames. Each frame consists of foreground and background. Since the background of video needs to be flexible enough to accommodate changes in the scene, it is natural to model it as approximately low rank. Foreground objects, such as cars or pedestrians, occupy a relatively small fraction of the image pixels, and hence can be treated as sparse errors. One basic imaging task in video surveillance is to separate the background from foreground. However, in real application, the video may include missing and noise pixels. Thus, only a fraction of noised entries can be obtained. A natural question is: can we extract the background, i.e., the low-rank part, from the foreground even with missing and noise observations? To see how this problem can be reflected by the model (8.2), C is the matrix representation of a sequence of video frames where each column represents a frame; the index set Ω (assumed known) indicates the locations of observed pixels, i.e., pixels outside Ω are missing; E represents the foreground while A denotes the background; and δ denotes the magnitude of Gaussian noise of corrupted pixels.

In our experiments, we test two sequences of video downloadable at the website.[2] One is a sequence of 150 greyscale frames of size 128×160 taken in a lobby, and the other is a sequence of 200 greyscale frames of size 144×176 taken in an airport. The data matrix C is formed by stacking each frame into a column. Thus, $C \in \Re^{20480\times 150}$ for the first video and $C \in \Re^{25344\times 200}$ for the second one. The first video has 20% missing pixels and the second has 30% missing pixels. The index of observed entries, i.e., Ω, is determined randomly by the MATLAB built-in function `randperm`. The Gaussian noise is generated with a zero mean and a standard deviation of $\sigma = 10^{-3}$.

As in Tao & Yuan (2011), let

$$\text{RelChg} := \frac{\|(A^{k+1}, E^{k+1}) - (A^k, E^k)\|_F}{\|(A^k, E^k)\|_F + 1} \tag{8.6}$$

measure the relative change of the recovered low-rank and sparse components. Our stopping criterion to implement the mentioned methods is

$$\text{RelChg} \leqslant 10^{-3}. \tag{8.7}$$

[2] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html.

TABLE 2 *Recovery results for Background extraction*

	It.	Time (s)	Rank	#SVD	$\|\hat{A}\|_*$ $(e+5)$	$\|\hat{E}\|_1$ $(e+6)$	obj $(e+5)$
Lobby Video, $128\times160\times150$, 20% missing data							
ADMM	44	58.1	8	44	3.86	20	5.26
HTY	47	59.1	6	47	3.85	20	5.26
PSALM	49	67.2	11	49	3.88	20	5.28
Airport Video, $144\times176\times200$, 30% missing data							
	It.	Time (s)	Rank	#SVD			
ADMM	41	114.9	23	41	1.66	4.39	1.94
HTY	43	115.8	18	43	1.63	4.39	1.91
PSALM	49	143.2	33	49	1.70	4.39	1.98

All methods start their iterations with the initial iterate $(A^0, E^0, Z^0) = (0, 0, 0)$. Again, we denote by $(\hat{A}, \hat{E})$ the iterate when the stopping criterion (8.7) is achieved.

In (8.2), we take $\tau = 1/\sqrt{l}$ and $\delta = \sqrt{l + \sqrt{8l}}\sigma$. Recall $l = 20480$ for the first video and 25344 for the second. To implemented HTY with $H = \beta \cdot I_l$, throughout we choose $\beta = 0.01(|\Omega|/\|P_\Omega(C)\|_1)$ where $|\Omega|$ denotes the cardinality of Ω. For the parameter μ, recall it suffices to choose $\mu > 2$. We take $\mu = 2.01$. The choice of H for ADMM and PSALM is the same as HTY. Moreover, for PSALM the additional parameter γ in its correction step is set as 0.8.

In Table 2, we report the numerical performance of these three tested methods, including the number of iterations ('It.'), the computing time in seconds ('Time (s)'), the rank of the recovered low-rank component ('Rank'), $\|\hat{A}\|_*$, $\|\hat{E}\|_1$ and the objective function ('obj') when the stopping criterion (8.7) is achieved. Since the computation at each iteration of all the tested methods is dominated by a singular value decomposition (SVD), we also report the numbers of SVD ('#SVD').

Data in Table 2 show that HTY performs very competitively with the extended ADMM scheme (1.4), whose convergence is still unclear, and it outperforms PSALM, which is as effective as exploiting properties of individual functions, while it requires an additional correction step to correct the output of (1.4). This verifies empirically our theoretical motivation of the proposed algorithm.

Due to the space limitation, we reported the extracted background and foreground only for HTY, the results of the other two compared methods are similar. More specifically, in Figs 2 and 3 we show the 10th, 40th and 80th frames of the tested video and their extracted background and foreground by implementing HTY.

8.2 *Image inpainting problem*

Then, we focus on image inpainting problems. Some background of image inpainting is provided and more details can be found in the literature, see, e.g., Chan & Shen (2005). More specifically, image inpainting refers to filling in missing or damaged regions in images, either in the pixel domain or in a transformed domain. Because of its pivotal role in many image processing tasks, the topic of image inpainting has been studied extensively in the literature. Let $\bar{x}$ be an unknown image. Without loss of generality, we assume that $\bar{x}$ is an n-by-n square image. Following the conventional treatment, we vectorize a two-dimensional image into a one-dimensional vector, e.g., in lexicographic order. Therefore, throughout this paper we treat n-by-n images as vectors in $\Re^{n^2}$, i.e., $\bar{x} \in \Re^{n^2}$. The model of image

FIG. 2. Lobby video. Left column: corrupted frames; Middle column: extracted background; Right column: extracted foreground.

inpainting we consider here is

$$f = S \cdot (K\bar{x} + \omega), \tag{8.8}$$

where $K \in \Re^{n^2 \times n^2}$ is a blurring (or convolution) operator, $S \in \Re^{n^2 \times n^2}$ is a mask operator, i.e., a diagonal matrix whose zero entries denote missing pixels and identity entries indicate observed pixels, $\omega \in \Re^{n^2}$ contains additive noise introduced in the observation process, '$\cdot$' denotes componentwise multiplication, and $f \in \Re^{n^2}$ denotes the observed image.

It is well known that the system of equations (8.8) is ill-conditioned. Therefore, instead of recovering $\bar{x}$ from f by solving (8.8) directly, we need to utilize some prior information such as adding a regularizer to ensure $f = SK\bar{x}$, i.e., the data fidelity. As in Chan & Shen (2002) and Chan *et al.* (2006), when the total variation (TV) regularization proposed in Rudin *et al.* (1992) is utilized, the model of image inpainting

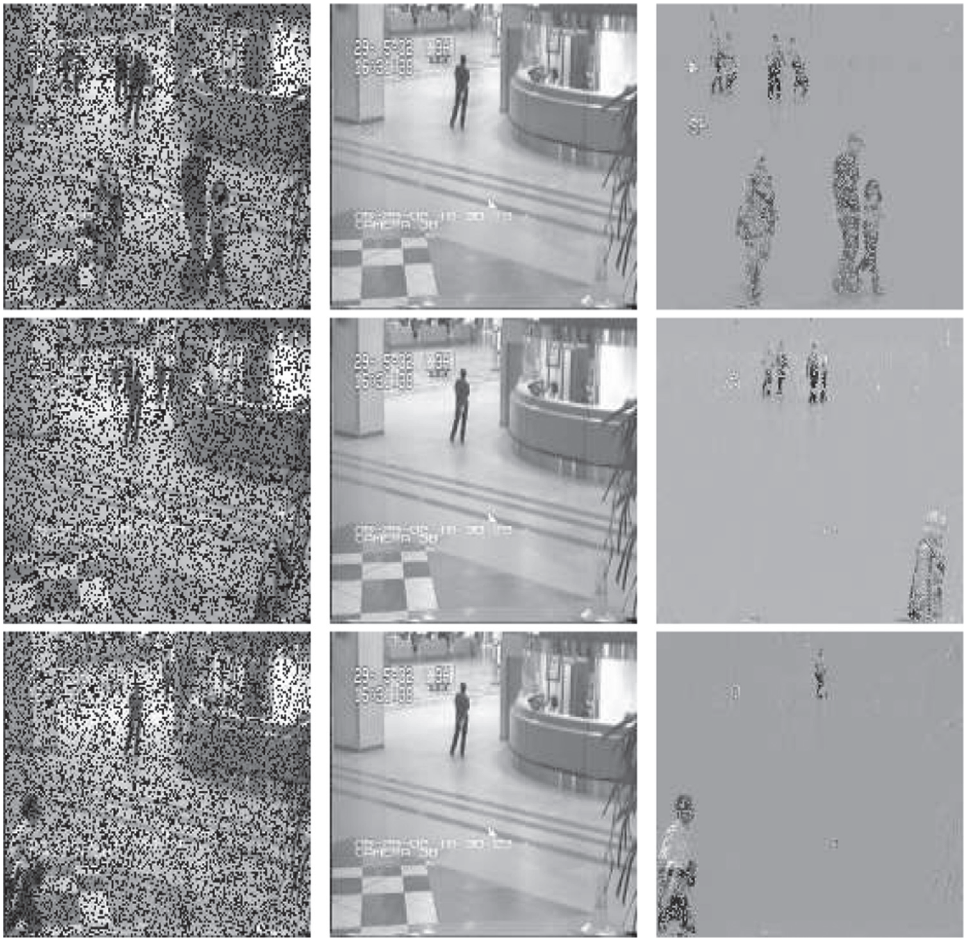

FIG. 3. Airport video. Left column: corrupted frames; Middle column: extracted background; Right column: extracted foreground.

with TV regularization turns out to be

$$\min_{x} \||\nabla x|\|_1 + \frac{\tau}{2}\|SKx - f\|^2. \tag{8.9}$$

In (8.9), $\nabla := (\partial_1, \partial_2)$, where $\partial_1 : \Re^{n^2} \to \Re^{n^2}$ and $\partial_2 : \Re^{n^2} \to \Re^{n^2}$ denotes the discretized derivatives in the horizontal and vertical directions, respectively; the constant $\tau > 0$ measures the trade-off between the fidelity to f and the amount of regularization; $\|x\|_1 := \sum_{i=1}^{n^2} |x_i|$ for $x \in \Re^{n^2}$; and for $y := (y_1, y_2) \in \Re^{n^2} \times \Re^{n^2}$, $|y|$ denotes a vector in $\Re^{n^2}$ whose entries are given by

$$|y|_i = \sqrt{(y_1)_i^2 + (y_2)_i^2}, \quad i = 1, 2, \ldots, n^2.$$

Note that our analysis below is also applicable for the case of the anisotropic discretization (i.e., the 2-norm is used in the TV-regularizer).

In the rest, we show that the model (8.9) can be easily reformulated as a special case of (1.1), and thus the proposed method is applicable for the TV image inpainting problem. In fact, by introducing two auxiliary variables y and z, (8.9) can be rewritten as

$$\begin{aligned} \min \quad & \||y|\|_1 + \frac{\tau}{2}\|Sz - f\|^2 \\ \text{s.t.} \quad & y = \nabla x, \\ & Kx = z. \end{aligned} \tag{8.10}$$

Then, (8.10) is a special case of (1.1) with the specification

$$x = (x_1, x_2, x_3) = (y, x, z) \in \Re^{2n^2} \times \Re^{n^2} \times \mathcal{R}^{n^2};$$
$$\theta_1(y) = \||y|\|_1, \quad \theta_2(x) = 0, \quad \theta_3(z) = \frac{\tau}{2}\|Sz - f\|^2$$

and

$$A = (A_1, A_2, A_3) = \begin{pmatrix} I & -\nabla & 0 \\ 0 & -K & I \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Now, we analyse the resulting subproblems when the proposed method is applied to solve the reformulation (8.10), and show that they are all simple enough to have closed-form solutions. In addition to the trivial tasks of updating the Lagrange multiplier (3.2) and (3.4), there are three main subproblems at each iteration of the proposed method, and we discuss them one by one. Throughout, we choose $H = \mathrm{diag}\{\beta_1 \cdot I_{2n^2}, \beta_2 \cdot I_{n^2}\}$ with $\beta_1 > 0$ and $\beta_2 > 0$ in the implementation of the proposed method.

- The y-related subproblem (3.1) amounts to solving

$$\operatorname{argmin}_y \left\{ \||y|\|_1 + \frac{\beta_1}{2}\left\| y - \nabla x^k - \frac{1}{\beta_1}\lambda_1^k \right\|^2 \right\},$$

  whose closed-form solution is given by

$$y^{k+1} = \mathcal{S}_{1/\beta_1}\left(\nabla x^k + \frac{1}{\beta_1}\lambda_1^k\right).$$

  Here, the shrinkage operator $\mathcal{S}$ is defined by

$$\mathcal{S}_\beta(y) = y - \min(\beta, |y|) \cdot \frac{y}{|y|}, \tag{8.11}$$

  where $0 \cdot (0/0) = 0$ is assumed.

- The x-related subproblem in (3.3) amounts to solving

$$\operatorname{argmin}_x \left\{ (\tilde{\lambda}_1^k)^{\mathrm{T}} \nabla x + (\tilde{\lambda}_2^k)^{\mathrm{T}}(Kx) + \frac{\beta_1 \mu}{2}\| - \nabla(x - x^k)\|^2 + \frac{\beta_2 \mu}{2}\| - K(x - x^k)\|^2 \right\},$$

  whose closed-form solution can be obtained via solving the system of linear equations

$$\mu(\beta_1 \nabla^{\mathrm{T}}\nabla + \beta_2 K^{\mathrm{T}}K)(x^{k+1} - x^k) = -\nabla^{\mathrm{T}}\tilde{\lambda}_1^k - K^{\mathrm{T}}\tilde{\lambda}_2^k. \tag{8.12}$$

When the blurring K is spatially invariant and periodic boundary conditions are used for the discrete differential operator, it is well known (see, e.g., Hansen *et al.*, 2006) that the matrices $K^{\mathrm{T}}K$ and $\nabla^{\mathrm{T}}\nabla$ can be diagonalized by the discrete Fourier transform (DFT), and faster solvers for (8.12) are available.

- The z-related subproblem in (3.3) is

$$\operatorname{argmin}_z \left\{ \frac{\tau}{2}\|Sz - f\|^2 + (-\tilde{\lambda}_2^k)^{\mathrm{T}} z + \frac{\beta_2\mu}{2}\|z - z^k\|^2 \right\},$$

whose closed-form solution is given by

$$z^{k+1} = \left(I + \frac{\tau}{\beta_2\mu} S^{\mathrm{T}} S \right)^{-1} \left(\frac{\tau}{\beta_2\mu} S^{\mathrm{T}} f + z^k + \frac{1}{\beta_2\mu}\tilde{\lambda}_2^k \right).$$

Note that, for the mask operator S, $S^{\mathrm{T}}S$ is a diagonal matrix whose entries are either 0 or 1. Thus, it is trivial to compute $(I + (\tau/\beta_2\mu)S^{\mathrm{T}}S)^{-1}$.

In addition that (8.9) can be reformulated as (8.10) and thus the proposed method is implementable, the model (8.9) is reducible to (1.2) and thus the original ADMM (1.3) becomes applicable. In fact, by grouping (y, z) as one variable, we obtain

$$\begin{aligned}
&(x_1, x_2) = ((y, z), x) \in (\Re^{2n^2} \times \Re^{n^2}) \times \mathcal{R}^{n^2};\\
&\theta_1(y, z) = \||y|\|_1 + \frac{\tau}{2}\|Sz - f\|^2, \quad \theta_2(x) = 0;\\
&A = (A_1, A_2) = \left(\begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}, \begin{pmatrix} -\nabla \\ -K \end{pmatrix} \right) \quad \text{and} \quad b = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.
\end{aligned}$$

Therefore, (8.9) is also a special case of (1.2). In the following, we elaborate on the detail of applying the original ADMM (1.3) directly to (8.9). Taking $H = \operatorname{diag}\{\beta_1 \cdot I_{2n^2}, \beta_2 \cdot I_{n^2}\}$ in (1.3), it is easy to specify the resulting subproblems as the follows.

- The (y, z)-related subproblem amounts to solving two independent problems. That is, the y-subproblem is

$$\operatorname{argmin}_y \left\{ \||y|\|_1 + \frac{\beta_1}{2}\left\| y - \nabla x^k - \frac{1}{\beta_1}\lambda_1^k \right\|^2 \right\},$$

whose solution is given by the closed form

$$y^{k+1} = \mathcal{S}_{\frac{1}{\beta_1}}\left(\nabla x^k + \frac{1}{\beta_1}\lambda_1^k \right),$$

where $\mathcal{S}$ is given in (8.11); and the z-related subproblem is

$$\operatorname{argmin}_z \left\{ \frac{\tau}{2}\|Sz - f\|^2 + \frac{\beta_2}{2}\left\| z - Kx^k - \frac{\lambda_2^k}{\beta_2} \right\|^2 \right\},$$

whose closed-form solution is given by

$$z^{k+1} = \left(I + \frac{\tau}{\beta_2} S^{\mathrm{T}} S\right)^{-1} \left(\frac{\tau}{\beta_2} S^{\mathrm{T}} f + K x^k + \frac{1}{\beta_2} \lambda_2^k\right).$$

- The x-related subproblem amounts to solving

$$x^{k+1} \in \operatorname{argmin}_x \left\{ \langle \lambda_1^k, \nabla x \rangle + \langle \lambda_2^k, Kx \rangle + \frac{\beta_1}{2} \|\nabla x - y^{k+1}\|^2 + \frac{\beta_2}{2} \|Kx - z^{k+1}\|^2 \right\},$$

whose closed-form solution can be obtained via solving the system of linear equations

$$\left(\nabla^{\mathrm{T}} \nabla + \frac{\beta_2}{\beta_1} K^{\mathrm{T}} K\right) x^{k+1} = \nabla^{\mathrm{T}} y^{k+1} + \frac{\beta_2}{\beta_1} K^{\mathrm{T}} z^{k+1} - \frac{1}{\beta_1} (\nabla^{\mathrm{T}} \lambda_1^k + K^{\mathrm{T}} \lambda_2^k).$$

Note that (λ_1, λ_2) is the Lagrange multiplier, and it is updated by

$$\lambda_1^{k+1} = \lambda_1^k + \beta_1 (\nabla x^{k+1} - y^{k+1})$$

and

$$\lambda_2^{k+1} = \lambda_2^k + \beta_2 (K x^{k+1} - z^{k+1}).$$

We are thus interested in the comparison between the implementation of the new method on the reformulation (8.10) with $m = 3$, and the implementation of the original ADMM (1.3) directly on (8.9) with $m = 2$.

Note that both HTY and ADMM are primal-dual based methods. Therefore, we can measure the accuracy of the solution in terms of the primal-infeasibility and dual-infeasibility. That is,

$$\max\{\beta_1 \|\nabla(x^{k+1} - x^k)\|, \beta_2 \|x^{k+1} - x^k\|\} < \epsilon, \tag{8.13}$$

and

$$\max\left\{\frac{1}{\beta_1} \|\lambda_1^{k+1} - \lambda_1^k\|, \frac{1}{\beta_2} \|\lambda_2^{k+1} - \lambda_2^k\|\right\} < \epsilon. \tag{8.14}$$

See, e.g., Boyd *et al.* (2010) and Yuan (2012), and the explanation before Lemma 7.2 for more details. In our numerical experiments, we take $\epsilon = 10^{-2}$ in (8.13) and (8.14).

We also compare the new method numerically with TwIST (Bioucas-Dias & Figueiredo, 2007) and FISTA (Beck & Teboulle, 2009), both of which handle the model (8.9) directly and are benchmarks in the imaging literature. For them, we terminate the iterations with the stopping criterion

$$\frac{\|f^{k+1} - f^k\|}{\max\{\|f^k\|, 1\}} < \epsilon, \tag{8.15}$$

where $\epsilon > 0$ is a given tolerance, and f^k represents the kth objective function value. We set $\epsilon = 1 \times 10^{-4}$ for (8.15) in our experiments.

We test the images 'peppers.png' (256×256) and 'lena.png' (256×256). The blurring operator K in (8.9) is generated with the *gaussian* kernel (`hsize` $= 5$ and `sigma` $= 14$). For 'peppers', the operator S in (8.9) is the characters mask; and for 'lena', S is the mask where 60% (randomly generated subject to the Gaussian distribution) of its pixels are missed. For both the images, ω is the additive

FIG. 4. original 'pepper', corrupted 'pepper' with 28.70% relative error, original 'lena', and corrupted 'lena' with 77.77% relative error.

zero-mean white noise with the standard deviation 10^{-3}. The relative error of an image with missing pixels is defined as

$$\text{RelErr} := \frac{\|f - \bar{x}\|^2}{\|\bar{x}\|^2}, \tag{8.16}$$

where f is the observed image and $\bar{x}$ is the true image. For the initial image of 'peppers', its relative error is 28.70%; and for 'lena', it is 77.72%. In Figure 4, we display the original and corrupted images of 'pepper' and 'lena'.

To measure the quality of restored images, in the literature the signal-to-noise ratio (SNR) in decibel (dB)

$$\text{SNR}(x) \triangleq 10 * \log_{10} \frac{\|\bar{x} - \tilde{x}\|^2}{\|\bar{x} - x\|^2} \tag{8.17}$$

is used, where $\bar{x}$ is the original image and $\tilde{x}$ is the mean intensity value of $\bar{x}$. In our experiments, the SNR values of the corrupted images 'peppers' and 'lena' are 2.81 and -5.99 dB, respectively.

In our experiments, we set $\tau = 10^4$ in the model (8.9). For the choice of the penalty matrix H, we choose $H = \text{diag}\{\beta_1 \cdot I_{2n^2}, \beta_2 \cdot I_{n^2}\}$ for HTY and ADMM. Throughout, we choose $(\beta_1 = 10, \beta_2 = 100)$ for HTY and $(\beta_1 = 10, \beta_2 = 200)$ for ADMM. Note that we tune the values of β_i for different methods, and these individual choices seem good enough to result in their own best numerical performance, according to our experiments. Other parameters of these methods are chosen as follows. For the parameter μ of HTY, since $m = 3$ for this application, it suffices to choose $\mu > 2$ and we again take $\mu = 2.01$. For TwIST and FISTA, we downloaded the Matlab codes from the authors' homepages, and thus values of all parameters are unchanged. Since a denoising subproblem is required to be solved at each iteration for both TwIST and FISTA, we employ the algorithm in Chambolle (2004) and allow for a maximal iterative number of 10 for this denoising subproblem. All the tested methods take the corrupted images as the initial iterate. The detailed results are also listed in Table 3 in terms of iteration numbers ('It'), relative error ('RelErr' defined in (8.16)), recovered SNR ('SNR', computing time in seconds ('Time (s)'), and the objective function value ('obj.'). Moreover, we report the primal-infeasibility ('Pr-infea') (defined in (8.13)) and dual-infeasibility ('Du-infea') (defined in (8.14)) of HTY and ADMM in Table 3.

In Fig. 5, we report the restored images by different methods. Relative errors of restored images ('RE'), SNR values ('SNR') and the computing time in seconds, when the stopping criterion (8.15) is achieved, are also reported. Furthermore, in Figs 6 and 7, we plot the evolutions of SNR values with respect to computing time for different methods.

TABLE 3 *Recovery results for image inpainting*

	It.	RelErr (%)	SNR	Time (s)	obj.	Pr-infea ($\times 10^{-3}$)	Du-infea ($\times 10^{-3}$)
			‘pepper’ with 28.70% relative error				
HTY	85	3.20	21.86	7.92	2852.99	3.038	5.358
ADMM	59	3.24	21.75	6.38	2853.61	1.999	4.929
TwIST	153	3.00	22.37	22.32	2894.97	—	—
FISTA	196	3.03	22.32	33.96	2935.88	—	—
			‘Lena’ with 77.77% relative error				
HTY	109	5.16	17.57	7.45	2294.23	1.918	2.697
ADMM	79	5.17	17.55	7.38	2294.41	2.208	2.851
TwIST	161	4.96	17.89	20.89	2337.87	—	—
FISTA	372	4.94	17.94	43.93	2378.68	—	—

FIG. 5. Restored images by ADMM, HTY, TwIST and FISTA, respectively. The first row: ‘peppers’; the second row: ‘lena’.

These numerical results show that FISTA achieves the best SNR values, while ADMM and HTY are able to achieve comparable SNR values with significantly less time. Moreover, although HTY is slightly slower in the restoration speed than ADMM, we see that HTY applied to the reformulation (8.10) with $m = 3$ is as effective as the original ADMM (1.3) directly applied on (8.9) with $m = 2$, in restoring images with almost the same quality. This fact indicates that even for a particular problem in the form of (1.1) with $m \geqslant 3$, but reducible to the case with $m = 2$, the proposed method is still very competitive with the ADMM (1.3). Therefore, together with the tested experiments for irreducible cases of (1.1) with $m \geqslant 3$, efficiency of the proposed method is further illustrated.

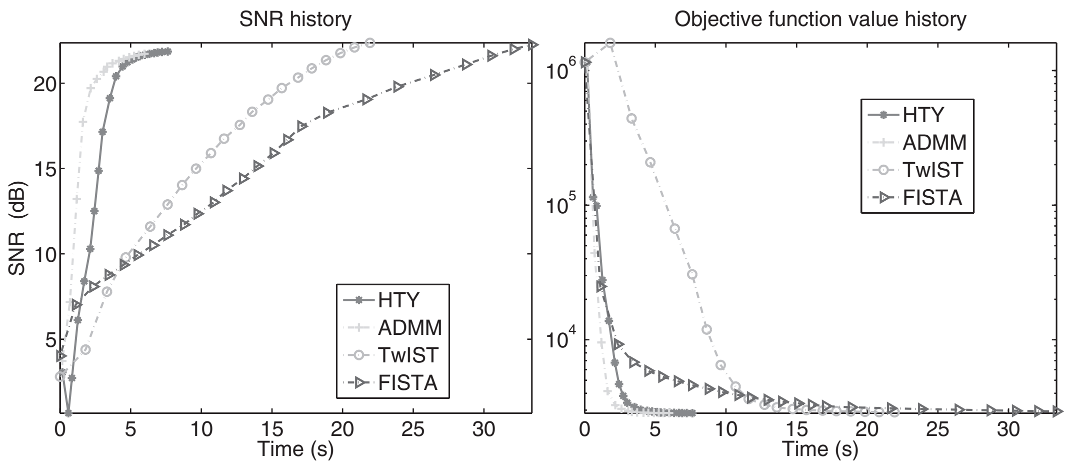

FIG. 6. Evolution of SNR and objective function value w.r.t. computing times for 'peppers'.

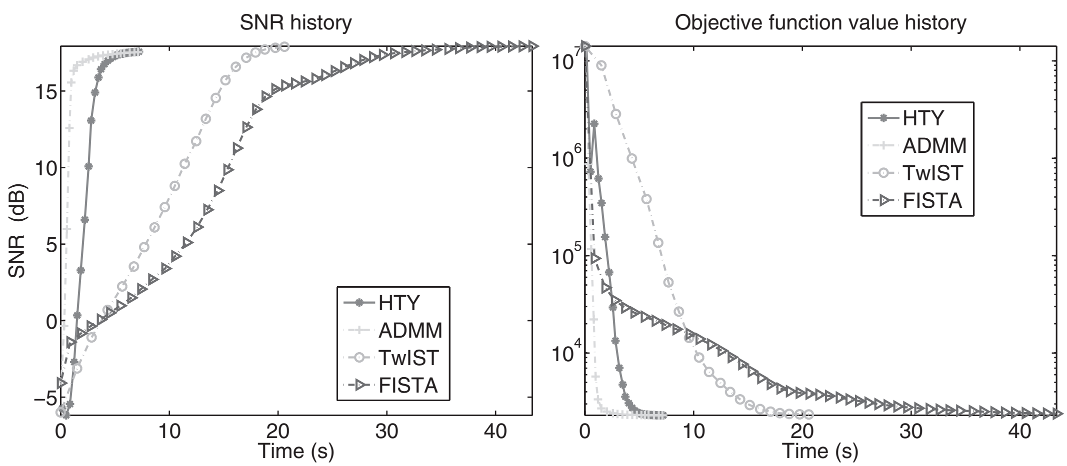

FIG. 7. Evolution of SNR and objective function value w.r.t. computing times for 'lena'.

9. Conclusions

In this paper, we propose a splitting method for solving a separable convex minimization problem with linear constraints, where the objective function is expressed as the sum of many individual functions without coupled variables. The new method is suitable for exploiting properties of these individual functions separably, resulting in subproblems which could easily enough have closed-form solutions if each individual function is simple. Moreover, an improvement of the new method over some pre-existing splitting methods is that no correction step is required. We verify these advantages numerically by some particular applications in image processing.

Funding

B.H. was supported by the NSFC grants 10971095 and 91130007, and the MOEC fund 20110091110004. X.Y. was supported by the Hong Kong General Research Fund: HKBU 203311.

References

Beck, A. & Teboulle, M. (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, **2**, 183–202.

Bioucas-Dias, J. & Figueiredo, M. (2007) A new TwIST: two-step iterative thresholding algorithm for image restoration. *IEEE Trans. Image Process.*, **16**, 2992–3004.

Blum, E. & Oettli, W. (1975) *Mathematische Optimierung, Econometrics and Operations Research XX*. Berlin: Springer.

Boyd, S., Parikh, N., Chu, E., Peleato, B. & Eckstein, J. (2010) Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, **3**, 1–122.

Candés, E. J., Li, X., Ma, Y. & Wright, J. (2011) Robust principal component analysis? *J. ACM*, **58**, 1–37.

Chambolle, A. (2004) An algorithm for total variation minimization and applications. *J. Math. Imaging Vision*, **20**, 89–97.

Chan, T. & Shen, J. (2002) Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math.*, **62**, 1019–1043.

Chan, T. & Shen, J. (2005) *Image Processing and Analysis: Variational, PDE, Wavelet, and Stochastic Methods*. Philadelphia, PA: SIAM.

Chan, T., Shen, J. & Zhou, H. (2006) Total variation wavelet inpainting. *J. Math. Imaging Vision*, **25**, 107–125.

Esser, E., Möller, M., Osher, S., Sapiro, G. & Xin, J. (2012) A convex model for non-negative matrix factorization and dimensionality reduction on physical space. *IEEE Trans. Image Process.*, **21**, 3239–3252.

Facchinei, F. & Pang, J.-S. (2003) *Finite-Dimensional Variational Inequalities and Complementarity problems*. Springer Series in Operations Research, vol. I. Berlin: Springer.

Gabay, D. & Mercier, B. (1976) A dual algorithm for the solution of nonlinear variational problems via finite-element approximations. *Comput. Math. Appl.*, **2**, 17–40.

Glowinski, R. & Marrocco, A. (1975) Approximation par éléments finis d'ordre un et résolution par pénalisation-dualité d'une classe de problèmes non linéaires. *R.A.I.R.O.*, **R2**, 41–76.

Han, D. R., Yuan, X. M. & Zhang, W. X. An augmented-Lagrangian-based parallel splitting method for separable convex programming with applications to image processing. *Math. Comput.* (to appear).

Hansen, P. C., Nagy, J. G. & O'Leary, D. P. (2006) *Deblurring Images: Matrices, Spectra, and Filtering*. Philadelphia; SIAM.

He, B. S. (2009) Parallel splitting augmented Lagrangian methods for monotone structured variational inequalities. *Comput. Optim. Appl.*, **42**, 195–212.

He, B. S. (2011) On the $O(1/t)$ convergence rate of the projection and contraction methods for variational inequalities with Lipschitz continuous monotone operators. Submitted.

He, B. S., Tao, M., Xu, M. H. & Yuan, X. M. (2013) Alternating directions based contraction method for generally separable linearly constrained convex programming problems. *Optimization*, **62**, 573–596.

He, B. S. & Yuan, X. M. (2012) On the $O(1/n)$ convergence rate of Douglas–Rachford alternating direction method. *SIAM J. Num. Anal.*, **50**, 700–709.

Hestenes, M. (1969) Multiplier and gradient methods. *J. Optim. Theory Appl.*, **4**, 303–320.

Kiwiel, K. C., Rosa, C. H. & Ruszczyński, A. (1999) Proximal decomposition via alternating linearization. *SIAM J. Optim.*, **9**, 668–689.

Larsen, R. M. (1998) Lanczos bidiagonalization with partial reorthogonalization. *Technical report, DAIMI PB-357*. Department of Computer Science, Aarhus University. Available at http://soi.stanford.edu/ rmunk/PROPACK.

Liang, X. J., Xia, Z., Zhang, L. W. & Wu, F. X. (2012) Inference of gene regulatory subnetworks from time course gene expression data. *BMC Bioinform.*, **13(Suppl. 9)**, S3.

PENG, Y. G., GANESH, A., WRIGHT, J., XU, W. L. & MA, Y. (2012) Robust alignment by sparse and low-rank decomposition for linearly correlated images. *IEEE Trans. Patt. Analy. Mach. Intell.*, **34**, 2233–2246.

POWELL, M. (1969) A method for nonlinear constraints in minimization problems. *Optimization* (R. Fletcher ed.). New York: Academic Press, pp. 283–298.

RUDIN, L., OSHER, S. & FATEMI, E. (1992) Nonlinear total variation based noise removal algorithms. *Physica D*, **60**, 259–268.

SETZER, S., STEIDL, G. & TEBUBER, T. (2010) Deblurring Poissonian images by split Bregman techniques. *J. Vis. Commun. Image Represent.*, **21**, 193–199.

TAO, M. & YUAN, X. M. (2011) Recovering low-rank and sparse components of matrices from incomplete and noisy observations. *SIAM J. Optim.*, **21**, 57–81.

TIBSHIRANI, R., SAUNDERS, M., ROSSET, S., ZHU, J. & KNIGHT, K. (2005) Sparsity and smoothness via the fused lasso. *J. R. Stat. Soc.*, **67**, 91–108.

YUAN, X. M. (2012) Alternating direction methods for covariance selection models. *J. Sci. Comput.*, **51**, 261–273.

ZHANG, X. Q., BURGER, M., BRESSON, X. & OSHER, S. (2010a) Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM J. Image Sci.*, **3**, 253–276.

ZHANG, X. Q., BURGER, M. & OSHER, S. (2010b) A unified primal-dual algorithm framework based on Bregman iteration. *J. Sci. Comput.*, **46**, 20–46.