Comput Optim Appl (2014) 59:135–161
DOI 10.1007/s10589-013-9616-x

Customized proximal point algorithms for linearly constrained convex minimization and saddle-point problems: a unified approach

Guoyong Gu · Bingsheng He · Xiaoming Yuan

Received: 30 January 2013 / Published online: 9 November 2013

Abstract This paper focuses on some customized applications of the proximal point algorithm (PPA) to two classes of problems: the convex minimization problem with linear constraints and a generic or separable objective function, and a saddle-point problem. We treat these two classes of problems uniformly by a mixed variational inequality, and show how the application of PPA with customized metric proximal parameters can yield favorable algorithms which are able to make use of the models' structures effectively. Our customized PPA revisit turns out to unify some algorithms including some existing ones in the literature and some new ones to be proposed. From the PPA perspective, we establish the global convergence and a worst-case $O(1/t)$ convergence rate for this series of algorithms in a unified way.

Keywords Convex minimization · Saddle-point problem · Proximal point algorithm · Convergence rate · Customized algorithms · Splitting algorithms

This paper is dedicated to Professor Masao Fukushima on the occasion of his 65th birthday.

G. Gu was supported by the NSFC grant 11001124. B. He was supported by the NSFC grant 91130007 and the MOEC fund 20110091110004. X. Yuan was supported by the General Research Fund from Hong Kong Research Grants Council: 203712.

G. Gu
Department of Mathematics, Nanjing University, Nanjing 210093, China
e-mail: ggu@nju.edu.cn

B. He
International Center of Management Science and Engineering and Department of Mathematics, Nanjing University, Nanjing 210093, China
e-mail: hebma@nju.edu.cn

X. Yuan (✉)
Department of Mathematics, Hong Kong Baptist University, Hong Kong, China
e-mail: xmyuan@hkbu.edu.hk

1 Introduction

Let Ω be a closed convex subset in $\Re^l$ and $F : \Re^l \to \Re^l$ be a monotone mapping; $\Re^{l_0}$ ($l_0 \leq l$) be a subspace of $\Re^l$ and $\theta : \Re^{l_0} \to \Re$ be a closed convex but not necessarily smooth function. Let u denote the sub-vector of w in $\Re^{l_0}$ for any $w \in \Omega$. We consider the mixed variational inequality (MVI): Find $w^* \in \Omega$ such that

$$\theta(u) - \theta\bigl(u^*\bigr) + \bigl(w - w^*\bigr)^T F\bigl(w^*\bigr) \geq 0, \quad \forall w \in \Omega. \tag{1.1}$$

The MVI (1.1) includes the ordinary variational inequality (see [15]) as a special case with $\theta \equiv 0$, and it has been well studied in various fields such as the partial differential equations, economics and mathematical programming [25, 44]. Throughout, the solution set of (1.1), denoted by Ω^*, is assumed to be nonempty.

We do not discuss the generic case of MVI (1.1) in this paper. Instead, we focus on some fundamental optimization models which all turn out to be special cases of (1.1) where there are specific properties/structures associated with the function θ, the mapping F and the set Ω. We thus discuss how to develop customized algorithms in accordance with these properties/structures for these optimization models. But, the MVI (1.1) serves as a unified mathematical model for our theoretical analysis, and it enables us to show the convergence results uniformly while presenting different algorithms for various models individually. More specifically, we consider two classes of problems: (a) the convex minimization problem with linear constraints and a generic or separable objective function, and (b) a saddle-point problem.

(1) The generic convex minimization problem with linear constraints

$$\min\bigl\{\theta(x) \bigm| Ax = b,\ x \in \mathcal{X}\bigr\}, \tag{1.2}$$

where $A \in \Re^{m \times n}$, $b \in \Re^m$, $\mathcal{X} \subseteq \Re^n$ is convex, and $\theta : \Re^n \to \Re$ is a closed convex but not necessarily smooth function. The objective function in (1.2) is generic, where no further separable structure is assumed.

(2) A particular separable case of (1.2) where the objective function is separable into two individual functions without coupled variables. For this case, by decomposing the linear constraints into two parts accordingly, we consider the model

$$\min\bigl\{\theta(x) := \theta_1(x_1) + \theta_2(x_2) \bigm| A_1 x_1 + A_2 x_2 = b,\ x = (x_1, x_2) \in \mathcal{X} := \mathcal{X}_1 \times \mathcal{X}_2\bigr\}, \tag{1.3}$$

where $A_1 \in \Re^{m \times n_1}$, $A_2 \in \Re^{m \times n_2}$, $b \in \Re^m$, $\mathcal{X}_1 \subseteq \Re^{n_1}$ and $\mathcal{X}_2 \subseteq \Re^{n_2}$ are convex sets, $n_1 + n_2 = n$, $\theta_1 : \Re^{n_1} \to \Re$ and $\theta_2 : \Re^{n_2} \to \Re$ are closed convex but not necessarily smooth functions. With the purposes of exploiting the separable structure effectively and developing more customized algorithms, the philosophy of algorithmic design for the separable case (1.3) is different from that for the generic case (1.2). Thus, the separable case (1.3) deserves a particular discussion.

(3) The general separable case of (1.2) where the objective function is separable into more than two individual functions without coupled variables. Again, by rewriting the linear constraints in accordance with the separable objective function, we

consider the model

$$\min\left\{\sum_{i=1}^{K}\theta_i(x_i)\,\middle|\,\sum_{i=1}^{K}A_ix_i=b,\ x_i\in\mathcal{X}_i,\ i=1,\dots,K\right\}, \tag{1.4}$$

where $A_i\in\Re^{m\times n_i}(i=1,\dots,K)$, $b\in\Re^m$, $\mathcal{X}_i\subseteq\Re^{n_i}(i=1,\dots,K)$ are convex sets, $\sum_{i=1}^{K}n_i=n$, and $\theta_i:\Re^{n_i}\to\Re(i=1,\dots,K)$ are closed convex but not necessarily smooth. Note that (1.3) is a special case of (1.4) with $K=2$. We consider (1.3) individually because of its own wide applications in various fields and its unique speciality in algorithmic design (as referred to Sect. 7) which are not extendable to the general case (1.4) with $K\geq 3$. Alternatively, with the purpose of exploiting the properties of θ_i's individually in the procedure of algorithmic design, the model (1.4) deserves specific attention mainly due to the failure of extending those algorithms applicable for (1.3) to (1.4) straightforwardly, see e.g. [24, 27, 28].

(4) The saddle-point problem

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\left\{\theta_1(x)-y^TAx-\theta_2(y)\right\}, \tag{1.5}$$

where $A\in\Re^{m\times n}$, $\mathcal{X}\subseteq\Re^n$, $\mathcal{Y}\subseteq\Re^m$, $\theta_1:\Re^n\to\Re$ and $\theta_2:\Re^m\to\Re$ are closed convex but not necessarily smooth functions. The saddle-point problem (1.5) captures a broad spectrum of applications in various fields such as image restoration problems with the total variation (TV) regularization introduced in [48] (see e.g. [8, 14, 52]), fluid dynamics or linear elasticity problems in the contexts of partial differential equations (see e.g. [1, 16]), and Nash equilibrium problems in game theory (see e.g. [36, 41]). In particular, finding a saddle point of the Lagrange function of the model (1.2) is a special case of (1.5).

In Sect. 2.1, we will specify how to reformulate the models (1.2)–(1.5) as special cases of the MVI (1.1) case by case.

The proximal point algorithm (PPA) dates back to [40] and it was introduced to the optimization community in [37]. PPA has been playing a fundamental role both theoretically and algorithmically in the optimization area, including of course the models (1.2)–(1.5) under our consideration, see e.g. [22, 47] for a few of seminal works. Let us make more concrete our motivation of relating PPA to the mentioned target models.

(1) For solving (1.2), the augmented Lagrangian method (ALM, [33, 43]) is a benchmark method in the literature. More specifically, the iterate scheme of ALM for (1.2) is

$$\begin{cases}x^{k+1}=\arg\min\{\theta(x)-(\lambda^k)^T(Ax-b)+\frac{\beta}{2}\|Ax-b\|^2\,|\,x\in\mathcal{X}\},\\ \lambda^{k+1}=\lambda^k-\beta(Ax^{k+1}-b),\end{cases} \tag{1.6}$$

where λ^k is the Lagrange multiplier and $\beta>0$ is the penalty parameter for the violation of the linear constraints. In [46], it was shown that the ALM (1.6) is exactly the application of PPA to the dual problem of (1.2).

(2) For solving (1.3), it is not wise to apply the generic-purpose ALM directly and a dominating method in the literature is the alternating direction method of multipliers (ADMM) proposed originally in [20] (see also [18]). The iterative scheme of ADMM for (1.3) is

$$\begin{cases} x_1^{k+1} = \arg\min\{\theta_1(x_1) - (\lambda^k)^T(A_1x_1 + A_2x_2^k - b) + \frac{\beta}{2}\|A_1x_1 \\ \qquad\qquad + A_2x_2^k - b\|^2 \mid x \in \mathcal{X}_1\}, \\ x_2^{k+1} = \arg\min\{\theta_2(x_2) - (\lambda^k)^T(A_1x_1^{k+1} + A_2x_2 - b) + \frac{\beta}{2}\|A_1x_1^{k+1} \\ \qquad\qquad + A_2x_2 - b\|^2 \mid x_2 \in \mathcal{X}_2\}, \\ \lambda^{k+1} = \lambda^k - \beta(A_1x_1^{k+1} + A_2x_2^{k+1} - b). \end{cases} \tag{1.7}$$

Obviously, the ADMM (1.7) is a splitting version of the ALM (1.6) in accordance with the separable structure of (1.3). By decomposing the ALM subproblem into two subproblems in the Gauss-Seidel fashion at each iteration, the variables x_1 and x_2 can be solved separably in the alternating order. Since the functions $\theta_1(x_1)$ and $\theta_2(x_2)$ often have specific properties for a particular application of (1.3), the decomposition treatment of ADMM makes it possible to exploit these particular properties separately. This feature has inspired many novel applications of ADMM in various areas, see e.g. [4, 13, 14, 17, 49] and references cited therein.

In [19], it was elucidated that the ADMM (1.7) is essentially the application of the Douglas-Rachford splitting method [11] (which is a special form of PPA, as demonstrated in [12]) to the dual problem of (1.3).

(3) For solving (1.4), a natural idea is to extend the ADMM (1.7) in a straightforward way, yielding the iterative scheme

$$\begin{cases} x_1^{k+1} = \arg\min\{\theta_1(x_1) + \frac{\beta}{2}\|(A_1x_1 + \sum_{j=2}^K A_jx_j^k - b) \\ \qquad\qquad - \frac{1}{\beta}\lambda^k\|^2 | x_1 \in \mathcal{X}_1\}; \\ x_2^{k+1} = \arg\min\{\theta_2(x_2) + \frac{\beta}{2}\|(A_1x_1^{k+1} + A_2x_2 + \sum_{j=3}^K A_jx_j^k - b) \\ \qquad\qquad - \frac{1}{\beta}\lambda^k\|^2 | x_2 \in \mathcal{X}_2\}; \\ \qquad \vdots \\ x_i^{k+1} = \arg\min\{\theta_i(x_i) + \frac{\beta}{2}\|(\sum_{j=1}^{i-1} A_jx_j^{k+1} + A_ix_i \\ \qquad\qquad + \sum_{j=i+1}^K A_jx_j^k - b) - \frac{1}{\beta}\lambda^k\|^2 | x_i \in \mathcal{X}_i\}; \\ \qquad \vdots \\ x_K^{k+1} = \arg\min\{\theta_K(x_K) + \frac{\beta}{2}\|(\sum_{j=1}^{K-1} A_jx_j^{k+1} + A_Kx_K - b) \\ \qquad\qquad - \frac{1}{\beta}\lambda^k\|^2 | x_K \in \mathcal{X}_K\}; \\ \lambda^{k+1} = \lambda^k - \beta(\sum_{j=1}^K A_jx_j^{k+1} - b). \end{cases} \tag{1.8}$$

The extended ADMM scheme (1.8), which comes from the straightforward splitting of the ALM subproblem in alternating order, preserves the advantages of the original ADMM scheme (1.7) such that θ_i's properties can be exploited individually and thus the subproblems might be easy. Unfortunately, the convergence of (1.8) remains a challenge without further assumptions on the model (1.4). This difficulty thus has inspired us to develop a series of splitting algorithms [24, 27, 28] recently, the common purpose of which is to preserve the decomposition nature as (1.8). Among these work, the splitting method in [27] is inspired by PPA.

(4) As analyzed in [8, 14, 52], the saddle-point problem (1.5) can be regarded as the primal-dual formulation of a nonlinear programming problem, and this fact has inspired a series of primal-dual algorithms in the particular literature of image restoration problems with total variational regularization. We refer to, e.g. [8, 14, 51, 52], for their numerical efficiency. In [31], we revisit these primal-dual algorithms from the PPA perspective. It turns out that this PPA revisit simplifies the convergence analysis for this type of algorithms substantially and makes it possible to relax the involved parameters (step sizes and proximal parameters) greatly, as acknowledged instantly by some most recent works (e.g. [10, 42, 50]).

Because of the aforementioned individual applications, we are interested in studying the PPA's applications for the models (1.2)–(1.5) in a unified way, by means of the unified model (1.1). Our aim is to show that by choosing the proximal parameters judiciously in accordance with the specific structures of the MVI reformulations of the models (1.2)–(1.5), a series of customized PPAs can be obtained. These customized PPAs are fully capable of taking advantage of the available structures of the models under consideration, and they are competitive with, or even more efficient than some benchmark methods designed particularly for these models. In addition, this customized PPA approach enables us to establish the global convergence and a worst-case $O(1/t)$ convergence rate uniformly for this series of algorithms.

To elucidate the application of PPA to the MVI (1.1), let $G \in \Re^{l\times l}$ be a symmetric positive definite matrix and the G-norm be defined as $\|w\|_G := \sqrt{w^T G w}$. With the given $w^k \in \Omega$, the PPA in the metric form for (1.1) generates the new iterate w^{k+1} via solving the subproblem

$$
\text{(PPA)}\quad \begin{aligned}
& w^{k+1} \in \Omega, \\
& \theta(u) - \theta\big(u^{k+1}\big) + \big(w - w^{k+1}\big)^T \big(F\big(w^{k+1}\big) + G\big(w^{k+1} - w^k\big)\big) \ge 0, \\
& \quad \forall w \in \Omega,
\end{aligned} \tag{1.9}
$$

where G is called the metric proximal parameter. A popular choice of G is $G = rI$ where $r > 0$ is a scalar and $I \in \Re^{l\times l}$ is the identity matrix. In the literature there are intensive investigations on how to determine a value of r to guarantee convergence theoretically or how to adjust it dynamically for numerical acceleration. This simplest choice of G essentially means that the proximal regularization makes no difference on different coordinates of w and thus all the coordinates of w are proximally regularized with equivalent weights. On the other hand, there are some impressive works on PPA with metric proximal regularization, e.g. [3, 5, 6, 35] and references therein, which mainly discuss theoretical restrictions or numerical choices on the metric proximal parameter.

For the generic form of (1.1) where particular properties/structures of θ, F and Ω are not specified, there is no hint to determine any customized choice for the metric proximal parameter and thus the simplest choice of G is enough on theoretical purposes. But, for the particular models (1.2)–(1.5) under our consideration, their MVI reformulations in the form of (1.1) enjoy favorable splitting structure in θ, F and Ω. Thus, customized choices of G in accordance with the separable structures of their MVI reformulations are potential to decompose the generic PPA task (1.9) into smaller and easier subproblems. Accordingly, it becomes possible to exploit fully the particular properties of the models (1.2)–(1.5) for algorithmic benefits. This is our philosophy of algorithmic design.

The rest of the paper is organized as follows. In Sect. 2, we provide some preliminaries which are necessary for further discussions. In Sect. 3, we present the conceptual algorithmic framework based on the relaxed PPA in [21] for the models (1.2)–(1.5). Then we establish uniformly the global convergence in Sect. 4 and a worst-case $O(1/t)$ convergence rate in Sect. 5 for the conceptual algorithmic framework. In Sect. 6, we elucidate the customization of this relaxed PPA for the model (1.2). The customization of this relaxed PPA for the model (1.3) is analyzed in Sect. 7. Similar discussions for the model (1.4) are completed in Sect. 8. Afterwards, we analyze the customization for the model (1.5) in Sect. 9. Finally, we make some conclusions in Sect. 10.

2 Preliminaries

In this section, we review some preliminaries which are useful later. First, we specify the MVI reformulation for the models (1.2)–(1.5). Then, we take a brief look at the generic application of PPA to this variational reformulation, and in particular, we recall the relaxed PPA in [21] which blends the original PPA with a simple relaxation step. After that, we show a characterization on the solution set of the MVI (1.1) which is a cornerstone for proving a worst-case $O(1/t)$ convergence rate for the algorithms to be proposed. Finally, we supplement some useful notations and preliminaries.

2.1 The MVI reformulations of (1.2)–(1.5)

We first show that all the models (1.2)–(1.5) can be reformulated as specifical cases of the MVI (1.1).

(1) Let $\lambda \in \Re^m$ be the Lagrange multiplier associated with the linear constraints in (1.2). It is easy to see that solving (1.2) amounts to finding $w^* = (u^*, \lambda^*)$ such that

$$w^* \in \Omega, \quad \theta(x) - \theta\big(x^*\big) + \big(w - w^*\big)^T F\big(w^*\big) \ge 0, \quad \forall w \in \Omega, \tag{2.1a}$$

where

$$w = \begin{pmatrix} x \\ \lambda \end{pmatrix}, \quad F(w) = \begin{pmatrix} -A^T\lambda \\ Ax - b \end{pmatrix} \quad \text{and} \quad \Omega = \mathcal{X} \times \Re^m. \tag{2.1b}$$

Therefore, (2.1a), (2.1b) is a special case of (1.1) with $l = m + n$, $u = x$, $w = (u, \lambda)$, $F(w)$ and Ω are given in (2.1b).

(2) Similarly, we have that solving the model (1.3) is equivalent to finding $w^* = (x_1^*, x_2^*, \lambda^*)$ such that

$$w^* \in \Omega, \quad \theta_1(x_1) - \theta_1(x_1^*) + \theta_2(x_2) - \theta_2(x_2^*) + (w - w^*)^T F(w^*) \geq 0, \quad \forall w \in \Omega, \tag{2.2a}$$

where

$$w = \begin{pmatrix} x_1 \\ x_2 \\ \lambda \end{pmatrix}, \quad F(w) = \begin{pmatrix} -A_1^T\lambda \\ -A_2^T\lambda \\ A_1x_1 + A_2x_2 - b \end{pmatrix} \quad \text{and} \quad \Omega = \mathcal{X}_1 \times \mathcal{X}_2 \times \Re^m. \tag{2.2b}$$

Therefore, (2.2a), (2.2b) is a special case of (1.1) with $l = n_1 + n_2 + m$, $u = (x_1, x_2)$, $\theta(u) = \theta_1(x_1) + \theta_2(x_2)$, $w = (x_1, x_2, \lambda)$, $F(w)$ and Ω are given in (2.2b).

(3) Also, the Lagrange function of (1.4) is

$$L(x_1, x_2, \dots, x_K, \lambda) = \sum_{i=1}^K \theta_i(x_i) - \lambda^T \left(\sum_{i=1}^K A_i x_i - b \right),$$

which is defined on

$$\Omega = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_K \times \Re^m. \tag{2.3}$$

It is evident that finding a saddle point of $\mathcal{L}(x_1, x_2, \dots, x_K, \lambda)$ is equivalent to finding a vector $w^* = (x_1^*, x_2^*, \dots, x_K^*, \lambda^*) \in \Omega$ such that

$$\begin{cases} \theta_1(x_1) - \theta_1(x_1^*) + (x_1 - x_1^*)^T(-A_1^T\lambda^*) \geq 0, & \forall x_1 \in \mathcal{X}_1, \\ \theta_2(x_2) - \theta_2(x_2^*) + (x_2 - x_2^*)^T(-A_2^T\lambda^*) \geq 0, & \forall x_2 \in \mathcal{X}_2, \\ \quad \vdots \\ \theta_K(x_K) - \theta_K(x_K^*) + (x_K - x_K^*)^T(-A_K^T\lambda^*) \geq 0, & \forall x_K \in \mathcal{X}_K, \\ (\lambda - \lambda^*)^T(\sum_{i=1}^K A_i x_i^* - b) \geq 0, & \forall \lambda \in \Re^m. \end{cases} \tag{2.4}$$

More compactly, (2.4) can be written into the following VI:

$$\theta(u) - \theta(u^*) + (w - w^*)^T F(w^*) \geq 0, \quad \forall w \in \Omega, \tag{2.5a}$$

which is a special case of (1.1) with $l = \sum_{i=1}^{K} n_i + m$,

$$u = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \end{pmatrix}, \qquad \theta(u) = \sum_{i=1}^{K} \theta_i(x_i),$$

$$w = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_K \\ \lambda \end{pmatrix}, \qquad F(w) = \begin{pmatrix} -A_1^T \lambda \\ -A_2^T \lambda \\ \vdots \\ -A_K^T \lambda \\ \sum_{i=1}^{K} A_i x_i - b \end{pmatrix} \tag{2.5b}$$

and Ω being given in (2.3).

(4) Note that solving (1.5) is equivalent to finding $w^* = (x^*, y^*)$ such that

$$w^* \in \Omega, \quad \theta_1(x) - \theta_1(x^*) + \theta_2(y) - \theta_2(y^*) + (w - w^*)^T F(w^*) \geq 0, \qquad \forall w \in \Omega, \tag{2.6a}$$

where

$$w = \begin{pmatrix} x \\ y \end{pmatrix}, \qquad F(w) = \begin{pmatrix} -A^T y \\ Ax \end{pmatrix}, \qquad \Omega = \mathcal{X} \times \mathcal{Y}. \tag{2.6b}$$

Therefore, the model (2.6a), (2.6b) is a special case of the MVI (1.1) with $l = n + m$, $u = w = (x, y)$, $\theta(u) = \theta_1(x) + \theta_2(y)$, $F(w)$ and Ω are given in (2.6b).

Note that it is trivial to verify that the operators $F(w)$ given in (2.1b), (2.2b), (2.5b) and (2.6b) are all monotone.

2.2 A characterization of the solution set of (1.1)

As in [7, 32], for the purpose of proving a worst-case $O(1/t)$ convergence rate, it is useful to follow Theorem 2.3.5 in [15] (see (2.3.2) in p. 159) to characterize the solution set of the MVI (1.1). The proof of the following Theorem 2.1 is similar as those in some existing literature, e.g. [7, 32]; we thus omit it.

Theorem 2.1 *The solution set of* (1.1), *i.e.*, Ω^*, *is convex and it can be represented by*

$$\Omega^* = \bigcap_{w \in \Omega} \left\{ \tilde{w} \in \Omega : \theta(u) - \theta(\tilde{u}) + (w - \tilde{w})^T F(w) \geq 0 \right\}. \tag{2.7}$$

Theorem 2.1 thus implies that $\tilde{w} \in \Omega$ is an approximate solution of the MVI (1.1) with the accuracy $\epsilon > 0$ if it satisfies

$$\theta(u) - \theta(\tilde{u}) + F(w)^T (w - \tilde{w}) \geq -\epsilon, \forall w \in \mathcal{D}(\tilde{u}),$$

where $\mathcal{D}(\tilde{w}) = \{w \in \Omega \mid \|w - \tilde{w}\| \le 1\}$. In other words, after t iterations of an algorithm, if we can find $\tilde{w} \in \Omega$ such that

$$\tilde{w} \in \Omega \quad \text{and} \quad \sup_{w \in \mathcal{D}(\tilde{w})} \left\{\theta(\tilde{u}) - \theta(u) + (\tilde{w} - w)^T F(w)\right\} \le \epsilon,$$

where $\epsilon = O(1/t)$, then a worst-case $O(1/t)$ convergence rate of this algorithm is derived. In the papers [7, 32], we have shown worst-case $O(1/t)$ convergence rates for some algorithms including the ADMM (1.7) and the split inexact Uzawa method in [51]. In this paper, we will follow this line of research and prove a worst-case $O(1/t)$ convergence rate for the application of PPA to MVI (1.1), and thus the worst-case $O(1/t)$ convergence rates of a series of algorithm are established uniformly.

2.3 Some additional notations

In this subsection we supplement some useful notations for the convenience of further analysis.

First, revisiting the iterative schemes of the ALM (1.6), we see that only the sequence $\{\lambda^k\}$ is required to execute the scheme and $\{x^k\}$ is not required at all. We thus call the variable x an intermediate variable, meaning it is not involved in the iteration, see e.g., [4]. Similarly, for the ADMM (1.7), the variable x_1 is an intermediate variable and only the sequence $\{x_2^k, \lambda^k\}$ is involved in the iteration. Thus, for the variable w in (1.1), we conceptually classify all the coordinates into two categories: the intermediate coordinate which means its variable is not involved in the iteration and essential coordinate which means its variable is required by the iteration. Here, we introduce the variable v, an appropriate sub-vector of w, to collect all the essential coordinates of w, i.e., v represents all the coordinates of w which are really involved in iterations. Accordingly, the intended meaning of the notations v^k, $\tilde{v}^k$, v^*, $\mathcal{V}$ and $\mathcal{V}^*$ should be clear from the context. For example, when the ALM (1.6) is considered, we have

$$\begin{aligned}
&v = \lambda; \quad && \mathcal{V} = \Re^m; \\
&v^k = \lambda^k; \quad && \tilde{v}^k = \tilde{\lambda}^k, \ \forall k \in \mathcal{N}; \\
&v^* = \lambda^*; \quad && \mathcal{V}^* = \left\{\lambda^* \middle| \left(x^*, \lambda^*\right) \in \Omega^*\right\}.
\end{aligned}$$

When the ADMM (1.7) is considered, we have

$$\begin{aligned}
&v = (x_2, \lambda); \qquad && \mathcal{V} = \mathcal{X}_2 \times \Re^m; \\
&v^k = \left(x_2^k, \lambda^k\right); \qquad && \tilde{v}^k = \left(\tilde{x}_2^k, \tilde{\lambda}^k\right), \quad \forall k \in \mathcal{N}; \\
&v^* = \left(x_2^*, \lambda^*\right); \qquad && \mathcal{V}^* = \left\{\left(x_2^*, \lambda^*\right) \middle| \left(x_1^*, x_2^*, \lambda^*\right) \in \Omega^*\right\}.
\end{aligned}$$

Later, for the algorithms to be proposed, we will establish the global convergence in the context of the essential coordinates v. That is, we shall show that the sequence $\{v^k\}$ generated by each of the algorithms to be proposed converges to a point in $\mathcal{V}^*$. The consideration of investigating the convergence on the essential coordinates is inspired by the well-known convergence results of the ALM in [46] and the ADMM in [4, 19, 29]. In these literatures, the convergence analysis for the ALM is conducted in term of the sequence $\{\lambda^k\}$ in [46], and for the ADMM in term of the sequence $\{x_2^k, \lambda^k\}$ in [4, 19, 29].

On the other hand, it is apparent that the proximal regularization on the intermediate coordinates of w is profitless or redundant in (1.9). Thus, we only need to regularize proximally the essential coordinates of w. In other words, the metric proximal parameter G in (1.9) are not necessarily square with the dimensionality $l \times l$. In fact, its number of columns corresponds to the dimensionality of the vector v while its number of rows are still in the dimensionality of l. Thus, in general, the metric proximal parameter could be a thin matrix in (1.1) under our consideration for the models (1.2)–(1.5). Accordingly, by denoting the partial metric proximal parameter as Q, we rewrite (1.9) as

$$w^{k+1} \in \Omega, \quad \theta(u) - \theta\big(u^{k+1}\big) + \big(w - w^{k+1}\big)^T \big\{F\big(w^{k+1}\big) + Q\big(v^{k+1} - v^k\big)\big\} \geq 0, \\ \forall w' \in \Omega. \tag{2.8}$$

Indeed, this idea of partially proximal regularization has been implemented in [7] for the particular case of (1.3).

Finally, we also recall the definition of the projection under the Euclidean norm. For a convex set $\Omega \subseteq \Re^l$, the projection onto Ω under the Euclidean norm is defined by

$$P_\Omega(w) = \operatorname{Arg\,min}\left\{ \frac{1}{2}\|z - w\|^2 \,\middle|\, z \in \Omega \right\}. \tag{2.9}$$

By the first-order optimality condition, we can easily derive the inequality

$$\big(z - P_\Omega(w)\big)^T \big(P_\Omega(w) - w\big) \geq 0, \quad \forall\, z \in \Omega,\ \forall w \in \Re^l, \tag{2.10}$$

which will be used often in the coming analysis.

3 Conceptual algorithmic framework

In this section, we propose the conceptual algorithmic framework by applying PPA to the MVI (1.1). In particular, we are interested in the relaxed PPA [21] which generates the new iterate by relaxing the output of the original PPA (1.9) appropriately. More specifically, let the solution of (1.9) be denoted by $\tilde{w}^k$, then the relaxed PPA in [21] yields the new iterate via

$$w^{k+1} = w^k - \gamma\big(w^k - \tilde{w}^k\big), \tag{3.1}$$

where $\gamma \in (0, 2)$ is a relaxation factor, and it is called an under-relaxation (when $\gamma \in (0, 1)$) or over-relaxation factor (when $\gamma \in (1, 2)$). Obviously, the relaxed PPA (3.1) reduces to the original PPA (1.9) when $\gamma = 1$. Our numerical results in [7, 30, 31] have already shown the effectiveness of acceleration contributed by this relaxed step (3.1) with $\gamma \in (1, 2)$ numerically.

Recall that we consider the PPA subproblem (2.8) with partial proximal regularization merely on the essential coordinates of w, i.e., the sub-vector v. Thus, instead of the full relaxation on w in (3.1), we only relax v^k in the relaxation step accordingly. Overall, we propose conceptually the following algorithmic framework for the MVI (1.1) by applying the relaxed PPA in [21] but with partial proximal regularization.

The conceptual algorithmic framework of relaxed PPA for (1.1) with partial proximal regularization.

Let the partial metric proximal parameter Q be positive semi-definite, the relaxation factor $\gamma \in (0, 2)$ and v be an appropriate sub-vector of w. With the initial iterate v^0, the iterate scheme is

1. **PPA step**: generate $\tilde{w}^k$ via solving

$$\tilde{w}^k \in \Omega, \quad \theta(u) - \theta\big(\tilde{u}^k\big) + \big(w - \tilde{w}^k\big)^T \big\{F\big(\tilde{w}^k\big) + Q\big(\tilde{v}^k - v^k\big)\big\} \geq 0, \quad \forall w \in \Omega. \tag{3.2}$$

2. **Relaxation step**: generate the new iterate v^{k+1} via

$$v^{k+1} = v^k - \gamma\big(v^k - \tilde{v}^k\big). \tag{3.3}$$

As we have mentioned, our convergence analysis will be mainly conducted in the context of the essential coordinates v. For each concrete algorithm to be proposed, the following requirement should be met for the convenience of theoretical analysis.

A requirement associated with the conceptual relaxed PPA (3.2)–(3.3).

We need to identify a positive semi-definite, square, and symmetric sub-matrix of Q, which is denoted by H, such that

$$w^T Q v = v^T H v, \quad \forall w \in \Omega. \tag{3.4}$$

Remark 3.1 Recall that the partial metric proximal parameter Q could be a thin matrix (when $v \neq w$), and it is not necessary to be square and positive definite. This is different from a traditional metric proximal parameter in the PPA literature, where the square and positive definiteness requirements are always assumed. However, if v coincides with w, i.e., all the coordinates of w are essential, then Q is square. For this case, we simply choose $H = Q$. Nevertheless, as we shall show later (see Remarks 6.1 and 7.1), if Q is a thin matrix, we can find H by simply removing the zero rows from Q.

Remark 3.2 We refer to [7, 30, 31] for some preliminary discussions on the relaxed PPA for the models (1.2), (1.3) and (1.5).

4 The global convergence

In this section, we establish the global convergence uniformly for all the proposed algorithms in the context of the unified form (3.2)–(3.3). Our coming analysis uses the notation $\|v\|_H$ to denote $\sqrt{v^T H v}$ for notational convenience.

The proof follows the framework of contraction type methods (see [2] for the definition of contraction methods). That is, we show that the sequence $\{v^k\}$ generated by the conceptual algorithm (3.2)–(3.3) is contractive with respect to the set $\mathcal{V}^*$ (see the definition in [2] or (4.4)). Recall that for a given proposed algorithm, v denotes the set of all essential coordinates which are truly required by the iteration; and all the proposed algorithms are concrete cases of the conceptual algorithm (3.2)–(3.3) with specified u, v, w, θ, F, Ω, Q and H defined before.

We first prove an inequality in Lemma 4.1 which is useful for establishing the global convergence.

Lemma 4.1 *The sequence* $\{v^k\}$ *generated by the scheme* (3.2)–(3.3) *satisfies*

$$\left(v^k - v^*\right)^T H\left(v^k - \tilde{v}^k\right) \geq \left\|v^k - \tilde{v}^k\right\|_H^2, \quad \forall v^* \in \mathcal{V}^*. \tag{4.1}$$

Proof Let $w^* \in \Omega^*$. Note that $w^* \in \Omega$. By setting $w = w^*$ in (3.2), we get

$$\left(\tilde{w}^k - w^*\right)^T Q\left(v^k - \tilde{v}^k\right) \geq \left(\tilde{w}^k - w^*\right)^T F\left(\tilde{w}^k\right) + \theta\left(\tilde{u}^k\right) - \theta\left(u^*\right) \geq 0, \quad \forall w^* \in \Omega^*.$$

By using the monotonicity of F and the fact that $w^* \in \Omega^*$, we obtain

$$\left(\tilde{w}^k - w^*\right)^T F\left(\tilde{w}^k\right) + \theta\left(\tilde{u}^k\right) - \theta\left(u^*\right) \geq \left(\tilde{w}^k - w^*\right)^T F\left(w^*\right) + \theta\left(\tilde{u}^k\right) - \theta\left(u^*\right) \geq 0.$$

Consequently, we have

$$\left(\tilde{w}^k - w^*\right)^T Q\left(v^k - \tilde{v}^k\right) \geq 0, \quad \forall w^* \in \Omega^*.$$

Finally, recall the requirement (3.4), we obtain

$$\left(\tilde{v}^k - v^*\right)^T H\left(v^k - \tilde{v}^k\right) \geq 0, \quad \forall v^* \in \mathcal{V}^*.$$

Therefore, the assertion (4.1) follows from the above inequality and the notation $\|v\|_H := \sqrt{v^T H v}$ immediately. □

With Lemma 4.1, we are ready to show an important inequality.

Lemma 4.2 *The sequence* $\{v^k\}$ *generated by the scheme* (3.2)–(3.3) *satisfies*

$$\left\|v^{k+1} - v^*\right\|_H^2 \leq \left\|v^k - v^*\right\|_H^2 - \gamma(2-\gamma)\left\|v^k - \tilde{v}^k\right\|_H^2, \quad \forall v^* \in \mathcal{V}^*. \tag{4.2}$$

Proof It follows from (3.3) that

$$\begin{aligned}
&\left\|v^k - v^*\right\|_H^2 - \left\|v^{k+1} - v^*\right\|_H^2 \\
&\quad = \left\|v^k - v^*\right\|_H^2 - \left\|\left(v^k - v^*\right) - \gamma\left(v^k - \tilde{v}^k\right)\right\|_H^2 \\
&\quad = 2\gamma\left(v^k - v^*\right)^T H\left(v^k - \tilde{v}^k\right) - \gamma^2\left\|v^k - \tilde{v}^k\right\|_H^2, \quad \forall v^* \in \mathcal{V}^*.
\end{aligned}$$

Recall (4.1). It follows from the last inequality that

$$\|v^k - v^*\|_H^2 - \|v^{k+1} - v^*\|_H^2 \geq \gamma(2-\gamma)\|v^k - \tilde{v}^k\|_H^2, \quad \forall v^* \in \mathcal{V}^*. \tag{4.3}$$

The assertion of the lemma follows immediately. □

Remark 4.1 According to (4.2), it is clear to require $\gamma \in (0, 2)$ in the relaxation step (3.3) for the purpose of ensuring the contraction of $\{v^k\}$ with respect to $\mathcal{V}^*$.

With Lemma 4.2, we show that the relaxation step (3.3) is actually effective for the contraction purpose. In other words, this step brings the new iterate v^{k+1} closer to the set $\mathcal{V}^*$ than v^k under the H-norm, making it true that the sequence $\{v^k\}$ generated by the scheme (3.2)–(3.3) be contractive with respect to the set $\mathcal{V}^*$. Hence, the techniques for establishing convergence in [2] apply.

Theorem 4.1 *The sequence $\{v^k\}$ generated by the scheme (3.2)–(3.3) is contractive with respect to $\mathcal{V}^*$, i.e.,*

$$\|v^{k+1} - v^*\|_H^2 \leq \|v^k - v^*\|_H^2 - \frac{2-\gamma}{\gamma}\|v^k - v^{k+1}\|_H^2, \quad \forall v^* \in \mathcal{V}^*. \tag{4.4}$$

Proof It follows from (3.3) that $v^k - \tilde{v}^k = \frac{1}{\gamma}(v^k - v^{k+1})$. Thus, the assertion (4.4) is an immediate conclusion of (4.2). □

Now, we are ready to establish the global convergence for the scheme (3.2)–(3.3). We first discuss the case where the matrix H chosen in (3.4) is positive definite.

Theorem 4.2 *Let the sequence $\{v^k\}$ be generated by the scheme (3.2)–(3.3) and the matrix H chosen in (3.4) be positive definite. Then we have*

(i) *the sequence* $\lim_{k\to\infty}\{\|v^k - \tilde{v}^k\|_H\} = 0$;
(ii) *any accumulation point of $\{\tilde{w}^k\}$ is a solution point of the generic MVI*;
(iii) *both $\{v^k\}$ and $\{\tilde{v}^k\}$ are bounded*;
(iv) *and there exists $v^\infty \in \mathcal{V}^*$ such that* $\lim_{k\to\infty} \tilde{v}^k = v^\infty$.

Proof Applying the assertion (4.4) for $k = 0, \dots, \infty$ and summarizing these inequalities, we obtain

$$\gamma(2-\gamma)\sum_{k=0}^{\infty}\|v^k - \tilde{v}^k\|_H^2 \leq \|v^0 - v^*\|_H^2,$$

which implies the assertion (i) immediately. When the block matrix H is symmetric positive definite, it follows from the assertion (i) that $\lim_{k\to\infty}(v^k - \tilde{v}^k) = 0$. By substituting $\lim_{k\to\infty}(v^k - \tilde{v}^k) = 0$ into (3.1), we obtain that

$$\tilde{w}^k \in \Omega, \quad \lim_{k\to\infty}\{\theta(u) - \theta(\tilde{u}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k)\} \geq 0, \quad \forall w \in \Omega,$$

which means that $\lim_{k\to\infty}\tilde{w}^k$ is a solution of the MVI (1.1). Hence, the assertion (ii) follows. The assertion (iii) is trivial by following the assertion (i). Let v^∞ be an accumulation point of $\{\tilde{v}^k\}$, i.e., there exists a subsequence $\{\tilde{v}^{k_j}\}$ that converges to v^∞. Then, it follows from assertion (ii) that $v^\infty \in \mathcal{V}^*$. As a consequence, Theorem 4.1 implies that

$$\|v^{k+1}-v^\infty\|_H^2 \le \|v^k - v^\infty\|_H^2 - \gamma(2-\gamma)\|v^k-\tilde{v}^k\|_H^2.$$

Together with the assertion (i), the last inequality ensures that the sequence $\{\tilde{v}^k\}$ cannot have any other accumulation point. Therefore, it must converge to $v^\infty \in \mathcal{V}^*$. The assertion (iv) is proved. □

Remark 4.2 For the case where the matrix H chosen in (3.4) is only positive semi-definite, we can apply a similar proof as Theorem 4.2 to the case $H+\epsilon I$ with $\epsilon > 0$; and then investigate the asymptotical convergence behavior of the scheme (3.2)–(3.3) when $\epsilon \to 0$. We omit the detail.

5 A worst-case $O(1/t)$ convergence rate

In this section, we establish a worst-case $O(1/t)$ convergence rate uniformly for all the proposed algorithms in the context of the conceptual algorithm (3.2)–(3.3). As we have mentioned in Sect. 2.2, for this purpose, we need to show that after t iterations of the scheme (3.2)–(3.3), we can find an approximate solution of MVI (1.1) with an accuracy of $\epsilon = O(1/t)$.

We first present an identity which will be often used in the proof. Since the proof is elementary, we omit it.

Lemma 5.1 *Let $H \in \Re^{l\times l}$ be positive semi-definite and we use the notation $\|v\|_H := \sqrt{v^T H v}$. Then, we have*

$$(a-b)^T H(c-d) = \frac{1}{2}\big(\|a-d\|_H^2 - \|a-c\|_H^2\big) + \frac{1}{2}\big(\|c-b\|_H^2 - \|d-b\|_H^2\big), \quad \forall a,b,c,d \in \Re^l. \tag{5.1}$$

Then, we show some inequalities in the following lemmas which are useful for the analysis of convergence rate.

Lemma 5.2 *The sequence generated by the scheme (3.2)–(3.3) satisfies*

$$\theta(u) - \theta(\tilde{u}^k) + (w-\tilde{w}^k)^T F(w) \ge (v-\tilde{v}^k)^T H(v^k - \tilde{v}^k), \quad \forall w \in \Omega. \tag{5.2}$$

Proof It follows from (3.2) that

$$\theta(u) - \theta(\tilde{u}^k) + (w-\tilde{w}^k)^T F(\tilde{w}^k) \ge (w-\tilde{w}^k)^T Q(v^k - \tilde{v}^k), \quad \forall w \in \Omega. \tag{5.3}$$

Because of the monotonicity of F, we have

$$\big(w-\tilde{w}^k\big)^T F(w) \ge \big(w-\tilde{w}^k\big)^T F\big(\tilde{w}^k\big),$$

and the requirement (3.4) implies that

$$\big(w-\tilde{w}^k\big)^T Q\big(v^k-\tilde{v}^k\big) = \big(v-\tilde{v}^k\big)^T H\big(v^k-\tilde{v}^k\big).$$

Hence, the assertion (5.2) follows from (5.3) directly. □

Lemma 5.3 *The sequence generated by the scheme* (3.2)–(3.3) *satisfies*

$$\gamma\big(v-\tilde{v}^k\big)^T H\big(v^k-\tilde{v}^k\big)+\frac{1}{2}\big(\|v-v^k\|_H^2-\|v-v^{k+1}\|_H^2\big)$$
$$\ge \gamma\left(1-\frac{\gamma}{2}\right)\|v^k-\tilde{v}^k\|_H^2, \quad \forall v\in\mathcal{V}. \tag{5.4}$$

Proof Recall (3.3). In order to show (5.4), we need only to prove

$$\big(v-\tilde{v}^k\big)^T H\big(v^k-v^{k+1}\big)+\frac{1}{2}\big(\|v-v^k\|_H^2-\|v-v^{k+1}\|_H^2\big)$$
$$\ge \gamma\left(1-\frac{\gamma}{2}\right)\|v^k-\tilde{v}^k\|_H^2, \quad \forall v\in\mathcal{V}. \tag{5.5}$$

For the term $(v-\tilde{v}^k)^T H(v^k-v^{k+1})$, by using the identity (5.1), we get

$$\big(v-\tilde{v}^k\big)^T H\big(v^k-v^{k+1}\big)=\frac{1}{2}\big(\|v-v^{k+1}\|_H^2-\|v-v^k\|_H^2\big)$$
$$+\frac{1}{2}\big(\|v^k-\tilde{v}^k\|_H^2-\|v^{k+1}-\tilde{v}^k\|_H^2\big). \tag{5.6}$$

Use the fact (3.3) again for the term $\frac{1}{2}(\|v^k-\tilde{v}^k\|_H^2-\|v^{k+1}-\tilde{v}^k\|_H^2)$, we obtain

$$\frac{1}{2}\big(\|v^k-\tilde{v}^k\|_H^2-\|v^{k+1}-\tilde{v}^k\|_H^2\big)=\frac{1}{2}\big(\|v^k-\tilde{v}^k\|_H^2-\|\big(v^k-\tilde{v}^k\big)-\gamma\big(v^k-\tilde{v}^k\big)\|_H^2\big)$$
$$=\frac{1}{2}\gamma(2-\gamma)\|v^k-\tilde{v}^k\|_H^2.$$

Substituting it into the right-hand side of (5.6), we obtain (5.5) and the lemma is proved. □

Now, with the assertions in Lemmas 5.2 and 5.3, we can show the $O(1/t)$ convergence rate for the scheme (3.2)–(3.3).

Theorem 5.1 *For any integer* $t>0$, *we define*

$$\tilde{w}_t=\frac{1}{t+1}\sum_{k=0}^{t}\tilde{w}^k, \tag{5.7}$$

where $\tilde{w}^k$ $(k = 1, 2, \ldots, t)$ *are generated by the scheme* (3.2)–(3.3). *Then, we have* $\tilde{w}_t \in \Omega$ *and*

$$\theta(\tilde{u}_t) - \theta(u) + (\tilde{w}_t - w)^T F(w) \leq \frac{1}{2\gamma(t+1)} \|v - v^0\|_H^2, \quad \forall w \in \Omega, \tag{5.8}$$

Proof First of all, combining (5.2) and (5.4), we get

$$\theta(u) - \theta\big(\tilde{u}^k\big) + \big(w - \tilde{w}^k\big)^T F(w) + \frac{1}{2\gamma}\|v - v^k\|_H^2 \geq \frac{1}{2\gamma}\|v - v^{k+1}\|_H^2, \quad \forall w \in \Omega. \tag{5.9}$$

Summing the inequality (5.9) over $k = 0, 1, \ldots, t$, we obtain

$$\left((t+1)\theta(u) - \sum_{k=0}^{t} \theta\big(\tilde{u}^k\big)\right) + \left((t+1)w - \sum_{k=0}^{t} \tilde{w}^k\right)^T F(w) + \frac{1}{2\gamma}\|v - v^0\|_H^2 \geq 0,$$
$$\forall w \in \Omega.$$

It follows that

$$\left(\sum_{k=0}^{t} \frac{\theta(\tilde{u}^k)}{t+1} - \theta(u)\right) + (\tilde{w}_t - w)^T F(w) \leq \frac{1}{2\gamma(t+1)}\|v - v^0\|_H^2, \quad \forall w \in \Omega. \tag{5.10}$$

Since

$$\tilde{u}_t = \frac{1}{t+1}\sum_{k=0}^{t} \tilde{u}^k \quad \text{and} \quad \theta(u) \text{ is convex},$$

we have

$$\theta(\tilde{u}_t) \leq \frac{1}{t+1}\sum_{k=0}^{t} \theta\big(\tilde{u}^k\big).$$

Substituting it into (5.10), the assertion (5.8) follows directly. □

It follows from Theorem 4.2 that the sequence $\{v^k\}$ generated by the scheme (3.2)–(3.3) is bounded. Thus there exists a constant $\mathcal{C} > 0$ such that $\|v^k - v^0\|_H^2 < \mathcal{C}$ for all k's. For any $w \in \mathcal{D}(\tilde{w}) = \{w \in \Omega \mid \|w - \tilde{w}\| \leq 1\}$, it follows from (5.8) that $\tilde{w}_t$ given by (5.7) satisfies

$$\theta(\tilde{u}_t) - \theta(u) + (\tilde{w}_t - w)^T F(w) \leq \epsilon,$$

where $\epsilon = \frac{\mathcal{C}}{2\gamma(t+1)}$. Recall the preliminary in Sect. 2.2. Then, a worst-case $O(1/t)$ convergence rate of the scheme (3.2)–(3.3) is established in ergodic sense.

Remark 5.1 We establish a worst-case $O(1/t)$ convergence rate for the general scheme (3.2)–(3.3) in ergodic sense. In the literature, the same convergence rate has been derived for some special cases of the scheme (3.2)–(3.3), such as those in [32, 39] for ADMM. Our technique here basically follows the approach in [32] and differs from that in [39].

Remark 5.2 The result (5.8) suggests us to choose aggressive values of γ which are close to 2, in order to induce a smaller right-hand side in (5.8). On the other hand, recall Remark 4.1. We thus need to take a balance between these two informative properties of γ. In practice, as shown in [7, 30, 31], we recommend $\gamma \in (1.5, 1.9)$.

6 Relaxed customized PPAs for (1.2)

Now, we start to specify the conceptual relaxed PPA (3.2)–(3.3) with customized metric proximal parameters in accordance with the structures of the models (1.2)–(1.5). In this section, we focus on the generic convex minimization model (1.2), and show that some scalable algorithms can be derived by choosing different forms of the metric proximal parameter Q in (3.2).

6.1 A relaxed augmented Lagrangian method

Recall that the ALM (1.6) is an application of PPA to the dual problem of (1.2). In this subsection, we demonstrate that the ALM (1.6) can be recovered by taking a specific metric proximal parameter in (3.2). Consequently, the ALM (1.6) itself is also a special case of the direct application of PPA to the primal problem (1.2). The PPA illustration of the ALM (1.6) thus makes it possible to combine the ALM (1.6) with a relaxation step, yielding immediately the relaxed ALM to be proposed.

Denoting by $(\tilde{x}^k, \tilde{\lambda}^k)$ the output of the ALM (1.6), we can rewrite the iterative scheme (1.6) of ALM as

$$\begin{cases} \tilde{x}^k = \operatorname{Arg\,min}\{\theta(x) + \frac{\beta}{2}\|Ax - b - \frac{1}{\beta}\lambda^k\|^2 | x \in \mathcal{X}\}, \\ \tilde{\lambda}^k = \lambda^k - \beta(A\tilde{x}^k - b). \end{cases} \tag{6.1}$$

According to the scheme (6.1), we have

$$\tilde{x}^k \in \mathcal{X}, \quad \theta(x) - \theta(\tilde{x}^k) + (x - \tilde{x}^k)^T \left\{\beta A^T \left[A\tilde{x}^k - b - \frac{1}{\beta}\lambda^k\right]\right\} \geq 0, \quad \forall x \in \mathcal{X},$$

or equivalently,

$$\tilde{x}^k \in \mathcal{X}, \quad \theta(x) - \theta(\tilde{x}^k) + (x - \tilde{x}^k)^T (-A^T \tilde{\lambda}^k) \geq 0, \quad \forall x \in \mathcal{X}. \tag{6.2}$$

Note that $\tilde{\lambda}^k = \lambda^k - \beta(A\tilde{x}^k - b)$ can be written as

$$(A\tilde{x}^k - b) + \frac{1}{\beta}(\tilde{\lambda}^k - \lambda^k) = 0. \tag{6.3}$$

Combining (6.2) and (6.3) together, we get $(\tilde{x}^k, \tilde{\lambda}^k) \in \Omega := \mathcal{X} \times \Re^m$ and

$$\theta(x) - \theta(\tilde{x}^k) + \begin{pmatrix} x - \tilde{x}^k \\ \lambda - \tilde{\lambda}^k \end{pmatrix}^T \left\{ \begin{pmatrix} -A^T\tilde{\lambda}^k \\ A\tilde{x}^k - b \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{\beta}(\tilde{\lambda}^k - \lambda^k) \end{pmatrix} \right\} \geq 0, \quad \forall (x, \lambda) \in \Omega. \tag{6.4}$$

Obviously, (6.4) coincides with (3.2) according to the specification given in (2.1b) and the metric proximal parameter given by

$$Q = \begin{pmatrix} 0 \\ \frac{1}{\beta} I_m \end{pmatrix}.$$

We thus can obtain a relaxed ALM for (1.2) by specifying the conceptual algorithmic framework of the relaxed PPA (3.2)–(3.3). Note that the scheme (6.1) only requires λ^k during its iterations. Thus, the essential coordinates of w is $v = \lambda$ in (6.4) and the metric proximal regularization matrix Q is thin. Accordingly, we only relax the coordinates λ in the relaxation step.

Algorithm 6.1 A relaxed augmented Lagrangian method for (1.2)

Let $\gamma \in (0, 2)$ and $\beta > 0$ be given. With the initial iterate λ^0, the iterate scheme is

1. **PPA step**: generate $\tilde{w}^k$ via solving

$$\begin{cases} \tilde{x}^k = \arg\min\{\theta(x) + \frac{\beta}{2}\|Ax - b - \frac{1}{\beta}\lambda^k\|^2 | x \in \mathcal{X}\}, \\ \tilde{\lambda}^k = \lambda^k - \beta(A\tilde{x}^k - b). \end{cases} \tag{6.5}$$

2. **Relaxation step**: generate the new iterate λ^{k+1} via

$$\lambda^{k+1} = \lambda^k - \gamma\big(\lambda^k - \tilde{\lambda}^k\big). \tag{6.6}$$

Remark 6.1 Note that the requirement (3.4) is met by choosing $H = \frac{1}{\beta} I_m$. In (6.4), the first n rows of the metric proximal parameter Q are all zero, which means the variable x (recall it is called intermediate variable) is not proximally regularized and only the variable λ is regularized. This coincides with an application of the idea of partial PPA in some references such as [23, 34, 38] to the MVI (2.1a), (2.1b).

Remark 6.2 Because of the simplicity of the relaxation step (6.6), the proposed Algorithm 6.1 and the ALM (1.6) are of the same difficulty to implement numerically.

6.2 The relaxed customized PPA for (1.2) in [30]

In various areas, we witness such a situation of (1.2) where the objective function $\theta(x)$ itself is easy in the sense that its resolvent operator has a closed-form representation or it can be efficiently solved up to a high precision. Here, the resolvent operator of the convex function θ is defined as

$$\left(I + \frac{1}{\beta}\partial\theta\right)^{-1}(a) = \operatorname{Arg\,min}\left\{\theta(x) + \frac{\beta}{2}\|x - a\|^2 \,\middle|\, x \in \Re^n\right\}, \tag{6.7}$$

for any given $a \in \Re^n$ and $\beta > 0$, see [45]. However, even for a function $\theta(x)$ whose resolvent operator is easy to evaluate, the evaluation of

$$\left(A^T A + \frac{1}{\beta}\partial\theta\right)^{-1}\big(A^T a\big) = \operatorname{Arg\,min}\left\{\theta(x) + \frac{\beta}{2}\|Ax - a\|^2 \,\middle|\, x \in \Re^n\right\},$$

could be still difficult provided that the matrix A is not identity. An illustrative example is the basis pursuit problem (see [9]) which falls exactly into the model (1.2) with $\theta(x) = \|x\|_1$ ($\|x\|_1 := \sum_{i=1}^n |u_i|$ for inducing the sparsity) and $A \in \Re^{m\times n}$ (with $m \ll n$). For such a scenario, although the ALM (1.6) alleviates the difficulty resulted by the linear constrains, the resulting ALM subproblem (1.6) is still difficult enough to require inner iterations to pursuit an approximate solution, and the implementation of the ALM (1.6) might be resistive from the numerical point of view. This difficulty, however, can be removed completely by the customized PPA developed in [30]. In [30], it was suggested to choose

$$Q = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix},$$

for (3.2) where the positive parameters r and s are required to satisfy $rs > \|A^TA\|$ for the purpose of ensuring the positive definiteness of Q. With this customized choice of Q, the PPA subproblem (3.2) reduces to

$$\begin{cases} \tilde{x}^k = \operatorname{Arg\,min}\{\theta(x) + \frac{r}{2}\|x - [x^k + \frac{1}{r}A^T\lambda^k]\|^2 \mid x \in \mathcal{X}\}, \\ \tilde{\lambda}^k = \lambda^k - \frac{1}{s}[A(2\tilde{x}^k - x^k) - b], \end{cases} \tag{6.8}$$

whose difficulty of implementation amounts to evaluating the resolvent operator of $\theta(x)$. Therefore, the customized PPA in [30] alleviates the difficulty of the ALM subproblem in (1.6) tangibly for some concrete applications.

Algorithm 6.2 The relaxed customized PPA for (1.2) in [30]

Let $\gamma \in (0, 2)$ be given, the positive scalars r and s be required to satisfy $rs > \|A^TA\|$. With the initial iterate $w^0 = (x^0, \lambda^0)$, the iterate scheme is

1. **PPA step**: generate $(\tilde{x}^k, \tilde{\lambda}^k)$ via

$$\begin{cases} \tilde{x}^k = \operatorname{Arg\,min}\{\theta(x) + \frac{r}{2}\|x - [x^k + \frac{1}{r}A^T\lambda^k]\|^2 \mid x \in \mathcal{X}\}, \\ \tilde{\lambda}^k = \lambda^k - \frac{1}{s}[A(2\tilde{x}^k - x^k) - b]. \end{cases} \tag{6.9}$$

2. **Relaxation step**: generate the new iterate w^{k+1} via

$$w^{k+1} = w^k - \gamma\big(w^k - \tilde{w}^k\big). \tag{6.10}$$

Note that $\tilde{x}^k$ and $\tilde{\lambda}^k$ are both required by the scheme (6.8). Accordingly, all the coordinates of w should be regularized proximally, the entire variable w should be relaxed, and the requirement (3.4) is met by taking $H = Q$. For completeness, we list the customized PPA in [30].

Remark 6.3 Note the update formula for $\tilde{\lambda}^k$ in the PPA step (6.9) is neither $\tilde{\lambda}^k = \lambda^k - \frac{1}{s}[Ax^k - b]$ nor $\tilde{\lambda}^k = \lambda^k - \frac{1}{s}[A\tilde{x}^k - b]$. As analyzed in [30], this particular form is a special case of (3.2) with the mentioned choice of Q.

7 A relaxed customized PPA for (1.3)

In this section, we review the generalized ADMM in [12] which can be regarded as customized applications of the relaxed PPA (3.2)–(3.3) for the separable model (1.3).

As we have mentioned, the ADMM (1.7) is a benchmark solver for (1.3) and it is more efficient than the straightforward application of the ALM (1.6) to (1.3). In [7], it is demonstrated that the generalized ADMM proposed in [12] can be recovered by the proposing relaxed customized PPA when the partial metric proximal parameter in (3.2) is chosen as

$$Q = \begin{pmatrix} 0 & 0 \\ \beta A_2^T A_2 & -A_2^T \\ -A_2 & \frac{1}{\beta} I \end{pmatrix}.$$

Note with this customized choice of Q, it is implied that $v = (x_2, \lambda)$ in (3.2). For completeness, we list the generalized ADMM in [12] below as Algorithm 7.1.

Algorithm 7.1 The generalized ADMM for (1.3) in [12]

Let $\gamma \in (0, 2)$ and $\beta > 0$ be given. With the initial iterate (x_2^0, λ^0), the iterate scheme is

1. **PPA step**: generate $\tilde{w}^k$ via solving

$$\begin{cases} \tilde{x}_1^k = \operatorname{Arg\,min}\{\theta_1(x_1) + \frac{\beta}{2}\|(A_1 x_1 + A_2 x_2^k - b) - \frac{1}{\beta}\lambda^k\|^2 \mid x_1 \in \mathcal{X}_1\}, \\ \tilde{\lambda}^k = \lambda^k - \beta(A_1 \tilde{x}_1^k + A_2 x_2^k - b), \\ \tilde{x}_2^k = \operatorname{Arg\,min}\{\theta_2(x_2) + \frac{\beta}{2}\|(A_1 \tilde{x}_1^k + A_2 x_2 - b) - \frac{1}{\beta}\tilde{\lambda}^k\|^2 \mid x_2 \in \mathcal{X}_2\}. \end{cases} \tag{7.1}$$

2. **Relaxation step**: generate the new iterate v^{k+1} by

$$\begin{pmatrix} x_2^{k+1} \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} x_2^k \\ \lambda^k \end{pmatrix} - \gamma \begin{pmatrix} x_2^k - \tilde{x}_2^k \\ \lambda^k - \tilde{\lambda}^k \end{pmatrix}. \tag{7.2}$$

Remark 7.1 Note that the requirement (3.4) is met for Algorithm 7.1 by choosing

$$H = \begin{pmatrix} \beta A_2^T A_2 & -A_2^T \\ -A_2 & \frac{1}{\beta} I \end{pmatrix},$$

which is a symmetric square sub-matrix of Q. In addition, it is easy to verify that

$$H = \frac{1}{\beta} \begin{pmatrix} \beta A_2^T \\ -I \end{pmatrix} \begin{pmatrix} \beta A_2 & -I \end{pmatrix}.$$

Thus, it is positive semi-definite for any $\beta > 0$. For this case, the first n rows of the metric proximal parameter Q are all zero, which means the variable x_1 is not proximally regularized and only the variables (x_2, λ) are regularized. This also coincides with an application of the idea of partial PPA in some references such as [23, 34, 38] to the MVI (2.2a), (2.2b).

8 A relaxed customized PPA for (1.4)

In this section, we discuss how to develop customized PPAs for the model (1.4) whose MVI reformulation is given by (2.5a), (2.5b). We show that the properties of θ_i's can be exploited individually by choosing the metric proximal parameters appropriately in (3.2), and consequently the relaxed PPA (3.2)–(3.3) can be specified into some concrete customized versions for the model (1.4). For simplicity, our discussion in this section only focuses on the circumstance where the tasks

$$\min\left\{\theta_i(x_i) + \frac{\beta}{2}\|A_i x_i - a\|^2 \,|\, x_i \in \mathcal{X}_i\right\}, \qquad i = 1, \ldots, K, \tag{8.1}$$

are easy for any given $\beta > 0$ and $a \in \Re^m$. More specifically, the relaxed customized PPA for (1.4) is summarized as Algorithm 8.1. where the primal variables x_i's are updated prior to the dual variable λ.

Algorithm 8.1 A relaxed customized PPA for (1.4)

Let $\gamma \in (0, 2)$ and $\beta > 0$ be given. With the initial iterate $w^0 = (x_1^0, \ldots, x_K^0, \lambda^0)$, the iterate scheme is

1. **PPA step:** generate $\tilde{w}^k$ via solving

$$\tilde{x}_i^k = \operatorname{Arg\,min}\left\{\theta_i(x_i) + \frac{\beta}{2K}\left\|\left[K A_i\left(x_i - x_i^k\right) + \sum_{j=1}^K A_j x_j^k - b\right] - \frac{1}{\beta}\lambda^k\right\|^2 \,\middle|\, x_i \in \mathcal{X}_i\right\}, \tag{8.2a}$$

where $i = 1, , \ldots, K$;

$$\tilde{\lambda}^k = \lambda^k - \beta\left(\sum_{j=1}^K A_j \tilde{x}_j^k - b\right); \tag{8.2b}$$

2. **Relaxation step**: generate the new iterate w^{k+1} by

$$\begin{pmatrix} x_1^{k+1} \\ \vdots \\ x_K^{k+1} \\ \lambda^{k+1} \end{pmatrix} = \begin{pmatrix} x_1^k \\ \vdots \\ x_K^k \\ \lambda^k \end{pmatrix} - \gamma \begin{pmatrix} x_1^k - \tilde{x}_1^k \\ \vdots \\ x_K^k - \tilde{x}_K^k \\ \lambda^k - \tilde{\lambda}^k \end{pmatrix}. \tag{8.3}$$

Now, we show that the step (8.2a), (8.2b) is essentially a special case of the PPA (3.2) with a customized choice of the metric proximal parameter Q. First, by deriving the first-order optimality condition of (8.2a), we have

$$\tilde{x}_i^k \in \mathcal{X}_i,$$

$$\theta_i(x_i) - \theta_i(\tilde{x}_i^k) + (x_i - \tilde{x}_i^k)^T \left\{ A_i^T \left[\beta \left(K A_i (\tilde{x}_i^k - x_i^k) + \sum_{j=1}^{K} A_j x_j^k - b \right) - \lambda^k \right] \right\} \geq 0,$$

$$\forall x_i \in \mathcal{X}_i,$$

for $i = 1, 2, \dots, K$. By using (8.2b), it can be rewritten as $\tilde{x}_i^k \in \mathcal{X}_i$ and

$$\begin{aligned} &\theta_i(x_i) - \theta_i(\tilde{x}_i^k) \\ &\quad + (x_i - \tilde{x}_i^k)^T \left\{ -A_i^T \tilde{\lambda}^k + \beta A_i^T \left[K A_i (\tilde{x}_i^k - x_i^k) - \sum_{j=1}^{K} A_j (\tilde{x}_j^k - x_j^k) \right] \right\} \geq 0, \\ &\quad \forall x \in \mathcal{X}. \end{aligned} \tag{8.4}$$

Note that the equation (8.2b) can be written as

$$\left(\sum_{j=1}^{K} A_j \tilde{x}_j^k - b \right) + \frac{1}{\beta} (\tilde{\lambda}^k - \lambda^k) = 0. \tag{8.5}$$

Combining (8.4) and (8.5) together, we get $\tilde{w}^k = (\tilde{x}^k, \dots, \tilde{x}_K^k, \tilde{\lambda}^k) \in \Omega$ such that

$$\theta(u) - \theta(\tilde{u}^k) + \begin{pmatrix} x_1 - \tilde{x}_1^k \\ \vdots \\ x_K - \tilde{x}_K^k \\ \lambda - \tilde{\lambda}^k \end{pmatrix}^T \left\{ \begin{pmatrix} -A_1^T \tilde{\lambda}^k \\ \vdots \\ -A_K \tilde{\lambda}^k \\ \sum_{i=1}^K A_i \tilde{x}_i^k - b \end{pmatrix} \right.$$
$$\left. + \begin{pmatrix} K\beta A_1^T A_1 (\tilde{x}_1^k - x_1^k) - \beta A_1^T \sum_{j=1}^K A_j (\tilde{x}_j^k - x_j^k) \\ \vdots \\ K\beta A_K^T A_K (\tilde{x}_K^k - x_K^k) - \beta A_K^T \sum_{j=1}^K A_j (\tilde{x}_j^k - x_j^k) \\ \frac{1}{\beta} (\tilde{\lambda}^k - \lambda^k) \end{pmatrix} \right\} \geq 0, \quad \forall w \in \Omega,$$

which coincides with (3.2) with the specification given in (2.3) and (2.5b), and the metric proximal parameter is

$$Q = \begin{pmatrix} K\beta A_1^T A_1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & K\beta A_K^T A_K & 0 \\ 0 & \cdots & 0 & \frac{1}{\beta} I_m \end{pmatrix} - \begin{pmatrix} \beta A_1^T A_1 & \cdots & \beta A_1^T A_K & 0 \\ \vdots & \ddots & \vdots & \vdots \\ \beta A_K^T A_1 & \cdots & \beta A_K^T A_K & 0 \\ 0 & \cdots & 0 & 0 \end{pmatrix}. \tag{8.6}$$

Obviously, the matrix

$$K \cdot \operatorname{diag}\left(A_1^T A_1, \ldots, A_K^T A_K\right) - \begin{pmatrix} A_1^T A_1 & \cdots & A_1^T A_K \\ \vdots & \ddots & \vdots \\ A_K^T A_1 & \cdots & A_K^T A_K \end{pmatrix}$$

is positive semi-definite. Hence, the matrix Q given in (8.6) is also positive semi-definite for any $\beta > 0$.

Remark 8.1 Note that all the iterate $(x_1^k, x_2^k, \ldots, x_K^k, \lambda^k)$ are required to implement Algorithm 8.1. Thus, all the coordinates of w need to be proximally regularized in (8.2a), (8.2b) and relaxed in (8.3). Accordingly, we take $H = Q$ and the requirement (3.4) can be met.

Remark 8.2 All the x_i-subproblems of Algorithm 8.1 are in the form of (8.1), and they are eligible for parallel computation. This feature is particularly favorable when K is large and the x_i-subproblems are of the same difficulty.

9 A relaxed customized PPA for (1.5)

In this section, we discuss how to design customized PPAs for the saddle-point problem (1.5). Recall the MVI reformulation (2.6a), (2.6b) of the model (1.5). Our purpose is to exploit the properties of θ_1 and θ_2 individually, and the resulting subproblems are of the same difficulty as evaluating the resolvent operators of θ_1 and θ_2 individually. Let us present the algorithm first in Algorithm 9.1.

Algorithm 9.1 A relaxed customized PPA for (1.5)

Let $\gamma \in (0, 2)$ be given, the positive scalars r and s be required to satisfy $rs \geq \|A^T A\|$. With the initial iterate $w^0 = (x^0, y^0)$, the iterate scheme is

1. **PPA step:** generate $\tilde{w}^k$ via solving

$$\tilde{x}^k = \operatorname{Arg\,min}\left\{\theta_1(x) + \frac{r}{2}\left\|x - \left[x^k + \frac{1}{r} A^T y^k\right]\right\|^2 \,\middle|\, x \in \mathcal{X}\right\}, \tag{9.1a}$$

$$\tilde{y}^k = \operatorname{Arg\,min}\left\{\theta_2(y) + \frac{s}{2}\left\|y - \left[y^k - \frac{1}{s} A\left(2\tilde{x}^k - x^k\right)\right]\right\|^2 \,\middle|\, y \in \mathcal{Y}\right\}. \tag{9.1b}$$

2. **Relaxation step**: generate the new iterate w^{k+1} by

$$\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} x^k \\ y^k \end{pmatrix} - \gamma \begin{pmatrix} x^k - \tilde{x}^k \\ y^k - \tilde{y}^k \end{pmatrix}. \tag{9.2}$$

Now, we show that the step (9.1a), (9.1b) is a special case of the PPA (3.2) with a customized metric proximal parameter. Hence, the relaxation step (9.2) is effective. In fact, by deriving the first-order optimality condition, it follows from (9.1a) that

$$\tilde{x}^k \in \mathcal{X}, \quad \theta_1(x) - \theta_1(\tilde{x}^k) + (x - \tilde{x}^k)^T \{r(\tilde{x}^k - x^k) - A^T y^k\} \geq 0, \quad \forall x \in \mathcal{X},$$

and it can be written as

$$\begin{aligned} &\tilde{x}^k \in \mathcal{X}, \\ &\theta_1(x) - \theta_1(\tilde{x}^k) + (x - \tilde{x}^k)^T \{(-A^T \tilde{y}^k) + r(\tilde{x}^k - x^k) + A^T(\tilde{y}^k - y^k)\} \geq 0, \\ &\quad \forall x \in \mathcal{X}. \end{aligned} \tag{9.3}$$

Similarly, from (9.1b), we have

$$\tilde{y}^k \in \mathcal{Y}, \quad \theta_2(y) - \theta_2(\tilde{y}^k) + (y - \tilde{y}^k)^T \{s(\tilde{y}^k - y^k) + A(2\tilde{x}^k - x^k)\} \geq 0, \quad \forall y \in \mathcal{Y},$$

and it can be written as

$$\begin{aligned} &\tilde{y}^k \in \mathcal{Y}, \\ &\theta_2(y) - \theta_2(\tilde{y}^k) + (y - \tilde{y}^k)^T \{A\tilde{x}^k + s(\tilde{y}^k - y^k) + A(\tilde{x}^k - x^k)\} \geq 0, \\ &\quad \forall y \in \mathcal{Y}. \end{aligned} \tag{9.4}$$

Combining (9.3) and (9.4) together, we get $\tilde{w}^k = (\tilde{x}^k, \tilde{y}^k) \in \Omega$ such that

$$\theta(u) - \theta(\tilde{u}^k) + \begin{pmatrix} x - \tilde{x}^k \\ y - \tilde{y}^k \end{pmatrix}^T \left\{ \begin{pmatrix} -A^T \tilde{y}^k \\ A\tilde{x}^k \end{pmatrix} + \begin{pmatrix} r(\tilde{x}^k - x^k) + A^T(\tilde{y}^k - y^k) \\ A(\tilde{x}^k - x^k) + s(\tilde{y}^k - y^k) \end{pmatrix} \right\} \geq 0,$$
$$\forall w \in \Omega,$$

which coincides with (3.2) with the specification given in (2.6a), (2.6b) and the metric proximal parameter is

$$Q = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix},$$

where the positive semi-definiteness of Q is ensured by the condition $rs \geq \|A^T A\|$.

Remark 9.1 Since the $(k+1)$-th iteration of Algorithm 9.1 requires both x^k and y^k, all the coordinates of w (i.e, x and y) need to be proximally regularized in the PPA step (9.1a), (9.1b) and relaxed in the relaxation step (9.2). Accordingly, it is easy to verify that the requirement (3.4) is met by choosing $H = Q$.

It is easy to verify that the PPA step (3.2) can be representable by finding $\tilde{w}^k = (\tilde{x}^k, \tilde{y}^k) \in \Omega$ such that

$$\theta(u) - \theta(\tilde{u}^k) + \begin{pmatrix} x - \tilde{x}^k \\ y - \tilde{y}^k \end{pmatrix}^T \left\{ \begin{pmatrix} -A^T \tilde{y}^k \\ A\tilde{x}^k \end{pmatrix} + \begin{pmatrix} r(\tilde{x}^k - x^k) - A^T(\tilde{y}^k - y^k) \\ -A(\tilde{x}^k - x^k) + s(\tilde{y}^k - y^k) \end{pmatrix} \right\} \geq 0,$$
$$\forall w \in \Omega,$$

which coincides with (3.2) with the specification given in (2.6a), (2.6b) and the metric proximal parameter is

$$Q = \begin{pmatrix} rI_n & -A^T \\ -A & sI_m \end{pmatrix},$$

where the positive semi-definiteness of Q is ensured by the condition $rs \geq \|A^T A\|$.

10 Conclusions

We study both the convex minimization problem with linear constraints and the saddle-point problem uniformly via their mixed variational inequality reformulations, and propose a unified methodology to design structure-exploited algorithms based on the classical proximal point algorithm (PPA). Our idea is to specify the PPA with customized choices of the metric proximal parameter in accordance with these models' MVI reformulations. The resulting algorithms are in the decomposition nature, with the possibility of exploiting the properties/structures of considered models effectively. This unified customized PPA approach makes it extremely easy to accelerate some existing benchmark methods (e.g. the augmented Lagrangian method, the alternating direction method of multipliers, the split inexact Uzawa method and a class of primal-dual methods), and to develop some customized algorithms for the considered models as well. The global convergence and a worst-case $O(1/t)$ convergence rate in ergodic sense for this series of algorithm can be established easily in a unified way.

In our analysis, we relax the conventional positive definiteness assumption on the metric proximal parameters in PPA literature to only positive semi-definiteness, and relax the full proximal regularization to only partial proximal regularization, see Q in (3.2). But, we still keep the symmetry requirement on H. That is, all the specified choices of the matrix H in Sects. 6–9 are forced to be symmetric. Inspired by the asymmetric proximal parameter in [26], we are interested in relaxing H to be asymmetric, and also relaxing the requirement (3.4) to identifying a possibly asymmetric square sub-matrix of Q (denoted by M) such that

$$w^T Qv = v^T Mv.$$

With these relaxed requirements on the customized choices of metric proximal parameter, a new series of algorithm based on the same idea of customizing PPA can be proposed for the models (1.2)–(1.5). Because of the relaxation of the symmetric requirement on M, we expect that the involved parameters may be further relaxed than that in the algorithms proposed in this paper.

References

1. Benzi, M., Golub, G.H., Liesen, J.: Numerical solution of saddle point problems. Acta Numer., 1–137 (2005)
2. Blum, E., Oettli, W.: Mathematische Optimierung. Grundlagen und Verfahren. Ökonometrie und Unternehmensforschung. Springer, Berlin (1975)

3. Bonnans, J.F., Gilbert, J.C., Lemarechal, C., Sagastizaba, C.A.: A family of variable-metric proximal methods. Math. Program. **68**, 15–47 (1995)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**, 1–122 (2010)
5. Burke, J.V., Qian, M.J.: A variable metric proximal point algorithm for monotone operators. SIAM J. Control Optim. **37**, 353–375 (1998)
6. Burke, J.V., Qian, M.J.: On the superlinear convergence of the variable metric proximal point algorithm using Broyden and BFGS matrix secant updating. Math. Program. **88**, 157–181 (2000)
7. Cai, X.J., Gu, G., He, B.S., Yuan, X.M.: A proximal point algorithm revisit on alternating direction method of multipliers. Sci. China Math. **56**(10), 2179–2186 (2013)
8. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problem with applications to imaging. J. Math. Imaging Vis. **40**, 120–145 (2011)
9. Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20**, 33–61 (1998)
10. Condat, L.: A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. J. Optim. Theory Appl. (to appear)
11. Douglas, J., Rachford, H.H.: On the numerical solution of the heat conduction problem in 2 and 3 space variables. Trans. Am. Math. Soc. **82**, 421–439 (1956)
12. Eckstein, J., Bertsekas, D.P.: On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. **55**, 293–318 (1992)
13. Eckstein, J., Fukushima, M.: Some reformulation and applications of the alternating directions method of multipliers. In: Hager, W.W., et al. (eds.) Large Scale Optimization: State of the Art, pp. 115–134. Kluwer Academic, Dordrecht (1994)
14. Esser, E., Zhang, X.Q., Chan, T.F.: A general framework for a class of first order primal-dual algorithms for convex pptimization in imaging science. SIAM J. Imaging Sci. **3**, 1015–1046 (2010)
15. Facchinei, F., Pang, J.S.: Finite-Dimensional Variational Inequalities and Complementarity Problems, vol. I. Springer Series in Operations Research. Springer, New York (2003)
16. Fortin, M., Brezzi, F.: Mixed and Hybrid Element Methods. Springer Series in Computational Mathematics. Springer, Berlin (1991)
17. Fukushima, M.: Application of the alternating directions method of multipliers to separable convex programming problems. Comput. Optim. Appl. **2**, 93–111 (1992)
18. Gabay, D., Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite-element approximations. Comput. Math. Appl. **2**, 17–40 (1976)
19. Gabay, D.: Applications of the method of multipliers to variational inequalities. In: Fortin, M., Glowinski, R. (eds.) Augmented Lagrange Methods: Applications to the Solution of Boundary-Valued Problems, pp. 299–331. North Holland, Amsterdam (1983)
20. Glowinski, R., Marrocco, A.: Approximation par éléments finis d'ordre un et résolution par pénalisation-dualité d'une classe de problémes non linéaires. RAIRO **2**, 41–76 (1975)
21. Gol'shtein, E.G., Tret'yakov, N.V.: Modified Lagrangian in convex programming and their generalizations. Math. Program. Stud. **10**, 86–97 (1979)
22. Güler, O.: On the convergence of the proximal point algorithm for convex minimization. SIAM J. Control Optim. **29**, 403–419 (1991)
23. Ha, C.D.: A generalization of the proximal point algorithm. SIAM J. Control Optim. **28**, 503–512 (1990)
24. 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)
25. Han, W.M., Reddy, B.D.: On the finite element method for mixed variational inequalities airising in elastoplasticity. SIAM J. Numer. Anal. **32**, 1778–1807 (1995)
26. He, B.S., Fu, X.L., Jiang, Z.K.: Proximal point algorithm using a linear proximal term. J. Optim. Theory Appl. **141**, 299–319 (2009)
27. He, B.S., Tao, M., Yuan, X.M.: A splitting method for separable convex programming. IMA J. Numer. Anal. (to appear)
28. He, B.S., Tao, M., Yuan, X.M.: Alternating direction method with Gaussian back substitution for separable convex programming. SIAM J. Optim. **22**, 313–340 (2012)
29. He, B.S., Yang, H.: Some convergence properties of a method of multipliers for linearly constrained monotone variational inequalities. Oper. Res. Lett. **23**, 151–161 (1998)
30. He, B.S., Yuan, X.M., Zhang, W.X.: A customized proximal point algorithm for convex minimization with linear constraints. Comput. Optim. Appl. (to appear)

31. He, B.S., Yuan, X.M.: Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective. SIAM J. Imaging Sci. **5**, 119–149 (2012)
32. He, B.S., Yuan, X.M.: On the $O(1/n)$ convergence rate of Douglas-Rachford alternating direction method. SIAM J. Numer. Anal. **50**, 700–709 (2012)
33. Hestenes, M.R.: Multiplier and gradient methods. J. Optim. Theory Appl. **4**, 302–320 (1969)
34. Konnov, I.V.: Partial proximal point method for nonmonotone equilibrium problems. Optim. Methods Softw. **21**, 373–384 (2006)
35. Kontogiorgis, S., Meyer, R.R.: A variable-penalty alternating directions method for convex optimization. Math. Program. **83**, 29–53 (1998)
36. Lemaire, B.: Saddle point problems in partial differential equations and applications to linear quadratic differential games. Ann. Sc. Norm. Super. Pisa, Class Sci. 3^0 Sér. **27**, 105–160 (1973)
37. Martinet, B.: Regularisation, d'inéquations variationelles par approximations succesives. Rev. Fr. Inform. Rech. Oper. **4**, 154–159 (1970)
38. Medhi, D., Ha, C.D.: Generalized proximal point algorithms for convex optimization. J. Optim. Theory Appl. **88**, 475–488 (1996)
39. Monteiro, R.D.C., Svaiter, B.F.: Iteration-complexity of block-decomposition algorithms and the alternating direction method of multipliers. SIAM J. Optim. **23**(1), 475–507 (2013)
40. Moreau, J.J.: Proximité et dualité dans un espace hilbertien. Bull. Soc. Math. Fr. **93**, 273–299 (1965)
41. Pang, J.S., Fukushima, M.: Quasi-variational inequalities, generalized Nash equilibria, and multi-leader-follower games. Comput. Manag. Sci. **2**, 21–56 (2005)
42. Pock, T., Chambolle, A.: Diagonal preconditioning for first order primal-dual algorithms in convex optimization. In: IEEE International Conference on Computer Vision (ICCV) (2011)
43. Powell, M.J.D.: A method for nonlinear constraints in minimization problems. In: Fletcher, R. (ed.) Optimization. Academic Press, New York (1969)
44. Reddy, B.D.: Mixed variational inequalities arising in elastoplasticity. Nonlinear Anal. **19**, 1071–1089 (1992)
45. Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton (1970)
46. Rockafellar, R.T.: Augmented Lagrangians and applications of the proximal point algorithm in convex programming. Math. Oper. Res. **1**, 97–116 (1976)
47. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**, 877–898 (1976)
48. Rudin, L., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**, 227–238 (1992)
49. Sun, J., Zhang, S.: A modified alternating direction method for convex quadratically constrained quadratic semidefinite programs. Eur. J. Oper. Res. **207**, 1210–1220 (2010)
50. Vũ, B.C.: A splitting algorithm for dual monotone inclusions involving cocoercive operators. Manuscript (2011)
51. Zhang, X.Q., Burger, M., Osher, S.: A unified primal-dual algorithm framework based on Bregman iteration. J. Sci. Comput. **46**, 20–46 (2010)
52. Zhu, M., Chan, T.: An efficient primal-dual hybrid gradient algorithm for total variation image restoration. CAM Reports 08-34, UCLA, Center for Applied Mathematics (2008)