



Adaptive Proximal Point Algorithms for Total Variation Image Restoration

Ying Chen¹, Jian Wu², and Gaohang Yu^{1, *}

¹*School of Mathematics and Computer Sciences, Gannan Normal University, China.*

²*College of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, China*

Received: 8 December 2014; Accepted: 4 February 2015

Editor: Yunhai Xiao

Abstract Image restoration is a fundamental problem in various areas of imaging sciences. This paper presents a class of adaptive proximal point algorithms (APPA) with contraction strategy for total variational image restoration. In each iteration, the proposed methods choose an adaptive proximal parameter matrix which is not necessary symmetric. In fact, there is an inner extrapolation in the prediction step, which is followed by a correction step for contraction. And the inner extrapolation is implemented by an adaptive scheme. By using the framework of contraction method, global convergence result and a convergence rate of $O(1/N)$ could be established for the proposed methods. Numerical results are reported to illustrate the efficiency of the APPA methods for solving total variation image restoration problems. Comparisons with the state-of-the-art algorithms demonstrate that the proposed methods are comparable and promising.

Keywords Total Variation, Image Restoration, Proximal Point Algorithm, Primal-Dual Method

AMS 2010 subject classifications 68U10, 90C30

DOI: 10.19139/soic.v3i1.122

1. Introduction

Image restoration is a fundamental problem in various areas of applied sciences such as medical imaging, microscopy, astronomy, film restoration, and image and video coding. Image reconstruction is to recover an original clean image from the degraded image,

$$f = Kw + n, \quad (1)$$

where w is the original image and f represents the given observed image, K is a line blurring operator, and n is an additive zero-mean Gaussian white noise. In general, a regularization method need to be used in image restoration since (1) is an ill-conditioned problem. One of the most popular regularization methods is total variational (TV) regularization technique which was due to Rudin, Osher, and Fatemi [29]. A discrete version of the TV deblurring problem is given by

$$\min_w \|w\|_{TV} + \frac{\lambda}{2} \|Kw - f\|^2, \quad (2)$$

where $\lambda > 0$ is a fit-to-data parameter. Here, $\|\cdot\|$ denotes the usual Euclidean norm (ℓ_2) of a vector. $\|w\|_{TV}$ is the discrete TV semi-norm defined by $\|w\|_{TV} = \sum_{1 \leq i, j \leq n} \|(\nabla w)_{i,j}\|$, and ∇w is the discrete gradient operator.

*Correspondence to: School of Mathematics and Computer Sciences, Gannan Normal University, Ganzhou, 341000, China. Email: maghyu@163.com

In [29], the authors proposed a time marching scheme to solve the associated Euler-Lagrange equation of (2). Vogel and Oman proposed in [34] a “lagged diffusivity” fixed point iteration method for solving the same Euler-Lagrange equation of (2) directly. Chambolle in [11] gave a semi-implicit gradient descent algorithm based on the dual formulation. In [36], Yu *et al* accelerated the Chambolle gradient projection method for total variation image restoration via choosing Barzilai-Borwein stepsize instead of the fix time step. In [38], the authors developed duality-based gradient projection algorithms for total variation image restoration problems. Beck and Teboulle in [7] proposed a fast iterative soft-thresholding algorithm (FISTA) for the TV-based image deblurring problem.

In recent years, many researchers [8, 9, 10, 12, 13, 18, 22, 28, 32] study the TV minimization via solving the following saddle-point problem:

$$\min_{y \in \mathcal{Y}} \max_{x \in \mathcal{X}} \Phi(x, y) := y^T Ax + \frac{\lambda}{2} \|By - z\|^2, \quad (3)$$

in which, the image matrix w (resp. f) is reordered row-wisely into a vector y (resp. z). $\mathcal{Y} \subset R^N$, x is the dual variable, $\mathcal{X} = \{x : x \in R^{2N}, \|x_l\| \leq 1, \text{ for } l = 1, 2, \dots, N\}$. The operator $A = [A_1, A_2, \dots, A_N]^T \in R^{N \times 2N}$ is the negative divergence operator. B is the discretization of the blurring operator K .

Since the seminal work of Arrow, Hurwicz and Uzawa [1, 4], classical methods based on the gradient/subgradient for solving the saddle point problem have been proposed. Particularly, to achieve fast convergence, a class of inexact Uzawa methods were proposed and analyzed in [4] for linear saddle points problem by introducing preconditioning matrices. In [37], the authors generalized Bregman operator splitting (BOS) algorithm in a primal-dual system, which can be called as a variant of inexact Uzawa method [1]. In [39], Zhu and Chan proposed a primal-dual hybrid gradient (PDHG) algorithm to solved the saddle-point problem. Recently, In [18], Esser, Zhang and Chan proposed a modified version of PDHG (PDHGMu and PDHGMp). More recently, Chen *et al* in [13] designed a primal-dual fixed point algorithm based on the proximity operator (PDFP²O_κ for κ ∈ [0, 1)), which was motivated by proximal forward-backward splitting (PFBS) proposed in [14] and fixed point algorithms based on the proximity operator (FP²O) for image denoising in [26].

Most of these work solve the saddle-point problem (3) via following primal-dual procedure:

Primal-Dual procedure for solving (3)

Let $\tau > 0, \sigma > 0$ and $\theta \in \mathcal{R}$. Given (x^k, y^k) , the new iterate (x^{k+1}, y^{k+1}) is generated by:

$$\begin{cases} y^{k+1} = \text{Arg min}_{y \in \mathcal{Y}} \{ \sigma \Phi(x^k, y) + \frac{1}{2} \|y - y^k\|^2 \}, & (4a) \\ \bar{y}^k = y^{k+1} + \theta(y^{k+1} - y^k), & (4b) \\ x^{k+1} = \text{Arg max}_{x \in \mathcal{X}} \{ \tau \Phi(x, \bar{y}^k) - \frac{1}{2} \|x - x^k\|^2 \}. & (4c) \end{cases}$$

If $\theta = 0$, the primal-dual procedure (4) reduces to the Arrow- Hurwicz algorithm in [1]. When $\theta = 0$ with τ, σ are made adaptive, the primal-dual procedure (4) become primal-dual hybrid gradient (PDHG) algorithm proposed in [39]. When $\theta \in [0, 1]$, the primal-dual procedure (4) covers the primal-dual algorithms proposed in [12, 18]. Moreover, both the adaptive parameters are presented in [12, 18] to accelerate the primal-dual algorithms. In [12], the authors show that for B being the identity, the primal-dual procedure (4) reduces to the Douglas Rachford splitting algorithm [16], and it can also be regarded as a preconditioned version of the alternating direction method of multipliers. In [22], a PPA perspective of (4) is given by He and Yuan. When $\theta = 1$, primal-dual procedure (4) can be viewed as a classical PPA, thus its convergence is guaranteed. Otherwise, the primal-dual procedure (4) become the prediction step, which is followed by a correction step for contraction/convergence. Recently, some nonlinear PPA methods were given and analysed in [35].

In this paper, we will study (4) from the perspective of PPA. The rest of this paper is organized as follows. In Section 2, we review some basic knowledge of variational inequalities (VI) and the PPA method. Section 3 presents an adaptive PPA, and establish its global convergence in the framework of contraction method. In Section 4, some preliminary numerical results and performance comparisons are reported. Finally, we have a conclusion section.

2. Preliminaries

2.1. The VI reformulation of (3)

Let (x^*, y^*) be a solution of the saddle-point problem (3) satisfies:

$$\max_{x \in \mathcal{X}} \Phi(x, y^*) \leq \Phi(x^*, y^*) \leq \min_{y \in \mathcal{Y}} \Phi(x^*, y).$$

Therefore, the saddle-point problem (3) can be transformed into a variational inequality problem $\text{VI}(\Omega, F)$ with F and Ω defined as follows. Finding $u^* \in \Omega$ such that:

$$\text{VI}(\Omega, F) : \quad (u - u^*)^T F(u^*) \geq 0, \quad \forall u \in \Omega, \quad (5)$$

where

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \quad F(u) = \begin{pmatrix} -A^T y \\ Ax + \lambda B^T (By - z) \end{pmatrix}, \quad \text{and } \Omega := \mathcal{X} \times \mathcal{Y}.$$

The mapping $F(u)$ is monotone, i.e. $(u - v)^T (F(u) - F(v)) \geq 0$, $\forall u, v \in \Omega$. Therefore, $\text{VI}(\Omega, F)$ is monotone and its solution set denoted by Ω^* is nonempty.

2.2. PPA method

One of the most well-known methods for $\text{VIP}(\Omega, F)$ is known as proximal point algorithm (PPA) [24], in which the new iterate point u^{k+1} is obtained by solving the following variational inequality problem

$$u^{k+1} \in \Omega, \quad (u - u^{k+1})^T (F(u^{k+1}) + \beta_k (u^{k+1} - u^k)) \geq 0, \quad \forall u \in \Omega. \quad (6)$$

However, solving the subproblem (6) exactly may be difficult or impossible in practice. So, it is essential to develop implementable algorithms [2, 5, 6, 17, 19, 20, 21, 30, 31, 33]. The PPA in the context of G-norm was proposed in [21]. The authors proposed a customized PPA for linearly constrained convex programming,

$$u^{k+1} \in \Omega, \quad (u - u^{k+1})^T (F(u^{k+1}) + G(u^{k+1} - u^k)) \geq 0, \quad \forall u \in \Omega,$$

where $G(u^{k+1} - u^k)$ is usually the gradient of a function and the corresponding Jacobian matrix. In [23], He *et al* proposed a class of linear PPA where the proximal parameter matrix G is fixed as a constant matrix. One main motivation in this paper is to develop an adaptive linear PPA.

3. Proposed Algorithm and Its Convergence

3.1. Algorithm

In this subsection, we present an adaptive proximal point algorithm model as follows.

APPA : Adaptive PPA method with Correction Step

Given an adaptive positive-definite matrix M_k , an arbitrary symmetric positive definite fixed matrix H and a constant $\gamma \in (0, 2)$.

Prediction Step: Produce a proximal point \tilde{u}^k which is the solution of the following VI:

$$u \in \Omega, \quad (u' - u)^T (F(u) + M_k(u - u^k)) \geq 0, \quad \forall u' \in \Omega. \quad (7a)$$

Correction Step: The new iterate is updated by

$$u^{k+1} = u^k - \alpha_k H^{-1} M_k (u^k - \tilde{u}^k), \quad (7b)$$

where

$$\alpha_k = \gamma \alpha_k^*, \quad \alpha_k^* = \frac{(u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k)}{\|H^{-1} M_k (u^k - \tilde{u}^k)\|_H^2}. \quad (7c)$$

In (7c), $\|\cdot\|_H^2$ is defined as $\|\cdot\|_H^2 = (\cdot)^T H (\cdot)$. If we translate the prediction step (7a) to the primal-dual procedure:

$$\begin{cases} \tilde{y}^k = \text{Arg min}_{y \in \mathcal{Y}} \{ \sigma \Phi(x^k, y) + \frac{1}{2} \|y - y^k\|^2 \} & (8a) \\ \bar{y}^k = \tilde{y}^k + \theta_k (\tilde{y}^k - y^k) & (8b) \\ \tilde{x}^k = \text{Arg max}_{x \in \mathcal{X}} \{ \tau \Phi(x, \bar{y}^k) - \frac{1}{2} \|x - x^k\|^2 \}, & (8c) \end{cases}$$

then the proximal parameter M_k and the H can be formulated as:

$$M_k = \begin{pmatrix} \frac{1}{\tau} I & -\theta_k A^T \\ -A & \frac{1}{\sigma} I \end{pmatrix}, \quad H = \begin{pmatrix} \frac{1}{\tau} I & \\ & \frac{1}{\sigma} I \end{pmatrix}, \quad (9)$$

and the positive parameters should satisfy the following requirement[22]:

$$\tau \sigma \frac{(1 + \theta_k)^2}{4} \|A^T A\| < 1, \quad \theta_k \in (-1, 1). \quad (10)$$

In order to establish the global convergence for the proposed APPA model, the adaptive metric proximal parameter matrix M_k need to satisfy the following assumptions:

A1: The sequence $\{M_k\}$ is uniformly positive-defined (not necessarily symmetric) matrix, i.e., there exist a constant c such that

$$u^t M_k u \geq c \|u\|^2, \quad \forall u \in R^n, k \in N^* \text{ and } u \neq 0. \quad (11)$$

A2: The sequence $\|M_k^T H^{-1} M_k\|$ is uniformly bounded, where H is an arbitrary symmetric positive definite fixed matrix.

Here and throughout the paper, $B : X \rightarrow Y$ is a continuous linear operator with induced norm

$$\|B\| = \max\{\|Bx\| : x \in X, \|x\| \leq 1\}.$$

As we can see in Appendix section, these conditions can be easily satisfied.

3.2. Convergence

This subsection is devoted to the convergence analysis of the proposed algorithm. The following theorem states that the APPA possesses Fejèr monotone contraction proposition.

Theorem 1

For any $\{u^*\} \in \Omega^*$, the sequence $\{u^k\}$ generated by the proposed APPA Algorithm satisfies

$$\|u^{k+1} - u^*\|_H^2 \leq \|u^k - u^*\|_H^2 - C \|u^k - \tilde{u}^k\|^2, \quad (12)$$

where $C > 0$ is a constant.

Proof. From (7b) and (7a), we have

$$\begin{aligned} & \|u^{k+1} - u^*\|_H^2 \\ &= \|(u^k - u^*) - \alpha_k H^{-1} M_k (u^k - \tilde{u}^k)\|_H^2 \\ &= \|u^k - u^*\|_H^2 - 2\alpha_k (u^k - u^*)^T M_k (u^k - \tilde{u}^k) + \alpha_k^2 \|H^{-1} M_k (u^k - \tilde{u}^k)\|_H^2 \\ &\leq \|u^k - u^*\|_H^2 - 2\alpha_k (u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k) + \alpha_k^2 \|H^{-1} M_k (u^k - \tilde{u}^k)\|_H^2 \\ &= \|u^k - u^*\|_H^2 - \gamma(2 - \gamma) \alpha_k^* (u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k). \end{aligned} \quad (13)$$

If the assumptions A1 and A2 hold, there respectively exist a constant c_0 and m such that

$$(u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k) \geq c_0 \|u^k - \tilde{u}^k\|^2, \quad \|M_k^T H^{-1} M_k\| \leq m.$$

Therefore

$$\begin{aligned} & \alpha_k^* (u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k) \\ &= \frac{(u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k)}{\|H^{-1} M_k (u^k - \tilde{u}^k)\|_H^2} (u^k - \tilde{u}^k)^T M_k (u^k - \tilde{u}^k) \\ &\geq \frac{c_0^2 \|u^k - \tilde{u}^k\|^4}{\|M_k^T H^{-1} M_k\| \|u^k - \tilde{u}^k\|^2} \\ &\geq \frac{c_0^2 \|u^k - \tilde{u}^k\|^2}{m}. \end{aligned} \tag{14}$$

By setting $C = \frac{\gamma(2-\gamma)c_0^2}{m}$, assertion (12) is proved. \square

The theorem above means the sequence $\{u^k\}$ generated by the proposed algorithm is Fejèr monotone with respect to Ω^* , hence the proposed algorithm (7) is a contraction method under the H -norm. He *et al* presented a analytic framework of contraction methods [22, 23], which simplifies convergence analysis. Now we are in a position to present the main convergence theorem of our algorithm.

Theorem 2

The sequence $\{u^k\}$ generated by the proposed algorithm (7) converges to some u^* which is a solution point of VIP(Ω , F).

Proof. It follows from (12) that $\{u^k\}$ is bounded and thus

$$\lim_{k \rightarrow \infty} \|u^k - \tilde{u}^k\| = 0.$$

Consequently, $\{\tilde{u}^k\}$ is also bounded. Let u^* be a cluster point of $\{\tilde{u}^k\}$ and there exist a subsequence $\{\tilde{u}_j^k\}$ that converges to u^* . For every \tilde{u}_j^k , we have

$$\tilde{u}_j^k \in \Omega, \quad (u - \tilde{u}_j^k)^T \{F(\tilde{u}_j^k) + M_k(u_j^k - \tilde{u}_j^k)\} \geq 0, \quad \forall u \in \Omega.$$

Since $\{\tilde{u}_j^k\} \rightarrow u^*$ and $\lim_{j \rightarrow \infty} \|u_j^k - \tilde{u}_j^k\| = 0$, we have

$$u^* \in \Omega, \quad (u - u^*)^T F(u^*) \geq 0, \quad \forall u \in \Omega,$$

and thus u^* is a solution point. Note that inequality (12) is true for all solution point of VI(Ω , F); hence, we have

$$\|u^{k+1} - u^*\|^2 \leq \|u^k - u^*\|^2, \quad \forall k > 0,$$

and thus the sequence $\{u^k\}$ converges to u^* . \square

3.3. Convergence rate

Classical linear PPA poses a convergence rate of $O(1/N)$ [15, 27]. We would show in this subsection that the proposed methods can reach an $O(1/N)$ convergence rate as well.

Lemma 1

Let the sequences $\{u^k\}$ and $\{\tilde{u}^k\}$ be generated by the proposed APPA Algorithm. Then, we have

$$(u - \tilde{u}^k)^T F(\tilde{u}^k) + \frac{1}{2\gamma\alpha_k^*} (\|u - u^k\|_H^2 - \|u - u^{k+1}\|_H^2) \geq (1 - \frac{1}{\gamma}) \|u^k - \tilde{u}^k\|_H^2. \tag{15}$$

Proof. It follows (7a) that

$$(u - \tilde{u}^k)^T F(\tilde{u}^k) \geq (u - \tilde{u}^k)^T M_k(u^k - \tilde{u}^k). \quad (16)$$

Form the correction step (7b) that $(u^k - \tilde{u}^k) = \frac{1}{\gamma\alpha_k^*} M_k^{-1} H(u^k - u^{k+1})$, we have

$$\begin{aligned} & (u - u^{k+1})^T M_k(u^k - \tilde{u}^k) \\ &= \frac{1}{\gamma\alpha_k^*} (u - u^{k+1})^T H(u^k - u^{k+1}) \\ &= \frac{1}{\gamma\alpha_k^*} (\|u - u^{k+1}\|_H^2 - \|u - u^k\|_H^2) + \frac{1}{\alpha_k^*} \|u^k - u^{k+1}\|_H^2. \end{aligned} \quad (17)$$

On the other hand, using the (7b) we have

$$\begin{aligned} & (u^{k+1} - \tilde{u}^k)^T M_k(u^k - \tilde{u}^k) \\ &= [(u^k - \tilde{u}^k) - \gamma\alpha_k^* H^{-1} M_k(u^k - \tilde{u}^k)]^T M_k(u^k - \tilde{u}^k) \\ &= \|(u^k - \tilde{u}^k)\|_{M_k}^2 - \gamma\alpha_k^* \|H^{-1} M_k(u^k - \tilde{u}^k)\|_H^2 \\ &= (1 - \gamma) \|(u^k - \tilde{u}^k)\|_{M_k}^2. \end{aligned} \quad (18)$$

By adding (17) and (18), using the the fact that $u^k - u^{k+1} = \gamma\alpha_k H^{-1} M_k(u^k - \tilde{u}^k)$, we obtain that

$$\begin{aligned} & (u - \tilde{u}^k)^T M_k(u^k - \tilde{u}^k) \\ &= \frac{1}{\gamma\alpha_k^*} (\|u - u^{k+1}\|_H^2 - \|u - u^k\|_H^2) \\ & \quad + \frac{1}{\alpha_k^*} \|u^k - u^{k+1}\|_H^2 + (1 - \gamma) \|(u^k - \tilde{u}^k)\|_{M_k}^2 \\ &= \frac{1}{\gamma\alpha_k^*} (\|u - u^{k+1}\|_H^2 - \|u - u^k\|_H^2) + (1 - \frac{\gamma}{2}) \|(u^k - \tilde{u}^k)\|_{M_k}^2. \end{aligned} \quad (19)$$

Combining (16) and (19), it is easy to derive the (15). Hence, the Lemma is proved.

Theorem 3

Let the sequences $\{\tilde{u}^k\}$ be generated by the proposed APPA Algorithm. For an integer $N > 0$, let

$$\bar{u}_N := \frac{1}{N} \sum_{k=1}^N \tilde{u}^k, \quad (20)$$

then $\bar{u}_N \in \Omega$ and

$$(\bar{u}_N - u)^T F(u) \leq \frac{m}{2c_0\gamma N} \|u - u^1\|. \quad (21)$$

Proof. It is easy to get that $\bar{u}_N \in \Omega$. In fact, $\bar{u}_N = \frac{1}{N} \sum_{k=1}^N \tilde{u}^k$ is a convex combination of \tilde{u}^k . Since $\gamma \in (0, 2)$, it follows from Lemma 1 that

$$(u - \tilde{u}^k)^T F(\tilde{u}^k) + \frac{1}{2\gamma\alpha_k^*} (\|u - u^k\|_H^2 - \|u - u^{k+1}\|_H^2) \geq 0, \forall u \in \Omega.$$

By combining the monotonicity of F with the last inequality, we obtain

$$(u - \tilde{u}^k)^T F(u) + \frac{1}{2\gamma\alpha_k^*} (\|u - u^k\|_H^2 - \|u - u^{k+1}\|_H^2) \geq 0, \forall u \in \Omega.$$

Summing the above inequality over $k = 1, 2, \dots, N$, we derive that

$$(Nu - \sum_{k=1}^N \tilde{u}^k)^T F(u) + \frac{1}{2\gamma\alpha_k^*} (\|u - u^1\|_H^2 - \|u - u^{k+1}\|_H^2) \geq 0, \forall u \in \Omega.$$

And it is obvious that

$$(Nu - \sum_{k=1}^N \tilde{u}^k)^T F(u) + \frac{1}{2\gamma\alpha_k^*} \|u - u^1\|_H^2 \geq 0, \forall u \in \Omega.$$

From Lemma A1 and Lemma A2, we have $\frac{1}{\alpha_k^*} \leq \frac{m}{c_0}$. Thus, we can obtain

$$\left(\frac{1}{N} \sum_{k=1}^N \tilde{u}^k - u\right)^T F(u) \leq \frac{m}{2c_0\gamma N} \|u - u^1\|.$$

The proof is completed. \square

For any given compact set $\mathcal{D} \in \Omega$, let $d := \sup\{\frac{m}{2c_0\gamma} \|u - u^1\|_H^2 | u \in \mathcal{D}\}$. By using Theorem 3, we know that after N iteration of the proposed algorithm, the point \bar{u}_N defined in (20) satisfies

$$\sup_{u \in \mathcal{D}} \{(\bar{u}_N - u)^T F(u)\} \leq \frac{d}{N},$$

which means that \bar{u}_N is an approximate solution of (5) with the accuracy $O(1/N)$.

4. Numerical Results for TV image restoration

In this section, we will test the proposed method for TV image deblurring problem, and compare it to the He and Yuan's Algorithm 1 (denote it as "HYPD" for short) in [22] and PDHG Algorithm in [39]. All the programs were coded in MATLAB R2007a and run on a personal computer with an Intel Core 2 Duo CPU at 2.4 GHz and 4GB of memory.

Firstly, we would like to present the APPA algorithm applying to TV image restoration as follows. We also call it Adaptive Primal-Dual Algorithm (APDA) with contraction strategy.

APDA : Adaptive Primal-dual Algorithm

Prediction Step: Generate the predictor $\tilde{u}^k = (\tilde{x}^k, \tilde{y}^k)$

$$\tilde{y}^k = \mathcal{F}^{-1} \left[\frac{\mathcal{F}(y^k - \sigma Ax^k) + \sigma \lambda \mathcal{F}(K)^* \odot \mathcal{F}(z)}{1 + \sigma \lambda \mathcal{F}(K)^* \odot \mathcal{F}(K)} \right]$$

$$\bar{y}^k = \tilde{y}^k + \theta_k (\tilde{y}^k - y^k)$$

$$\tilde{x}^k = P_{\mathcal{X}}(x^k + \tau A^T \bar{y}^k)$$

Correction Step: Let α_k be defined in (7c), and M_k, H be defined in (9). Correct the predictor and generate the new iterate $u^{k+1} = (x^{k+1}, y^{k+1})$ via:

$$u^{k+1} = u^k - \gamma \alpha_k^* H^{-1} M_k (u^k - \tilde{u}^k)$$

APDA1 Algorithm: $\theta_k^1 = \frac{t_{k-1}-1}{t_k}$ with $t_k = \frac{1}{2}(1 + \sqrt{1 + 4t_{k-1}^2})$, $t_1 = 1$.

APDA2 Algorithm: $\theta_k^2 = \frac{k+1}{k+2}$.

Remarks:

R1: The choice of θ_k^1 comes from Nesterov acceleration strategy [7], and the θ_k^2 is the same parameter used in adaptive PDHGMu for image denoising [18] (in which $\theta_k = \frac{\tau_k}{\tau_{k-1}}$ and $\tau_k = \frac{1}{k+2}$.)

R2: In fact, except from the above two choices, the sequence $\{\theta_k\}$ can also be chosen according to one of the following rules:

Rule 1: $\theta_k = \frac{t_{k-1}-1}{t_k}$, reference to [25, 3]

Rule 2: $\theta_k = \frac{t_k-1}{t_k}$, which is the modification of Rule 1,

with $\{t_k\}$ satisfying

$$t_{k+1}^2 - t_{k+1} \leq t_k^2, \quad t_k \geq \frac{k+2}{2}, \quad \forall k \geq 2, \quad t_1 = 1.$$

According to the above two rules, we can get some other choices for θ_k as follows:

- $\theta_k^3 = \frac{k-1}{k+2}$ which is taken $t_k = \frac{k+2}{2}$ according to rule 1;
- $\theta_k^4 = \frac{t_k-1}{t_k}$ and take $t_{k+1}^2 - t_{k+1} = t_k^2$ according to rule 2;
- $\theta_k^5 = \frac{k}{k+2}$ which is taken $t_k = \frac{k+2}{2}$ according to rule 2.

It is justified in Appendix that each choice for M_k could satisfy the conditions in **A1**, **A2**. In our experiments, we just report the numerical results for θ_k^1 and θ_k^2 .

We test the numerical performances of APDA1 and APDA2 and compare them to PDHG, HYPD on image restoration problems of three images of various sizes. The original and clean images are degraded by convolutions and the zero-mean Gaussian noise with the standard deviation 10^{-3} . The blur operator and the additive noise are generated by the respective scripts fspecial and imnoise in Matlab Image Processing Toolbox. We set medium and severe motion blur in the scenarios fspecial ('motion', 21, 135) and fspecial ('motion', 91, 135), respectively. And set medium and severe gaussian blur in the scenarios fspecial ('gaussian', 21, 5) and fspecial ('gaussian', 41, 10), respectively.

We use the following stopping criterion:

$$\frac{\|y^{k+1} - y^k\|}{\|y^{k+1}\|} < \text{Tol}, \quad (22)$$

where $\{y^k\}$ is the sequence generated by the test algorithms. We also use signal-to-noise ratio (SNR)

$$\text{SNR} := 20 \log_{10} \frac{\|y^*\|}{\|\bar{y} - y^*\|},$$

as a measurement to evaluate the performance of the tested algorithms, where \bar{y} is the image restored by certain algorithm and y^* is the original one.

Table 1. The value of parameters

| APDA1 | APDA2 | HYPD | PDHG |
|---|--------------------------------|-----------------|---|
| $\tau = 0.03$ | $\tau = 0.03$ | $\tau = 0.03$ | $\tau_k = 10 + 40k$ |
| $\sigma = 5.0$ | $\sigma = 5.0$ | $\sigma = 5.0$ | $\sigma_k = (1 - \frac{0.2}{k})/\tau_k$ |
| $\gamma = 1.3$ | $\gamma = 1.3$ | $\gamma = 1.6$ | |
| $\theta_k^1 = \frac{t_{k-1}-1}{t_k}$ | $\theta_k^2 = \frac{k+1}{k+2}$ | $\theta = -0.2$ | |
| $(t_k = \frac{1}{2}(1 + \sqrt{1 + 4t_{k-1}^2}), t_1 = 1)$ | | | |

To deblur these corrupted images, we take $\lambda = 250$ and $\lambda = 1000$ in (3), $\text{Tol}=10^{-4}$ and $\text{Tol}=5 \times 10^{-5}$ in (22) for the motion and gaussian blur cases, respectively. Table 1 presents parameters for all tested algorithms.



Figure 1. Original images: Cameraman.png, Barbara.png, and Man.tiff.

Table 2. Numerical results of images with motion blur.

| problem | algorithm | medium motion blur | | | severe motion blur | | |
|------------------------|-----------|--------------------|---------|-------|--------------------|---------|-------|
| | | Iter | Time(s) | SNR | Iter | Time(s) | SNR |
| Cameraman 256 × 256 | PDHG | 102 | 1.81 | 21.53 | 110 | 1.87 | 14.53 |
| | HYPD | 27 | 0.80 | 27.35 | 76 | 2.28 | 21.35 |
| | APDA1 | 28 | 0.83 | 27.36 | 46 | 1.54 | 20.88 |
| | APDA2 | 19 | 0.51 | 27.34 | 23 | 0.72 | 18.99 |
| Barbara 512 × 512 | PDHG | 100 | 9.33 | 21.76 | 104 | 9.47 | 15.96 |
| | HYPD | 26 | 3.92 | 24.53 | 71 | 11.17 | 18.95 |
| | APDA1 | 28 | 4.63 | 24.54 | 46 | 7.43 | 18.77 |
| | APDA2 | 17 | 2.65 | 24.61 | 20 | 3.06 | 18.03 |
| Man 1024 × 1024 | PDHG | 90 | 37.72 | 23.33 | 122 | 50.67 | 17.52 |
| | HYPD | 34 | 24.20 | 25.55 | 63 | 43.96 | 22.37 |
| | APDA1 | 29 | 20.11 | 25.54 | 40 | 28.70 | 22.20 |
| | APDA2 | 22 | 15.43 | 25.49 | 17 | 11.73 | 21.25 |

Table 3. Numerical results of images with Gaussian blur.

| problem | algorithm | medium gaussian blur | | | severe gaussian blur | | |
|------------------------|-----------|----------------------|---------|-------|----------------------|---------|-------|
| | | Iter | Time(s) | SNR | Iter | Time(s) | SNR |
| Cameraman 256 × 256 | PDHG | 280 | 5.02 | 15.88 | 241 | 4.15 | 14.86 |
| | HYPD | 90 | 2.64 | 17.89 | 156 | 4.18 | 16.28 |
| | APDA1 | 54 | 1.50 | 17.90 | 71 | 1.98 | 16.21 |
| | APDA2 | 40 | 1.12 | 17.90 | 55 | 1.47 | 16.14 |
| Barbara 512 × 512 | PDHG | 265 | 25.57 | 15.93 | 219 | 20.44 | 15.82 |
| | HYPD | 80 | 12.78 | 17.26 | 137 | 21.47 | 16.54 |
| | APDA1 | 54 | 8.47 | 17.28 | 69 | 11.22 | 16.57 |
| | APDA2 | 40 | 6.21 | 17.30 | 54 | 8.39 | 16.57 |
| Man 1024 × 1024 | PDHG | 313 | 142.46 | 16.10 | 264 | 118.86 | 15.87 |
| | HYPD | 83 | 58.03 | 18.84 | 138 | 97.75 | 17.13 |
| | APDA1 | 53 | 37.32 | 18.86 | 70 | 49.62 | 17.12 |
| | APDA2 | 40 | 27.67 | 18.87 | 55 | 39.34 | 17.10 |

Numerical results are shown in tables 2 and 3 for the motion blur cases and gaussian blur cases, respectively. As is demonstrated in these tables, the proposed algorithms have a comparative SNR with that of HYPD. But, in most

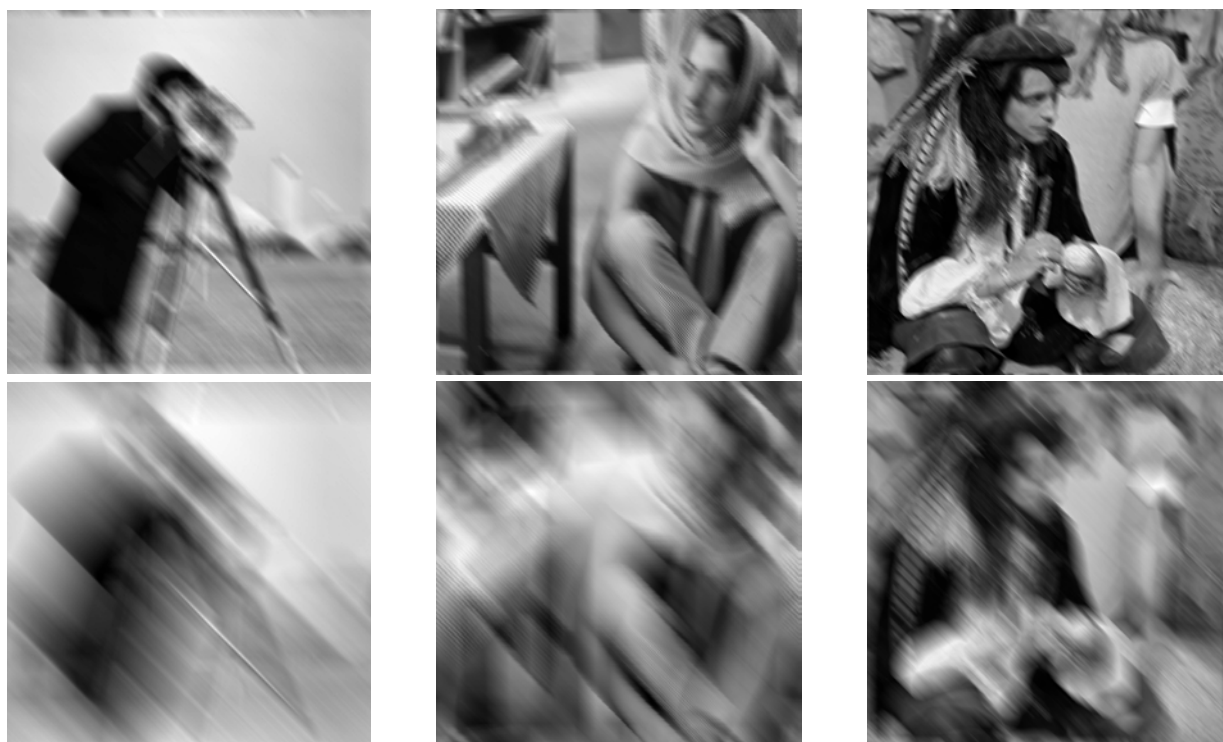


Figure 2. Corrupted images: Top: medium motion blur; Bottom: severe motion blur.



Figure 3. Corrupted images: Top: medium gaussian blur; Bottom: severe gaussian blur.

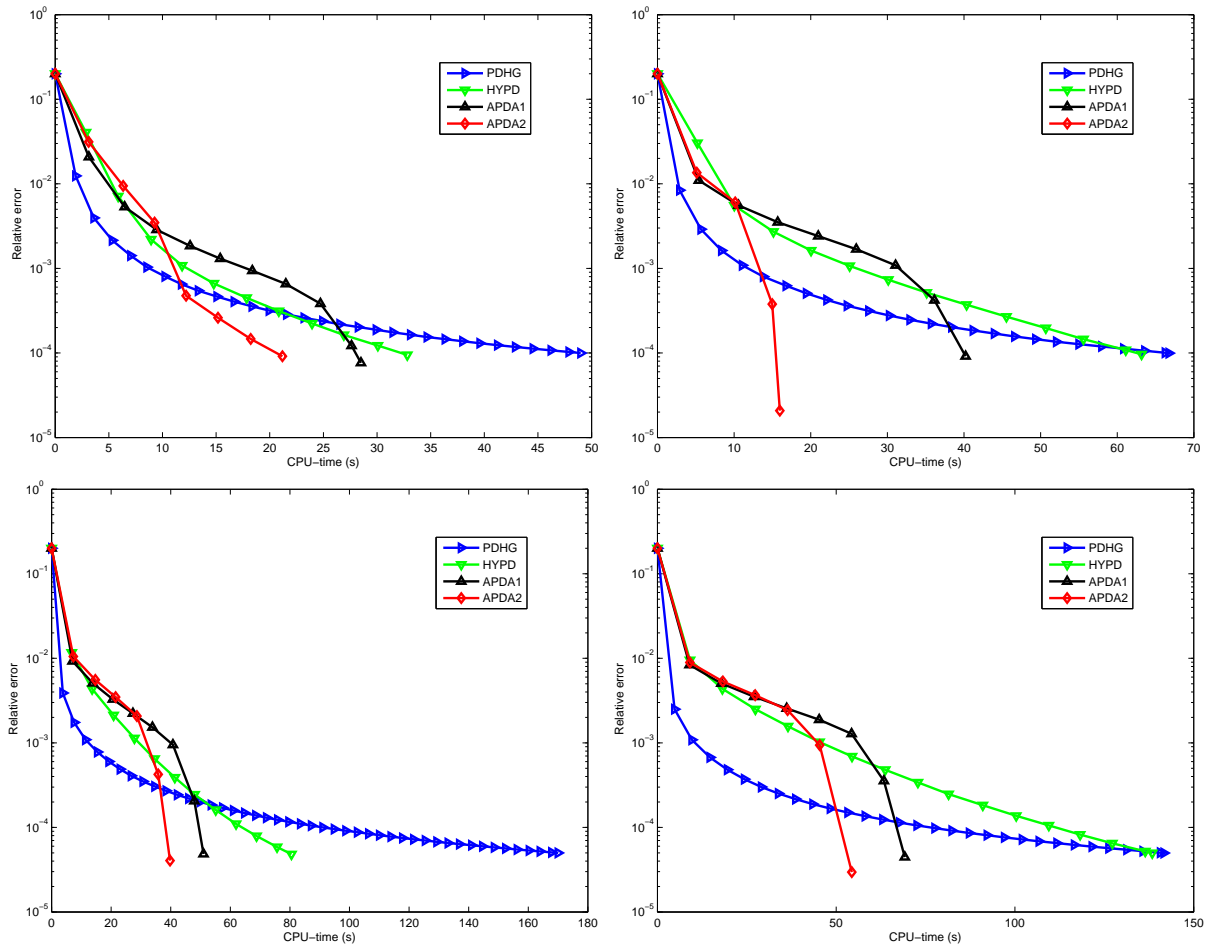


Figure 4. Evolutions of Relative error w.r.t computing times for Man.tiff.

cases, APDA methods, especially APDA2, use much less time and iterates than HYPD and PDHG. This advantage is more apparent for the case of severe deblurring.

Original images were shown in Figure 1: Cameraman.png(256×256), Barbara.png(512×512) and Man.tiff(1024×1024). Figure 2 and 3 display the blurred images corrupted with different scenarios.

Figure 4 plots the relative error against the CPU time for 'Man.tiff'. Figures 5-8 display the restored images by PDHG, HYPD, APDA1 and APDA2. As we can see from Table 2 and Table 3, the proposed method is competitive to HYPD and outperforms PDHG for image deblurring problems. According to Figure 4, we would like to point out that in some case, APDA may possess a better convergence rate (say $O(1/N^2)$).

5. Conclusion

In this paper, we propose a class of adaptive proximal point algorithms (APPA) with contraction strategy for solving TV image restoration problems. The proposed methods choose an adaptive proximal parameter matrix in each iteration. In fact, there is an inner extrapolation in prediction step and it is accelerated by an adaptive scheme such as Nesterov kind strategy. By using the framework of contraction method, global convergence and a $O(1/N)$ convergence rate for the proposed methods could be established. Numerical experiments verified that APPA methods are competitive to HYPD and outperform PDHG for the test problems.



Figure 5. From Left to Right: the medium motion blurred images restored by PDHG, HYPD, APDA1, APDA2, respectively.

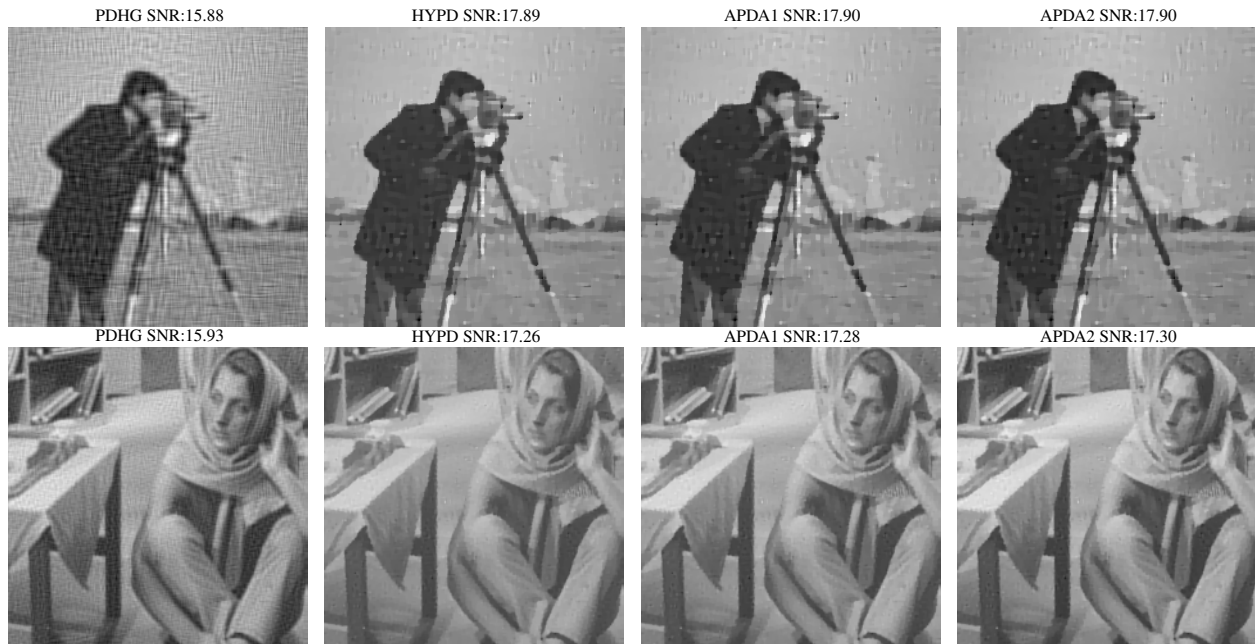


Figure 6. From Left to Right: the medium gaussian blurred images restored by PDHG, HYPD, APDA1, APDA2, respectively.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (No. 61262026), NCET Programm of the Ministry of Education (NCET 13-0738), JGZX programm of Jiangxi Province (20112BCB23027), Natural Science Foundation of Jiangxi Province (20132BAB201026), Science and technology

programm of Jiangxi Education Committee (LDJH12088) and the graduate student innovation project of Jianxi Province (YC2013-S264).

Appendix

This appendix is devoted to justifying that with M_k be defined in (9) assumptions **A1**, **A2** hold.

Lemma A1: If the step sizes τ, σ and θ_k are defined as in Table 1, then for the matrix M_k we have

$$(u - \tilde{u})^T M_k (u - \tilde{u}) \geq \frac{\delta_k}{1 + \delta_k} \|u - \tilde{u}\|_H^2 > 0, \forall u \neq \tilde{u}, \quad (23)$$

where

$$\delta_k = \frac{2}{1 + \theta_k} \sqrt{\frac{1}{\tau\sigma\|A^T A\|}} - 1, \quad \text{and } \delta_k > 0. \quad (24)$$

Proof. From(24), we have that

$$\tau(1 + \delta_k)\|A^T A\| \frac{(1 - \theta_k)^2}{4} = \frac{1}{\sigma(1 + \delta_k)}.$$

Therefore, for any $u \neq \tilde{u}$,

$$\begin{aligned} (u - \tilde{u})^T M_k (u - \tilde{u}) &= \|u - \tilde{u}\|_H^2 - (1 + \theta_k)(y - \tilde{y})^T A(x - \tilde{x}) \\ &= \|u - \tilde{u}\|_H^2 + 2\left(\frac{1 - \theta_k}{2}(\tilde{y} - y)^T\right) A(x - \tilde{x}) \\ &\geq \|u - \tilde{u}\|_H^2 - (\tau(1 + \delta_k)\|A^T A\|) \frac{(1 + \theta_k)^2}{4} \|y - \tilde{y}\|^2 \\ &\quad - \frac{1}{\tau(1 + \delta_k)\|A^T A\|} \|A(x - \tilde{x})\|^2 \\ &= \|u - \tilde{u}\|_H^2 - \frac{1}{\sigma(1 + \delta_k)} \|y - \tilde{y}\|^2 \\ &\quad - \frac{1}{\tau(1 + \delta_k)\|A^T A\|} \|A(x - \tilde{x})\|^2 \\ &\geq \|u - \tilde{u}\|_H^2 - \frac{1}{1 + \delta_k} \left(\frac{1}{\tau} \|x - \tilde{x}\|^2 + \frac{1}{\sigma} \|y - \tilde{y}\|^2\right) \\ &= \frac{\delta_k}{1 + \delta_k} \|u - \tilde{u}\|_H^2. \end{aligned}$$

Since $\|u - \tilde{u}\|_H^2 > 0$ and $\delta_k > 0$, thus the Lemma A1 is proved. Therefore, the M_k is positive-defined. \square

Lemma A2: Let M_k be defined in (9) and H be an arbitrary symmetric positive definite fixed matrix, then there exist a constant m , such that

$$\|M_k^T H^{-1} M_k\| \leq m.$$

Proof. From the defined of M_k , we have that

$$M_k^T H^{-1} M_k = \begin{pmatrix} \frac{1}{\tau} I + \sigma A^T A & -(\theta_k + 1) A^T \\ -(\theta_k + 1) A & \frac{1}{\sigma} I + \tau \theta_k^2 A A^T \end{pmatrix}. \quad (25)$$

With the fact $\lim_{k \rightarrow \infty} \theta_k = 1$, it is easy to get that

$$\lim_{k \rightarrow \infty} M_k^T H^{-1} M_k = \begin{pmatrix} \frac{1}{\tau} I + \sigma A^T A & -2A^T \\ -2A & \frac{1}{\sigma} I + \tau A A^T \end{pmatrix}.$$

For convenience, we denoted $M = \begin{pmatrix} \frac{1}{\tau}I + \sigma A^T A & -2A^T \\ -2A & \frac{1}{\sigma}I + \tau A A^T \end{pmatrix}$.

With the continuity of the norm, we have

$$\lim_{k \rightarrow \infty} \|M_k^T H^{-1} M_k\| = \|M\|.$$

Then, exists a positive integer N , when $k > N$, we have

$$\lim_{k \rightarrow \infty} \|M_k^T H^{-1} M_k\| \leq \|M\| + 1.$$

Let $m = \max\{\|M_1^T H^{-1} M_1\|, \|M_2^T H^{-1} M_2\|, \dots, \|M_N^T H^{-1} M_N\|, \|M\| + 1\}$, which complete the proof. \square

REFERENCES

1. K. J. Arrow, L. Hurwicz and H. Uzawa (1958), *Studies in linear and non-linear programming*, Stanford: Stanford University Press.
2. A. Auslender and M. Teboulle (2000), Lagrangian duality and related multiplier methods for variational inequality problems, *SIAM Journal on Optimization*, 10(4), 1097-1115.
3. D. P. Bertsekas (2009), *Convex optimization theory*, Cambridge: Athena Scientific.
4. J. H. Bramble, J. E. Pasciak and A. T. Vassilev (1997), Analysis of the inexact Uzawa algorithm for saddle point problems, *SIAM Journal on Numerical Analysis*, 34(3), 1072-1092.
5. R. S. Burachik, A. N. Iusem and B. F. Svaiter (1997), Enlargement of monotone operators with applications to variational inequalities, *Set-Valued Analysis*, 5(2), 159-180.
6. J. V. Burke and M. Qian (2000), On the superlinear convergence of the variable metric proximal point algorithm using Broyden and BFGS matrix secant updating, *Mathematical Programming*, 88(1), 157-181.
7. A. Beck and M. Teboulle (2009), Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems, *IEEE Transactions on Image Processing*, 18(11), 2419-2434.
8. C. Brune, A. Sawatzky and M. Burger (2011), Primal and dual Bregman methods with application to optical nanoscopy, *International Journal of Computer Vision*, 92(2), 211-229.
9. T. F. Chan, G. H. Golub and P. Mulet (1999), A nonlinear primal-dual method for total variation-based image restoration, *SIAM Journal on Scientific Computing*, 20(6), 1964-1977.
10. X. Cai, D. Han and L. Xu (2012), An improved first-order primal-dual algorithm with a new correction step, *Journal of Global Optimization*, 1-10.
11. A. Chambolle (2004), An algorithm for total variation minimization and applications, *Journal of Mathematical Imaging and Vision*, 20(1-2), 89-97.
12. A. Chambolle and T. Pock (2011), A first-order primal-dual algorithm for convex problems with applications to imaging, *Journal of Mathematical Imaging and Vision*, 40(1), 120-145.
13. P. Chen, J. Huang and X. Zhang (2013), A primal-dual fixed point algorithm for convex separable minimization with applications to image restoration, *Inverse Problems*, 29(2), 025011.
14. P. L. Combettes and V. R. Wajs (2005), Signal recovery by proximal forward-backward splitting, *Multiscale Modeling and Simulation*, 4(4), 1168-1200.
15. X. Cai, G. Gu, B. He and X. Yuan (2011), A relaxed customized proximal point algorithm for separable convex programming, *Optimization Online*.
16. J. Douglas and H. H. Rachford (1956), On the numerical solution of heat conduction problems in two and three space variables, *Transactions of the American mathematical Society*, 82(2), 421-439.
17. J. Eckstein (1998), Approximate iterations in Bregman-function-based proximal algorithms, *Mathematical Programming*, 83(1-3), 113-123.
18. E. Esser, X. Zhang and T. Chan (2009), A general framework for a class of first order primal-dual algorithms for TV minimization, *UCLA CAM Report*, 09-67.
19. D. Han and B. He (2001), A new accuracy criterion for approximate proximal point algorithms, *Journal of Mathematical Analysis and Applications*, 263(2), 343-354.
20. D. Han (2003), A new hybrid generalized proximal point algorithm for variational inequality problems, *Journal of Global Optimization*, 26(2), 125-140.
21. B. He and X. Yuan (2010), A contraction method with implementable proximal regularization for linearly constrained convex programming, www.optimization-online.org/DB_HTML/2010/11/2817.html
22. B. He and X. Yuan (2012), Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective, *SIAM Journal on Imaging Sciences*, 5(1), 119-149.
23. B. He, X. L. Fu and Z. K. Jiang (2009), Proximal-point algorithm using a linear proximal term, *Journal of Optimization Theory and Applications*, 141(2), 299-319.
24. B. Martinet (1970), Brve communication Rgularisation d'inquations variationnelles par approximations successives, *ESAIM: Mathematical Modelling and Numerical Analysis-Modlisation Mathmatique et Analyse Numrique*, 4(R3), 154-158.
25. Y. Nesterov (1983), A method of solving a convex programming problem with convergence rate $O(1/k^2)$, *In Soviet Mathematics Doklady*, 27(2), 372-376.

26. C. A. Micchelli, L. Shen and Y. Xu (2011), Proximity algorithms for image models: denoising, *Inverse Problems*, 27(4), 045009.
27. A. Nemirovski (2004), Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems, *SIAM Journal on Optimization*, 15(1), 229-251.
28. T. Pock and A. Chambolle (2011), Diagonal preconditioning for first order primal-dual algorithms in convex optimization, In *2011 IEEE International Conference on Computer Vision (ICCV)*, pp. 1762-1769.
29. L. I. Rudin, S. Osher and E. Fatemi (1992), Nonlinear total variation based noise removal algorithms, *Physica D: Nonlinear Phenomena*, 60(1), 259-268.
30. R. T. Rockafellar (1976), Monotone operators and the proximal point algorithm, *SIAM Journal on Control and Optimization*, 14(5), 877-898.
31. M. V. Solodov and B. F. Svaiter (2000), An inexact hybrid generalized proximal point algorithm and some new results on the theory of Bregman functions, *Mathematics of Operations Research*, 25(2), 214-230.
32. S. Setzer (2011), Operator splittings, Bregman methods and frame shrinkage in image processing, *International Journal of Computer Vision*, 92(3), 265-280.
33. M. Teboulle (1997), Convergence of proximal-like algorithms, *SIAM Journal on Optimization*, 7(4), 1069-1083.
34. C. R. Vogel and M. E. Oman (1996), Iterative methods for total variation denoising, *SIAM Journal on Scientific Computing*, 17(1), 227-238.
35. J. Wu and G. Yu (2014), On the convergence and $O(1/N)$ complexity of a class of nonlinear proximal point algorithms for monotonic variational inequalities, *Statistics, Optimization and Information Computing*, 2(2), 105-113.
36. G. Yu, L. Qi and Y. Dai (2009), On nonmonotone Chambolle gradient projection algorithms for total variation image restoration, *Journal of Mathematical Imaging and Vision*, 35(2), 143-154.
37. X. Zhang, M. Burger and S. Osher (2011), A unified primal-dual algorithm framework based on Bregman iteration, *Journal of Scientific Computing*, 46(1), 20-46.
38. M. Zhu, S. J. Wright and T. F. Chan (2010), Duality-based algorithms for total-variation-regularized image restoration, *Computational Optimization and Applications*, 47(3), 377-400.
39. M. Zhu and T. F. Chan (2008), An efficient primal-dual hybrid gradient algorithm for total variation image restoration, *UCLA CAM Report*, 08-34.