The PPADMM Method for Solving Quadratic Programming Problems

Hai-Long Shen 1,* and Xu Tang 2

1 Department of Mathematics College of Sciences, Northeastern University Shenyang, Shenyang 100819, China
2 Northwest Institute of Mechanical and Electrical Engineering Xianyang, Xianyang 712000, China;
* Correspondence: shenhailong@mail.neu.edu.cn; Tel.: +86-024-83683382

Abstract: In this paper, a preconditioned and proximal alternating direction method of multipliers (PPADMM) is established for iteratively solving the equality-constraint quadratic programming problems. Based on strictly matrix analysis, we prove that this method is asymptotically convergent. We also show the connection between this method with some existing methods, so it combines the advantages of the methods. Finally, the numerical examples show that the algorithm proposed is efficient, stable, and flexible for solving the quadratic programming problems with equality constraint.

Keywords: quadratic programming problem; global convergence; preconditioning and proximal terms; iterative methods; convex problems

1. Introduction

This manuscript introduces the following convex optimization model with linear constraints and a separable objective function:

\[
\begin{aligned}
\min_{x,y} & \quad f_1(x) + f_2(y), \\
\text{s.t.} & \quad Ax + By = b,
\end{aligned}
\]

where \( A \in \mathbb{R}^{p \times n} \) and \( B \in \mathbb{R}^{p \times m} \) are two matrices, and \( b \in \mathbb{R}^p \) is a known vector. The objective function \( f_1 : \mathbb{R}^n \to \mathbb{R} \) and \( f_2 : \mathbb{R}^m \to \mathbb{R} \) are two quadratic functions defined by

\[
\begin{aligned}
f_1(x) &= \frac{1}{2} x^T F x + x^T f, \\
f_2(y) &= \frac{1}{2} y^T G y + y^T g,
\end{aligned}
\]

where \( F \in \mathbb{R}^{n \times n} \), \( G \in \mathbb{R}^{m \times m} \) are symmetric positive semidefinite matrices and \( f \in \mathbb{R}^n \), \( g \in \mathbb{R}^m \) are the known vectors. The class of convex minimization problems arises in many areas of computational science and engineering applications such as compressed sensing [1], financial [2,3], image restoration [4–6], network optimization problems [7,8], and traffic planning convex problems [9–12]. The model (1)–(2) captures many applications in different areas—see the \( \ell_1 \)-norm regularized least-squares problems in [12,13], the total variation image restoration in [13–16], and the standard quadratic programming problems in [7,13].

Let \( H \in \mathbb{R}^{p \times p} \) be the symmetric positive definite matrix, \( \langle \cdot, \cdot \rangle_H \) represents the weighted inner product with the weighting matrix \( H \), \( \| \cdot \| \) stands for the Euclidean norm, and \( \| \cdot \|_H \) stands for the analogous weighted matrix norm. Note that for the vectors \( u, v \in \mathbb{R}^p \) and the matrix \( X \in \mathbb{R}^{p \times p} \), it holds that \( \langle u, v \rangle_H = \langle Hu, v \rangle, \| u \|_H = \| H^{1/2} u \| \) and \( \| X \|_H = \| H^{1/2} X H^{-1/2} \| \).

If \( \langle u, v \rangle_H = 0 \), we say that \( u, v \in \mathbb{R}^p \) are \( H \)-orthogonal, which is denoted by \( u \perp_H v \). In particular, if \( H \) is the identity matrix, then the vectors \( u \) and \( v \) are orthogonal, which is simply denoted by \( u \perp v \). For \( \zeta \in \mathbb{C}, \zeta \) stands for its conjugate complex.
The problem (1)–(2) is mathematically equivalent to the unconstraint optimization problem [7]

$$\max_{x,y} \min_{\lambda} \psi(x, y, \lambda)$$

(3)

$$\psi(x, y, \lambda)$$ is the augmented Lagrangian function defined as

$$\psi(x, y, \lambda) = f_1(x) + f_2(y) - \langle Ax + By - b, \lambda \rangle + \frac{\beta}{2} \|Ax + By - b\|^2,$$  

(4)

where $\lambda$ is the Lagrangian multiplier, and $\beta$ is a regularization parameter. In a word, a point $(x_\star, y_\star)$ is the solution to the problem (1)–(2) if and only if there exists $\lambda_\star \in \mathbb{R}^p$ such that the point $(x_\star, y_\star, \lambda_\star) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ is the solution to the problem (3)–(4) [7].

The most common method to solve the problem (3)–(4) is the alternating direction method of multipliers (ADMM) [7], in which each iteration of the augmented Lagrangian method (ALM) [17,18] has a Gauss–Seidel decomposition. The scheme of the ADMM method for (1)–(2) is

$$\begin{align*}
x^{(k+1)} &= \arg\min_x \psi(x, y^{(k)}, \lambda^{(k)}), \\
y^{(k+1)} &= \arg\min_y \psi(x^{(k+1)}, y, \lambda^{(k)}), \\
\lambda^{(k+1)} &= \lambda^{(k)} - \beta(Ax^{(k+1)} + By^{(k+1)} - b).
\end{align*}$$  

(5)

A significant advantage of ADMM is that each subproblem involves only one of the functions $f_1$ and $f_2$ in the original problem. Therefore, the variables $x$ and $y$ are treated separately during iterations, which makes solving the subproblems in (3)–(4) evidently easier than the original problem (1)–(2). The ADMM’s easy implementation and impressive efficiency has recently received extensive attention from many different areas. The ADMM is a very effective method to solve the convex optimization problem. Glowinski and Marrocco [19] were the first batch of scientists to describe it. Gabay [20], Glowinski and Le Tallec [21], as well as Eckstein and Bertsekas [22] studied some convergence results related to ADMM. It is suitable to solve convex programming problems with separable variables [7,23,24], and they are widely used in image processing, statistical learning, and machine learning [4–9].

For the convex optimization problems, some methods based on gradient operator are sensitive to the choice of the iteration step. If the parameters are not properly selected, the algorithm may not be convergent. In contrast, the ADMM is robust to the choice of parameters: under some mild conditions, the method can guarantee convergence for any positive parameter of its single parameter. When the objective function is a quadratic function, its global convergence is proved [13,24], and this method is linearly convergent. For example, the ADMM converges linearly to the optimal solution for problems (1) and (2). Although the convergence of the ADMM is perfectly solved, the accurate estimation of its convergence rate is still in its early stages; see, for example, [13,14].

Since the classical ADMM algorithm is inefficient in solving the accuracy of subproblem, Deng and Wo proposed a generalized ADMM in the literature [25], which adds the proximal terms $\frac{1}{2}\|x - x^{(k)}\|^2_P$ and $\frac{1}{2}\|y - y^{(k)}\|^2_T$ to the $x$- and $y$-subproblems, respectively. Its scheme for (1)–(2) is

$$\begin{align*}
x^{(k+1)} &= \arg\min_x \psi(x, y^{(k)}, \lambda^{(k)}) + \frac{1}{2}\|x - x^{(k)}\|^2_P, \\
y^{(k+1)} &= \arg\min_y \psi(x^{(k+1)}, y, \lambda^{(k)}) + \frac{1}{2}\|y - y^{(k)}\|^2_T, \\
\lambda^{(k+1)} &= \lambda^{(k)} - \alpha\beta(Ax^{(k+1)} + By^{(k+1)} - b),
\end{align*}$$  

(6)

where $\alpha \in (0, \frac{1+\sqrt{5}}{2})$, and the matrices $P$ and $T$ are symmetric positive semidefinite.

Deng and Wo solved the problem of global and linear convergence of the generalized ADMM and gave the mathematical proof where $P$ and $T$ are symmetric positive semidefi-
nite matrices. What is more, in order to make the subproblems of the generalized ADMM easier to solve and more efficient to run, the ultimate goal is to choose the adequate $P$ and $T$.

In order to iteratively solve the linear constraint quadratic programming problem (1)–(2), Bai and Tao [18] proposed a class of preconditioned alternating variable minimization with multiplier (PAVMM) methods by the matrix preconditioning strategy and utilizing a parameter accelerating technique, which is based on a weighted inner product and the homologous weighted norm. The iteration scheme is as follows:

$$
\begin{align*}
\tilde{x}^{(k+1)} &= \arg\min \left\{ \tilde{\psi}(x, y^{(k)}, \lambda^{(k)}) \right\}, \\
\tilde{y}^{(k+1)} &= \arg\min \left\{ \tilde{\psi}(\tilde{x}^{(k+1)}, y, \lambda^{(k)}) \right\}, \\
\lambda^{(k+1)} &= \lambda^{(k)} - aQ^{-1}W^{-1}(Ax^{(k+1)} + By^{(k+1)} - b), 
\end{align*}
$$

(7)

where $\tilde{\psi}(x, y, \lambda)$ is the following weighted augmented Lagrangian function:

$$
\tilde{\psi}(x, y, \lambda) = f_1(x) + f_2(y) - \lambda^T(Ax + By - b) + \frac{\beta}{2}\|Ax + By - b\|_W^2, 
$$

(8)

the matrices $W$ and $Q$ are symmetric positive semidefinite and nonsingular, and $a$ is the relaxation parameter. Actually, the PAVMM method is a class of preconditioned alternating direction method of multipliers (PADMM). Therefore, in this manuscript, the PAVMM method is recorded as PADMM. In particular, the ADMM is a special case of the PADMM. If $W = Q = I$ and $a = \beta$, the PADMM automatically reduces to the ADMM. Later, Bai and Tao [26] also establish a preconditioned and relaxed alternating variable minimization with multiplier (PRAVMM) method based on the PAVMM method, and the scheme is

$$
\begin{align*}
\tilde{g}^{(k+1)} &= \arg\min \left\{ \tilde{\psi}(x, y^{(k)}, \lambda^{(k)}) \right\}, \\
\tilde{x}^{(k+1)} &= \alpha\tilde{g}^{(k+1)} + (1 - \alpha)x^{(k)}, \\
\tilde{y}^{(k+1)} &= \alpha\tilde{g}^{(k+1)} + (1 - \alpha)y^{(k)}, \\
\lambda^{(k+1)} &= \lambda^{(k+\frac{1}{2})} - aQ^{-1}W^{-1}(Ax^{(k+1)} + By^{(k+1)} - b), 
\end{align*}
$$

(9)

where $\alpha$, $\tau$, and $\alpha$ are positive constants. As above, we also rewrite the PRAVMM into PRADMM. In order to achieve acceleration, it is obvious that the PRADMM adds two relaxation parameters to the iterative process. Hence, the PADMM and ADMM are the special cases of the PRADMM. When $\alpha = \tau = 1$, the PRADMM degenerates into the PADMM; when $\alpha = \tau = 1$, $W = Q = I$ and $a = \beta$; then, the PRADMM reduces to the ADMM.

To further generalize the PADMM and maximally promote the convergence speed of the PADMM, in this manuscript, we establish a preconditioned and proximal alternating direction of multiplier (PPADMM) to solve the problem (1)–(2) by the iterative methods. Assuming that the matrices $F \in \mathbb{R}^{n \times n}$ and $G \in \mathbb{R}^{m \times m}$ are symmetric positive semidefinite, we should choose reasonably the parameters, the weighting, and the preconditioning matrices; the PPADMM is convergent to the unique solution of the problem (1.1)–(1.2) for any initial guess. Obviously, the PPADMM proposed in this paper is an extension of the PADMM. Hence, the ADMM and PADMM are special cases of the PPADMM. In addition, we test the robustness and effectiveness of the PPADMM by using numerical experiments. The experimental results indicate that this method performs better than ADMM, PADMM, and PRADMM when they are employed to solve the convex optimization programming problem (1)–(2).

The paper is organized as follows. In Section 2, the computational properties of PPADMM was established. We introduce some necessary concepts for analyzing the asymptotic convergence of the PPADMM in Section 3. Then, based on the analysis in
Section 3, the asymptotic convergence of the PPADMM was demonstrated in Section 4. In Section 5, we have some test by solving the image deblurring problem to illustrate that our method is effective. Finally, we give some concluding remarks in Section 5.

2. The PPADMM Method

In this section, we will introduce the PPADMM proposed in this paper. At the $k$-th iteration step of the PPADMM, we add the proximal term $\frac{1}{2}\|x - x^{(k)}\|_P^2$ and $\frac{1}{2}\|y - y^{(k)}\|_T^2$ to the $x$- and $y$-subproblems during computing minimal points $x^{(k+1)}$ and $y^{(k+1)}$ of the minimization subproblems. In order to solve the problem (1)–(2) by the iterative methods, we are able to establish the preconditioned and proximal alternating direction method of multipliers (PPADMM) as follows:

$$
\begin{align*}
\chi^{(k+1)} &= \arg \min \left\{ \psi(x, y^{(k)}, \lambda^{(k)}) + \frac{1}{2}\|x - x^{(k)}\|_P^2 \right\}, \\
y^{(k+1)} &= \arg \min \left\{ \psi(x^{(k+1)}, y, \lambda^{(k)}) + \frac{1}{2}\|y - y^{(k)}\|_T^2 \right\}, \\
\lambda^{(k+1)} &= \lambda^{(k)} - \alpha\beta Q^{-1}W^{-1} \left( Ax^{(k+1)} + By^{(k+1)} - b \right),
\end{align*}
$$

where $P \in \mathbb{R}^{n \times n}$ and $T \in \mathbb{R}^{m \times m}$ are two symmetric positive semidefinite matrices, and $\alpha$ and $\beta$ are positive constants. We choose $P = \tau_1 I_n - \beta A^T A$ with the requirement $\tau_1 > \beta \|A^T A\|$ and $T = \tau_2 I_m - \beta B^T B$ with the requirement $\tau_2 > \beta \|B^T B\|$. In the same way, the matrices $W, Q$ are generated as follows:

$$
W = \xi_1 I_n - \beta A^T A, \quad Q = \xi_2 I_m - \beta B^T B
$$

where $\xi_1 \in (0, \frac{1}{\lambda_{\max}(A^T A)}), \xi_2 \in (0, \frac{1}{\lambda_{\max}(B^T B)})$.

It is easy to get the derivative of the quadratic functions $f_1(x)$ and $f_2(y)$ in (2) as follows:

$$
f_1'(x) = Fx + f \quad \text{and} \quad f_2'(y) = Gy + g.
$$

By making full use of Equation (10), the first and second for formulas in Equation (10) are differentiated for $x$ and $y$, respectively, with a simple manipulation. Afterwards, the iteration scheme of PPADMM can be rewritten as the following mathematically equivalent matrix–vector form:

$$
\begin{align*}
Fx^{(k+1)} + f - A^T W^{-1} \lambda^{(k)} + \beta A^T W^{-1} (Ax^{(k+1)} + By^{(k+1)} - b) + P(x^{(k+1)} - x^{(k)}) &= 0, \\
y^{(k+1)} + g - B^T W^{-1} \lambda^{(k)} + \beta B^T W^{-1} (Ax^{(k+1)} + By^{(k+1)} - b) + T(y^{(k+1)} - y^{(k)}) &= 0,
\end{align*}
$$

which can be equivalently reformulated as

$$
\begin{align*}
(F + \beta A^T W^{-1} A + P)x^{(k+1)} &= P x^{(k)} + A^T W^{-1} \beta( b - By^{(k)}) + \lambda^{(k)} - f, \\
(G + \beta B^T W^{-1} B + T)y^{(k+1)} &= B^T W^{-1} \beta( b - Ax^{(k+1)}) + \lambda^{(k)} - g, \\
\lambda^{(k+1)} &= \lambda^{(k)} - \alpha\beta Q^{-1}W^{-1} \left( Ax^{(k+1)} + By^{(k+1)} - b \right),
\end{align*}
$$

Since $F, G, P, T$ are symmetric positive definite matrices, $W$ is a symmetric positive definite matrix, and $\alpha > 0, \beta > 0$; thus, we know that the matrices $F + \beta A^T W^{-1} A + P$ and $G + \beta B^T W^{-1} B + T$ are symmetric positive definite if and only if the following statements are true [18]:

(a) $\text{null}(F) \cap \text{null}(A) = \{0\}$ or $\text{null}(F) \cap \text{null}(P) = \{0\}$ or $\text{null}(A) \cap \text{null}(P) = \{0\}$.

(b) $\text{null}(G) \cap \text{null}(B) = \{0\}$ or $\text{null}(G) \cap \text{null}(T) = \{0\}$ or $\text{null}(B) \cap \text{null}(T) = \{0\}$.

Therefore, when these above null-space conditions of PPADMM are well satisfied, measuring the costs of computation in the iteration scheme (13), it is obvious that the main costs of the PPADMM are to solve the linear systems with coefficient matrices $F + \beta A^T W^{-1} A + P$ and $G + \beta B^T W^{-1} B + T$. When the sizes $n, m, \text{and/or } p$ are relatively small, some direct methods such as the Cholesky factorization are able to effectively solve
these systems of linear equations. However, when the sizes \( n, m, \) and/or \( p \) are huge, it will take a lot of time to solve these linear systems by the direct methods. So, the iterative methods are used to solve the problems, e.g., the preconditioned conjugate methods. Of course, the weighted matrices \( W, P, \) and \( T, \) and penalty parameters \( a, \beta \) should be reasonably selected so that both the matrices \( F + \beta A^T W^{-1} A + P \) and \( G + \beta B^T W^{-1} B + T \) have the better conditions than the original matrices \( F \) and \( G, \) respectively. It will make the linear equations with the coefficient matrices \( F + \beta A^T W^{-1} A + P \) and \( G + \beta B^T W^{-1} B + T \) accurate, fast, and robust at each step of the PPADMM.

For the linear systems \( Ax = b, \) we define

\[
A = D - C_L - C_U,
\]

where \( A = (a_{ij}) \in \mathbb{C}^{m \times n} \) \((n \leq m), D = \text{diag}(A_{11}, \ldots , A_{m \times m}), C_L \) and \( C_U \) are strictly lower triangular matrix and strictly upper triangular matrix, respectively. Let \( L = D^{-1} C_L, U = D^{-1} C_U, \) then, the block SOR iterative method is described as

\[
x^{(k+1)} = L_w x^{(k)} + w(1-wL)^{-1}D^{-1}b, \\
L_w = (I - wL)^{-1}[(1-w)I + wU].
\]

According to the iterative scheme (13), we can see that the PPADMM can be classified as a modified block SOR iterative method for solving the following linear systems.

\[
\begin{cases}
(F + \beta A^T W^{-1} A)x - \beta A^T W^{-1}By - A^T W^{-1}z = \beta A^T W^{-1}b - f \\
\beta B^T W^{-1}Ax + (G + \beta B^T W^{-1} B)y - B^T W^{-1}z = \beta B^T W^{-1}b - g \\
W^{-1}Ax + W^{-1}By = W^{-1}b
\end{cases}
\]

For the properties that a saddle point of the weighted augmented Lagrangian function \( \psi(x, y, \lambda) \) defined in (3) possess, we refer to the literature [18] for more details.

**Theorem 1.** [7] Let \( F \in \mathbb{R}^{n \times n}, G \in \mathbb{R}^{m \times m} \) be symmetric positive semidefinite matrices and \( A \in \mathbb{R}^{p \times n}, B \in \mathbb{R}^{p \times m} \) be two arbitrary matrices. Let

\[
A(\beta) = \begin{bmatrix}
F + \beta A^T W^{-1} A & \beta A^T W^{-1} B & -A^T W^{-1} \\
\beta B^T W^{-1} A & G + \beta B^T W^{-1} B & -B^T W^{-1} \\
W^{-1} A & W^{-1} B & 0
\end{bmatrix}, \quad b(\beta) = \begin{pmatrix}
\beta A^T W^{-1} b - f \\
\beta B^T W^{-1} b - g \\
W^{-1} b
\end{pmatrix}, \quad x = \begin{pmatrix}
x \\
y \\
\lambda
\end{pmatrix}.
\]

Then, the following results are obtained:

(i) For \( x_s \in \mathbb{R}^n, y_s \in \mathbb{R}^m \) and \( \lambda_s \in \mathbb{R}^p, \) then the point \( (x_s, y_s, \lambda_s) \) is a saddle point of the weighted augmented Lagrangian function \( \psi(x, y, \lambda) \) defined in (3) if and only if \( x_s = (x_s^T, y_s^T, \lambda_s^T)^T \) is a solution of the linear system \( A(\beta)x = b(\beta); \)

(ii) The matrix \( A(\beta) \in \mathbb{R}^{(n+m+p) \times (n+m+p)} \) is nonsingular if and only if

\[
(a) \quad (\text{null}(F) \oplus \text{null}(G)) \cap \text{null}((A B)) = \{0\}
\]

\[
(b) \quad \text{null}(A^T) \oplus \text{null}(B^T) = \{0\}.
\]

In addition, for the complex quadratic polynomial equation, the determinant criterion for locations of its two roots is shown in Lemma 1. It is indispensable to prove the asymptotic convergence of the PPADMM in Section 4.

**Lemma 1.** [16,27,28] Assume \( \eta \) and \( \zeta \) are two complex constants; then, both roots of the complex quadratic polynomial equation \( \lambda^2 + \zeta \lambda + \eta = 0 \) have a modulus of less than one if and only if

\[
|\zeta - \overline{\eta}| + |\eta|^2 < 1. \quad (14)
\]

In particular, if both \( \eta \) and \( \text{null}(G) \cap \text{null}(B) = \{0\} \) are real constants, the condition (14) degrades into \( |\eta| < 1 \) and \( |\zeta| < 1 + \eta. \)
3. The Asymptotic Convergence of the PPADMM

In this section, we will prove that the PPADMM is globally and asymptotically convergent, and its corresponding asymptotic convergence rate was also estimated.

First, we define the matrices

\[
M_\beta(\alpha) = \begin{bmatrix}
F + \beta A^T W^{-1} A + P & 0 & 0 \\
\beta B^T W^{-1} A & G + \beta B^T W^{-1} B + T & 0 \\
W^{-1} A & W^{-1} B & 1_{\beta p} Q
\end{bmatrix}, N_\beta(\alpha) = \begin{bmatrix}
P - \beta A^T W^{-1} B & A^T W^{-1} \\
0 & T & B^T W^{-1} \\
0 & 0 & 1_{\beta p} Q
\end{bmatrix}.
\]

Obviously, the following matrix equations are true:

\[
A(\beta) = M_\beta(\alpha) - N_\beta(\alpha) = \begin{bmatrix}
F + \beta A^T W^{-1} A & \beta B^T W^{-1} A & -A^T W^{-1} \\
\beta B^T W^{-1} A & G + \beta B^T W^{-1} B & -B^T W^{-1} \\
W^{-1} A & W^{-1} B & 0
\end{bmatrix}.
\]

Hence, the iteration scheme (13) of the PPADMM can be rewritten as the equivalent matrix-vector form

\[
M_\beta(\alpha)x^{(k+1)} = N_\beta(\alpha)x^{(k)} + b(\beta).
\]

Multiplying \(M_\beta^{-1}(\alpha)\) by both sides of Equation (16), we obtain

\[
x^{(k+1)} = M_\beta^{-1}(\alpha)N_\beta(\alpha)x^{(k)} + M_\beta^{-1}(\alpha)b(\beta).
\]

It is easy to know that \(L_\beta(\alpha) = M_\beta^{-1}(\alpha)N_\beta(\alpha)\) is the iterative matrix of (17). For the matrix splitting of (14), the iterative scheme of the PPADMM can be equivalently reconstructed into the matrix splitting iteration method (16), which can solve the linear equation \(A(\beta)x = b(\beta)\). Therefore, the PPADMM is asymptotically and globally convergent if and only if the spectral radius of the iterative matrix \(L_\beta(\alpha)\) is less than one, i.e., \(\rho(L_\beta(\alpha)) < 1\).

We define the weighted matrices

\[
\hat{A} = W^{-\frac{1}{2}} A, \hat{B} = W^{-\frac{1}{2}} B, \hat{Q} = W^{\frac{1}{2}} Q W^{\frac{1}{2}},
\]

the augmented matrices

\[
\hat{F} = F + \beta A^T W^{-1} A + P, \hat{G} = G + \beta B^T W^{-1} B + T,
\]

and the compounded matrices

\[
\hat{R} = \hat{A}(P - \lambda \hat{F})^{-1} \hat{A}^T, \hat{S} = \hat{B}(T - \lambda \hat{G})^{-1} \hat{B}^T.
\]

Therefore, \(M, N\) can be rewritten as

\[
M_\beta(\alpha) = \begin{bmatrix}
\hat{F} & 0 & 0 \\
\beta \hat{B}^T \hat{A} & \hat{G} & 0 \\
W^{-\frac{1}{2}} \hat{A} & W^{-\frac{1}{2}} \hat{B} & 1_{\beta p} Q
\end{bmatrix}, N_\beta(\alpha) = \begin{bmatrix}
P - \beta \hat{A}^T \hat{B} & \hat{A}^T W^{-\frac{1}{2}} \\
0 & T & \hat{B}^T W^{-\frac{1}{2}} \\
0 & 0 & 1_{\beta p} Q
\end{bmatrix}.
\]

Here, we define the block diagonal matrix

\[
H = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & W^{\frac{1}{2}}
\end{bmatrix}.
\]

After preconditioning \(M_\beta(\alpha)\) and \(N_\beta(\alpha)\), respectively, we can obtain the block lower-triangular matrix \(\tilde{M}_\beta(\alpha)\) and the block upper-triangular matrix \(\tilde{N}_\beta(\alpha)\) as follows:
\[ \mathcal{M}_\beta(\alpha) = HM_\beta(\alpha)H = \begin{bmatrix} \hat{F} & 0 & 0 \\ \beta\hat{B}^T\hat{A} & \hat{C} & 0 \\ \hat{A} & \hat{B} & \frac{1}{\alpha}\hat{Q} \end{bmatrix}, \quad \mathcal{N}_\beta(\alpha) = HN_\beta(\alpha)H = \begin{bmatrix} P & -T & \hat{A}^T \\ -T & 0 & \hat{B}^T \\ \hat{A} & \hat{B} & \frac{1}{\alpha}\hat{Q} \end{bmatrix}. \]

It follows that \( \tilde{L}_\beta(\alpha) = \mathcal{M}_\beta(\alpha)^{-1}\mathcal{N}_\beta(\alpha) = H^{-1}L_\beta(\alpha)H \). It is obvious that \( \tilde{L}_\beta(\alpha) \) and \( L_\beta(\alpha) \) are similar matrices, so they have the same eigenvalues. When \( p(L_\beta(\alpha)) < 1 \), the PPADMM globally converges to the optimal solution of the problem (1)-(2).

**Theorem 2.** Suppose that \( F \in \mathbb{R}^{n \times n} \) and \( G \in \mathbb{R}^{m \times m} \) are symmetric positive semidefinite matrices, \( A \in \mathbb{R}^{p \times n} \) and \( B \in \mathbb{R}^{p \times m} \) are two matrices such that:

(a) \( (\text{null}(F) \oplus \text{null}(G)) \cap \text{null}(AB) = \{0\} \) and

(b) \( \text{null}(A^T) \oplus \text{null}(B^T) = \{0\} \).

Define the scaled matrices \( \hat{A}, \hat{B}, \hat{Q}, \hat{F}, \hat{G} \) and \( \hat{R}, \hat{S} \) as in (18)-(19). Suppose that \( \lambda \) is a nonzero eigenvalue of the PPADMM iteration matrix \( L_\beta(\alpha) \). If the matrices \( I + \beta\lambda\hat{S}, P - \lambda\hat{F}, \) and \( T - \lambda\hat{G} \) are nonsingular, then \( \lambda \) is an eigenvalue of the following eigenvalue problem \( (\lambda^2\beta^2\hat{R}\hat{S}\hat{Q} + \lambda E + \hat{Q})\hat{w} = 0, \) where

\[
E = a\beta\hat{S} - \hat{Q} + a\beta^2\hat{R}\hat{S} - \beta^2\hat{R}\hat{S}\hat{Q} + a\beta\hat{R},
\]

\[
\begin{cases}
\hat{Q} = (I + \beta\lambda\hat{S})^{-1}\hat{Q}(I + \beta\lambda\hat{S}), \\
\hat{S} = (I + \beta\lambda\hat{S})^{-1}\hat{S}(I + \beta\lambda\hat{S}), \\
\hat{w} = (I + \beta\lambda\hat{S})^{-1}\hat{w}.
\end{cases}
\]

**Proof.** In order to analyze the nonzero eigenvalues of \( L_\beta(\alpha) \), we first need to analyze the nonzero eigenvalues of \( \tilde{L}_\beta(\alpha) \) because the matrices \( \tilde{L}_\beta(\alpha) \) and \( L_\beta(\alpha) \) are similar. Obviously, the conditions (a) and (b) guarantee the nonsingularity of the matrix \( A(\beta), \lambda = 1 \) is not an eigenvalue of \( \tilde{L}_\beta(\alpha) \).

Assume \( \lambda \) is a nonunit and nonzero eigenvalue of the matrix \( \tilde{L}_\beta(\alpha), u = (u^T, v^T, w^T)^T \in \mathbb{C}^{n+m+p}, \) with \( u \in \mathbb{C}^n, v \in \mathbb{C}^m \) and \( w \in \mathbb{C}^p \) is the corresponding eigenvector, i.e., \( \tilde{L}_\beta(\alpha)u = \lambda u \). Then, it holds that \( \tilde{N}_\beta(\alpha)u = \lambda \tilde{M}_\beta(\alpha)u \), or equivalently,

\[
\begin{cases}
(P - \lambda\hat{F})u = \beta\hat{A}^T\hat{B}v - \hat{A}^Tw, \\
(T - \lambda\hat{G})v = \lambda\beta\hat{B}^T\hat{A}u - \hat{B}w, \\
\frac{1}{\alpha}\hat{Q}w = \lambda(\hat{A}u + \hat{B}v).
\end{cases}
\]

Since the matrices \( I + \beta\lambda\hat{S}, P - \lambda\hat{F}, \) and \( T - \lambda\hat{G} \) are nonsingular, by simplifying the first and second equations in (21), we can get

\[
\begin{cases}
\hat{A}u = \hat{R}(\beta\hat{B}v - w), \\
\hat{B}v = \hat{S}(\lambda\beta\hat{A}u - w).
\end{cases}
\]

Obviously, \( \beta v = \frac{1 - \lambda}{\alpha \beta \lambda} \hat{Q}w - \hat{A}u \). Substituting the equation into the first and second equation in (21), it holds that

\[
\begin{cases}
(I + \beta\hat{R})\hat{A}u = \hat{R}(1 - \lambda)(\hat{Q} - I)w, \\
(I + \beta\lambda\hat{S})\hat{A}u = (\frac{1 - \lambda}{\alpha \beta \lambda} \hat{Q} + \hat{S})w.
\end{cases}
\]

We can get the following simplified equation

\[
(I + \beta\hat{R})[(1 - \lambda)\hat{Q} + a\beta\lambda\hat{S}]\hat{w} = \beta\hat{R}(I + \beta\lambda\hat{S})[(1 - \lambda)\hat{Q} - a\lambda \lambda]\hat{w}.
\]
By combining the same terms in (25) according to the power exponents of \( \lambda \), we can get the following equation

\[
\{ \lambda^2 \beta^2 R \tilde{S} Q + \lambda [(I + \beta R)(\alpha \beta \tilde{S} - \tilde{Q}) - \beta^2 R \tilde{S} Q + \beta R \tilde{Q} + \alpha \beta R] + \tilde{Q} \} \tilde{w} = 0. \tag{26}
\]

As a result of

\[
(I + \beta \tilde{R})(\alpha \beta \tilde{S} - \tilde{Q}) - \beta^2 R \tilde{S} Q + \alpha \beta R = \alpha \beta \tilde{S} - \tilde{Q} + \alpha \beta^2 \tilde{R} \tilde{S} - \beta^2 R \tilde{S} Q + \alpha \beta \tilde{R}, \tag{27}
\]

we can rewrite (26) as

\[
(\lambda^2 \beta^2 \tilde{R} \tilde{S} \tilde{Q} + \lambda E + \tilde{Q}) \tilde{w} = 0. \tag{28}
\]

In accordance with Theorem 2, we can instantly get the following sufficient condition, which guarantees the globally asymptotic convergence of the PPADMM.

**Theorem 3.** Let the conditions of Theorem 2 be satisfied. For any non-zero vector \( \tilde{w} \in \mathbb{C}^n \), define

\[
\begin{align*}
\mu_s &= \frac{\tilde{w}^* \tilde{S} \tilde{w}}{\tilde{w}^* \tilde{R} \tilde{S} \tilde{Q} \tilde{w}}, & \mu_q &= \frac{\tilde{w}^* \tilde{Q} \tilde{w}}{\tilde{w}^* \tilde{R} \tilde{S} \tilde{Q} \tilde{w}}, & \chi &= \frac{\tilde{w}^* \tilde{R} \tilde{S} \tilde{w}}{\tilde{w}^* \tilde{R} \tilde{S} \tilde{Q} \tilde{w}}, & \mu_r &= \frac{\tilde{w}^* \tilde{R} \tilde{w}}{\tilde{w}^* \tilde{R} \tilde{S} \tilde{Q} \tilde{w}}. \tag{29}
\end{align*}
\]

If \(|\kappa - \pi \eta| + |\eta|_2 < 1\) is satisfied, then the iteration sequence \( \{x^{(k)}\}_{k=0}^\infty \) generated by the iteration scheme of the PPADMM in (10) converges to the optimal solution of the problem (1)–(2), where

\[
\begin{align*}
\kappa &= \frac{\delta \mu_s - \mu_q + \delta \beta^2 \chi - \beta^2 + \alpha \beta \mu_r}{\delta^2}, \\
\eta &= \frac{\mu_q}{\delta^2}.
\end{align*}
\]

Moreover, the convergence factor of the PPADMM is given by

\[
\sigma(\alpha, \beta) = \max\{\lambda_+^{(\text{max})}, \lambda_-^{(\text{max})}\},
\]

where

\[
\lambda_+^{(\text{max})} = \max_{\tilde{w} \in \mathbb{C}^n \setminus \{0\}} \left\{ \frac{|-\kappa \pm \sqrt{\kappa^2 - 4\eta}|}{2} \right\}.
\]

**Proof.** According to Theorem 2, we know that the PPADMM generates the following rational eigenvalue problem

\[
(\lambda^2 \beta^2 \tilde{R} \tilde{S} \tilde{Q} + \lambda E + \tilde{Q}) \tilde{w} = 0.
\]

The two sides of the equation are multiplied by \( \tilde{w}^* \in \mathbb{C}^n \setminus \{0\} \) at the same time, and we can get

\[
\lambda^2 \beta^2 \tilde{w}^* \tilde{R} \tilde{S} \tilde{Q} \tilde{w} + \lambda \tilde{w}^* E \tilde{w} + \tilde{w}^* \tilde{Q} \tilde{w} = 0 \tag{30}
\]

where \( E = \alpha \beta \tilde{S} - \tilde{Q} + \alpha \beta^2 \tilde{R} \tilde{S} - \beta^2 R \tilde{S} \tilde{Q} + \alpha \beta \tilde{R} \), we can rewrite the Equation (30) as

\[
\beta^2 \lambda^2 + \lambda (\alpha \beta \mu_s - \mu_q + \alpha \beta^2 \chi - \beta^2 + \alpha \beta \mu_r) + \mu_q = 0 \tag{31}
\]

With the notation in (28), Equation (30) is organized into

\[
\lambda^2 + \lambda \kappa + \eta = 0. \tag{32}
\]

According to the Lemma 1, if the root of the quadratic function about the eigenvalue is less than one, the coefficient must satisfy the following conditions:

\[
|\kappa - \pi \eta| + |\eta|_2 < 1.
\]
Last but not least, it is easy to acquire two roots of the quadratic Equation (32). The two roots are listed:

$$
\lambda_{\pm}^{(\text{max})} = \max_{\omega \in C^{d}} \left\{ \frac{-\omega \pm \sqrt{\omega^2 - 4\eta}}{2} \right\}.
$$

So, we can get the convergence factor of the PPADMM as follows:

$$
\sigma(a, \beta) = \max\{\lambda_+^{(\text{max})}, \lambda_-^{(\text{max})}\}. \quad \square
$$

4. Numerical Results

In this section, numerical examples are adopted to illustrate the performance of the PPADMM for the image deblurring problem. In [26], the PRADMM was shown to be more efficient than the PADMM. Hence, we only need to compare the PPADMM that we proposed with the PRADMM and ADMM in this section.

Image deblurring is a classical and significant subject study in image processing, which is usually an inverse problem. It is comprehensively concerned by many scholars engaged in computer vision. Normally, the goal of image deblurring is to recover an unknown original image \( u \in \mathbb{R}^n \) from a noisy image \( y \in \mathbb{R}^n \) that is often modeled as \( y = Bu + n \), where \( B \) is a linear operator, and \( n \) is a white Gaussian noise with variance \( \sigma^2 \).

In general, the blurring matrix \( B \) is a highly ill-conditioned matrix, so the image deblurring problem is an ill-conditioned inverse problem. A common method for solving the ill-conditioned problems is the regularization methods. Therefore, in practical applications, the image deblurring problem is converted into the following optimization problem:

$$
\min_{x} \left\{ \frac{1}{2} \|Ax - c\|^2 + \frac{\epsilon^2}{2} \|Kx\|^2 \right\} \quad (33)
$$

where \( A \) is a blurring operator, \( K \) is a regularization operator, \( \epsilon \) is a regularization parameter, and \( c \) is the observed image (see [16, 29–33]). The mathematical expression (33) can be equivalently reformulated into the equality-constraint quadratic programming problem (1)–(2) as follows:

$$
\begin{aligned}
\min_{x, y} & \frac{\epsilon^2}{2} \|Kx\|^2 + \frac{1}{2} \|y\|^2, \\
\text{s.t.} & \ Ax - y = b.
\end{aligned} \quad (34)
$$

As it was mentioned above, the blurring image \( c \) can be described as \( b = Ax^t + \omega r \), where \( x^t \) is the true image, \( r \) is the noise generated by the random vector, and \( \omega \) is the level of noise. In our simulations of image deblurring, in order to generate blurred images, first, the original images are blurred by a blurring kernel, and then followed by an additive Gaussian white noise. As in [15], we set \( \epsilon = 0.1, \beta = 0.1, \) and \( \omega = 3 \).

In our simulations, \( K \) takes the Laplacian operator. Under the periodic boundary conditions, \( K \) and \( A \) are the block circulant matrices with circulant blocks. Therefore, both \( K^T K \) and \( A^T A \) are also block circulant matrices with circulant blocks, so that they can be diagonalized by the FFT2 (2 Dimension Fast Fourier Transform). See, e.g., [34]. Therefore, the three methods experimented require \( o(n \log n) \) operations per iteration.

The PRADMM is used to solve the problem (34), and its corresponding iteration scheme is as follows:

$$
\begin{aligned}
&\begin{cases}
(\epsilon^2 K^T K + \beta A^T W^{-1} A)\hat{x}^{(k+1)} = A^T W^{-1}[\beta(c + y^{(k)}) + \lambda^{(k)}], \\
\chi^{(k+1)} = \omega \hat{x}^{(k+1)} + (1 - \omega)\chi^{(k)};
\end{cases} \\
&\begin{cases}
(I + \beta W^{-1})\hat{y}^{(k+1)} = W^{-1}[\beta(Ax^{(k+1)} - c) + \lambda^{(k)}], \\
y^{(k+1)} = \tau \hat{y}^{(k+1)} + (1 - \tau)y^{(k)};
\end{cases} \\
&\lambda^{(k+1)} = \lambda^{(k)} - a Q^{-1} W^{-1} (Ax^{(k+1)} - y^{(k+1)} - c).
\end{aligned} \quad (35)
$$
Applying the PPADMM to (33), we obtain the following iterative scheme:

\[
\begin{aligned}
&\begin{cases}
(\varepsilon^2K^T\mathbf{K} + \beta\mathbf{A}^T\mathbf{W}^{-1}\mathbf{A} + \mathbf{P})\mathbf{x}^{(k+1)} = \mathbf{P}\mathbf{x}^{(k)} + \mathbf{A}^T\mathbf{W}^{-1}[\beta(\mathbf{c} + \mathbf{y}^{(k)}) + \lambda^{(k)}], \\
(\mathbf{I} + \beta\mathbf{W}^{-1} + \mathbf{T})\mathbf{y}^{(k+1)} = \mathbf{W}^{-1}[\beta(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{c}) + \lambda^{(k)}] + \mathbf{T}\mathbf{y}^{(k)}, \\
\lambda^{(k+1)} = \lambda^{(k)} - \alpha\beta\mathbf{Q}^{-1}\mathbf{W}^{-1}(\mathbf{A}\mathbf{x}^{(k+1)} - \mathbf{y}^{(k+1)} - \mathbf{c}).
\end{cases}
\end{aligned}
\] (36)

In order to reduce the computational cost of iterative schemes (35) and (36), we choose the following term to generate the positive definite proximal term matrices \(\mathbf{P}\) and \(\mathbf{T}\) as \(\mathbf{P} = \frac{\beta_1}{\tau_1} \mathbf{I} - \beta \mathbf{A}^T \mathbf{A}, \mathbf{T} = (\frac{\beta_2}{\tau_2} - \beta)\mathbf{I}\), where \(\tau_1 \in (0, \frac{1}{\lambda_{\text{max}}(\mathbf{A}^T \mathbf{A})})\), \(\tau_2 \in (0, 1)\). For the PRADMM and PPADMM, \(\mathbf{W} = (\frac{\beta_1}{\tau_1} \mathbf{I} - \beta \mathbf{A}^T \mathbf{A})^{-1}, \mathbf{Q} = (\frac{\beta_2}{\tau_2} - \beta)\mathbf{I}\), where \(\gamma_1 \in (0, \frac{1}{\lambda_{\text{max}}(\mathbf{A}^T \mathbf{A})}), \gamma_2 \in (0, 1)\). In our simulations, \(\tau_1 = 0.9, \tau_2 = 0.04\), and \(\gamma_1 = \gamma_2 = 0.1\). It is obvious that the matrices \((\varepsilon^2K^T\mathbf{K} + \beta\mathbf{A}^T\mathbf{W}^{-1}\mathbf{A}), (\mathbf{I} + \beta\mathbf{W}^{-1}), (\mathbf{I} + \beta\mathbf{W}^{-1} + \mathbf{T})\), \((\varepsilon^2K^T\mathbf{K} + \beta\mathbf{A}^T\mathbf{W}^{-1}\mathbf{A} + \mathbf{P})\) are the block circulant matrices with circulant blocks so that their inverse matrices can be easily obtained by the FFT2. It will greatly reduce the computational cost of schemes (35) and (36).

In our experiments, the methods have been implemented in MATLAB (version R2014a) on a PC with Core i5-4590, 3.30 GHz CPU, and 8.00 GB RAM. We test the 256-by-256 grayscale images Capsule, Baboon, House, and Cameraman [35], which are shown in Figure 1. We state that the image Capsule is a .jpg file and the other three images are all .pgm files. For 8-bit images, their pixel values are all integers in \([0, 255]\). Hence, for these four images, we have \(m = n = p = 256 \times 256 = 65,536\) in the problem (34). In our experiments of deblurring, we test two types of blurring kernels: Type I (fspecial(average, 13)) and Type II (fspecial(Gaussian, [9, 9], 3)). For the three methods in the experiment, all initial values are chosen as the blurred image \(\mathbf{c}\), i.e., \(\mathbf{x}^{(0)} = \mathbf{c}\), which is terminated if the current iterates satisfy \(\frac{||\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}||}{||\mathbf{x}^{(k-1)}||} \leq 10^{-5}\) and are performed in MATLAB with a machine precision \(10^{-16}\).

![Image](image1.png)

(a) Capsule (b) Baboon

(c) House (d) Cameraman

**Figure 1.** The original images; (a) Capsule. (b) Baboon. (c) House. (d) Cameraman.

The peak signal-to-noise ratio (PSNR) is the most common and widely used objective measurement method for evaluating the quality of an image. It is widely used in the
literature to investigate the performance of the methods solving the image restoration problem. The PSNR is defined as follows:

$$\text{PSNR}(x) = 20 \log_{10} \frac{255}{\sqrt{\text{var}(x, x^t)}}$$

with$$\text{var}(x, x^t) = \frac{1}{n} \sum_{j=0}^{n-1} (x^t_j - x_j)^2$$

where $x^t_j$ is the $j$-th pixel value of the original image $x^t$, and $x_j$ is the $j$-th pixel values of the corresponding estimated image $y = Bu + n$. PSNR is a very important index in image deblurring, which can measure the difference between the restored image and the original image. In general, the PSNR value increases with the quality of restoration. In image deblurring, if the PSNR is larger, the effect of image deblurring is better. In order to obtain the high value of PSNR, it will be worth taking a long time. Table 1 shows the optimal parameters for the PPADMM and the PRADMM in the experiment, respectively. These optimal parameters are obtained experimentally by maximizing the corresponding PSNR.

Table 1. The optimal iteration parameters computed by experiment for the preconditioned and proximal alternating direction method of multipliers (PPADMM) and preconditioned and relaxed alternating direction method of multipliers (PRADMM).

| Image | Capsule | Baboon | House | Cameraman |
|-------|---------|--------|-------|-----------|
| blur type | I | II | I | II | I | II | I | II |
| PPADMM | $a_\tau$ | 2.0 | 2.1 | 2.1 | 2.1 | 2.1 | 2.1 | 2.1 |
| PRADMM | $\omega_\tau$ | 0.8 | 0.8 | 0.8 | 0.8 | 0.8 | 0.8 | 0.8 |
| | $\tau_\omega$ | 0.6 | 0.6 | 0.6 | 0.6 | 0.6 | 0.6 | 0.6 |
| | $a_\omega$ | 0.23 | 0.23 | 0.23 | 0.23 | 0.23 | 0.26 | 0.25 |

In order to illustrate the effectiveness and robustness of the PPADMM to the different images, we show the experimental results for the PPADMM, PRADMM, and ADMM to solve the image deblurring problem in Table 2. Table 2 shows the elapsed CPU time in seconds (denoted as “CPU”) and the peak signal-to-noise ratio measured in the described (dB) (denoted as “PSNR”) image deblurring problem (34). In order to be accurate, the experimental results are the average value of the three repeatedly executed results. In short, the PPADMM, PRADMM, and ADMM are denoted as PPAD, PRAD, and AD. From Table 2, we can see that the PPADMM acquires the highest value of PSNR for the four images of two types of the kernel. The difference with respect to the PSNR ranges from 0.38 to 0.94 dB. In order to know the relationship between PSNR and iteration, Figure 2 takes the capsule blurred by Type I as an example, which shows the PSNR versus iterations for the PPADMM, PRADMM, and ADMM. We can see that for the image capsule blurred by Type I, the PPADMM was able to solve the problem (34) in 83 iterations and achieved a PSNR value of 25.05 dB. The PRADMM and ADMM took 18 iterations and 47 iterations to solve the problem (33) and obtained a PSNR value of 24.5 dB. Clearly, although the PPADMM took more iterations than the PRADMM and ADMM, the value of PSNR that the PPADMM calculated is much larger than those of the other two methods. Above all, the PPADMM is much more effective than the PRADMM and the ADMM in the qualities of the deblurred images, and it is robust.
Table 2. The elapsed CPU time (CPU), peak signal-to-noise ratio (PSNR) and valofun for the PPAD (PPADMM), PRAD (PRADMM) and AD (alternating direction method of multipliers, or ADMM) with respect to optimal parameters.

| Image     | Blur Type | Cpu Ppad | Prad Ppad | Ad Ppad | Prad Prad | Ad Ad |
|-----------|-----------|----------|-----------|---------|-----------|------|
| Capsule   | I         | 1.018558 | 0.211038  | 0.449924| 25.0501   | 24.4955   | 24.4942   |
|           | II        | 0.732373 | 0.198781  | 0.425788| 26.5942   | 26.0038   | 26.0026   |
| Baboon    | I         | 0.670735 | 0.190145  | 0.428676| 20.8321   | 20.4507   | 20.4500   |
|           | II        | 0.625869 | 0.180112  | 0.411398| 21.0639   | 20.6318   | 20.6311   |
| House     | I         | 0.604927 | 0.188013  | 0.433318| 26.6212   | 26.1505   | 26.1484   |
|           | II        | 0.544012 | 0.161778  | 0.404065| 27.3147   | 26.3739   | 26.3706   |
| Cameraman | I         | 0.677334 | 0.169664  | 0.421255| 23.7530   | 23.2583   | 23.2566   |
|           | II        | 0.538381 | 0.168758  | 0.416063| 24.4705   | 23.6654   | 23.6632   |

Figure 2. Pictures of PSNR versus iterations for the PPADMM, PRADMM, and ADMM.

Finally, for the blurred (Type I) and noised (B&N) image capsule, we deblur it by the ADMM, PRADMM, and PPADMM, respectively, and we show the deblurred images in Figure 3. We can see that the PRADMM and ADMM produced over-smoothed results and eliminated many images details. Instead, the PPADMM ensured that the restored images have better visual quality. On the one hand, it can remove effectively the blurring effects and noise. On the other hand, it is also able to reconstruct more image edges than the two other methods.

In summary, it can be concluded from Table 2 and Figures 1 and 2 that for the image deblurring problem (33), the proposed PPADMM is clearly more effective than the PRADMM and ADMM in obtaining comparable qualities of the deblurred images.
5. Conclusions

In this paper, we have proposed an efficient PPADMM for solving the linear constraint quadratic programming problem (1)–(2). This algorithm is a proximal generalization of the PADMM, and it extrapolates the block variables and the block multiplier in each new iteration. In fact, the PPADMM is naturally a generalized and modified block SOR iterative method in the viewpoint of matrix computation. Its theoretical properties such as global convergence and convergence factor are established. Meanwhile, our numerical results verify the efficiency of the proposed method in comparison with PRADMM and ADMM. In addition, it is easy to be applied to construct the similar method for solving convex optimization problems of several block variables and of several equality and inequality constraints. However, how to choose the optimal parameters is still a challenging problem that deserves further discussion.

Author Contributions: Conceptualization, H.-L.S. and X.T.; methodology, H.-L.S. and X.T.; software, X.T.; validation, H.-L.S. and X.T.; formal analysis, H.-L.S. and X.T.; investigation, H.-L.S.; resources, H.-L.S.; data curation, X.T.; writing—original draft preparation, H.-L.S. and X.T.; writing—review and editing, H.-L.S.; visualization, H.-L.S.; supervision, H.-L.S. and X.T.; project administration, H.-L.S.; funding acquisition, H.-L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Liaoning Province (No. 20170540323), Central University Basic Scientific Research Business Expenses Special Funds (N2005013).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The codes related to numerical examples refer to the link: https://github.com/Klarustx/ADMM/tree/master/code (accessed on 6 July 2020).
Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wang, H.; Li, G.; Tsai, C.L. Regression coefficient and autoregressive order shrinkage and selection via the lasso. J. R. Stat. Soc. 2007, 69, 63–78.
2. Markowitz, H.M. Portfolio Selection: Efficient Diversification of Investment. J. Inst. Actuar. 1992, 119, 243–265.
3. André, P.; Harry, M. Markowitz, Sparsity and Piecewise Linearity in Large Portfolio Optimization Problems. In Sparse Matricies and Their Uses; Academic Press: Cambridge, MA, USA, 1981; pp. 89–108.
4. Recht, B.; Fazel, M.; Parrilo, P.A. Guaranteed Minimum-Rank Solutions of Linear Matrix Equations via Nuclear Norm Minimization. SIAM Rev. J. 2010, 52, 471–501. [CrossRef]
5. Haber, E.; Modersitzki, J. Numerical methods for volume preserving image registration. Inverse Probl. J. 2004, 20, 1621–1638. [CrossRef]
6. Han, D.; Yuan, X.; Zhang, W. An augmented Lagrangian based parallel splitting method for separable convex minimization with applications to image processing. J. Math. Comput. 2014, 83, 2263–2291. [CrossRef]
7. Boyd, S.; Parikh, N.; Chu, E. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. J. Found. Trends Mach. Learn. 2010, 3, 1–122. [CrossRef]
8. Erseghe, T.; Zennaro, D.; Dall’Anese, E.; Vangelista, L. Fast Consensus by the Alternating Direction Multipliers Method. J. IEEE Trans. Signal Process. 2011, 59, 5523–5537. [CrossRef]
9. He, B.; Xu, M.; Yuan, X. Solving Large-Scale Least Squares Semidefinite Programming by Alternating Direction Methods. SIAM J. Matrix Anal. Appl. 2011, 32, 136–152. [CrossRef]
10. Amir, F.; Farajzadeh, A.; Petrot, N. Proximal Point Algorithm for Differentiable Quasi-Convex Multiobjective Optimization. J. Optim. 2020, 34, 2367–2376. [CrossRef]
11. Goodrich, C.S.; Ragusa, M.A. Holder continuity of weak solutions of p-Laplacian PDEs with VMO coefficients. J. Nonlinear Anal. Theory Methods Appl. 2019, 185, 336–355. [CrossRef]
12. Suantai, S.; Kankam, K.; Cholamjiak, P. A Novel Forward-Backward Algorithm for Solving Convex Minimization Problem in Hilbert Spaces. J. Math. 2020, 8, 42. [CrossRef]
13. Daniel, B. Local Linear Convergence of the Alternating Direction Method of Multipliers on Quadratic or Linear Programs. SIAM J. Optim. 2013, 23, 2183–2207.
14. He, B.; Yuan, X. On the O(1/n) Convergence Rate of the Douglas-Rachford Alternating Direction Method. SIAM J. Numer. Anal. 2012, 50, 700–709. [CrossRef]
15. Morini, B.; Porcelli, M.; Chan, R.H. A reduced Newton method for constrained linear least-squares problems. J. Comput. Appl. Math. 2010, 233, 2200–2212. [CrossRef]
16. Miller, J.H. On the Location of Zeros of Certain Classes of Polynomials with Applications to Numerical Analysis. J. Inst. Math. Appl. 1971, 8, 397–406. [CrossRef]
17. Powell, M.J.D. A method for nonlinear constraints in minimization problems. J. Optim. 1969, 5, 283–298.
18. Bai, Z.Z.; Tao, M. Rigorous convergence analysis of alternating variable minimization with multiplier methods for quadratic programming problems with equality constraints. J. Bit Numer. Math. 2016, 56, 399–422. [CrossRef]
19. Glowinski, R.; Marroco, A. Sur l’approximation, par pénalisation-dualité, d’une classe de problèmes de Dirichlet non linéaires. J. Equine Vet. Sci. 1975, 2, 41–76. [CrossRef]
20. Gabay, D.; Mercier, B. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. Comput. J. Math. Appl. 1976, 2, 17–40. [CrossRef]
21. Oden, J.T. Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics; Glowinski, R., Tallec, R.L.M., Eds.; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 1989.
22. Eckstein, J.; Bertsekas, D.P. On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators. J. Math. Program. 1992, 55, 293–318. [CrossRef]
23. Bertsekas, D.P. Constrained Optimization and Lagrange Multiplier Methods; Academic Press: Cambridge, MA, USA, 1982.
24. Gabay, D. Chapter ix applications of the method of multipliers to variational inequalities. In Studies in Mathematics and Its Applications; Elsevier: Amsterdam, The Netherlands, 1983; Volume 15, pp. 299–331.
25. Deng, W.; Yin, W. On the Global and Linear Convergence of the Generalized Alternating Direction Method of Multipliers. J. Sci. Comput. 2016, 66, 889–916. [CrossRef]
26. Bai, Z.Z.; Tao, M. On Preconditioned and Relaxed AVMM Methods for Quadratic Programming Problems with Equality Constraints. J. Linear Algebra Its Appl. 2016, 516, 264–285. [CrossRef]
27. Bai, Z.Z.; Parlett, B.N.; Wang, Z.Q. On generalized successive overrelaxation methods for augmented linear systems. J. Numer. Math. 2005, 102, 1–38. [CrossRef]
28. Young, D.M. Iterative Solution of Large Linear Systems; Academic Press: New York, NY, USA, 1971.
29. Xie, S.; Rahardja, S. Alternating Direction Method for Balanced Image Restoration. J. IEEE Trans. Image Process. 2012, 21, 4557–4567.
30. Hearn, T.A.; Reichel, L. Image Denoising via Residual Kurtosis Minimization. J. Numer. Math. Theory Methods Appl. 2015, 8, 406–424. [CrossRef]
31. Hochstenbach, M.E.; Noschese, S.; Reichel, L. Fractional regularization matrices for linear discrete ill-posed problems. *J. Eng. Math.* **2015**, *93*, 1–17. [CrossRef]

32. Hearn, T.A.; Reichel, L. Application of denoising methods to regularization of ill-posed problems. *J. Numer. Algorithms* **2014**, *66*, 761–777. [CrossRef]

33. Hageman, L.A.; Young, D.M. Iterative Solution of Large Linear Systems. *J. Am. Math. Mon.* **1973**, *80*, 92. [CrossRef]

34. Chan, R.H.; Ng, M.K. Conjugate Gradient Methods for Toeplitz Systems. *SIAM Rev.* **1995**, *38*, 427–482. [CrossRef]

35. Available online: http://decsai.ugr.es/cvg/dbimagenes/ (accessed on 13 March 2014).