(c) http://onlinelibrary.wiley.com/doi/10.1002/nla.2040/pdf
Numerical Linear Algebra with Applications
Volume 23, Issue 3, 2016

Preconditioning Techniques for an Image Deblurring **Problem**

Ke Chen[1], Faisal Fairag[2], Adel Al-Mahdi[3]

Abstract

In this paper, we consider the solution of a large linear system of equations which is obtained from discretizing the Euler Lagrange equations associated with the image deblurring problem. The coefficient matrix of this system is of the generalized saddle point form with high condition number. One of the blocks of this matrix has the block Toeplitz with Toeplitz block (BTTB) structure. This system can be efficiently solved using the minimal residual (MINRES) iteration method with preconditioners based on the fast Fourier transform (FFT). Eigenvalue bounds for the preconditioner matrix are obtained. Numerical results are presented.

Keywords: Preconditioning technique, Saddle-point problems, Image deblurring, Krylov subspace method, TV Regularization, Primal dual formulation, BTTB matrix, FFT.
2010 MSC: 65F10, 65F08, 65F15, 65N22, 68U10, 94A08

---

1. Introduction

Image deblurring problem requires solving a large, dense, ill-conditioned linear system of equations. For example an image with 256×256 resolution requires solving system of size 256^2. The suitable choice of linear solver is an iterative method such as a Krylov subspace method. Unfortunately, Krylov subspace methods such as the conjugate gradient (CG) method or

*Corresponding author
Email addresses: K.Chen@liverpool.ac.uk (Ke Chen), ffairag@kfupm.edu.sa (Faisal Fairag), almahdi@kfupm.edu.sa (Adel Al-Mahdi)
[1]Department of Mathematical Sciences, University of Liverpool, Liverpool L697ZL, UK
[2]Department of Mathematics and statistics, KFUPM, Dhahran, 31261 Saudi Arabia
[3]Department of Mathematics and statistics, KFUPM, Dhahran, 31261 Saudi Arabia

the minimal residual (MINRES) method are very slow with ill-conditioned linear system of equations. One technique to overcome this slowness property is using an appropriate preconditioner. A good preconditioner which accelerates the convergence needs to be easy to construct and cheap to invert. Moreover, the preconditioned matrix should have eigenvalues clustering behavior. Many preconditioners in [4] are developed for a saddle point problem. In this research work, we convert the linear system resulted from image deblurring problem into a saddle point problem and we develop block preconditioners with two parameters. These preconditioners are of Murphy, Golub and Wathen type [20]. Moreover, we give a bounds on all positive and negative eigenvalues. These bounds depend on the values of the parameters. The selection of these two parameters will affect the clustering behavior of the eigenvalues. Our preconditioners involve a Schur complement matrix which contains a product of a Toeplitz matrix with Toeplitz blocks (BTTB) and its transpose. This product may not be a BTTB. So, we approximate this product by a symmetric BTTB matrix [23]. The benefit of this approximation is to reduce the storage and operation numbers due to fast Fourier transformation (FFT) and the convolution theorem. This paper is organized as follows: In section 2, we introduce the problem and invert it into a saddle point problem. We develop a block diagonal preconditioner with two parameters in section 3. We drive bounds on the eigenvalues of the preconditioned matrix in section 4. Approximation of the blurring matrix is given in section 5. We give some numerical examples and show the algorithm's performance in sections 6. Finally, we give a short summary in section 7.

2. Problem Setup

To deblur an image, we need a mathematical model of how it was blurred. Blurring and noise affect the quality of the received image. The recorded image z and the original image u are related by the equation

$$z = \mathbf{K}u + \varepsilon, \tag{1}$$

where $\mathbf{K}$ denotes the blurring operator and ε denotes the noise function. $\mathbf{K}$ is typically a Fredholm integral operator of the first kind,

$$(\mathbf{K}u)(x) = \int_\Omega k(x, x')u(x')dx', \qquad x \in \Omega \tag{2}$$

with translational invariance, the kernel $k(x, x') = k(x - x')$ is known as the point spread function (PSF). The operator $\mathbf{K}$ is compact, so problem (1) is ill-posed [17]. Ω will denote a square in $\mathcal{R}^2$ on which the image intensity function u is defined. $x = (\underline{x}, \underline{y})$ denotes the location in Ω; $\mid x \mid = \sqrt{\underline{x}^2 + \underline{y}^2}$ denotes the Euclidean norm, and $\| \cdot \|$ denotes the norm in $\mathcal{L}^2(\Omega)$. To stabilize problem (1) the total variation (TV) regularization functional, which was introduced in [24] by Rudin, Osher, and Fatemi, is often used. The problem is then to find a u which minimizes the functional

$$T(u) = \frac{1}{2} \| \mathbf{K}u - z \|^2 + \alpha J(u), \tag{3}$$

with positive parameter α and the total variational functional [1] is given by

$$J(u) = \int_\Omega \mid \nabla u \mid . \tag{4}$$

However, the derivative of the integrand in equation (4) does not exist at zero. One remedy of this issue [17] is to add a constant β as follows

$$J_\beta(u) = \int_\Omega \sqrt{\mid \nabla u \mid^2 + \beta^2}. \tag{5}$$

Then the functional to be minimized is

$$T(u) = \frac{1}{2} \| \mathbf{K}u - z \|^2 + \alpha \int_\Omega \sqrt{\mid \nabla u \mid^2 + \beta^2}, \tag{6}$$

with $\alpha,\ \beta > 0$. The well-posedness of this minimization is established in [1]. The Euler-Lagrange equations associated with the above minimization problem are

$$\mathbf{K}^*(\mathbf{K}u - z) + \alpha L(u)u = 0 \qquad x \in \Omega, \tag{7}$$

$$\frac{\partial u}{\partial n} = 0 \qquad x \in \partial\Omega, \tag{8}$$

where $\mathbf{K}^*$ is the adjoint operator of the integral operator $\mathbf{K}$. The differential operator $L(u)$ is given by

$$L(u)w = -\nabla.\left(\frac{1}{\sqrt{\mid \nabla u \mid^2 + \beta^2}} \nabla w\right). \tag{9}$$

Note that (7) is a nonlinear integro-differential equation of elliptic type. Equations (7-8) can be expressed as a nonlinear first order system [10]

$$\mathbf{K}^*\mathbf{K}u - \alpha\nabla.\vec{v} = \mathbf{K}^*z, \tag{10}$$

$$-\nabla u + \sqrt{|\nabla u|^2 + \beta^2}\,\vec{v} = \vec{0}, \tag{11}$$

with the dual, or flux, variable

$$\vec{v} = \frac{\nabla u}{\sqrt{|\nabla u|^2 + \beta^2}}. \tag{12}$$

To discretize (10) and (11), we start by dividing the square domain $\Omega = (0,1) \times (0,1)$ into n_x^2 equals squares (cells) where n_x denotes the number of equispaced partitions in the x or y directions. The cell centers are denoted by (x_i, y_j) and given by

$$\begin{array}{ll} x_i = (i - \frac{1}{2})h & i = 1, ..., n_x, \\ y_j = (j - \frac{1}{2})h & j = 1, ..., n_x, \end{array}$$

where $h = \frac{1}{n_x}$. The midpoints of cell edges are given by $(x_{i\pm\frac{1}{2}}, y_j)$ and $(x_i, y_{j\pm\frac{1}{2}})$ where

$$\begin{array}{ll} x_{i\pm\frac{1}{2}} = x_i \pm \frac{h}{2} & i = 1, ..., n_x, \\ y_{j\pm\frac{1}{2}} = y_j \pm \frac{h}{2} & j = 1, ..., n_x. \end{array}$$

The set

$$e_{ij} = \{(x, y) : x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}],\ y \in [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]\},$$

represents a cell with (x_i, y_j) as a center. Let

$$\chi_i(x) = \begin{cases} 1, & \text{if } x \in (x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}); \\ 0, & \text{otherwise.} \end{cases}$$

$$\chi_j(y) \begin{cases} 1, & \text{if } y \in (y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}); \\ 0, & \text{otherwise,} \end{cases}$$

and

$$\begin{array}{l} \phi_i(x_{l+\frac{1}{2}}) = \delta_{il}, \\ \phi_j(y_{k+\frac{1}{2}}) = \delta_{jk}. \end{array}$$

Approximate u as

$$u(x,y) \simeq U(x,y) = \sum_{i=1}^{n_x} \sum_{j=1}^{n_x} u_{ij} \chi_i(x) \chi_j(y),$$

where $U(x_i, y_j) = u_{ij}$ and represent the data z as

$$z(x,y) \simeq Z(x,y) = \sum_{i=1}^{n_x} \sum_{j=1}^{n_x} z_{ij} \chi_i(x) \chi_j(y),$$

where z_{ij} may be calculated as cell averages. Also, approximate v by

$$v(x,y) \simeq \sum_{i=1}^{n_x-1} \sum_{j=1}^{n_x} V_{ij}^x \begin{pmatrix} \phi_i(x)\chi_j(y) \\ 0 \end{pmatrix} + \sum_{i=1}^{n_x-1} \sum_{j=1}^{n_x} V_{ij}^y \begin{pmatrix} 0 \\ \phi_i(y)\chi_j(x) \end{pmatrix}$$

Now, applying Galerkin's method to (10-11) together with midpoint quadrature for the integral term and cell centered finite difference method (CCFD) for the derivative part (see [15], [30]), one obtains the following system

$$K_h^* K_h U + \alpha B_h^* V = K_h^* Z, \tag{13}$$

$$\alpha B_h U - \alpha D_h V = 0 \tag{14}$$

Here K_h is a matrix of size $n \times n$ and B_h is a matrix of size $m \times n$. D_h is a matrix of $m \times m$ (here $n = n_x^2$ and $m = 2n_x(n_x - 1)$). For simplicity we eliminate the subscript h equipped with the matrices in (13,14) and then one can re-write them after rearrangement the unknowns as

$$\begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^* K \end{bmatrix} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} 0 \\ -K^* Z \end{bmatrix}, \tag{15}$$

Both K^*K and $L = B^* D^{-1} B$ are symmetric positive semi definite matrices [1]. The matrix K is a BTTB matrix. The matrix D is a diagonal with positive diagonal entries

$$D = \begin{bmatrix} D^x & 0 \\ 0 & D^y \end{bmatrix},$$

where D^x is an $(n_x - 1) \times n_x$ and D^y an $n_x \times (n_x - 1)$ matrices with diagonal entries obtained by discretize the expression $\sqrt{|\nabla u|^2 + \beta^2}$. The matrix B is given by

$$B = \frac{1}{h} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix},$$

where the matrices B_1 $(n_x(n_x-1)\times n)$ and B_2 $(n_x(n_x-1)\times n)$ have the following structures

$$B_1 = \begin{bmatrix} -I & I & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -I & I \end{bmatrix},$$

where I is the identity matrix of size n_x by n_x.

$$B_2 = \begin{bmatrix} E & 0 & 0 & 0 & 0 \\ 0 & E & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & 0 & E \end{bmatrix},$$

where E $((n_x-1)\times n_x)$ is given by

$$E = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}.$$

Note that one can eliminate V from (13) and (14) to get the following primal system

$$(K^*K + \alpha L)U = K^*Z. \tag{16}$$

If Tikhonov regularization is used then (16) becomes

$$(K^*K + \alpha I)U = K^*Z, \tag{17}$$

where I is the identity matrix of the same size of K. The linear system (15) can be seen as a generalized saddle point version of (16). Another generalized saddle point version of (16) is

$$\begin{bmatrix} I & K \\ -K^* & \alpha L \end{bmatrix} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} Z \\ 0 \end{bmatrix}. \tag{18}$$

We note that (15), (16) and (18) are equivalent. In the next paragraph, we discuss several iterative methods for solving these three equivalent systems.

In [30], Vogel and Oman introduced product preconditioner for the system (16) with approximating the BTTB matrix by (block circulant with circulant block) BCCB. Chan et. al in [8] introduced cosine-transform based preconditioners for the (TV) deblurring problem. Donatelli in [12] used another solver for the deblurring problem with Dirichlet and periodic boundary conditions. The blurring matrices are BTTB and BCCB. He solved the resulting systems by applying a multigrid method and he showed an optimality property with $O(n)$ arithmetic operations where n is the linear system size. In [13], Donatelli and Hanke introduced an iterative scheme similar to nonstationary iterated Tikhonov regularization for (17). The rapid convergence of their method is determined by adaptive strategy of selecting the regularization parameters. For the second version of the generalized saddle point problem (18), NG and Pan in [21] developed new Hermitian and skew-Hermitian splitting (HSS) preconditioners for solving such system with weighted matrix. They gave a strategy to choose the HSS parameters to force all eigenvalues of the preconditioned matrices to be clustered around one and hence the Krylov subspace method converges very quickly. Axelsson and Neytcheva [3, 2] introduced a block diagonal preconditioner (P_{AN}) for generalized saddle point problem in the same structure as (15). They derived bounds on the eigenvalues of the preconditioned matrix. In this research work, we introduce block diagonal preconditioners for (15). Our proposed preconditioners can be seen as an Axelsson and Neytcheva preconditioner with two parameters. For more detail on iterative methods for image deblurring we refer to [5].

3. The Preconditioner

Let us denote the matrix equation given in (15) by $Ax = b$, one can note that the coefficient matrix A is symmetric but not positive definite. So, the suitable Krylov subspace method for such matrices is the MINRES method. Unfortunately, the convergence is slow. To overcome this slowness, we introduce the following symmetric positive definite preconditioner.

$$\bar{P} = \begin{bmatrix} \alpha\gamma_1 D & 0 \\ 0 & \gamma_2 \bar{S} \end{bmatrix}, \tag{19}$$

where $\bar{S}$ is an approximation to the matrix $S = (K^*K + \alpha L)$. This matrix, S, is the Schur complement of the matrix A. γ_1 and γ_2 are positive parameters. The preconditioner $\bar{P}$ in (19) is an approximation of the following exact preconditioner matrix

$$P = \begin{bmatrix} \alpha\gamma_1 D & 0 \\ 0 & \gamma_2 S \end{bmatrix}.$$

In case $\gamma_1 = \gamma_2 = 1$, the preconditioner matrix P is the Axelsson and Neytcheva preconditioner P_{AN}. Since the coefficient matrix A is symmetric and indefinite, the appropriate iterative method is the preconditioned MINRES [22]. In general, MINRES minimizes the residual over the shifted Krylov subspace. For more detail in preconditioning technique we refer to see [4], [20] and [6].

4. Eigenvalues Estimates

In order to have information of the spectral properties of the preconditioned matrix $\bar{P}^{-1}A$, we need to study the spectral properties of the preconditioned matrix $P^{-1}A$. In this section we give a bound for the positive and negative eigenvalues of the preconditioned matrix $P^{-1}A$ but before doing that, we start by discussing the number of the positive and negative eigenvalues of the preconditioned matrix $P^{-1}A$. Note that the preconditioned matrix $P^{-1}A$ is similar to the matrix $P^{-1/2}AP^{-1/2}$. The matrix $P^{-1/2}AP^{-1/2}$ can be decomposed into

$$\begin{bmatrix} I_m & 0 \\ -\sqrt{\frac{\alpha\gamma_1}{\gamma_2}} S^{-1/2}B^* D^{-1/2} & I_n \end{bmatrix} \begin{bmatrix} \frac{1}{\gamma_1} I_m & 0 \\ 0 & -\frac{1}{\gamma_2} I_n \end{bmatrix} \begin{bmatrix} I_m & -\sqrt{\frac{\alpha\gamma_1}{\gamma_2}} D^{-1/2}BS^{-1/2} \\ 0 & I_n \end{bmatrix},$$

where I_m and I_n are the identities matrices of size $m \times m$ and $n \times n$ respectively. The above decomposition is known as the congruence transformations of the matrix $P^{-1/2}AP^{-1/2}$. By Sylvesters law of inertia (page 403 in [16]), congruence transformations preserve the signs of the eigenvalues [14]. It follows that the number of the positive eigenvalues of $P^{-1}A$ is m and the number of the negatives is n (here $m > n$). Several bounds on the eigenvalues of the generalized saddle point matrix are established in [25, 28] and [3, 2]. Here we state the bounds in [2].

Theorem 4.1 (Theorem 1 in [2] p 4). *Let* $\hat{A} = \begin{bmatrix} \hat{M} & \hat{B}^T \\ \hat{B} & -\hat{C} \end{bmatrix}$, *where* $\hat{M}$ *and* $\hat{S} = \hat{C} + \hat{B}\hat{M}^{-1}\hat{B}^T$ *are symmetric and positive definite. Let* $0 < \hat{\mu_1} \leq \hat{\mu_2} \leq ... \leq \hat{\mu_n}$, $0 < \hat{\sigma_1} \leq \hat{\sigma_2} \leq ... \leq \hat{\sigma_m}$ *be the eigenvalues of* $\hat{M}$ *and* $\hat{B}\hat{M}^{-1}\hat{B}^T$, *respectively, and let* $\gamma^2 = \rho(\hat{S}^{-1/2}\hat{B}\hat{M}^{-1}\hat{B}^T\hat{S}^{-1/2})$, *the spectral* *radius. Then the eigenvalues* (λ_i) *of* $\hat{A}$ *are located in the two intervals:* $[-\lambda_{max}(\hat{S}), \frac{-\lambda_{min}(\hat{S})}{1+\frac{\gamma^2}{\hat{\mu_1}}\lambda_{min}(\hat{S})}] \cup [\hat{\mu_1}, \hat{\mu_n} + \hat{\sigma_m}]$. *If* $\hat{C}$ *is positive semidefinite then the upper bound can be replaced by the more accurate bound* $\hat{\mu_n}\frac{1+\sqrt{1+\frac{4\hat{\sigma_m}}{\hat{\mu_n}}}}{2}$.

In the following theorem, we give upper and lower bounds for the positive and negative eigenvalues of $P^{-1}A$.

Theorem 4.2. *The* $m+n$ $(\mu_{-n} \leq \mu_{-n+1} \leq ... \leq \mu_{-1} < 0 < \mu_1 \leq \mu_2 \leq ... \leq \mu_m)$ *eigenvalues of the generalized eigenvalue problem,*

$$\begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^*K \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} \alpha\gamma_1 D & 0 \\ 0 & \gamma_2 S \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{20}$$

satisfy the following:

$$\mu_i \in \left[\frac{1}{\gamma_1}, \frac{1+\sqrt{1+\frac{4\alpha\gamma_1}{\gamma_2}\sigma_m}}{2\gamma_1}\right] \quad i = 1, ..., m, \tag{21}$$

$$\mu_{-j} \in \left[-\frac{1}{\gamma_2}, -\frac{1}{\gamma_2+\alpha\gamma_1\tau}\right] \quad j = 1, ..., n, \tag{22}$$

where γ_1 *and* γ_2 *are positive parameters.* σ_m *is the maximum eigenvalue of* $S^{-1/2}LS^{-1/2}$ *and* $\tau = \rho(S^{-1/2}LS^{-1/2})$*, the spectral radius.*

Proof: We start expressing the preconditioned matrix $P^{-1}A$ in a generalized saddle point matrix. $P^{-1}A$ is similar to $P^{\frac{1}{2}}(P^{-1}A)P^{-\frac{1}{2}} = P^{-\frac{1}{2}}AP^{-\frac{1}{2}} =$

$$= \begin{bmatrix} \frac{1}{\sqrt{\alpha\gamma_1}}D^{-\frac{1}{2}} & 0 \\ 0 & \frac{1}{\sqrt{\gamma_2}}S^{-\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^*K \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{\alpha\gamma_1}}D^{-\frac{1}{2}} & 0 \\ 0 & \frac{1}{\sqrt{\gamma_2}}S^{-\frac{1}{2}} \end{bmatrix} \tag{23}$$

$$= \begin{bmatrix} \frac{\alpha}{\sqrt{\alpha\gamma_1}}D^{\frac{1}{2}} & \frac{-\alpha}{\sqrt{\alpha\gamma_1}}D^{-\frac{1}{2}}B \\ \frac{-\alpha}{\sqrt{\gamma_2}}S^{-\frac{1}{2}}B^* & \frac{-1}{\sqrt{\gamma_2}}S^{-\frac{1}{2}}K^*K \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{\alpha\gamma_1}}D^{-\frac{1}{2}} & 0 \\ 0 & \frac{1}{\sqrt{\gamma_2}}S^{-\frac{1}{2}} \end{bmatrix} \tag{24}$$

$$= \begin{bmatrix} \frac{1}{\gamma_1}I & -\sqrt{\frac{\alpha}{\gamma_1\gamma_2}}D^{-\frac{1}{2}}BS^{-\frac{1}{2}} \\ -\sqrt{\frac{\alpha}{\gamma_1\gamma_2}}S^{-\frac{1}{2}}B^*D^{-\frac{1}{2}} & \frac{-1}{\gamma_2}S^{-\frac{1}{2}}K^*KS^{-\frac{1}{2}} \end{bmatrix} \tag{25}$$

$$= \begin{bmatrix} \hat{M} & \hat{B}^* \\ \hat{B} & -\hat{C} \end{bmatrix} = \hat{\mathcal{A}}. \tag{26}$$

Now one can use Theorem 4.1 with the matrices

$$\hat{M} = \frac{1}{\gamma_1} I, \qquad \hat{B} = -\sqrt{\frac{\alpha}{\gamma_1\gamma_2}} S^{-\frac{1}{2}} B^* D^{-\frac{1}{2}},$$
$$\hat{C} = \frac{1}{\gamma_2} S^{-\frac{1}{2}} K^* K S^{-\frac{1}{2}}, \qquad \hat{S} = \frac{1}{\gamma_2} I_n,$$

and

$$\lambda_{max}(\hat{S}) = \frac{1}{\gamma_2}, \qquad \lambda_{min}(\hat{S}) = \frac{1}{\gamma_2},$$
$$\hat{\mu_1} = \frac{1}{\gamma_1}, \qquad \hat{\mu_n} = \frac{1}{\gamma_1},$$
$$\hat{\sigma_m} = \text{maximum eigenvlaue of } \frac{\alpha}{\gamma_2} S^{-\frac{1}{2}} L S^{-\frac{1}{2}},$$
$$\gamma^2 = \rho(\alpha S^{-1/2} L S^{-1/2}),$$

to obtain the bound given in (21) and (22).

Remark 4.1. *In the above theorem and its proof, since both P and S are positive definite then $P^{-1/2}$, $P^{1/2}$ and $S^{-1/2}$ are well defined.*

Remark 4.2. *If $\gamma_1 = \gamma_2 = 1$ (P_{AN}), then (21) and (22) are given by*

$$\mu_i \in \left[1, \frac{1+\sqrt{1+4\alpha\sigma_m}}{2}\right] \quad i = 1, ..., m,$$
$$\mu_{-j} \in \left[-1, -\frac{1}{1+\alpha\tau}\right] \quad j = 1, ..., n.$$

Remark 4.3. *From (21) and (22), one can note that the smaller value of $\frac{\gamma_1}{\gamma_2}$ yields the smaller length of both intervals. This means that we have a good clustering behavior for the negative and positive eigenvalues. Hence, we expect fast convergence.*

4.1. Numerical results for the eigenvalues analysis

Our aim is to verify that the bounds given in Theorem 4.2 are matched with the following numerical example. In this example we take $n_x = 4$, $\beta = 1$ and $\alpha = 8 \times 10^{-5}$ with the kernel described in (2). Table 1 shows the upper and lower (positive/negative) bounds of the intervals given in the above theorem. Also it shows the maximum and the minimum (positive/negative) eigenvalues of the preconditioned matrix $P^{-1}A$. These eigenvalues are computed using the built-in Matlab command *eig*.

γ_1, γ_2	Bounds in (21,22)	Computed eigenvalues
P_{AN}	$[-1, -6.42e-1] \cup [1, 1.39]$	$[-1, -7.59e-1] \cup [1, 1.31]$
$1e-3, 1$	$[-1, -9.99444e-1] \cup [1e+3, 1.0005555e+3]$	$[-1, -9.99445e-1] \cup [1e+3, 1.0005552e+3]$
$1e-6, 1$	$[-1, -9.999994441e-1]$ $\cup$ $[1e+6, 1.0000005558257e+6]$	$[-1, -9.999994442e-1]$ $\cup$ $[1e+6, 1.0000005558255e+6]$

Table 1: Bounds on eigenvalues of the preconditioned matrix $P^{-1}A$

In Table 1, observe that all intervals in the third column are contained in the second column. This observation verifies the bounds given in Theorem 4.2.

It is known that the PMINRES convergence estimate [14] can be written as

$$\frac{\| r^{(k)} \|_{P^{-1}}}{\| r^{(0)} \|_{P^{-1}}} \leq \min_{q_k \in \Pi_k \; q_k(0)=1} \; \max_{\lambda \in \sigma(P^{-1}A)} \mid q_k(\lambda) \mid, \tag{27}$$

where Π_k is the space of all polynomial of degree less than or equals k and $\| r^{(0)} \|^2_{P^{-1}} = {r^{(0)}}^T P^{-1} r^{(0)}$. To minimize the right hand side of the above inequality (27) we need to cluster both the positive and negative eigenvalues. This can be obtained by reducing the lengths of the intervals in (21) and (22).

5. Approximating K^*K

An $n \times n$ matrix M is Toeplitz if the entries along each diagonal are the same. A circulant matrix is a Toeplitz matrix for which each column is a circular shift of the elements in the preceding column (so that the last entry becomes the first entry). In our problem, K is BTTB matrix and it has the block form

$$K = \begin{bmatrix} T_0 & T_{-1} & \cdots & T_{1-n} \\ T_1 & T_0 & T_{-1} & \cdots \\ \vdots & \ddots & \ddots & T_{-1} \\ T_{n-1} & \cdots & T_1 & T_0 \end{bmatrix}, \tag{28}$$

where each block T_j is a Toeplitz matrix. The first row and the first column uniquely define a Toeplitz matrix. Circulant preconditioning for Toeplitz systems was introduced by Strang [29] and extended by others to block Toeplitz systems [11]. Many researchers use a Toeplitz preconditioners and block Toeplitz preconditioners for Toeplitz systems see for instance [9] and [18]. Band Toeplitz preconditioner and band BTTB preconditioner are proposed in Chan [7] and Serra [27]. In [19], BTTB preconditioners for BTTB systems are discussed.
In our preconditioner $\bar{P}$ given in (19), note that K is a BTTB matrix but K^*K need not be BTTB. So, we follow [23] to approximate K^*K given in the preconditioner matrix P by a symmetric BTTB matrix T. Symmetric BTTB matrices can always be extended to form symmetric BCCB matrices. The benefit of this approximation is that the matrix-vector products that involve $n_x^2 \times n_x^2$ matrices can be computed in $O\ (n_x^2 \log n_x)$ operations due to the FFT's and the Convolution Theorem. Moreover, all that is needed for computation is the first column of the matrix, which decreases the amount of required storage. Note that, the preconditioner $\bar{P}$ requires the solution of $T + \alpha L$. We use the Conjugate Gradient (CG) method to solve the system $(T + \alpha L)x = y$. In CG, we need only matrix-vector product in the form $(T + \alpha L)v$ which is simply $Tv + \alpha Lv$.

6. Numerical Experiments

In this section, we solve the linear system (15) using preconditioned MINRES method (PMINRES) with $\bar{P}$ in (19) as a preconditioner. PMINRES needs to compute matrix-vector product in the form $(K^*K + \alpha L)v$ which is simply $K^*(Kv) + \alpha Lv$. Here, both K^*q and $q = Kv$ can be done quickly. In each PMINRES iteration, a linear system of type $\bar{P}v = w$ needs to be solved. The (1,1)-block in the matrix $\bar{P}$ is a diagonal matrix and hence easy to invert. Conjugate Gradient (CG) method is used to solve the second part of $\bar{P}v = w$. We present some results based on the image data given in Figure 1. The blurred image is shown in Figure 3. Figure 2 shows the kernel described in (2). In this experiment, we used PMINRES as a linear solver, we take $n_x = 256$, the resulting system has $256^2 \approx 6.55 \times 10^4$ unknowns. We take $\alpha = 8 \times 10^{-5}$, $\beta = 1$, $tol = 1e - 4$ and we fix $\gamma_1 = \gamma_2 = 1$ in the preconditioner matrix $\bar{P}$. Figure 4 displays the deblurred image achieved by using the total variation (TV) reconstruction algorithm.

Figures (5-6) show the efficiency of the different preconditioners. In each

Figure 1: The exact image.

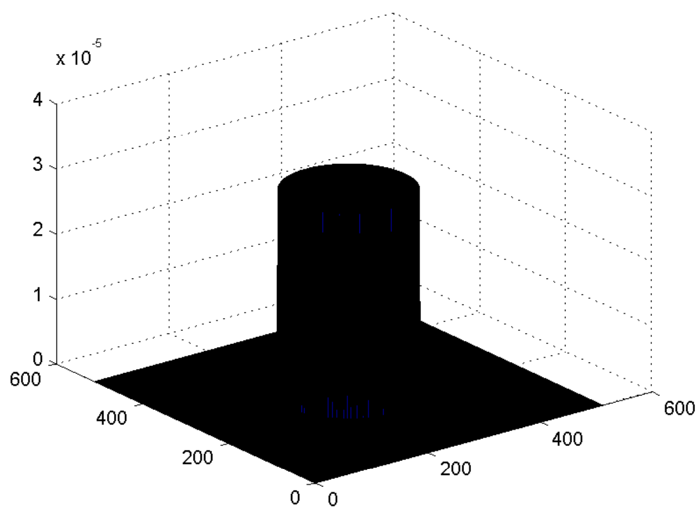

Figure 2: The kernel.

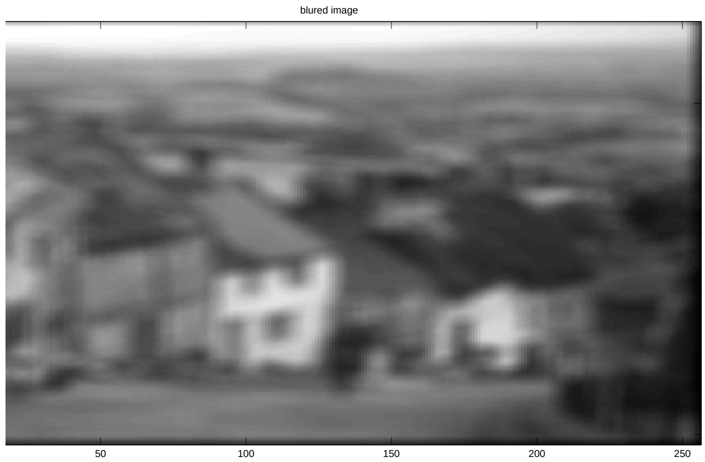

Figure 3: The blurred image.

Figure 4: The deblurred image.

PMINRES iteration, the logarithm of $\frac{\|r^{(n)}\|_{\bar{P}^{-1}}}{\|r^{(0)}\|_{\bar{P}^{-1}}}$ ($\bar{P}$ is the preconditioner and r is the residual) is calculated and then plotted in Figures (5-6). In this comparison study we have chosen $n_x = 128$ and $\beta = 0.01$. Here P_0 refers to no-preconditioner, P_{11} (where P_{11} ia an approximation to P_{AN}) to the preconditioner $\bar{P}$ with $\gamma_1 = \gamma_2 = 1$, P_{12} to the preconditioner $\bar{P}$ with $\gamma_1 = 1,\ \gamma_2 = 10$, P_2 to the preconditioner $\bar{P}$ with $\gamma_1 = 1e-3,\ \gamma_2 = 1$ and finally P_3 refers to the preconditioner $\bar{P}$ with $\gamma_1 = 1e-6,\ \gamma_2 = 1$. In Figures 5 and 6 observe that unpreconditioned MINRES converged most slowly, followed by PMINRES P_{11} and then both P_0 and P_{11} are followed by P_{12}. We note that PMINRES P_3 is the fastest one. This has the smallest value of the parameter γ_1 which leads to the best clustering behavior of the eigenvalues (see Remark 4.3 and Table 1). The CPU time and the measure of image quality, Peak Signal-to-Noise Ratio, (PSNR) for the preconditioners P_{11}, P_{12}, P_2 and P_3 are given in Table 2. In this table, we compute the CPU time for 15 iterations for P_{11} to reach $tol = 1e-3$, 10 iterations for P_{12} to reach $tol = 1e-3$, 7 iterations for P_2 to reach $tol = 1e-3$ and 6 iterations for P_3 to reach the same tolerance (see Figure 5). Through this comparison, we find that the PSNR for the blurred image is (21.2004) while the PSNR for deblurred image can be seen in Table 2.

	P_{11}	P_{12}	P_2	P_3
CPU(in second)	23.59	14.52	12.53	11.24
PSNR for deblurred image (in decibels)	26.6606	26.6673	26.6609	26.6609

Table 2: The CPU time and the PSNR for the preconditioners P_{11}, P_{12}, P_2 and P_3

Condition number for the preconditioned matrices with several values of $\gamma_1 \in [1e-6, 9e-6]$ and $\gamma_2 \in [0.4, 1.3]$ are computed and plotted in Figure 7.

7. PMINRES .vs. FGMRES

Using a Krylov subspace method as a preconditioner within a different kyrlov subspace method may lead to a changing preconditioner. In such cases, the preconditioner matrix changes from step to step. Flexible GMRES (FGMRES) [26], allows the preconditioner to vary from step to step. For sake of comparison, we solve the linear system (15) using PMINRES and

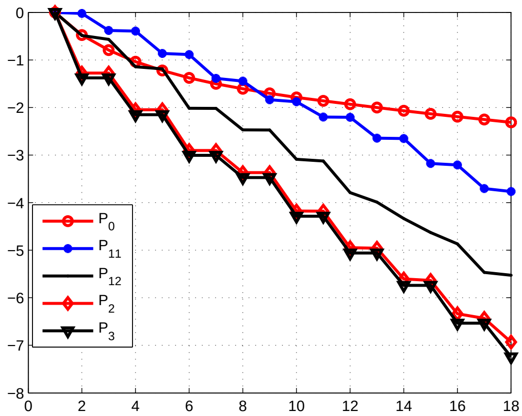

Figure 5: Residual .vs. iteration $\alpha = 8e - 5$

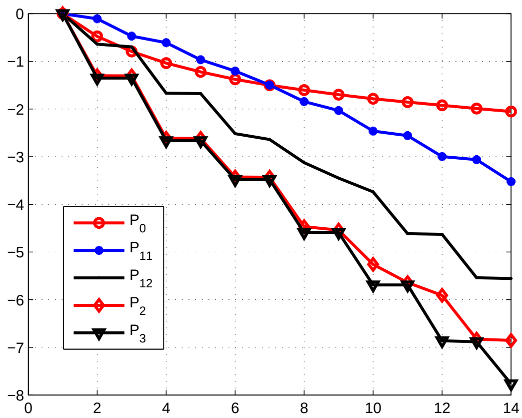

Figure 6: Residual .vs. iteration $\alpha = 8e - 4$

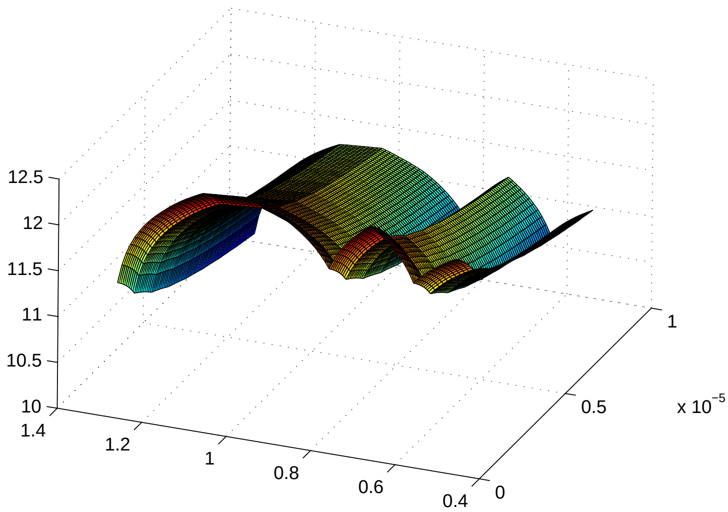

Figure 7: Condition numbers with different γ_1 and γ_2.

FGMRES with $\bar{P}$ given in (19)as a preconditioner. In this experiment, we take $n_x = 128$, $\alpha = 8e-4$, $\beta = 0.1$ and $tol = 1e-7$. The results are based on the image data given in Figure 1. In each PMINRES iteration, the logarithm of $\frac{\|r^{(n)}\|_{\bar{P}^{-1}}}{\|r^{(0)}\|_{\bar{P}^{-1}}}$ is calculated and then plotted in Figure 8. In each FMINRES iteration, the logarithm of $\frac{\|r^{(n)}\|_2}{\|r^{(0)}\|_2}$ is calculated and then plotted in Figure 9. In these figures, we observe that in both PMINRES and FGMRES, the unpreconditioned converged most slowly, followed by $\bar{P}_{AN}$ and then both P_0 and P_{11} are followed by P_{12}. We note that P_3 is the fastest one.

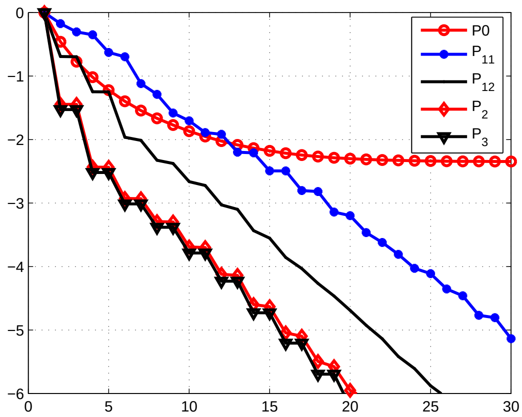

Figure 8: Residual .vs. iteration (PMINRES)

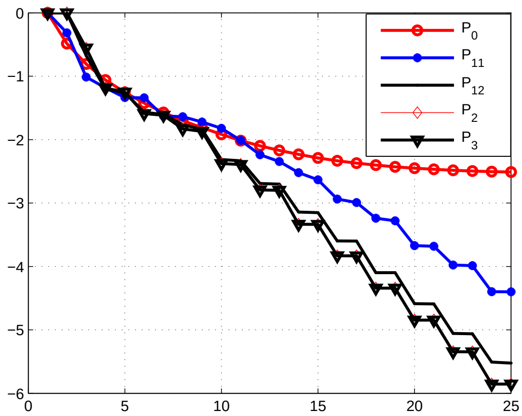

Figure 9: Residual .vs. iteration (FGMRES)

8. Summary

Block diagonal preconditioning techniques for the image deblurring problem using primal-dual formulation are presented. Bounds on the eigenvalues of the preconditioned matrix are obtained and verified with several experiments. The proposed preconditioner is used to accelerate the reconstruction of the blurred image. Residual plot shows fast convergence of the method.

9. Acknowledgment

The authors would like to thank the editor and the referees for their helpful comments and suggestions which helped to improve the presentation of the present paper. This research work was undertaken while the second author was visiting the Centre for Mathematical Imaging Techniques (CMIT) at University of Liverpool supported by King Abdulaziz City for Science and Technology (KACST). He expresses thanks to CMIT for their hospitality and support. He would like to acknowledge the support provided by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) for funding this work through project No. IN131044.

References

[1] Robert Acar and Curtis R Vogel. Analysis of bounded variation penalty methods for ill-posed problems. *Inverse problems*, 10(6):1217, 1994.

[2] Owe Axelsson. Eigenvalue estimates for preconditioned saddle point matrices. In *Large-Scale Scientific Computing*, pages 3–16. Springer, 2004.

[3] Owe Axelsson and Maya Neytcheva. Eigenvalue estimates for preconditioned saddle point matrices. *Numerical Linear Algebra with Applications*, 13(4):339–360, 2006.

[4] Michele Benzi, Gene H Golub, and Jörg Liesen. Numerical solution of saddle point problems. *Acta numerica*, 14(1):1–137, 2005.

[5] Sebastian Berisha and James G Nagy. Iterative methods for image restoration. *Academic Press Library in Signal Processing: Image, Video Processing and Analysis, Hardware, Audio, Acoustic and Speech Processing*, 4:193–247, 2014.

[6] Zhi-Hao Cao. A note on constraint preconditioning for nonsymmetric indefinite matrices. *SIAM Journal on Matrix Analysis and Applications*, 24(1):121–125, 2002.

[7] Raymond H Chan. Toeplitz preconditioners for Toeplitz systems with nonnegative generating functions. *IMA journal of numerical analysis*, 11(3):333–345, 1991.

[8] Raymond H Chan, Tony F Chan, and Chiu-Kwong Wong. Cosine transform based preconditioners for total variation deblurring. *Image Processing, IEEE Transactions on*, 8(10):1472–1478, 1999.

[9] Raymond H Chan and Kwok-Po Ng. Toeplitz preconditioners for Hermitian Toeplitz systems. *Linear algebra and its applications*, 190:181–208, 1993.

[10] Tony F Chan, Gene H Golub, and Pep Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, 20(6):1964–1977, 1999.

[11] Tony F Chan and Julia A Olkin. Circulant preconditioners for Toeplitz-block matrices. *Numerical Algorithms*, 6(1):89–101, 1994.

[12] M Donatelli. A multigrid for image deblurring with Tikhonov regularization. *Numerical linear algebra with applications*, 12(8):715–729, 2005.

[13] Marco Donatelli and Martin Hanke. Fast nonstationary preconditioned iterative methods for ill-posed problems, with application to image deblurring. *Inverse Problems*, 29(9):095008, 2013.

[14] Howard Elman, David Silvester, and Andy Wathen. *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*. Oxford University Press, 2014.

[15] Richard E Ewing and Jian Shen. A multigrid algorithm for the cell-centered finite difference scheme. In *NASA Conference Publication*, pages 583–583. NASA, 1993.

[16] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, third edition, 1996.

[17] Charles W Groetsch and CW Groetsch. *Inverse problems in the mathematical sciences*, volume 52. Springer, 1993.

[18] Fu-Rong Lin. Preconditioners for block Toeplitz systems based on circulant preconditioners. *Numerical Algorithms*, 26(4):365–379, 2001.

[19] Fu-Rong Lin and Chi-Xi Wang. BTTB preconditioners for BTTB systems. *Numerical Algorithms*, 60(1):153–167, 2012.

[20] Malcolm F Murphy, Gene H Golub, and Andrew J Wathen. A note on preconditioning for indefinite linear systems. *SIAM Journal on Scientific Computing*, 21(6):1969–1972, 2000.

[21] Michael K Ng and Jianyu Pan. Weighted Toeplitz regularized least squares computation for image restoration. *SIAM Journal on Scientific Computing*, 36(1):B94–B121, 2014.

[22] Christopher C Paige and Michael A Saunders. Solution of sparse indefinite systems of linear equations. *SIAM Journal on Numerical Analysis*, 12(4):617–629, 1975.

[23] Kyle Lane Riley. *Two-level preconditioners for regularized ill-posed problems*. PhD thesis, Montana State University-Bozeman, 1999.

[24] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[25] Torgeir Rusten and Ragnar Winther. A preconditioned iterative method for saddlepoint problems. *SIAM Journal on Matrix Analysis and Applications*, 13(3):887–904, 1992.

[26] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[27] Stefano Serra. Preconditioning strategies for asymptotically ill-conditioned block Toeplitz systems. *BIT Numerical Mathematics*, 34(4):579–594, 1994.

[28] David Silvester and Andrew Wathen. Fast iterative solution of stabilised Stokes systems part ii: using general block preconditioners. *SIAM Journal on Numerical Analysis*, 31(5):1352–1367, 1994.

[29] Gilbert Strang. A proposal for Toeplitz matrix calculations. *Studies in Applied Mathematics*, 74(2):171–176, 1986.

[30] Curtis R Vogel and Mary E Oman. Fast, robust total variation-based reconstruction of noisy, blurred images. *Image Processing, IEEE Transactions on*, 7(6):813–824, 1998.