Performance of Three Preconditioners for Image Deblurring Problem in Primal-Dual Formulation

F. Fairag and A. Al-Mahdi

Abstract—**In this paper, we consider the generalized saddle point linear system of equations which is obtained from discretizing the Euler Lagrange equations associated with image debulrring problem. This system is ill-conditioned and is of huge size. Moreover, the (2,2) block of the coefficient matrix of this system contains summation of two terms. One of these terms is a product of a Toepelitz matrix with Toepelitz blocks (BTTB) and its transpose. This structure needs a prconditioner to speed up the iterative method such us the minimal residual (MINRES) method. Hence, we devolve three block diagonal preconditioners which are of Murphy, Golub and Wathen [MGW] type for this saddle point system. The first preconditioner is based on approximating the product of the (BTTB) matrix and its transpose by a symmetric BTTB matrix while in the second one, we approximate the BTTB matrix by the Strang circulant. In the the third preconditioner, we approximate the BTTB matrix by the optimal circulant. We investigate the efficiency of these three preconditioners by several numerical computations in term of CPU time, iteration numbers and the quality of the reconstruction images.**

Index Terms—**Preconditioning technique, saddle-point problems, Image deblurring, Krylov subspace method, Strang circulant, optimal circulant.**

I. Introduction

IN this paper, we consider the following generalized saddle point system

$$\underbrace{\begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^*K \end{bmatrix}}_{C} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} 0 \\ -K^*Z \end{bmatrix}, \qquad (1)$$

resulted from discretizing the Euler Lagrange equations associated with image debulrring problem. Here Z is a given data, $[V\ U]^T$ is the solution vector. In the coefficient matrix, C, of the above system (1), the matrices K, B and D are of sizes $n \times n$, $m \times n$ and $m \times m$, respectively. Here $n = n_x^2$ and $m = 2n_x(n_x - 1))$ where n_x denotes the number of equispaced partitions in the x or y directions and the parameter $\alpha > 0$ is small number which is called the regularities parameter. Both K^*K and $L = B^*D^{-1}B$ are symmetric positive semi definite matrices [21]. The matrix K and its transpose have the block Toeplitz with Toeplitz block (BTTB) structures but K^*K need not be BTTB. It is know that image debulrring 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 256^4. Hence the only choice of linear solver to the above system is the iterative method such as Krylov subspace method. Unfortunately, Krylov subspace method as conjugate gradient (CG) method or minimal residual (MINRES) method are very slow with ill-conditioned linear system of equations. One technique to overcome this slowness properties 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 preconditions in [2] are developed for a spital linear system such as a saddle point problem. In this research work, we convert the linear system resulted from image debulrring problem into a saddle point problem and then we develop three block preconditioners for this saddle point problem. These preconditioners are of Murphy, Golub and Wathen [MGW] type [15] and they involve a Schur complement matrix which contains a product of a Toeplitz matrix with Toeplitz blocks (BTTB) and its transpose. The first preconditioner is based on approximating this product, this product may not be a BTTB, by a symmetric BTTB matrix [17] while the second is based on approximating a Toepelitz matrix by Strang circulant (see [20], [7]). In the third preconditioner, we use the optimal circulant approximation for a Toeplitz matrix [8]. In this paper, we investigate the efficient of these preconditioners by several numerical computations in term of CPU time, iteration numbers and the quality of the reconstruction image. This paper is organized as follows: In section II, we introduce the problem and invert it into a saddle point problem. In section III, we devolve a block diagonal preconditioner that involves a Schur complement matrix which contains a product of a Toeplitz matrix with Toeplitz blocks (BTTB) and its transpose. In sections IV, we give some numerical examples and show the algorithm performance. Finally, in section V, we give some conclusions.

II. Problem setup

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

$$z = \mathbf{K}u + \varepsilon \qquad (2)$$

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

$$(\mathbf{K}u)(x) = \int_\Omega k(x, x')u(x')dx', \qquad x \in \Omega \qquad (3)$$

with translation 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 (2) is ill-posed [1], [21]. Ω will denote a square in $\mathcal{R}^2$ on which the image intensity function u is

Manuscript received December 28, 2014. This work was supported by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) under Grant No. IN131044.

F. Fairag is with the Department of Mathematics and Statistics, KFUPM, 31261 Saudi Arabia e-mail: (ffairag@kfupm.edu.sa).

A. Al-Mahdi is with the Department of Mathematics and Statistics, KFUPM, 31261 Saudi Arabia e-mail: (almahdi@kfupm.edu.sa).

Proceedings of the International MultiConference of Engineers and Computer Scientists 2015 Vol I,
IMECS 2015, March 18 - 20, 2015, Hong Kong

defined. $x = (\underline{x}, \underline{y})$ denotes location in Ω; $\mid x \mid = \sqrt{\underline{x}^2 + \underline{y}^2}$ denotes Euclidean norm, and $\| \cdot \|$ denotes the norm in $\mathcal{L}^2(\Omega)$. To stabilize problem (2) the total variation (TV) regularization functional, which was introduced in [18] by Rudin, Osher, and Fatemi, is often used. The problem is then to find a u which minimizes the functional

$$T(u) = \frac{1}{2} \| \mathbf{K}u - z \|^2 + \alpha J(u), \tag{4}$$

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

$$J(u) = \int_\Omega \mid \nabla u \mid . \tag{5}$$

However, the derivative of integrand in equation (5) does not exist at zero. One remedy of this issue [1], [21] is to add a constant β as follows

$$J_\beta(u) = \int_\Omega \sqrt{\mid \nabla u \mid^2 + \beta^2}. \tag{6}$$

Then the functional to be minimized is

$$T(u) = \frac{1}{2} \| \mathbf{K}u - z \|^2 + \alpha \int_\Omega \sqrt{\mid \nabla u \mid^2 + \beta^2}, \tag{7}$$

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

$$\mathbf{K}^*(\mathbf{K}u - z) + \alpha L(u)u = 0 \qquad x \in \Omega, \tag{8}$$

$$\frac{\partial u}{\partial n} = 0 \qquad x \in \partial\Omega \tag{9}$$

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

$$L(u)w = -\nabla.(\frac{1}{\sqrt{\mid \nabla u \mid^2 + \beta^2}} \nabla w). \tag{10}$$

Note that (8) is a nonlinear integro-differential equation of elliptic type. It can be expressed as a nonlinear first order system [9]

$$\mathbf{K}^*\mathbf{K}u - \alpha \nabla.\vec{v} = \mathbf{K}^* z \tag{11}$$

$$-\nabla u + \sqrt{\mid \nabla u \mid^2 + \beta^2}\vec{v} = \vec{0}, \tag{12}$$

with the dual, or flux, variable

$$\vec{v} = \frac{\nabla u}{\sqrt{\mid \nabla u \mid^2 + \beta^2}} \tag{13}$$

Applying Galerkin's method to (11-12) together with mid-point quadrature for the integral term and cell center finite difference method (CCFD) for the derivative part [11], one obtain the following dual-system

$$K_h^* K_h U + \alpha B_h^* V = K_h^* Z, \tag{14}$$

$$\alpha B_h U - \alpha D_h V = 0 \tag{15}$$

For the simplicity we eliminate the subscript h equipped with the matrices in (14-15) and then one can re-write them after rearrangement the unknowns as

$$\underbrace{\begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^*K \end{bmatrix}}_{\text{C}} \begin{bmatrix} V \\ U \end{bmatrix} = \begin{bmatrix} 0 \\ -K^*Z \end{bmatrix}, \tag{16}$$

Here Z is the given data, $[V\ U]^T$ is the solution vector. The matrix K is a dense matrix and has the BTTB structure while the matrix K^*K may not be BTTB. 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 $\sqrt{\mid \nabla u \mid^2 + \beta^2}$. The matrix B is given by

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

where $h = \frac{1}{n_x}$ and 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

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

Since the coefficient matrix C is symmetric but not positive definite, the suitable Krylov subspace method for this system is MINRES method [16] but the convergence of this method well be slow. This slowness is resulted from the huge size of C. To over come this problem the preconditioning technique is needed. In our paper, we provide three block digonal preconditioners for the generalized saddle point problem given in (16).

Note that one can eliminate V from (14,15) to get the following primal system

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

where K^*K is full matrix and $L = B^*D^{-1}B$ is sparse matrix . In (17), the matrix $H = K^*K + \alpha L$ is symmetric positive definite (SPD) but it is extremely large. Since H is (SPD), then on can use the CG method but the convergence of this method well be slow. This slowness is resulted from the huge size of H. Hence, preconditioning technique is needed. In [21], Vogel et al. introduced product preconditioner for the system (17) with approximating the BTTB matrix by block circulant with circulant block (BCCB). Chan et al. [5] introduced cosine-transform based preconditioners for the system (17).

III. The block diagonal preconditioners

At the beginning, we introduce the following symmetric positive definite preconditioner

$$P = \begin{bmatrix} \alpha D & 0 \\ 0 & S \end{bmatrix}, \tag{18}$$

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

where $S = (K^*K + \alpha L)$ is the Schur complement of the matrix C. Hence, the appropriate iterative method is preconditioned MINRES which minimizes the residual over the shifted Krylov subspace. For more detail in preconditioning technique we refer to [2] and [3].

The following theorem [12] gives upper and lower bounds for the positive and negative eigenvalues of the preconditioned matrix $P^{-1}C$.

Theorem 3.1: 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,

$$\begin{bmatrix} \alpha D & -\alpha B \\ -\alpha B^* & -K^*K \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} \alpha D & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (19)$$

satisfy the following:

$$\mu_i \in \left[1, \frac{1+\sqrt{1+4\alpha\sigma_m}}{2}\right] \quad i = 1, ..., m, \quad (20)$$

$$\mu_{-j} \in \left[-1, \frac{-1}{1+\alpha\tau}\right] \quad j = 1, ..., n, \quad (21)$$

where 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.
The above cluster of the preconditioned matrix $P^{-1}C$ gives grantees of the convergence of the PMINRES method. It is known in each PMINRES iteration we need to solve a linear system of the form $Px = y$. Since, the (2,2) block of P contains a full matrix K^*K, then we lock for some circulant approximations to K^*K to ease the computations and to reduce the storages. The mean contribution of our paper is that we introduce three circulant approximations to the product K^*K in the preconditioner given in (18). The advantage of these approximations is to apply the fast Fourier transformation (FFT) and the Convolution Theorem.

A. 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

$$K = \begin{bmatrix} T_0 & T_{-1} & \cdots & T_{1-n} \\ T_1 & T_0 & T_{-1} & \cdots \\ \vdots & \ddots & \ddots & T_{-1} \\ T_{n-1} & \cdots & T_1 & T_0 \end{bmatrix} \quad (22)$$

where each block T_j is a Toeplitz matrix. The first row and the first column are uniquely define a Toeplitz matrix. Circulant preconditioning for Toeplitz systems was introduced by Strang [20] and extended by others to block Toeplitz systems [10]. Many researchers use a Toeplitz preconditioners and block Toeplitz preconditioners for Toeplitz systems see for instance [6] and [13]. Band Toeplitz preconditioner and band BTTB preconditioner are proposed in Chan [4] and Serra Capizzano [19]. In [14], BTTB preconditioners for BTTB systems are discussed. In our preconditioner given in (18), note that K is a BTTB matrix but K^*K may not a BTTB. So, the first approach that we follow is to approximate K^*K given in the preconditioner matrix P by a symmetric BTTB matrix T (see [17]) because 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 $n \times n$ matrices can be computed in $O\ (n \log n)$ operations instead of $O\ (n^2)$. This reduction is due to the FFT 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. The second approach that we follow is that we approximate the n by n Toeplitz matrix K given in the preconditioner matrix P by the well known Strang circulant matrix S with diagonals s_k by copying the central diagonals of K and circulate them around to complete the circulant (see [7] page 17–18). More precisely, if $n = 2m+1$ the diagonals s_k of S are given by

$$s_k = \begin{cases} k_k, & 0 \leq k \leq m, \\ k_{k-n}, & m < k < n-1, \\ \bar{s}_{-k}, & 0 < -k < n-1, \end{cases} \quad (23)$$

where k_i is the ith diagonal of the matrix K. If $n = 2m$, we get the Strang matrix S as above. In this case, we define $s_m = 0$ or $s_m = \frac{k_m + k_{-m}}{2}$. In the last approach, we also approximate the Toeplitz matrix K given in the preconditioner matrix P by an optimal circulant matrix C.

If C_n denote the set of $n \times n$ circulant matrices. The optimal circulant approximation to $K \in \mathbb{C}^{n \times n}$ in the Frobenius norm is given by $C = \arg \min_{B \in C_n} \parallel B - K \parallel_{Fro}$. In this case, the value of the entries c_k of the matrix C is obtained by averaging the corresponding diagonal of K (extended to length n by a wraparound [8]). Then the matrix C, whose entries are given by $c_k = \frac{kT_{-(n-k)} + (n-k)T_k}{n}$, $k = -(n-1), ..., 0, ..., (n-1)$. So our preconditioner is being in the forms

$$P_T = \begin{bmatrix} \alpha D & 0 \\ 0 & T + \alpha L \end{bmatrix}, \quad P_S = \begin{bmatrix} \alpha D & 0 \\ 0 & S^*S + \alpha L \end{bmatrix}, \quad P_C = \begin{bmatrix} \alpha D & 0 \\ 0 & C^*C + \alpha L \end{bmatrix}, \quad (24)$$

IV. Numerical Experiment

The aim of this section is to investigate the efficiency of the three preconditioners described above for several blurry images. Two images are considered the first one is a retinal image of a diabetic patient (see Fig. 2(a)) and the second one is goldhill image (see Fig. 3(a)). We start by blurred the two images by a certain kernel given in Figs. 2(f) or in Figs. 3(f) and then we deblurred the images back and solving the linear system by preconditioned MINRES method using these three preconditioners and we use the well known fixed point iteration method to linearized the non-linear term. We watch the cpu time and the number of MINRES iterations. It is known that in each PMINRES iteration, we solve a linear system of the form $Px = y$. To solve this system, we use the conjugate gradient method (CG) for the (2,2) block. All numerical computations were obtained using MATLAB 7 installed on HP-laptop with intel Core 2 Duo CPU processer and with RAM of 4 GB.

Example 1: In this example, we calculate the iterations number of PMINRES by using the preconditioners P_T, P_S and P_C. We fix the maximum iteration to be 100, the tolerance $1e-2$, $beta = 0.01$, $alpha = 0.00008$, and we use the retinal image (blurred image) given in Fig. 1(b) as a data with PSNR = 20.5548.

TABLE I: The Preconditioner P_T

Fixed Point Iteration Number	nx	dof	PMINRES Iteration	PSNR
1	128	48896	> 100	40.6813
2	128	48896	> 100	42.2709
3	128	48896	14	42.5842
4	128	48896	3	42.5841
5	128	48896	1	42.5841

TABLE II: The Preconditioner P_S

Fixed Point Iteration Number	nx	dof	PMINRES Iteration	PSNR
1	128	48896	> 100	40.6510
2	128	48896	78	42.6645
3	128	48896	6	42.6688
4	128	48896	1	42.6688
5	128	48896	1	42.6688

TABLE III: The Preconditioner P_C

Fixed Point Iteration Number	nx	dof	PMINRES Iteration	PSNR
1	128	48896	> 100	40.6493
2	128	48896	81	42.6535
3	128	48896	6	42.6577
4	128	48896	1	42.6577
5	128	48896	1	42.6577

TABLE IV: Comparison between P_T, P_S and P_C

nx	dof	CPU Time of P_T	CPU Time of P_S	CPU Time of P_C
128	48896	74.706	39.243	42.653

Firstly, we start by using the preconditioner P_T. Table I shows the degree of freedom (dof), the PMINRES iterations and the PSNR in each iteration of the fixed point method.

Secondly, we use preconditioner P_S with the same blurred image and the same parameters given above. Table II shows the degree of freedom (dof), the PMINRES iterations and the PSNR in each iteration of the fixed point method.

Finally, we use preconditioner P_C with the same bulurred image and the same parameters given above. Table III show the degree of freedom (dof), the PMINRES iterations and the PSNR in each iteration of the fixed point method. For the qualities of the reconstruction images using these three preconditioners, see Fig. 2(c-e). In this example, the second computations carried out for the second data (blurred image) given in Fig. 3(b) which is blurred by the same kernel given in Fig. 3(f). The qualities of the reconstruction images are shown in Fig. 3(c-e).

Example 2: In this example we compare the CPU-time of the three PMINRES preconditioned. In Table (IV), we list the CPU-time of the PMINRES spends to do 5 fixed point iterations.

Example 3: In this example, we compare the iteration numbers of the three PMINRES preconditioners with the same bulurred image and the same parameters given in the above examples. Fig. 1 shows the convergence of the methods. From Fig. 1, it can be seen that the preconditioner P_S is the fastest one followed by P_C and then P_T. It is clear that P_S needs 78 iterations to reach the $tol = 1e-2$, P_C needs 81 while P_T needs more than 100 iterations to reach the same tolerance. Note that we take the PMINRES iterations for these three preconditioners at the second iteration of the fixed point iteration method.

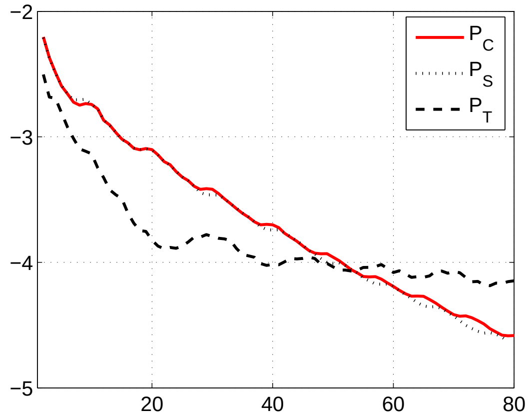


Fig. 1: Iterations Number v.s. the Residual

Example 4: In this example, we use the true image given in Fig. 3(a) and the blurred images given in Fig. 3(b), (it is blurred by using the kernel given in Fig. 3(f)), and we fix the preconditioner to be P_T. We watch the quality of the deblurred images in each fixed point iteration. Fig. 4(a-d) show the deblurred images for iteration 1, 5, 10 and 13. The second computations carried out for different values of the regularization parameters α. Fig. 5(a-d) show the deblurred images for $\alpha = 8e-2, 8e-4, 8e-7, 8e-8$.

V. Conclusion

Three different preconditioners for the generalized saddle point system resulted from discretizing the Euler Lagrange equations associated with image debulrring problem are presented. In these preconditioners, three approximations for the product of the BTTB matrix and its transpose are considered. From the computations, we observe that the P_S preconditioner is the most effective one followed by P_C and then by P_T.

Acknowledgment

The authors would like to thank the support provided by the Deanship of Scientific Research (DSR) at King Fahd University of Petroleum and Minerals (KFUPM) for funding this work.

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] Michele Benzi, Gene H Golub, and Jorg Liesen. Numerical solution of saddle point problems. *Acta numerica*, 14(1):1–137, 2005.
[3] Zhi-Hao Cao. A note on constraint preconditioning for nonsymmetric indefinite matrices. *SIAM Journal on Matrix Analysis and Applications*, 24(1):121–125, 2002.
[4] Raymond H Chan. Toeplitz preconditioners for toeplitz systems with nonnegative generating functions. *IMA journal of numerical analysis*, 11(3):333–345, 1991.

[5] 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.

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

[7] Raymond Hon-Fu Chan and Xiao-Qing Jin. *An introduction to iterative Toeplitz solvers*, volume 5. SIAM, 2007.

[8] Tony F Chan. An optimal circulant preconditioner for toeplitz systems. *SIAM journal on scientific and statistical computing*, 9(4):766–771, 1988.

[9] 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.

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

[11] 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.

[12] F Fairag Ke chen and A Al-Mahdi. Preconditioning technique for image deblurring problem. *Numerical Linear Algebra*, 60(1):153–167, 2014.

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

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

[15] 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.

[16] 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.

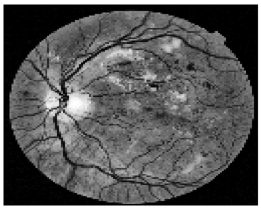
[17] Kyle Lane Riley. *Two-Level Preconditioners For Regularized Ill-Posed Problems*. PhD thesis, Montana State University-Bozeman, 1999.

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

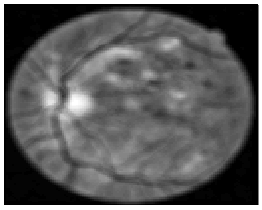
[19] Stefano Serra. Preconditioning strategies for asymptotically ill-conditioned block toeplitz systems. *BIT Numerical Mathematics*, 34(4):579–594, 1994.

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

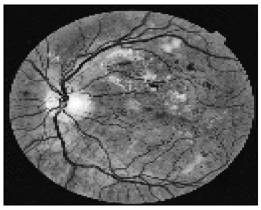
[21] Curtis R Vogel and Mary E Oman. Fast, robust total variation-based reconstruction of noisy, blurred images. *Image Processing, IEEE Transactions on*, 7(6):813–824, 1998.



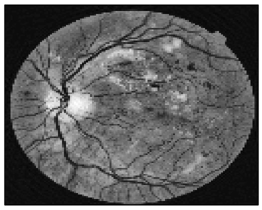
(a) True Image



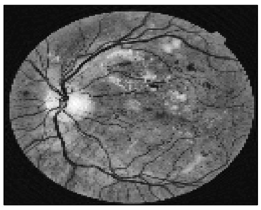
(b) Blurred Image



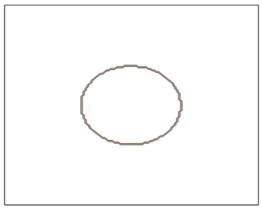
(c) Deblured Image P_T



(d) Deblured Image P_S



(e) Deblured Image P_C



(f) Kernel

Fig. 2: Retinal Image of a Diabetic Patient

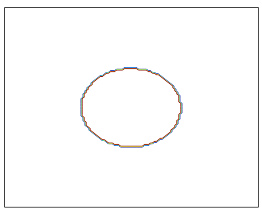
(a) True Image 2

(b) Blurred Image

(c) Deblured Image P_T

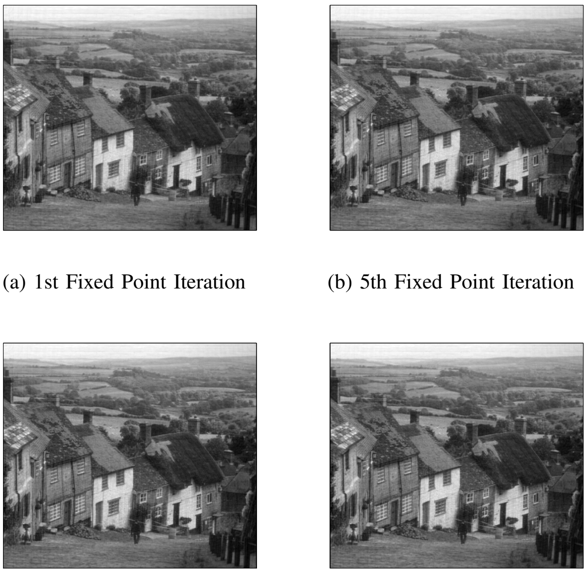
(d) Deblured Image P_S

(e) Deblured Image P_C



(f) Kernel

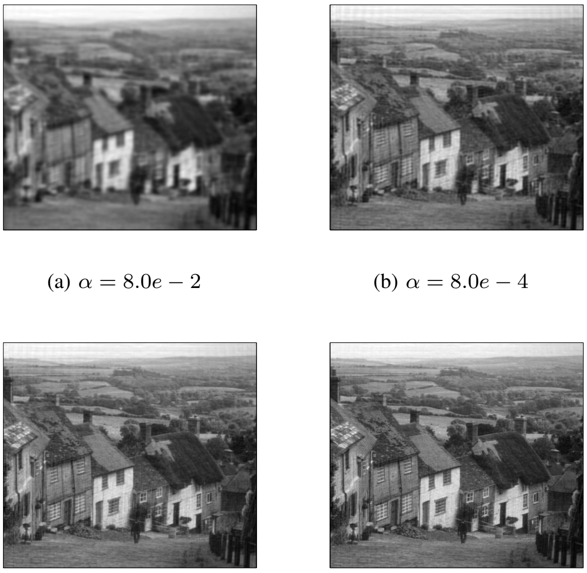
Fig. 3: Goldhill Image



(a) 1st Fixed Point Iteration (b) 5th Fixed Point Iteration

(c) 10th Fixed Point Iteration (d) 13th Fixed Point Iteration

Fig. 4: Fixed Point Iteration



(a) $\alpha = 8.0e - 2$ (b) $\alpha = 8.0e - 4$

(c) $\alpha = 8.0e - 7$ (d) $\alpha = 8.0e - 8$

Fig. 5: Regularization Parameter

ISBN: 978-988-19253-2-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)