

Fast wave-front reconstruction by solving the Sylvester equation with the alternating direction implicit method

Hongwu Ren and Richard Dekany

Caltech Optical Observatories,
Physics, Math and Astronomy Division,
California Institute of Technology, Pasadena, CA 91125

hren@astro.caltech.edu, rgd@astro.caltech.edu

Abstract: Large degree-of-freedom real-time adaptive optics (AO) control requires reconstruction algorithms that are computationally efficient and readily parallelized for hardware implementation. In particular, we find the wave-front reconstruction for the Hudgin and Fried geometry can be cast into a form of the well-known Sylvester equation using the Kronecker product properties of matrices. We derive the filters and inverse filtering formulas for wave-front reconstruction in two-dimensional (2-D) Discrete Cosine Transform (DCT) domain for these two geometries using the Hadamard product concept of matrices and the principle of separable variables. We introduce a recursive filtering (RF) method for the wave-front reconstruction on an annular aperture, in which, an imbedding step is used to convert an annular-aperture wave-front reconstruction into a square-aperture wave-front reconstruction, and then solving the Hudgin geometry problem on the square aperture. We apply the Alternating Direction Implicit (ADI) method to this imbedding step of the RF algorithm, to efficiently solve the annular-aperture wave-front reconstruction problem at cost of order of the number of degrees of freedom, $O(n)$. Moreover, the ADI method is better suited for parallel implementation and we describe a practical real-time implementation for AO systems of order 3,000 actuators.

©2004 Optical Society of America

OCIS codes: (010.1080) Adaptive Optics; (010.7350) Wave-front Sensing

References

1. R. Dekany, J. E. Nelson and B. Bauman, "Design considerations for CELT adaptive optics," in *Optical Design, Materials, Fabrication, and Maintenance*, P. Dierickx, ed., Proc. SPIE **4003**, 212-225 (2000)
2. R. H. Hudgin, "Wave-front reconstruction for compensated imaging," J. Opt. Soc. Am. **67**, 375-378 (1977)
3. D. L. Fried, "Least-square fitting a wave-front distortion estimate to an array of the phase difference measurements," J. Opt. Soc. Am. **67**, 370-375 (1977)
4. K. R. Freischlad and C. L. Koliopoulos, "Modal estimation of a wave front from difference measurements using the discrete Fourier transform," J. Opt. Soc. Am. A **3**, 1852-1861 (1986)
5. L. A. Poyneer, D.T. Gavel and J.M. Base, "Fast wave-front reconstruction in large adaptive optics systems using the Fourier transform," J. Opt. Soc. Am. A **19**, 2100-2111 (2002)
6. L. Gilles, C. R. Vogel, and B. L. Ellerbroek, "A multigrid preconditioned conjugate gradient method for large-scale wave-front reconstruction," J. Opt. Soc. Am. A **19**, 1817-1822 (2002)
7. L. Gilles, "Order-N sparse minimum-variance open-loop reconstructor for extreme adaptive optics," Opt. Lett. **28**, 1927-1929 (2003)
8. D. G. MacMartin, "Local, hierachic, and iterative reconstructors for adaptive optics," J. Opt. Am. A **20**, 1084-1093 (2003)

9. F. Shi, D. G. MacMartin, M. Troy, G. L. Brack, R. S. Burruss, and R. G. Dekany, "Sparse matrix wave-front reconstruction: simulations and experiments," in *Adaptive optical system technologies II*, P. Wizinowich, ed., Proc. SPIE **4839**, 1035-1044 (2002)
10. B. R. Hunt, "Matrix formulation for the reconstruction of phase values from phase differences," *J. Opt. Soc. Am.* **69**, 393-399 (1979)
11. R. A. Horn, C. R. Johnson, *Topics in matrix analysis*, (Cambridge University Press, New York, 1991)
12. H. Ren, R. Dekany, "wave-front reconstruction for extreme adaptive optics based on fast recursive filtering," submitted to *applied optics*
13. A. Lu and E. L. Wachspress, "Solution of Lyapunov equations by alternating direction implicit iteration," *Computers Math. Applic.* **21(9)**, 43-58 (1991)
14. R. H. Bartels and G. W. Stewart, "Solution of the matrix equation $AX + XB = C$," *Comm. of ACM* **15**, 820-826 (1972)
15. D. S. Watkins, *Fundamentals of Matrix Computations, second edition* (John Wiley & Sons, inc., New York, 2002)
16. Rao and P. Yip, *Discrete Cosine Transform, Algorithm, Advantages, Applications* (Academic Press, inc, San Diego, 1990)
17. R. G. Lane, A. Glindemann, and J.C. Dainty, "Simulation of a Kolmogorov phase screen," *Waves in Random Media* **2**, 209-224 (1992)

1. Introduction

Adaptive optics is an ideal technology developed for compensation of aberrations in optical systems or due to atmospheric turbulence to achieve unprecedented quality of astronomical observations and measurements. Wave-front reconstructors are an indispensable element of most AO systems. Whatever configuration used in these systems, such as extreme AO or multi-conjugate AO, fast and efficient algorithms allowing for hardware implementations are required to realize real time reconstruction and control for future giant telescope [1].

The most commonly used geometries for wave-front reconstruction are Hudgin and Fried geometries [2-3]. We will consider both geometries and develop a new wave-front reconstruction algorithm - the RF algorithm. Freischalad *et al* [4] have used the 2-D discrete Fourier transform (DFT) domain filtering method to perform wave-front estimation for various types of wave-front slope sampling geometry on a rectangular aperture, and Poyneer *et al* [5] have demonstrated further this algorithm using a curl-free boundary method to reduce the period boundary error, and derived a 2-D inverse filtering formula in the DFT domain for the Fried geometry. These algorithms scale as $O(n \log n)$ when implemented with fast Fourier transforms, where n is the degree of the freedom for the wave-front sensor. Gilles *et al* [6] have described an $O(n \log n)$ multigrid preconditioned conjugate gradient method, as well as an $O(n)$ sparse minimum-variance open-loop wave-front reconstructor based on multigrid preconditioning with a symmetrical Gauss-Seidel smoothing function [7]. MacMartin [8] has proposed hierarchic iterative wave-front reconstructors which scale linearly using a local influence function assumption. Shi *et al* have validated them experimentally at Palomar Observatory [9].

The fundamental algebraic structure of the wave-front reconstruction problem for the Hudgin geometry has been studied using a matrix representation and the least square principle by Hunt [10]. However, the reconstructor he developed is slow due to the particular iterative methods he adopted. By casting the reconstruction problem into the language of traditional image processing, we can create fast algorithms that have been specifically designed for astronomical adaptive optics systems, and overcome this limitation. In this theoretical context, we write the gradient matrix into a Kronecker product [11] form and use it with the least square wave-front reconstructor to derive a novel reconstructor formulated as the Sylvester Equation. We then derive the inverse filtering formulas in the 2-D DCT domain for the Hudgin and Fried geometries wavefront reconstruction using the spectral theorem for symmetrical real matrices [12, 15], the Hadamard product definition of matrices [11], and the principle of separation of variables. We have recently developed a RF algorithm [12]. In this algorithm, we exploited an imbedding step to change the annular-aperture wave-front reconstruction problem for the Fried geometry to a square-aperture wave-front reconstruction

problem for the Hudgin geometry, and then the square-aperture wave-front reconstruction problem was solved by the method of inverse filtering performed in the 2-D DCT domain and the multigrid method performed in the original image domain. The RF algorithm developed there scales as $O(n \log n)$ and $O(n)$ respectively when these two methods are used. In this investigation, we extend the RF algorithm by combining the square-aperture wave-front reconstructor based on the Sylvester equation with the RF algorithm, and then we solve the imbedding step in the RF algorithm using both ADI method [13] and exact Bartels-Stewart (BS) method [14]. We show that the ADI method is an $O(n)$ method, and compare the performance of both the ADI and the BS methods by simulating the wave-front reconstruction process for a large adaptive optics system using the Monte Carlo method. We will not consider the phase branch-point problem here, but it is a problem that needs to be solved when adaptive optics is used in the airborne laser and laser communication systems.

2. The Sylvester equation for wave-front reconstruction on a square aperture

We will solve the following wave-front slope measurement equation for a single-conjugate adaptive optics system [10],

$$s = P\phi + \eta, \quad (1)$$

where s is the wave-front phase difference vector, which is calculated from the centroid measurements in the subapertures of a Shack-Hartmann wave-front sensor (and interpreted according to the assuming sensor geometry), ϕ is the incident wave-front phase vector, η is a vector representing noise on the slopes measurement. The matrix P is the influence matrix representing the gradient operator that transforms phase into phase difference. For convenience, the vectors s , ϕ and η are arranged in a lexicographical order along the row direction of the corresponding 2-D images as described in Ref. [10]. The least squares estimation of Eq. (1) is equivalent to solving the following equation [10],

$$P^T P \phi = P^T s. \quad (2)$$

The matrix P can be expressed with two partitions,

$$P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}, \quad (3)$$

where P_1 and P_2 are sparse phase difference matrices along the x and y directions respectively. In this study, we concentrate mainly on the algebraic structure of the left-hand side of the Eq. (2). The right-hand side can be calculated directly using sparse matrix technique, though we will find it convenient to rewrite this as

$$r = P^T s = P_1^T s_x + P_2^T s_y, \quad (4)$$

where s_x and s_y are wave-front phase differences along the row and column directions respectively.

We will derive the Sylvester equation for wave-front reconstruction using Eq. (2) with the wave-front phase ϕ defined on a $N \times N$ square-aperture sampling grid as defined in Ref. [10], for two different sensor geometries, those introduced by Hudgin [2] and Fried [3].

2.1 The Hudgin geometry

Assuming we solve Eq. (1) for the Hudgin geometry, we can write P_1 and P_2 in Eq. (3) as the following form,

$$\begin{aligned} P_1 &= I \otimes D_1, \\ P_2 &= D_1 \otimes I, \end{aligned} \quad (5)$$

in which, \otimes is the Kronecker product [11]. For a matrix Y of size $m \times n$ and a matrix Z of size $j \times k$, $Y \otimes Z$ is defined as

$$Y \otimes Z = \begin{bmatrix} y_{11}Z & y_{12}Z & \cdots & y_{1n}Z \\ y_{21}Z & y_{22}Z & \cdots & y_{2n}Z \\ \vdots & \vdots & \ddots & \vdots \\ y_{m1}Z & y_{m2}Z & \cdots & y_{mn}Z \end{bmatrix}, \quad (6)$$

$Y \otimes Z$ is seen to be a matrix of size $mj \times nk$. In Eq. (5), I is an identity matrix of order N , and the matrix D_1 is a 1-D phase difference matrix of size $(N-1) \times N$ for the Hudgin geometry [10],

$$D_1 = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \\ & & & & 1 & -1 \end{bmatrix}. \quad (7)$$

From equations (3) and (5), we obtain

$$P^T P = (I \otimes D_1)^T (I \otimes D_1) + (D_1 \otimes I)^T (D_1 \otimes I). \quad (8)$$

Using the transpose property of the Kronecker product [11],

$$(Y \otimes Z)^T = Y^T \otimes Z^T, \quad (9)$$

and its mixed product property [11],

$$(Y \otimes Z)(W \otimes X) = (YW) \otimes (ZX), \quad (10)$$

we can rewrite Eq. (8) as,

$$P^T P = I \otimes A + A \otimes I, \quad (11)$$

where $A = D_1^T D_1$, is a matrix of size $N \times N$ having the following form [10],

$$A = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}. \quad (12)$$

This matrix is a tridiagonal matrix obtained from the 1-D central second differences. From Eq. (2) and (4), we immediately find that the least squares solution of the wave-front estimation Eq. (1) is equivalent to solution of the following equation in Kronecker product form,

$$(I \otimes A + A \otimes I)\phi = r. \quad (13)$$

For compactness, we now define the *vec* operator as [11]

$$\text{vec}A = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix}, \quad (14)$$

which creates a large column vector from matrix A by stacking together all of its column vectors. The *vec* operator therefore reorders an image in a lexicographical order along the

column direction. Because we previously defined ϕ and r to be vectors formed by stacking the represented images in a lexicographical order along their row direction, therefore from Eq. (13), we obtain the following representation of Eq. (2),

$$(I \otimes A + A \otimes I) \text{vec}(\Phi^T) = \text{vec}(R^T), \quad (15)$$

where Φ and R are 2-D matrices of size $N \times N$ representing the solution and residue images respectively, and with the image matrix row and column index defined as in Ref. [10]. From Eq. (15), using the symmetry properties of A and the vec operator property of the Kronecker product [11] defined as

$$\text{vec}(AYB) = (B^T \otimes A) \text{vec}Y, \quad (16)$$

we finally obtain the following Sylvester equation for the wave-front reconstruction problem assuming a Hudgin geometry,

$$A \Phi + \Phi A = R. \quad (17)$$

This form of equation is also sometimes referred to as a Lyapunov equation.

2.2 The Fried geometry

We use a similar method as above to derive the Sylvester equation for wave-front reconstruction on an $N \times N$ square-aperture sampling grid for the Fried sensor geometry. However, to account for the character of the Fried geometry, we introduce a 1-D interpolation matrix F of size $(N-1) \times N$ as follows,

$$F = \begin{bmatrix} 0.5 & 0.5 & & & & & \\ & 0.5 & 0.5 & & & & \\ & & \ddots & \ddots & & & \\ & & & 0.5 & 0.5 & & \\ & & & & 0.5 & 0.5 & \\ & & & & & 0.5 & 0.5 \end{bmatrix}. \quad (18)$$

We use the Kronecker product technique along the row and column directions for the Fried geometry, and then we find that P_1 and P_2 in Eq. (3) have the following forms,

$$\begin{aligned} P_1 &= F \otimes D_1, \\ P_2 &= D_1 \otimes F, \end{aligned} \quad (19)$$

where matrix D_1 is the same $(N-1) \times N$ matrix as in Eq. (7) for the Hudgin geometry. The matrices P_1 and P_2 are sparse phase difference matrices along the row and column directions respectively for the Fried geometry. From Eq. (3) and (19), we obtain

$$P^T P = (F \otimes D_1)^T (F \otimes D_1) + (D_1 \otimes F)^T (D_1 \otimes F). \quad (20)$$

Once again, using the transpose property and mixed product property of Kronecker product as in Eq. (9) and (10), we obtain

$$P^T P = H \otimes A + A \otimes H, \quad (21)$$

in which, $H = F^T F$ is a matrix of size $N \times N$, having the following form

$$H = \frac{1}{4} \begin{bmatrix} 1 & 1 & & & & & \\ 1 & 2 & 1 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & & 1 & 2 & 1 & \\ & & & & 1 & 1 & \end{bmatrix}, \quad (22)$$

and the matrix A takes the same form as in Eq. (12). From Eq. (2) and (4), we immediately find the following Kronecker product form of the wave-front estimation equation,

$$(H \otimes A + A \otimes H)\phi = r. \quad (23)$$

Using the *vec* operator definition in Eq. (14), and the procedures described in the previous subsection, we obtain the generalized Sylvester equation for the Fried-geometry wave-front reconstruction as

$$A \Phi H + H \Phi A = R, \quad (24)$$

in which, Φ and R are the matrices representing solution and residue images as described in previous subsection for those in Eq. (17).

3. Solution of the Sylvester equation for wave-front reconstruction in 2-D DCT domain

Several computationally efficient techniques for solving Eq. (17) and (24) are available. In this section, we derive the Hudgin and Fried Filters formulas and the filtering relationship in 2-D DCT domain for the wave-front reconstruction on a square aperture.

3.1 DCT and the Hudgin filter for the Hudgin geometry

An efficient fast transform method for solving the Sylvester Equation (17), corresponding to the Hudgin geometry, can be derived by exploiting the spectral theorem for real symmetrical matrices [12, 15], and the fact that matrix A in Eq. (12) is real symmetric. Accordingly, we know that A can be diagonalized by an orthonormal real matrix M of size $N \times N$, such that,

$$A = M \Lambda_A M^T. \quad (25)$$

Substituting Eq. (25) into (17) and using the definition of an orthonormal real matrix, namely that $M^{-1} = M^T$ or $MM^T = M^T M = I$, we obtain

$$\Lambda_A M^T \Phi M + M^T \Phi M \Lambda_A = M^T R M. \quad (26)$$

In the real orthonormal matrix M , every column is a real orthonormal eigenvector (basis) of matrix A , and matrix Λ_A is a real eigenvalue matrix whose every diagonal value is a real eigenvalue of matrix A , we call it the spectrum matrix (or spectrum) of matrix A .

On the right-hand side of Eq. (26), right multiplication of R with a transform matrix M is equivalent to transforming it with respect to its rows; left multiplication of R with a transposed transform matrix M^T is equivalent to transforming it with respect to its columns. According to the property of the separable transform [16], the right-hand side is just a 2-D orthonormal transform of R , but implemented by applying two 1-D orthonormal transforms to its rows and columns sequentially. Applying the same argument to the left-hand side of Eq. (26), we know $M^T \Phi M$ is also equivalent to a 2-D orthonormal transform of the wave-front image Φ into a spectral image in this transform domain. The left multiplication of $M^T \Phi M$ by Λ_A is equivalent to multiplying every row of the 2-D spectrum with the eigenvalue in the corresponding row of matrix Λ_A . The right multiplication of $M^T \Phi M$ by Λ_A is equivalent to multiplying every column of the 2-D spectrum with the eigenvalue in the corresponding column of matrix Λ_A , so we can write Eq. (26) as,

$$(\lambda u^T + u \lambda^T) \circ (M^T \Phi M) = M^T R M, \quad (27)$$

where $u = [1, 1, \dots, 1]^T$, is a column vector of size N full of ones in it; and $\lambda = \text{diag}(\Lambda_A)$, is a column vector of size N , obtained from the diagonal of the eigenvalue matrix; and the symbol \circ represents the Hadamard product [11], which is an entrywise product defined as $(Y \circ Z)_{ij} =$

$Y_{ij} * Z_{ij}$ for matrix Y and Z . We can see from Eq. (27) that its left-hand side is an entrywise multiplication of two matrices. We can solve for Φ by first dividing the spectral image $M^T R M$ by the filter image $\lambda u^T + u \lambda^T$ entrywisely, and then transform the result back into the original domain by an inverse 2-D orthonormal transform. The solution process is represented by the following equation,

$$\Phi = M \left(\frac{M^T R M}{\lambda u^T + u \lambda^T} \right) M^T = M \left(\frac{M^T R M}{T^H} \right) M^T. \quad (28)$$

We refer to $(T^H)^{-1}$ as the Hudgin filter in a 2-D orthonormal transform domain. In order to solve equation (28), we need to know λ and M . From Ref. [16], matrix A in Eq. (12) is the inverse of the covariance matrix of the Markov-1 signal when the adjacent correlation coefficient ρ approaching to 1. However, from physical interpretation, matrix A is also the Laplacian matrix corresponding to the 1-D central second differences, therefore from these two perspectives, we know that matrix A can be diagonalized by the 1-D DCT matrix M , whose elements are defined as

$$M_{mn} = \sqrt{\frac{2}{N}} \kappa_m \cos \left[\frac{m(2n+1)\pi}{2N} \right], \quad (m, n = 0, 1, \dots, N-1), \quad (29)$$

in which, $\kappa_m = 1/\sqrt{2}$ for $m = 0$, otherwise $\kappa_m = 1$. This transform matrix is an orthonormal matrix, in which every column vector is an eigenvector of matrix A , the eigenvalues correspondingly are defined as

$$\lambda_m = 4 \sin^2 \left(\frac{m\pi}{2N} \right), \quad (m = 0, 1, \dots, N-1), \quad (30)$$

for the eigenvalue column vector λ . From the above equation, the filter, $\lambda u^T + u \lambda^T$, can be rewritten as,

$$T_{mn}^H = 4 \left[\sin^2 \left(\frac{m\pi}{2N} \right) + \sin^2 \left(\frac{n\pi}{2N} \right) \right], \quad (m, n = 0, 1, \dots, N-1). \quad (31)$$

The 2-D Hudgin filter $(T^H)^{-1}$ is obtained using an element-by-element inverse of the filter image T_{mn}^H in the above equation. Because T_{00}^H corresponds to the piston mode, we suppress this singular value by assigning a zero value to the corresponding zero frequency component of the 2-D Hudgin filter.

The wave-front reconstruction Eq. (28), describes the following procedure:

- 2-D DCT the right-hand-side image R to obtain the spectrum image
- Multiply the spectrum image with the (pre-computed) 2-D Hudgin filter
- Recover the original wave-front by another inverse 2-D DCT transform.

The 2-D DCT transform and its inverse can be implemented using fast methods described in Ref. [16]. According to the principle of separation of variables, we can reduce the 2-D DCT transform to two 1-D DCT transforms, which means that the 2-D DCT transform can be implemented using 1-D FFT with computation cost scales as $O(n \log n)$ when exploiting the relationship between DCT and DFT described in Ref. [16].

3.2 DCT and the Fried filter for the Fried geometry

For the Fried geometry, there is no exact transform filtering relationship as that obtained for the Hudgin geometry above. Deriving the filtering relationship for Fried-geometry wave-front reconstruction from Eq. (24) is more complex than that for the Hudgin geometry. Because the matrices A and H don't satisfy the commutative relationship $AH = HA$, they can not be diagonalized by the same orthonormal transform matrix. Therefore, we can not find an exact 2-D filtering relationship for the Fried geometry, but an approximate one from the properties of matrix H . Matrix H can be exactly diagonalized by the following matrix S , whose elements are defined as

$$S_{mn} = \sqrt{\frac{2}{N}} \kappa_m \sin \left[\frac{(m+1)(2n+1)\pi}{2N} \right], \quad (m, n = 0, 1, \dots, N-1), \quad (32)$$

where $\kappa_m = 1/\sqrt{2}$ for $m = N-1$, otherwise $\kappa_m = 1$. This orthonormal transform matrix is the 1-D Discrete Sine Transform (DST) matrix [16], while the elements of the eigenvalue vector σ for matrix H are

$$\sigma_m = \cos^2 \left[\frac{(m+1)\pi}{2N} \right], \quad (m = 0, 1, \dots, N-1). \quad (33)$$

The above DST matrix and the DCT matrix in Eq. (29) are not commutative, in order to find a filtering relationship as in Eq. (28), we need to compromise the requirement of exact commutative relationship as above. Fortunately, the matrix H can be approximately diagonalized by the 1-D DCT matrix M , which means that we can obtain an approximate filtering relationship for the Fried geometry. The elements of the approximate eigenvalue vector corresponding to 1-D DCT transform matrix M are,

$$\tau_m = \cos^2 \left(\frac{m\pi}{2N} \right), \quad (m = 0, 1, \dots, N-1). \quad (34)$$

Therefore we can approximately write H as

$$H = M \Lambda_H M^T. \quad (35)$$

Substituting Eq. (25) and (35) into (24), using the same procedures as described in the above subsection for the Hudgin geometry, we can obtain the filtering relationship in the 2-D DCT domain for Fried-geometry wave-front reconstruction,

$$\Phi = M \left(\frac{M^T R M}{\lambda \tau^T + \tau \lambda^T} \right) M^T. \quad (36)$$

In 2-D DCT domain, the denominator of the above equation $\lambda \tau^T + \tau \lambda^T$ can be rewritten as

$$T_{mn}^F = 4 \left[\sin^2 \left(\frac{m\pi}{2N} \right) \cos^2 \left(\frac{n\pi}{2N} \right) + \sin^2 \left(\frac{n\pi}{2N} \right) \cos^2 \left(\frac{m\pi}{2N} \right) \right], \quad (37)$$

$(m, n = 0, 1, \dots, N-1).$

The 2-D Fried filter is obtained using an element by element inverse of the image T_{mn}^F in the above equation. Because T_{00}^F and $T_{N-1, N-1}^F$ correspond to the piston and waffle modes respectively, we suppress these singular values by assigning zero values to the corresponding grid points in the 2-D Fried filter.

The wave-front reconstruction for the Fried geometry can be performed using the same procedures described at the end of the last subsection for the Hudgin geometry. The 2-D DCT

inverse filtering can also be combined with a regularization scheme to take into account the prior covariance information of the atmospheric turbulence and the noise of the wave-front slopes measurement. In another work of ours [12], this regularization scheme was discussed and the cost of this filtering method was shown scales as $O(n \log n)$. We will next develop a faster method to solve the Sylvester Equation (17) and (24) for wave-front reconstruction on an annular aperture using the RF algorithm found recently by us.

4. RF algorithm

Most telescope entrance pupils take on an annular shape. Thus, we must in general find ways to solve the wave-front reconstruction problem on an annular aperture. The strategy we adopt here is to embed the desired annular aperture in a square aperture to allow application of standard fast methods for solution of the Sylvester Equation (17) and (24).

The RF algorithm for open-loop wavefront reconstruction is described in the flowchart shown in Fig. 1. In this flowchart, x_0 and o_n are the initializing solution and null vectors of size n respectively; G is the gradient matrix generated on an annular aperture using the corresponding annular versions of Eq. (3) and (19) for the Fried geometry, or those of Eq. (3) and (5) for the Hudgin geometry; A is the matrix shown in Eq. (12) and it is multiplied with the solution image Φ_k on a square aperture for the Hudgin geometry; the matrix W is an annular mask image which multiplies entrywisely with the solution image Φ_k to transform it back into the annular aperture; the symbol $\| \cdot \|$ represents the ℓ_2 norm operator; ε is the relative error and k_{\max} is the maximum iteration number, specified for the convergence. In the imbedding step, the Sylvester equation from Eq. (17) is used for the Hudgin-geometry wave-front reconstruction on a square aperture. For convenience of discussions, we repeat it here,

$$A\Phi_k + \Phi_k A = R_k . \quad (38)$$

This imbedding step in the RF algorithm also serves to accelerate the convergence rate for the iterative solution process. In this step, first convert the residue vector r_k to a residue image R_k ; and then using the following ADI method to obtain the solution update image Φ_k from the above equation; and then convert it back to a vector ϕ_k . In the iterative solution process, we can directly update the solution vector x_k using ϕ_{k-1} or update it with α_k and vector p_k using the conjugate gradient (CG) procedure. However, for the wavefront reconstructor test purpose, we update the new residue vector using b_k and the current solution estimation x_k in this study,

$$r_k = b_k - Cx_k , \quad (39)$$

where $b_k = G^T s_k$, and s_k is the current phase slope vector; and $C = G^T G$ is a matrix acting on an annular aperture; and x_k is the current solution estimation. This residue update step is different from the CG method used elsewhere [6-7]. For closed-loop wavefront reconstruction, we should use $r_k = b_k$ instead of Eq. (39) in the RF algorithm, and s_k is generated using the optical phase output after phase compensation. The new imbedding strategy permits the fast methods designed for the Hudgin-geometry wave-front reconstruction on a square aperture to be used with the Fried-geometry wave-front reconstruction problem on an annular aperture.

5. ADI method for the RF algorithm

5.1 ADI method for the imbedding step

If matrix A is a nonsingular matrix, the Sylvester Equation (38) can usually be exactly solved using the BS method [14], which uses the Householder and QR transformations [15]. The BS method requires matrix A in Eq. (38) to be a nonsingular matrix, therefore before using it, we need to regularize the singular matrix A using an identity matrix scaled down with a small coefficient, taking into account of the signal to noise ratio (SNR) in the slopes measurement.

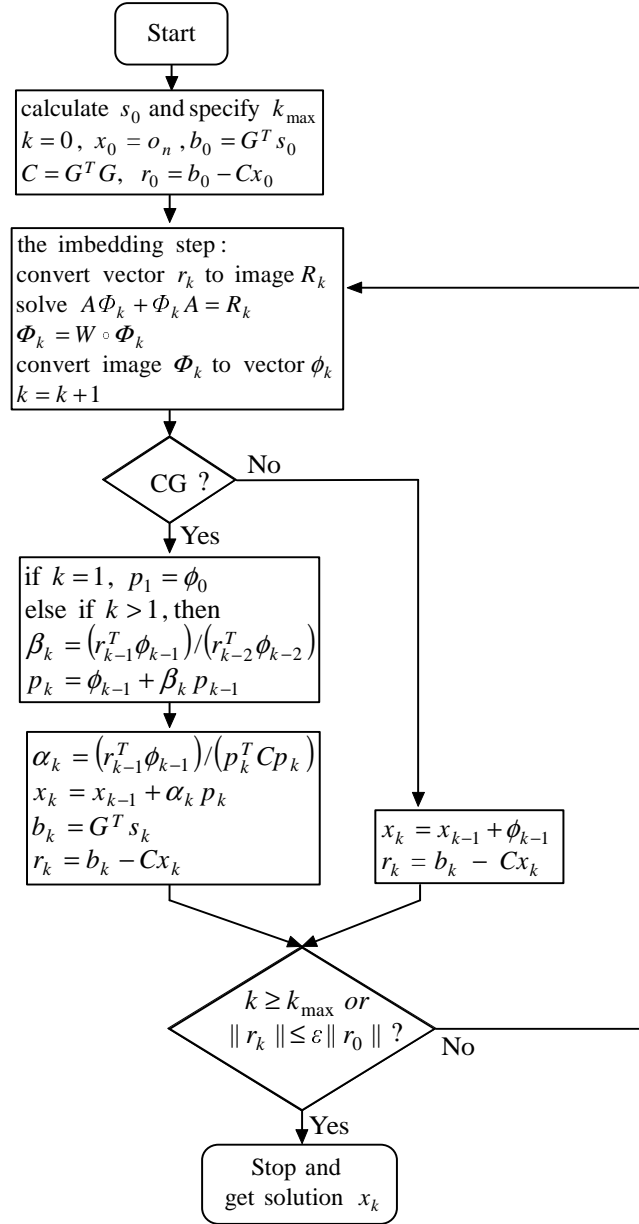


Fig. 1. Flowchart for the RF in which the preconditioning solution step is solved through the Sylvester equation to accelerate the convergence rate of the iterative process.

This regularization procedure changes the condition of the matrix A into a nonsingular matrix. The computation cost of the BS method scales as $O(n^{3/2})$. Another efficient method for solving the Sylvester Equation (38) is the ADI method [13]. We use this method for wave-front method when applied to Eq. (38) is described by the following two iterative equations,

$$(A + \rho_j I) \Phi_k^{j-1/2} = R_k - \Phi_k^{j-1} (A - \rho_j I), \quad (40)$$

$$\Phi_k^j (A + \rho_j I) = R_k - (A - \rho_j I) \Phi_k^{j-1/2}. \quad (41)$$

In these two equations, ρ_j is an optimal parameter. To accelerate the convergence rate of the iterative solution process, it is adjusted to a different value for each iteration step j , where $j = 1, 2, \dots, J$, where J is the total ADI iteration number. Because the iterative ADI method is used in the imbedding step of the RF algorithm, which is also an iterative process, therefore two iterative loops are involved in the RF algorithm. To solve the above two iterative ADI equations, we need to first determine the optimum parameter ρ_j and the total iteration number J , then solve the tridiagonal systems of Eq. (40) and (41) for every iteration step j iteratively until the total iteration number J is reached. The parameter ρ_j and J are determined by solving the ADI minimax problem [13], which is discussed in next subsection. The tridiagonal equation solver is addressed in Subsection 5.3.

5.2 The ADI minimax problem

We use similar numerical technique exploited in Ref. [13] to determine the optimum parameter ρ_j and the total iteration number J for the ADI method. The procedures are described as follows. From Eq. (30), we can see that the spectrum of matrix A is real positive when excluding its singular spectrum ($m = 0$), therefore the α parameter in the definition of elliptic function [13] is zero for the case here, and only limited field of the positive real axis [ϖ_a, ϖ_b] is required to consider for the ADI minimax problem. For the finite number of spectrum modes from $m = 1$ to $N-1$, we define the minimum and maximum spectrum bounds ϖ_a and ϖ_b as follows,

$$\begin{aligned} \varpi_a &= 4 \sin^2 \left(\frac{\pi}{2N} \right), \\ \varpi_b &= 4 \sin^2 \left[\frac{(N-1)\pi}{2N} \right]. \end{aligned} \quad (42)$$

The ADI parameters ρ_j and J are found using the following procedures. First defining a parameter h as,

$$h = \frac{1}{2} \left(\frac{\varpi_a}{\varpi_b} + \frac{\varpi_b}{\varpi_a} \right), \quad (43)$$

and then because $h \geq 1$, defining

$$\begin{aligned} g' &= \frac{1}{h + \sqrt{h^2 - 1}}, \\ g &= \sqrt{1 - g'^2}. \end{aligned} \quad (44)$$

The approximation for the total ADI iteration number is given by

$$J = \frac{\ln \frac{4}{\epsilon} \ln \frac{4}{g'}}{\pi^2}. \quad (45)$$

The optimal parameter ρ_j is given by

$$\rho_j = \sqrt{\frac{\varpi_a \varpi_b}{g'}} \operatorname{dn}(v_j, g), \quad (j = 1, 2, \dots, J), \quad (46)$$

where $\text{dn}(v_j, g)$ is defined as

$$\begin{aligned} \text{dn}(v_j, g) &= \frac{2q^{v_j/2}(1+q^{1-v_j}+q^{1+v_j})}{(1+2q)(1+q)}, \quad (v_j \geq 0.5), \\ \text{dn}(v_j, g) &= \frac{g'}{\text{dn}(v_{J-j+1}, g)}, \quad (v_j < 0.5), \end{aligned} \quad (47)$$

where v_j and q are defined as,

$$\begin{aligned} v_j &= \frac{2j-1}{2J}, \\ q &= \left(\frac{g'}{4}\right)^2 \left(1 + \frac{g'^2}{2}\right). \end{aligned} \quad (48)$$

5.3 The tridiagonal solver

To solve Eq. (40), we separate its right-hand side residue matrix and its left-hand side solution matrix into N columns respectively. In this way, the problem is changed to solving N independent tridiagonal equations of the following form,

$$Qx = d, \quad (49)$$

where Q is a tridiagonal matrix of order N , shown as follows,

$$Q = \begin{bmatrix} a_1 & f_1 & & & \\ c_2 & a_2 & f_2 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{N-1} & a_{N-1} & f_{N-1} \\ & & & c_N & a_N \end{bmatrix}. \quad (50)$$

The tridiagonal matrix Q can be decomposed as $Q = LU$ with the LU decomposition method, where L is a lower bidiagonal matrix and U is an upper bidiagonal matrix shown as in the following two equations,

$$L = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & \ddots & \ddots & & \\ & & l_{N-1} & 1 & \\ & & & l_N & 1 \end{bmatrix}, \quad (51)$$

$$U = \begin{bmatrix} u_1 & f_1 & & & \\ & u_2 & f_2 & & \\ & & \ddots & \ddots & \\ & & & u_{N-1} & f_{N-1} \\ & & & & u_N \end{bmatrix}. \quad (52)$$

The unknown value for l_i and u_i is computed using the following iteration equations,

$$\begin{aligned} u_1 &= a_1, \\ l_i &= c_i/u_i, \\ u_i &= a_i - l_i f_{i-1}, \\ (i &= 2, 3, \dots, N). \end{aligned} \quad (53)$$

Let $y = Ux$, we can solve $Ly = d$ first using the following forward substitution equations,

$$\begin{aligned} y_1 &= d_1, \\ y_i &= d_i - l_i y_{i-1}, \\ (i &= 2, 3, \dots, N), \end{aligned} \quad (54)$$

and then solving x using the following backward substitution equations ,

$$\begin{aligned} x_N &= y_N / u_N, \\ x_i &= (y_i - x_{i+1} f_i) / u_i, \\ (i &= N-1, N-2, \dots, 1). \end{aligned} \quad (55)$$

After solving this tridiagonal equation for multiple right-hand side columns, we obtain the intermediate solution image by assembling all the solution columns together. Eq. (41) is solved using the same technique after transpose operations on its two sides are performed.

5.4 Computation complexity of the ADI method for the RF algorithm

We analyze the computation complexity of solving Φ_k using the ADI iterative equations (40) and (41). We need to solve Eq. (40) first. On the right-hand sides of Eq. (40), because the matrix $A - \rho_j I$ is a tridiagonal sparse matrix, $3n$ multiplications and n additions are required for every iteration j , where $n = N^2$, is the total number of the sampling grid for the solution image. We decompose the matrix equation (40) into multiple tridiagonal equations. For one of right-hand side vectors, and one of the solution vectors, Eq. (40) is reduced to the form of Eq. (49). Therefore its computation complexity can be analyzed using Eq. (49). Because the upper diagonal and lower diagonal elements of matrix $A + \rho_j I$ are all value 1, and correspond to the f and c elements in Eq. (50) respectively, therefore we save N multiplication in the substitution process shown in Eq. (53) and (55). If we precompute the l_i and u_i value, then only $2N$ multiplications and $2N$ additions are required to solve Eq. (54) and (55) for this reduced tridiagonal equations, but we have N independent right hand-side column vectors, so totally we need only $2n$ multiplications and $2n$ additions to solve Eq. (40). Next we need to perform a transpose operation on two sides of Eq. (41), and then using the same procedure used for Eq. (40), we can solve N independent systems of tridiagonal equations. Thus in all, we need $10n$ multiplications and $6n$ additions and some overhead computation for the matrix transpose for every ADI iteration step j . The computation cost to complete the ADI iterations scales as $10Jn$, where J is the total number of the ADI iterations in the imbedding step. Because it takes approximately $15n$ multiplications to update the solution when using a CG procedure and to calculate the residue vector, and about $5n$ to do that when not using a CG procedure for every recursive iteration step k , so the total operation complexity for the recursive algorithm is about $10JKn + 16Kn$, and $10JKn + 6Kn$ when also including the masking operation using the mask W . This computation complexity estimation neither considers the overhead computation resulting from the matrix transpose operation nor the vector to image conversion and vice versa. If considering these factors, the computation complexity should be larger than this estimation. But for single processor, the transpose and conversion operation are memory access processes, the computation time for these operations is determined by the speed of memory access processes. The transpose operation permits solving the tridiagonal equation along the row direction of the right-hand side image sequentially after solving along its column direction. In Eq. (40) and (41), if we separate the right-hand side residue matrix and its left-hand side solution matrix into N columns respectively and then solve N independent tridiagonal equations for these columns, so the solution speed is can be accelerate N times if we use N processors at the same time. If the wavefront reconstruction is conducted in this way, we could use a one-dimensional mask for the masking step in the imbedding procedure, so the vector to matrix conversion and vice versa is not required, because now Φ_k is just ϕ_k , and in the solution and residue update procedure, the vectors are reduced to small vectors corresponding to one column of the whole image rather to the whole image, therefore the

operation count complexity of the RF algorithm can reduce to $O(n^{1/2})$ when the ADI method is used in its imbedding step. However, the complexity analysis is complicated by the transpose operations involved in the ADI iterative solution process, because the time for matrix transpose operation is determined by the memory access patterns for a single processor computer, and also by the communication speed between the multiprocessors for multiprocessors computer. Therefore, the final effective computation complexity should be between $O(n^{1/2})$ and $O(n)$ depending on what kind of computer architectures is used.

6. Performance metrics

We introduce the residue phase error (RPE) and the relative total root mean squared (RMS) error as the performance metrics to quantify the performance of various wave-front reconstruction algorithms. The definitions are the same as that in Ref. [12] for convenience of comparing with the results there. For every iteration step of the RF algorithm, the RPE is defined as

$$e_r = w_0 \cdot * [e - (v_1^T e) v_1], \quad (56)$$

where w_0 is the annular pupil plane masking vector converted from the annular pupil mask, v_1 is the piston mode vector, e is the phase error vector for the estimation defined as $e = \hat{x} - x$, where \hat{x} is the reconstructed wave-front phase for every temporal step and x is the original input wave-front phase. The RMS and total RMS error are defined using RPE defined in Eq. (56) as shown in the following two equations,

$$\text{RMS} = (e_r \cdot * e_r)^{1/2}, \quad (57)$$

$$\text{TRMS} = \left[\frac{1}{n} e_r^T e_r \right]^{1/2}, \quad (58)$$

where the symbol $\cdot *$ represents the component-wise product of two vectors, and n is the total number of sub-apertures on the annular solution aperture. From TRMS, the relative (e.g. fractional) TRMS error is defined as

$$\text{rTRMS} = \left[\frac{e_r^T e_r}{x^T x} \right]^{1/2}. \quad (59)$$

As defined in Eq. (56), RPE is a vector, but it is typically converted into an image when used as a performance metric. When we study the convergence rate of the RF algorithms, we will use rTRMS as the performance metric.

7. Simulations and discussions

Performance of our wavefront reconstruction algorithm, when applied in the single-conjugate AO system in an optical telescope, can be modeled using Monte Carlo simulations. The original wave-front was generated using the subharmonics method¹⁷ and considering the von Karman power spectrum of the optical turbulence. The wave-front phase difference vector corresponding to the centroid measurements of the Shack-Hartmann wave-front sensor is generated using annular-aperture version of Eq. (1), (3) and (19) as described in Section 4 for the Fried geometry. The Gaussian noise is added to the simulated wave-front slope to produce varying levels of subaperture slope SNR. We will apply the BS method to the regularized Sylvester equation, and then the ADI method to the non-regularized Sylvester equation in the imbedding step of the RF algorithm, and then will compare their performance using both the RPE and the rTRMS performance metrics. For wave-front reconstruction simulations, the

vector b_k in Eq. (39) for every iteration step of the RF algorithm is only required to be fixed to the original constant vector b_0 , which is equal to $G^T s_0$.

Numerical simulations were performed for a telescope having a 17 meters diameter annular aperture. The Shack-Hartmann wave-front sensor adopted has subaperture diameter of $r_0/3$, when mapped back onto the telescope entrance pupil, where r_0 is the Fried coherence length, and takes 0.2 meter in our simulations. This results in an annular pupil with 48816 sub-apertures. We embed this annular aperture in the square-aperture sampling grid of dimension 255×255 for all the simulations. The typical phase screen generated for these simulations is shown in Fig. 2. The piston mode is already removed from this phase screen, so it can act as the initial RPE image for the iterative solution process.

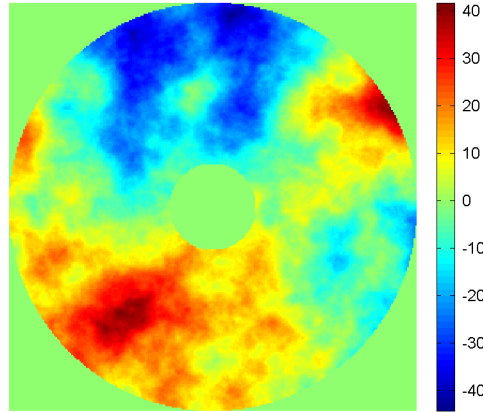


Fig. 2. Annular wave-front phase screen embedded in a 255×255 size square-aperture sampling grid. The colormap is shown in radians.

To use the ADI method in the RF algorithm, the ADI minimax problem needs to be solved first. In our simulations, the expected rTRMS error lower bound ε is taken to be 5×10^{-4} and $N = 255$, using the ADI minimax procedures described in Subsection 5.2, the J parameter we obtained is 11, and the ρ_j we obtained are shown in table 1.

Table 1. The ρ_j parameter for the ADI method when $\varepsilon = 5 \times 10^{-4}$ and $N = 255$

j	1	2	3	4	5	6	7	8	9	10	11
ρ_j	3.50	1.58	5.74×10^{-1}	2.02×10^{-1}	7.05×10^{-2}	2.46×10^{-2}	8.61×10^{-3}	3.01×10^{-3}	1.06×10^{-3}	3.83×10^{-4}	1.73×10^{-4}

For the convenience of discussion, we use the notation RFCG and RFD respectively for the RF algorithm with a CG procedure and the RF algorithm used directly without a CG procedure. The convergence rate curve of the rTRMS error for RFCG algorithm, using the BS and the ADI methods to solve the Sylvester equation in the imbedding step and when SNR of the wave-front slope are 2, 8, 32, 128, respectively, are shown together in Fig. 3. From it, we can see that the convergence rates of the rTRMS error defined as in Eq. (59) are similar for these two methods. After a few iterations the curves converge. The RPE vector define as in Eq. (56) for the 50th step are converted to images and shown in Fig. 4, for wave-front reconstruction using RFCG algorithm, and with the imbedding preconditioning step done by the ADI method, for the case of subaperture slope SNR equal to 2, 8, 32 and 128. The

corresponding results for the RPE and rTRMS error performance when the RFD algorithm is used are shown in Fig. 5 and 6 under the same conditions.

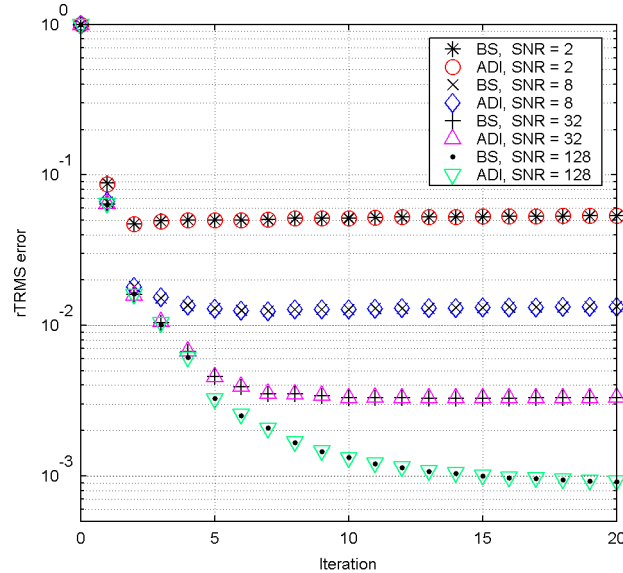


Fig. 3. Comparison of the rTRMS performance of the RFCG algorithm for the Fried geometry, when the imbedding step is solved using the ADI and BS methods respectively. The subaperture slope SNR is equal to 2, 8, 32 and 128, respectively, for both methods from top curves to bottom ones.

We can see from Fig. 3 and 5 that the RFCG algorithm converges fast for all SNR cases, but the RFD algorithm converges fast only for the cases of SNR less than 30. It starts to slow down when SNR is larger than 30 in the simulation conditions that has just been described. From Fig. 4 and 6 we can see that the RPE performance after 50 iterations is similar for both the RFCG and the RFD algorithms when SNR less than 30. The RPE performance for the RFD algorithm becomes inferior to the RFCG method around the boundary region with SNR equal to 128, so we infer the slow convergence rate in Fig. 5 as due to the error introduced around the boundary of the aperture. From Fig. 4 and 6, we also can see the minimum and the maximum limit values of the RPE images decrease when SNR increases.

From our experience, J is usually a number smaller than 15 for the case of $N \leq 1024$, and $\varepsilon \geq 5 \times 10^{-4}$, which is the rTRMS range that can be achieved when $\text{SNR} \leq 200$ for the slopes measurements. Thus when we solve the Sylvester equation in the imbedding step using the ADI method, the RFCG algorithm has a computation cost of less than $166Kn$ plus that of both the matrix transposition and the vector-image inter-conversion operations. The RFD algorithm has a computational cost of less than $156Kn$ for this situation. Therefore the RFCG algorithm should be a competitive algorithm for real time implementations in future extreme AO and multi-conjugate AO systems and the RFD algorithm should also perform well when SNR for the subaperture slopes is low.

We have also completed simulations for wavefront reconstruction when applying the Hudgin filtering in the 2-D DCT domain, as discussed in Subsection 3.1, to the imbedding step of the RF algorithm. The results obtained are similar to those shown in Fig. 3 and 4 when the same simulation conditions are used, so are not included here. Additional results using a regularization scheme for the wave-front reconstruction are presented in another publication [12].

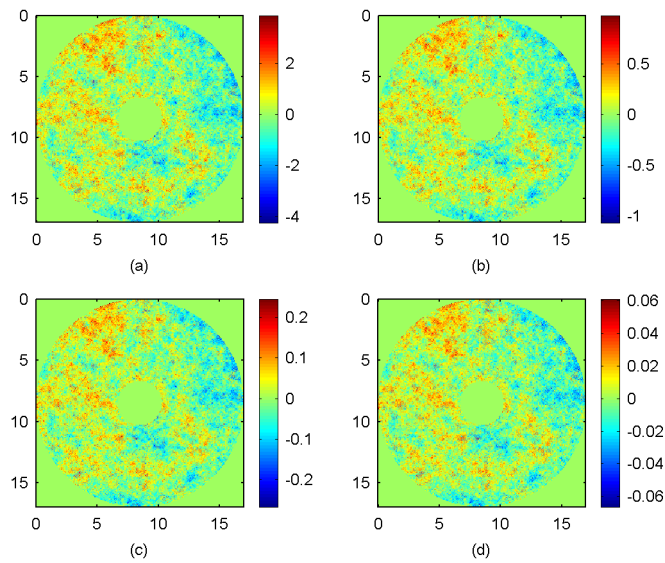


Fig. 4. RPE (in radians) after 50 iterations when solving the wave-front reconstruction for the Fried geometry using the RFCG algorithm, and the imbedding step is solved using the ADI method. The wave-front reconstruction is done in a 255x255 sized sampling grids and the subaperture slope SNR is equal to 2, 8, 32 and 128 for the RPE in image (a), (b), (c) and (d), respectively.

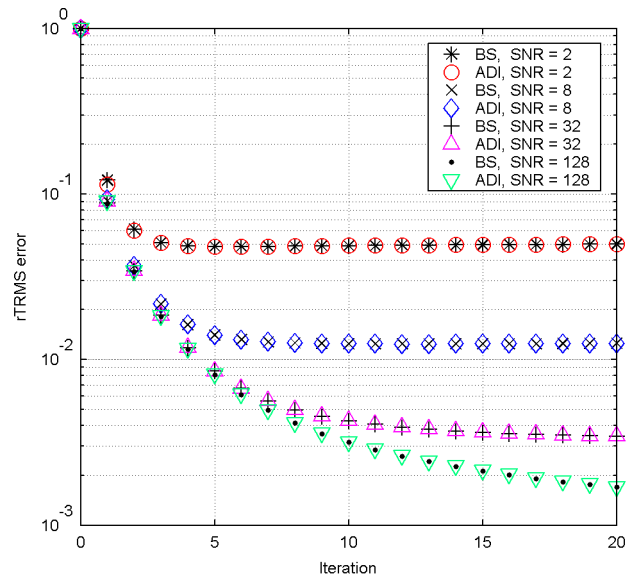


Fig. 5. Comparison of the rTRMS performance of the RFD algorithm for the Fried geometry, when the imbedding step is solved using the ADI and BS methods respectively. The subaperture slope SNR is equal to 2, 8, 32 and 128, respectively, for both methods from top curves to bottom ones.

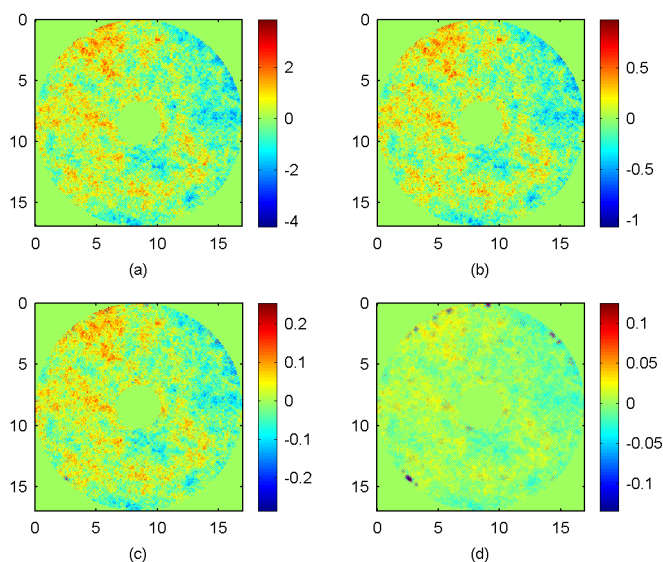


Fig. 6. RPE (in radians) after 50 iterations when solving the wave-front reconstruction for the Fried geometry using the RFD algorithm, and the imbedding step is solved using the ADI method. The wave-front reconstruction is done in a 255x255 sized sampling grids and the subaperture slope SNR is equal to 2, 8, 32 and 128 for the RPE in image (a), (b), (c) and (d), respectively.

6. Conclusion

Using the property of the Kronecker product of matrices, we have found a novel least square wave-front reconstructor based on the Sylvester equations for both the Hudgin and Fried geometry. Using the spectrum theorem for symmetrical real matrices and the Hadamard product concept, we have obtained the Hudgin and Fried filters and the filtering relationship in the 2-D DCT domain for square-aperture wave-front reconstruction problem. Furthermore, to accelerate the convergence rate of the RF algorithm when it is used for the Fried-geometry wave-front reconstruction on an annular aperture, we have exploited the Sylvester equation for the Hudgin-geometry wavefront reconstruction in the imbedding step of the RF algorithm. We use both the ADI method and the BS method to solve this Sylvester equation on a square aperture in the imbedding step, and then compare their performance when they are used with and without a CG procedure in the RF algorithm. We have solved the ADI minimax problem to determine the ADI parameter, and implemented a tridiagonal equation solver based on the LU decomposition method, and then used this tridiagonal solver to solve the tridiagonal systems of equations encountered in the ADI iterative solution process. The tridiagonal equation solver allows a parallel and highly efficient implementation using real-time computing hardware.

Acknowledgements

We appreciate Marcia Brown for improving the presentation of this paper. We acknowledge the financial support of the National Science Foundation through the Center for Adaptive Optics.