LANCZOS-BASED FAST BLIND DECONVOLUTION METHODS

L. DYKES*, R. RAMLAU†, L. REICHEL‡, K. M. SOODHALTER§, AND R. WAGNER¶

Abstract. The task of restoring an image that has been contaminated by blur and noise arises in many applications. When the blurring matrix (or equivalently, the point-spread function) is explicitly known, this task commonly is referred to as deconvolution. In many applications only an approximation of the blurring matrix is available. The restoration task then is referred to as blind deconvolution. This paper describes a family of blind deconvolution methods that allow a user to adjust the blurring matrix used in the computation to achieve an improved restoration. The methods are inexpensive to use; the major computational effort required for large-scale problems is the partial reduction of an available large symmetric approximate blurring matrix by a few steps of the symmetric Lanczos process. A real-time application to adaptive optics that requires fast blind deconvolution is described.

Key words. image restoration, ill-posed problem, Lanczos tridiagonalization, discrepancy principle.

1. Introduction. The need to restore images that have been contaminated by blur and noise arises in many applications, including medical imaging and astronomy. The blur may be caused by object motion, calibration error of the imaging device, or random fluctuations of the medium, e.g., the atmosphere. Let the vector $\boldsymbol{b}^\delta \in \mathbb{R}^n$ represent an available blur- and noise-contaminated $(p \times q)$-pixel image with $n = pq$. For instance, we may store the pixel values for the available noise- and blur-contaminated image column-wise in $\boldsymbol{b}^\delta$. We would like to determine an accurate approximation of the associated unknown blur- and noise-free image, which we represent by the vector $\boldsymbol{x}_{true} \in \mathbb{R}^n$.

Let $A \in \mathbb{R}^{n\times n}$ denote the matrix that models the blurring in the image $\boldsymbol{b}^\delta$, and let the vector $\boldsymbol{e}^\delta \in \mathbb{R}^n$ represent the noise in $\boldsymbol{b}^\delta$. We will assume that a bound $\delta > 0$ for $\|\boldsymbol{e}^\delta\|_2$ is known, i.e.,

$$\|\boldsymbol{e}^\delta\|_2 \leq \delta. \tag{1.1}$$

Here and throughout this paper $\|\cdot\|_2$ denotes the Euclidean vector norm or spectral matrix norm. We also will use the 1-norm of a vector, denoted by $\|\cdot\|_1$, and the Frobenius matrix norm; it is defined by $\|M\|_F = (\mathrm{trace}(M^T M))^{1/2}$ for a matrix M. The available contaminated image $\boldsymbol{b}^\delta$ and the desired blur- and noise-free image $\boldsymbol{x}_{true}$ are assumed to be related by the linear degradation model

$$\boldsymbol{b}^\delta = A\boldsymbol{x}_{true} + \boldsymbol{e}^\delta. \tag{1.2}$$

Many kinds of blur that arise in applications, such as Gaussian blur, can be modeled by a symmetric blurring matrix A. Let the eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ of A be ordered so that $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|$. For many blurring matrices, the magnitude $|\lambda_j|$

*University School, 2785 SOM Center Road, Hunting Valley, OH 44022, USA. E-mail: ldykes@math.kent.edu.

†Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenbergerstr. 69, A-4040 Linz, Austria. E-mail: ronny.ramlau@oeaw.ac.at.

‡Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: reichel@math.kent.edu.

§School of Mathematics, Trinity College Dublin, The University of Dublin, College Green, Dublin 2, Ireland. E-mail: ksoodha@maths.tcd.ie.

¶Johann Radon Institute for Computational and Applied Mathematics, Austrian Academy of Sciences, Altenbergerstr. 69, A-4040 Linz, Austria. E-mail: roland.wagner@ricam.oeaw.ac.at.

decreases quite quickly to zero as j increases. We are concerned with blurring matrices of this kind. Throughout this paper, we assume that the eigenvalues of a matrix are ordered in decreasing magnitude.

The task of determining an approximation of $\boldsymbol{x}_{true}$ when the contaminated image $\boldsymbol{b}^\delta$ and the associated blurring matrix A are explicitly known is commonly referred to as *deconvolution*. Thus, given A and $\boldsymbol{b}^\delta$, deconvolution amounts to determining an approximation of $\boldsymbol{x}_{true}$ by computing a suitable approximate solution of the least-squares problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \|A\boldsymbol{x} - \boldsymbol{b}^\delta\|_2. \tag{1.3}$$

A nice introduction to deconvolution is provided by Hansen et al. [16].

In many image restoration applications only an approximation, $A_0 \in \mathbb{R}^{n\times n}$, of the matrix A in (1.2) is known. Since A is symmetric, it is natural that A_0 would also be symmetric. Simply replacing A by A_0 in (1.3) gives, in many situations, a poor approximation of the desired image $\boldsymbol{x}_{true}$. The determination of a better approximation of $\boldsymbol{x}_{true}$ typically requires that A_0 be replaced by a more accurate approximation of A. It may be possible to compute such an approximation simultaneously with the determination of an approximation of $\boldsymbol{x}_{true}$. The task of determining both a restored image and an improved approximation of the blurring matrix is commonly referred to as *blind deconvolution*.

The dominating computational work for the blind deconvolution method of this paper is the partial reduction of a large symmetric matrix A_0 to a small symmetric matrix by carrying out a few steps of the symmetric Lanczos process. The number of steps required depends on how quickly the eigenvalues of A_0 decay to zero in magnitude as their index number increases and how well A_0 approximates the actual blurring matrix A. This reduction only has to be computed once. Therefore, our blind deconvolution method is quite inexpensive to use. Our solution method has two regularization parameters. This allows for the selection of a blurring matrix from a family of matrices determined by A_0. One of these parameters is determined by the discrepancy principle; the other one is chosen interactively by the user. Alternatively, the user may specify multiple input parameter pairings, as the reconstruction for each pair can be computed fairly inexpensively from the same Krylov subspace.

Many approaches to blind deconvolution have been described in the literature; see, e.g., [3, 4, 5, 6, 7, 8, 27, 29] and references therein. Some of the available methods are computationally expensive, because they require the evaluation of a large number of matrix-vector products with the large matrix A_0 or with related blurring matrices that are determined during the computations. Justen and Ramlau [20] proposed a fast non-iterative blind deconvolution method for $n \times n$ block circulant blurring matrices with circulant blocks. The structure of the blurring matrices makes it possible to reduce the blind deconvolution problem to a problem with a diagonal matrix. The method in [20] allows a user to choose two parameters to adjust a minimization problem to be solved. These parameters determine the blurring matrix used for the restoration. For each value of the parameter pair, the dominant computational effort for computing the associated restoration is the evaluation of a fast Fourier transformation in $\mathcal{O}(n \log n)$ arithmetic floating point operations (flops). The method by Justen and Ramlau [20] imposes periodic boundary conditions. This method is fast, but due to the imposed periodicity, the computed restorations may be contaminated by artifacts. A related blind deconvolution method that is based on a wavelet decomposition is described in [21], and an extension of the method in [20] that allows reflective and anti-reflective

boundary conditions, but requires more computational work, is discussed in [17].

It is the purpose of this paper to present a new blind deconvolution method that is well suited for real-time applications that arise in adaptive optics. Speed of execution is very important for this application, because the point-spread function (PSF) has to be updated every 2 ms (or even more frequently). The use of fast numerical methods therefore is imperative.

Our scheme first reduces the given large blind deconvolution problem to a small one by applying a few steps of the symmetric Lanczos process to reduce A_0. This is followed by spectral factorization of the reduced matrix. The reduced matrix, generally, can be applied to the restoration of a sequence of blur- and noise-contaminated images. The dominating computational effort is the reduction of A_0. Each step of the Lanczos process requires one matrix-vector product evaluation with A_0. Typically, A_0 has a structure, such as block Toeplitz with Toeplitz blocks, that allows the evaluation of each matrix-vector product in only $\mathcal{O}(n \log n)$ flops; see, e.g., Section 5.1. When the spectral factorization of the reduction of A_0 is available, the restoration of an image requires only $\mathcal{O}(n)$ flops. The low flop count makes the proposed method fast and applicable to real-time large-scale blind image restoration problems. We will apply our method to a real-time active optics image restoration problem. This application is described in Section 5.4.

This paper is organized as follows. Section 2 describes a minimization problem and several simplifications. The solution of the latter yields restored images. Section 3 discusses the solution of the minimization problem. A discussion on how to choose the values of the regularization parameters can be found in Section 4, and Section 5 presents a few computed examples. These include experiments from an astronomical imaging application with a large ground-based telescope. Section 6 contains concluding remarks.

2. Minimization problems. Let the symmetric matrix $A_0 \in \mathbb{R}^{n\times n}$ denote an available approximation of the unknown symmetric blurring matrix $A \in \mathbb{R}^{n\times n}$ in (1.2), and let $\boldsymbol{x}_0 \in \mathbb{R}^n$ denote an available approximation of the desired blur- and noise-free image $\boldsymbol{x}_{true} \in \mathbb{R}^n$. For instance, we may choose $\boldsymbol{x}_0$ to be $\boldsymbol{b}^\delta$ or simply the zero vector. Introduce the two-parameter functional

$$F(\alpha_1,\alpha_2) := \min_{\substack{\boldsymbol{x}\in\mathbb{R}^n \\ A\in\mathcal{A}}} \left\{ \|A\boldsymbol{x}-\boldsymbol{b}^\delta\|_2^2 + \alpha_1\|\boldsymbol{x}-\boldsymbol{x}_0\|_2^2 + \alpha_2\|A-A_0\|_F^2 \right\}, \tag{2.1}$$

where $\alpha_1 > 0$ and $\alpha_2 > 0$ are user-specified regularization parameters, and $\mathcal{A}$ is a set of symmetric matrices in $\mathbb{R}^{n\times n}$ for which (2.1) has unique minimizers $\{\alpha_1,\alpha_2\}$ (e.g., a set of matrices which are simultaneously diagonalizeable with A_0).

Example 2.1. Assume that n is small or that the symmetric matrix $A_0 \in \mathbb{R}^{n\times n}$ has a structure that makes it feasible to compute its spectral factorization

$$A_0 = U_0\Lambda_0 U_0^T. \tag{2.2}$$

The former situation arises when $\boldsymbol{b}^\delta$ represents a signal in one space-dimension; the latter case may arise when $\boldsymbol{b}^\delta$ represents a signal (image) in two or more space-dimensions and A_0 is the Kronecker product of small matrices. The matrix $\Lambda_0 \in \mathbb{R}^{n\times n}$ in (2.2) is diagonal, $U_0 \in \mathbb{R}^{n\times n}$ is orthogonal, and the superscript ^{T} denotes transposition.

Define the n-parameter family of matrices

$$\mathcal{A} := \{A \in \mathbb{R}^{n\times n} :\ A = U_0\Lambda U_0^T,\ \Lambda = \mathrm{diag}[\lambda_1,\lambda_2,\ldots,\lambda_n] \in \mathbb{R}^{n\times n}\},$$

in which each element matrix is determined by the parameters $\lambda_1, \lambda_2, \ldots, \lambda_n$. Thus, the real eigenvalues of A are the parameters; the eigenvectors are prescribed to be those of A_0. With this choice of the set $\mathcal{A}$, the minimization problem (2.1) can be simplified to

$$F(\alpha_1, \alpha_2) = \min_{\substack{\boldsymbol{y}\in\mathbb{R}^n \\ \Lambda\in\mathbb{R}^{n\times n} \\ \text{diagonal}}} \left\{ \|\Lambda\boldsymbol{y} - \tilde{\boldsymbol{b}}^{\delta}\|_2^2 + \alpha_1\|\boldsymbol{y} - \boldsymbol{y}_0\|_2^2 + \alpha_2\|\Lambda - \Lambda_0\|_F^2 \right\}, \tag{2.3}$$

where

$$\boldsymbol{y} = U_0^T\boldsymbol{x}, \quad \boldsymbol{y}_0 = U_0^T\boldsymbol{x}_0, \quad \tilde{\boldsymbol{b}}^{\delta} = U_0^T\boldsymbol{b}^{\delta}.$$

Because the matrices Λ and Λ_0 are diagonal, each non-trivial entry of Λ and each element of the vector $\boldsymbol{y}$ in the solution of the minimization problem (2.3) can be computed independently by a nonlinear solver for each pair of positive parameters α_1 and α_2. The solution of (2.3) therefore only requires $\mathcal{O}(n)$ flops. Since the computation of each diagonal entry of Λ and associated component of $\boldsymbol{y}$ can be carried out independently, this makes efficient execution in parallel possible. We will discuss properties of the solution of (2.3) and its computation in Section 3. □

Example 2.2. If n is large or A_0 does not have a structure that makes it possible to compute its spectral factorization rapidly, then we can reduce A_0 to a small matrix by orthogonal projection using the symmetric Lanczos process. Application of $\ell \ll n$ steps of the symmetric Lanczos process to A_0 with initial unit vector $\boldsymbol{v}_1 \in \mathbb{R}^n$ yields the partial Lanczos tridiagonalization

$$A_0V_\ell = V_\ell T_{0,\ell} + \beta_{\ell+1}\boldsymbol{v}_{\ell+1}\boldsymbol{e}_\ell^T, \tag{2.4}$$

where the matrix $V_\ell = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_\ell] \in \mathbb{R}^{n\times\ell}$ has orthonormal columns that span the Krylov subspace

$$\mathcal{K}_\ell(A_0, \boldsymbol{v}_1) = \text{span}\{\boldsymbol{v}_1, A_0\boldsymbol{v}_1, \ldots, A_0^{\ell-1}\boldsymbol{v}_1\}, \tag{2.5}$$

the matrix

$$T_{0,\ell} = \begin{bmatrix} \alpha_1 & \beta_2 & & \\ \beta_2 & \alpha_2 & \ddots & \\ & \ddots & \ddots & \beta_\ell \\ & & \beta_\ell & \alpha_\ell \end{bmatrix} \in \mathbb{R}^{\ell\times\ell}$$

is symmetric and tridiagonal, $\boldsymbol{e}_\ell$ denotes the ℓth Cartesian basis vector, and $\beta_{\ell+1} \geq 0$; see, e.g., [15, 28] for details on the Lanczos process. We will use the initial vector

$$\boldsymbol{v}_1 = \frac{\boldsymbol{b}^{\delta}}{\|\boldsymbol{b}^{\delta}\|_2} \tag{2.6}$$

in the computed examples of Section 5.

The Lanczos process is said to break down at step ℓ if $\beta_{\ell+1} = 0$ in (2.4). Then the spectrum of $T_{0,\ell}$ is a subset of the spectrum of A_0 and the computations with the Lanczos process cannot be continued. One then may either use the matrix $T_{0,\ell}$ or restart the Lanczos process with an initial unit vector that is orthogonal to the columns

of V_ℓ. The occurrence of breakdown is rare and, therefore, will not be discussed further. We remark that instead of the Lanczos process, one may apply the symmetric block Lanczos process. This can be advantageous in certain computing environments; see, e.g., [15] for a discussion on the symmetric block Lanczos process and [12] for an overview of efficient implementation of numerical methods on a parallel computer.

We remark that in our application, generally, only very few steps, ℓ, of the Lanczos process are required. This reduces the possibility of breakdown. It also eliminates the need to restart the Lanczos algorithm. Since the matrix A_0 only is an approximation of the actual unknown blurring matrix, it is not necessary to determine the eigenpairs of A_0 to high accuracy. Using the eigenpairs of $T_{0,\ell}$ generally suffices. In particular, it is not necessary to improve the quality of the eigenpairs by applying a restarted Lanczos method. We comment on the choice of ℓ below.

Computing the spectral factorization of $T_{0,\ell}$, we obtain

$$T_{0,\ell} = U_{0,\ell}\Lambda_{0,\ell}U_{0,\ell}^T, \tag{2.7}$$

where the matrix $U_{0,\ell} \in \mathbb{R}^{\ell\times\ell}$ is orthogonal and

$$\Lambda_{0,\ell} = \mathrm{diag}[\lambda_{0,1}, \lambda_{0,2}, \ldots, \lambda_{0,\ell}] \in \mathbb{R}^{\ell\times\ell}. \tag{2.8}$$

Define the vectors

$$\boldsymbol{y}_\ell = U_{0,\ell}^T V_\ell^T \boldsymbol{x}, \quad \boldsymbol{y}_{0,\ell} = U_{0,\ell}^T V_\ell^T \boldsymbol{x}_0, \quad \tilde{\boldsymbol{b}}_\ell^\delta = U_{0,\ell}^T V_\ell^T \boldsymbol{b}^\delta, \tag{2.9}$$

and the ℓ-parameter family of matrices

$$\mathcal{A}_\ell := \{A \in \mathbb{R}^{n\times n} : A = V_\ell U_{0,\ell}\Lambda_\ell U_{0,\ell}^T V_\ell^T,\ \Lambda_\ell = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_\ell] \in \mathbb{R}^{\ell\times\ell}\} \tag{2.10}$$

determined by the parameters $\lambda_1, \lambda_2, \ldots, \lambda_\ell$, which are to be chosen. All matrices in this set are of rank at most ℓ and have the same eigenspace. The minimization problem (2.1) with $\mathcal{A} := \mathcal{A}_\ell$ can be written as the ℓ-dimensional minimization problem with diagonal matrices,

$$F(\alpha_1, \alpha_2) := \min_{\substack{\boldsymbol{y}_\ell\in\mathbb{R}^\ell \\ \Lambda_\ell\in\mathbb{R}^{\ell\times\ell} \\ \text{diagonal}}} \left\{ \|\Lambda_\ell \boldsymbol{y}_\ell - \tilde{\boldsymbol{b}}_\ell^\delta\|_2^2 + \alpha_1\|\boldsymbol{y}_\ell - \boldsymbol{y}_{0,\ell}\|_2^2 + \alpha_2\|\Lambda_\ell - \Lambda_{0,\ell}\|_F^2 \right\}. \tag{2.11}$$

This functional is analogous to (2.3). We will show in Section 3 that the minimization problem (2.11) is easy to solve. In numerous applications, we found the choice $\boldsymbol{x}_0 = \boldsymbol{0}$, which yields $\boldsymbol{y}_0 = \boldsymbol{0}$, to be suitable. We found this choice typically to yield more accurate approximations of $\boldsymbol{x}_{true}$ than $\boldsymbol{x}_0 = \boldsymbol{b}^\delta$.

Let $\{\boldsymbol{y}_\ell^*, \Lambda_\ell^*\}$ minimize (2.11). The restored image then is given by

$$\boldsymbol{x}^* = V_\ell U_{0,\ell} \boldsymbol{y}_\ell^*. \tag{2.12}$$

In many applications, the matrix A_0 has a structure that allows fast evaluation of matrix-vector products. For instance, when $\boldsymbol{b}^\delta$ represents an image in two space-dimensions that has been contaminated by space-invariant Gaussian blur, $A_0 \in \mathbb{R}^{n\times n}$ can be chosen to be a symmetric block Toeplitz matrix with Toeplitz blocks. The evaluation of a matrix-vector product with such a matrix requires only $\mathcal{O}(n \log n)$ flops. The number of steps ℓ of the Lanczos process should be large enough so that the largest eigenvalues and eigenvectors of A_0 can be approximated fairly accurately by

Ritz values and Ritz vectors defined by the Lanczos decomposition (2.4). This choice is motivated by the model assumption that the invariant subspace of A_0 associated with the largest eigenvalues does not greatly depart from the corresponding invariant subspace of the true operator A. In our practical adaptive optics example, cf., Figures 5.8–5.9, one sees that this is a reasonable assumption. Taking the eigenvalues to be listed in order of descending magnitude, we observe that the eigenvalues many blurring matrices A_0 decay quite quickly to zero in magnitude. Thus, the number of Lanczos steps ℓ generally can be chosen independently of n. Therefore, the flop count for the Lanczos process typically is $\mathcal{O}(n \log n)$. The choice of ℓ is also influenced by how much the eigenvectors of A_0 depart from those of the true operator A. This behavior is illustrated in the first part of Section 5.

3. Solution of the minimization problem. This section discusses the solution of the minimization problem (2.11). Let the diagonal matrices $\Lambda_{0,\ell}$ and Λ_ℓ have the nontrivial entries $\lambda_{0,j}$ and λ_j, $1 \leq j \leq \ell$, respectively; see (2.8) and (2.10). Let

$$\begin{aligned} \boldsymbol{y}_\ell &= [y_1, y_2, \ldots, y_\ell]^T \in \mathbb{R}^\ell, \\ \boldsymbol{y}_{0,\ell} &= [y_{0,1}, y_{0,2}, \ldots, y_{0,\ell}]^T \in \mathbb{R}^\ell, \\ \tilde{\boldsymbol{b}}_\ell^\delta &= [\tilde{b}_1^\delta, \tilde{b}_2^\delta, \ldots, \tilde{b}_\ell^\delta]^T \in \mathbb{R}^\ell. \end{aligned}$$

Then the minimization problem (2.11) is equivalent to the ℓ decoupled minimization problems

$$\min_{y_j, \lambda_j \in \mathbb{R}} \left\{ (\lambda_j y_j - \tilde{b}_j^\delta)^2 + \alpha_1 (y_j - y_{0,j})^2 + \alpha_2 (\lambda_j - \lambda_{0,j})^2 \right\}, \quad j = 1, 2, \ldots, \ell. \tag{3.1}$$

We first note that we do not consider the case $\alpha_1 = \alpha_2 = 0$. Clearly, the minimization problems (3.1) do not have a unique solution in this situation. Also when α_1 and α_2 are positive, the solution might not be unique. This is illustrated in Example 3.2 below. However, this case was not encountered in any of the practical problems we tested.

Example 3.1. Let $\tilde{b}_j^\delta = 1$, $\alpha_1 = \alpha_2 = 1/10$, and $y_{0,j} = 1$, and $\lambda_{0,j} = 0$ in (3.1). Figure 3.1 displays the function

$$f_j(y_j, \lambda_j) = (\lambda_j y_j - 1)^2 + \frac{1}{10}((y_j - 1)^2 + \lambda_j^2), \qquad -2 \leq y_j, \lambda_j \leq 2.$$

The minimum of f_j at approximately $y_j = 1.4$ and $\lambda_j = 0.7$ is marked by a red star. □

One convenient choice in applications is to enforce a linear dependence between the parameters, with $\alpha_1 := \alpha$ and $\alpha_2 := \alpha t$ in (3.1), where $t \geq 0$ is a fixed, user-defined constant and $\alpha \geq 0$ is a regularization parameter to be determined. This simplifies the solution of the minimization (3.1) and allows the user to determine t based on knowledge of the specific reconstruction problem. Substitution into (3.1) yields

$$f_j(y, \lambda) := (\lambda y - \tilde{b}_j^\delta)^2 + \alpha((y - y_{0,j})^2 + t(\lambda - \lambda_{0,j})^2), \qquad j = 1, 2, \ldots, \ell. \tag{3.2}$$

The gradient and Hessian of f_j are given by

$$\nabla f_j(y, \lambda) = \begin{bmatrix} \dfrac{\partial f_j}{\partial y}(y, \lambda) \\ \dfrac{\partial f_j}{\partial \lambda}(y, \lambda) \end{bmatrix} = \begin{bmatrix} 2(y\lambda - \tilde{b}_j^\delta)\lambda + 2\alpha(y - y_{0,j}) \\ 2(y\lambda - \tilde{b}_j^\delta)y + 2\alpha t(\lambda - \lambda_{0,j}) \end{bmatrix} \tag{3.3}$$

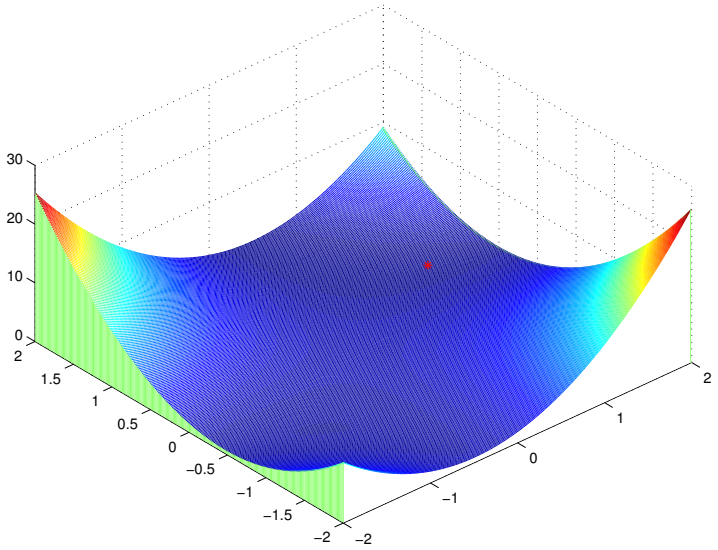

Fig. 3.1: Generic function $f_j(y,\lambda)$ for $-2 \le y \le 2$ and $-2 \le \lambda \le 2$.

and

$$\Delta f_j(y,\lambda) = \begin{bmatrix} 2(\lambda^2+\alpha) & 4y\lambda - 2\tilde{b}_j^\delta \\ 4y\lambda - 2\tilde{b}_j^\delta & 2(y^2+\alpha t) \end{bmatrix}. \tag{3.4}$$

PROPOSITION 3.1. *Let $\alpha > 0$ and $t \ge 0$. Then the Hessian (3.4) has (at least) one positive eigenvalue. If $y \ne 0$ or $t > 0$, then the other eigenvalue is positive for $\alpha > 0$ sufficiently large, in which case the function f_j is convex in a neighborhood of (y,λ).*

Proof. The trace of the Hessian is positive. Therefore, the sum of the eigenvalues is positive. The product of the eigenvalues is given by the determinant,

$$\det\left(\Delta f_j(y,\lambda)\right) = 4(\lambda^2+\alpha)(y^2+\alpha t) - 4(2y\lambda - \tilde{b}_j^\delta)^2,$$

and may be negative, in which case the Hessian is indefinite. For fixed $\{y, \lambda, \tilde{b}_j^\delta\}$, the Hessian can be made positive definite by choosing α sufficiently large provided that $y \ne 0$ or $t > 0$. ∎

To find y- and λ-values that minimize $f_j(y,\lambda)$, we first consider values such that $\nabla f_j(y,\lambda) = 0$. We obtain from $\frac{\partial f_j}{\partial \lambda}(y,\lambda) = 0$ that

$$\lambda = \frac{\tilde{b}_j^\delta y + \alpha t \lambda_{0,j}}{y^2 + \alpha t}. \tag{3.5}$$

Let us assume that the conditions on α, t, and y of Proposition 3.1 hold. Then $y^2 + \alpha t > 0$. Substituting (3.5) into $\frac{\partial f_j}{\partial y}(y,\lambda) = 0$ yields the equation $p(y) = 0$ for a polynomial in y of degree five,

$$\begin{aligned} p(y) = {} & y^5 - y_{0,j}y^4 + 2\alpha t y^3 + (\lambda_{0,j}\tilde{b}_j^\delta t - 2y_{0,j}\alpha t)y^2 \\ & + (\alpha^2 t^2 - (\tilde{b}_j^\delta)^2 t + \lambda_{0,j}^2 \alpha t^2)y - y_{0,j}\alpha^2 t^2 - \tilde{b}_j^\delta \lambda_{0,j} \alpha t^2. \end{aligned} \tag{3.6}$$

Having computed its zeros $y_j^{(k)}$, $k = 1, 2, \ldots, 5$, we determine the associated λ-values from (3.5), i.e., from

$$\lambda_j^{(k)} = \frac{\tilde{b}_j^\delta y_j^{(k)} + \alpha t \lambda_{0,j}}{(y_j^{(k)})^2 + \alpha t}, \qquad k = 1, 2, \ldots, 5.$$

We are only interested in the real zeros. The points $(y_j^{(k)}, \lambda_j^{(k)})$ associated with the real zeros $y_j^{(k)}$ of p are possible minima of $f_j(y, \lambda)$. We evaluate the function f_j at these points to determine all solutions of (2.11) with $\alpha_1 = \alpha$ and $\alpha_2 = \alpha t$. If $f_j(y, \lambda)$ achieves the minimum at more than one point, then we choose a point that yields the largest $|\lambda_j^{(k)}|$ to obtain a matrix A with eigenvalues of the largest possible magnitude. We note that if $t^2(\alpha^2 y_{0,j} + \alpha \tilde{b}_j^\delta \lambda_{0,j}) = 0$, then $p(y) = yq(y)$ for a polynomial q of degree four. This gives the zero $y = 0$ and the associated value $\lambda = \lambda_{0,j}$. We also obtain

$$y_{0,j} = -\frac{\tilde{b}_j^\delta}{\alpha} \lambda_{0,j}.$$

This relation can be used to check whether $y = 0$ is a zero (in the case that $t > 0$ so that the conditions of Proposition 3.1 are fulfilled).

Under certain conditions, the solution of $\nabla f_j(y, \lambda) = 0$ can be reduced to the determination of the zeros of a polynomial of degree three in y. We will discuss this situation and use it to illustrate that the minimization problem (3.1) may have more than one solution. Assume that $y\lambda \neq 0$ and let $t = 1$. Multiplying $\partial f_j / \partial y$ and $\partial f_j / \partial \lambda$ by y and λ, respectively, gives the nonlinear system of equations

$$\begin{aligned} \lambda^2 y^2 + \alpha y^2 &= \tilde{b}_j^\delta \lambda y + \alpha y_{0,j} y, \\ \lambda^2 y^2 + \alpha \lambda^2 &= \tilde{b}_j^\delta \lambda y + \alpha \lambda_{0,j} \lambda. \end{aligned}$$

Subtracting the second equation from the first one yields

$$y^2 - \lambda^2 = y_{0,j} y - \lambda_{0,j} \lambda, \tag{3.7}$$

where we have used that $\alpha > 0$. Assume that we have the special situation that $\lambda_{0,j} = y_{0,j}$. Then (3.7) becomes $y^2 - \lambda^2 = y_{0,j}(y - \lambda)$. Thus, either $\lambda = y$ or $\lambda = y_{0,j} - y$. Substitution into (3.5) yields two types of cubic polynomial equations

$$\begin{aligned} \lambda = y: &\quad y^3 + (\alpha - \tilde{b}_j^\delta) y - \alpha y_{0,j} = 0, \\ \lambda = y_{0,j} - y: &\quad y(y^2 - y_{0,j} y + \alpha + \tilde{b}_j^\delta) = 0. \end{aligned}$$

In the second cubic, $y = 0$ is an extraneous solution. It is easily verified that the product of the first cubic and the quadratic factor of the second cubic will give the original fifth degree polynomial when $t = 1$.

Example 3.2. Let $\tilde{b}_j^\delta = 1$, $\alpha = 1/10$, $t = 1$, and $y_{0,j} = \lambda_{0,j} = 0$. Then the above discussion yields two third degree polynomials in y. The real solutions are $\{0, \pm 3/\sqrt{10}\}$. The minimum of $f_j(y, \lambda)$ is achieved at $\{y, \lambda\}$ for $y = \lambda = \pm \frac{3}{\sqrt{10}}$.

4. Selection of parameters. The blind deconvolution method described in the previous sections requires the selection of three parameters: two regularization parameters α_1 and α_2, as well as the number of Lanczos steps ℓ. In computations, we

will determine the parameter $\alpha_1 := \alpha$ with the aid of the discrepancy principle (see below), and let $\alpha_2 := \alpha t$ for some user-chosen fixed parameter $t \geq 0$. This parameter balances the influence of the regularization terms in (3.1). For instance, letting $\alpha > 0$ and $t = \infty$ gives (standard) Tikhonov regularization restricted to a subset of Ritz vectors associated with A_0. This is an appropriate regularization method when the blur in the available blur- and noise-contaminated image is accurately modeled by the matrix $A := A_0$. In many of the computed examples of Section 5, we let

$$t = \|\tilde{\boldsymbol{b}}_\ell^\delta\|_2^2 / \|\Lambda_0\|_F^2. \tag{4.1}$$

This choice of t balances the influence of the regularization terms in (3.1). If one knows nothing else about the problem, this is a sound choice, but for particular reconstructions, one may find that a different value of t works better, cf., Section 5.4.

The third parameter required is the dimension ℓ of the subspace (2.10), which is used to determine the blurring matrix A. This dimension equals the number of steps of the symmetric Lanczos process applied to A_0. The value of ℓ should be chosen large when A_0 is an accurate approximation of the true blurring matrix than when it is not; the value also should be chosen larger when the available image $\boldsymbol{b}^\delta$ is contaminated by a significant amount of noise than when it is not. In our experience, ℓ generally can be chosen quite small. It is difficult to determine a suitable value of ℓ without some experimentation for each kind of problem of interest. In the adaptive optics problems of Subsection 5.4, the available matrices A_0 can be chosen to be the same for all restoration problems, and the amount of error in the vector $\boldsymbol{b}^\delta$ also is about the same for all problems. This allows us to determine a suitable value of ℓ by carrying out some experiments. We then can use this ℓ-value for all restoration problems. We use the initial vector (2.6) for the Lanczos process.

The discrepancy principle is a popular approach to determine the regularization parameter in (standard) Tikhonov regularization when a fairly accurate bound (1.1) for the norm of the "noise" vector $\boldsymbol{e}^\delta$ in $\boldsymbol{b}^\delta$ is known; see, e.g., [10]. As a first step in our blind deconvolution method, we solve a (standard) Tikhonov regularization problem with the available blurring matrix A_0. Thus, we solve

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \{\|A_0\boldsymbol{x} - \boldsymbol{b}^\delta\|_2^2 + \alpha\|\boldsymbol{x}\|_2^2\}. \tag{4.2}$$

This minimization problem has a unique solution for any $\alpha > 0$. Denote the solution by $\boldsymbol{x}_\alpha$. The discrepancy principle prescribes that $\alpha > 0$ be chosen such that

$$\|A_0\boldsymbol{x}_\alpha - \boldsymbol{b}^\delta\|_2 = \tau\delta, \tag{4.3}$$

where $\tau > 1$ is a user-specified constant independent of δ; see [10] for details. When the matrix A_0 is small enough so that it is feasible to compute its spectral factorization (2.2), the solution $\boldsymbol{x}_\alpha$ of (4.2) that satisfies the constraint (4.3) easily can be computed. For large-scale problems, determining α via the discrepancy principle using $\boldsymbol{b}^\delta$ and A_0 can be done efficiently using an Arnoldi-Tikhonov iteration, in particular because we already iteratively generate the Krylov subspace in our proposed algorithm. We denote the regularization parameter determined in this manner by α_0. The parameter ℓ has to be chosen large enough so that the discrepancy principle can be satisfied.

We present this process as Algorithm 1, using the parameter choice rule described in eq. (4.1) and where $\alpha > 0$ is chosen in advance or according to a discrepancy principle of the form (4.3).

Algorithm 1: Lanczos-based blind deconvolution

Input: $A_0 \in \mathbb{R}^{n\times n}$, $\mathbf{x}_0$, $\mathbf{b}^\delta \in \mathbb{R}^n$, image regularization parameter $\alpha > 0$, noise level $\delta > 0$
Set t according to (4.1);
Use Lanczos process to build $\mathcal{K}_\ell\left(A_0, \boldsymbol{b}^\delta\right)$ producing $T_{0,\ell}$ and V_ℓ;
Compute eigendecomposition $U_0\Lambda_0U_0^T$ with eigenvalues in descending order by magnitude;
Set $j \leq \ell$ to be the last index such that $|\lambda_{j,0}| > \delta$;
Truncate $U_0 \leftarrow U_0(:, 1:j)$ and $\Lambda_0 \leftarrow \Lambda_0(1:j, 1:j)$;
for $i = 1, 2, \ldots j$ **do**
    Compute zeros of $p(y)$ from (3.6) using $\lambda_{0,j}$, $y_{0,j}$, and $\tilde{b}_{0,j}$;
    for *each y a root of $p(y)$* **do**
        Compute associated λ according to (3.5);
    end
    Determine which of these computed pairs (y, λ) minimizes (3.2);
    Set $(y_i, \lambda_i) \leftarrow (y, \lambda)$
end
Set $\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_j \end{bmatrix}^T$ and $\Lambda = \operatorname{diag}\{\lambda_1, \lambda_2, \ldots, \lambda_j\}$;
Set $\mathbf{x}_\ell = \mathbf{x}_0 + V_\ell U_0 \mathbf{y}$ and $A_\ell = V_\ell U_0 \Lambda \left(V_\ell U_0\right)^T$;

5. Computed examples. This section presents two sets of computed examples. All examples were executed on a MacBook Pro with a 3.1 Ghz Intel Core i5 processor and 8GB of 2133 MHz LPDDR3 main memory and a solid state drive. Except for in the adaptive optics examples in Section 5.4, the regularization parameter α was chosen to satisfy the discrepancy principle, either directly or using an Arnoldi-Tikhonov [23] iteration, applied to the perturbed operator and noisy right-hand side. The parameter t was chosen according to (4.1). In the first set of examples, reported in Section 5.1, we carry out experiments using a sample Gaussian blurring problem with perturbed eigenvalues. The second set of tests, shown in Section 5.4, discuss an adaptive optics application problem. Comparisons with a method described by Buccini et al. [2] are presented in Sections 5.2 and 5.3. However, we omit comparisons in some experiments for brevity. Our experience has been that the algorithm presented in [2] produces reconstructions of higher quality than the method presented in this paper, but it is much slower as it is more computationally intensive.

In our computed examples, we show the relative error of our reconstruction, which for a particular reconstruction $\boldsymbol{x}_{recon}$ is calculated as

$$\frac{\|\boldsymbol{x}_{true} - \boldsymbol{x}_{recon}\|_2}{\|\boldsymbol{x}_{true}\|_2},$$

and we also compute the peak signal-to-noise ratio (PSNR),

$$\mathrm{PSNR}(\boldsymbol{x}_{recon}, \boldsymbol{x}_{true}) = 10\log_{10}\left(\frac{255^2}{\|\boldsymbol{x}_{recon} - \boldsymbol{x}_{true}\|_2^2}\right).$$

The constant 255 in the numerator stems from that each image is represented by n 8-bit pixels, which take on interger values in the interval $[0, 255]$. The range limitation is not imposed during the solution of the minimization problem described in the previous sections. Moreover, the PSF should not create or damp light. We determine

a reconstruction that approximately satisfies this requirement, as well as the range restriction as follows: We first set negative entries of the restoration computed as described in the previous sections to zero, then scale the vector so obtained so that its 1-norm agrees with $\|\boldsymbol{b}^{\delta}\|_1$, and finally set entries larger than 255 of the latter vector to 255. This gives the reconstruction $\boldsymbol{x}_{recon}$, which approximates $\boldsymbol{x}_{true}$.

In the following experiments, the initial approximation is always chosen to be the zero vector, i.e., an empty black image. The initial operator approximation is the given operator A_0.

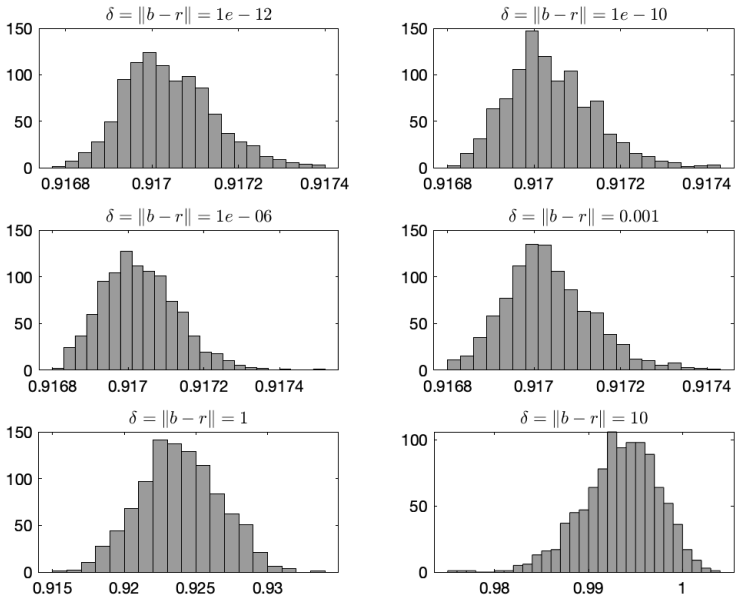

Fig. 5.1: For eigenvalue perturbations of the Gaussian blurring matrix, we plot the relative quality of the reconstructed operator (as measured by (5.1)) at different distances from $\boldsymbol{b}$. Each histogram shows the results for 1000 experiments.

5.1. Applications to a Gaussian blur matrix. We first show some experiments that involve simple perturbations of a Gaussian blurring matrix. Let A denote the (true) Gaussian blurring matrix and A_0 a modified blurring matrix that is obtained by perturbing the eigenvalues of A. Each eigenvalue is perturbed by numbers generated by the Matlab function `rand()` (so uniformly distributed on the interval $[0, 1]$) scaled by 10^{-2}. Thus, A_0 has the same eigenvectors as A.

We cannot expect the computed blurring matrix $\widetilde{A}$ to be an accurate approximation of the true blurring matrix A for all initial vectors $\boldsymbol{b}^{\delta}$ for the symmetric Lanczos process. With the proposed method, one builds $\widetilde{A}$ using a Krylov subspace determined by the matrix A_0 and the particular initial vector $\boldsymbol{b}^{\delta}$. We therefore can expect $\widetilde{A}$ to be most accurate in a neighborhood of the initial vector $\boldsymbol{b}^{\delta}$. As an analogy, one can consider the accuracy of the Fréchet derivative developed around a point as one moves away from that point. This idea motivates the following experiment.

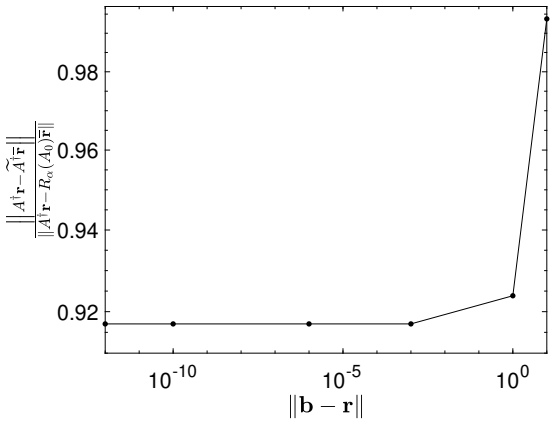

Fig. 5.2: For eigenvalue perturbations of the Gaussian blurring matrix, we plot the relative quality of the reconstructed matrix at for different distances from the right-hand side $\boldsymbol{b}$. Each data point displays the average over 1000 experiments.

For a given right-hand side $\boldsymbol{b}$ and initial matrix A_0, the proposed method generates a low-rank matrix $\widetilde{A}$. The goal now is to test the accuracy of this operator in various neighborhoods of $\boldsymbol{b}$. We define accuracy in terms how well the action of $\widetilde{A}^\dagger$ applied to a noise-polluted vector in that neighborhood approximates the action of $A^\dagger$ applied to the nonpolluted vector. This is judged relative to the accuracy of the optimal Tikhonov regularized solution produced using A_0 (denoted by $R_\alpha(A_0)$). Thus, for $\boldsymbol{r}$ in a neighborhood of $\boldsymbol{b}$ and $\boldsymbol{r}^\delta$ a noisy perturbation thereof, we compute

$$err(\tilde{A}, \boldsymbol{r}) = \frac{\|A^\dagger \boldsymbol{r} - \widetilde{A}^\dagger \boldsymbol{r}^\delta\|_2}{\|A^\dagger \boldsymbol{r} - R_\alpha(A_0)\boldsymbol{r}^\delta\|_2}. \tag{5.1}$$

We must, however, be careful when generating vectors $\boldsymbol{r}$ in the neighborhood. The vector is an unperturbed vector in the range of the blurring matrix A. We would like to consider perturbations $\boldsymbol{r}$ of $\boldsymbol{b}$ that also are in the range of A. We therefore set $\boldsymbol{r} = \boldsymbol{b} + A\overline{\boldsymbol{r}}$, where $\overline{\boldsymbol{r}}$ is a vector with normally distributed random entries with zero mean scaled so that $\boldsymbol{r}$ is in the desired neighborhood of $\boldsymbol{b}$. In other words, we choose $\boldsymbol{r}$ from the desired neighborhood of $\boldsymbol{b}$ such that $\|A^\dagger \boldsymbol{r}\|$ does not blow up.[1]

Figures 5.1 and 5.2 illustrate the quality of the operator $\widetilde{A}$ as measured according to the tests we just described. The figures demonstrate that when measuring the quality of the reconstructed operator, the distance $\|\boldsymbol{r} - \boldsymbol{b}\|_2$ may play a role. Figure 5.1 shows histograms of the relative norms (5.1) for several distances of $\boldsymbol{r}$ from $\boldsymbol{b}$. Each histogram is for 1000 runs with different random vectors $\boldsymbol{r}$. Figure 5.2 displays the mean relative norm for each distance.

5.2. Grain problem. Here we tested with the *Grain* image from [1, 25] and the same exact and perturbed PSFs used in the experiments from [2, Figures 8–9]. The method proposed in [2] produces a higher quality solution, but requires a $\mathcal{O}(10)$ more time to run to completion on the same computer. Neither code has been optimized; thus we only report general performance comparisons to illustrate the strength of

[1] In an infinite-dimensional setting, we could simply say that we assure that $\boldsymbol{r}$ is in the range of A.

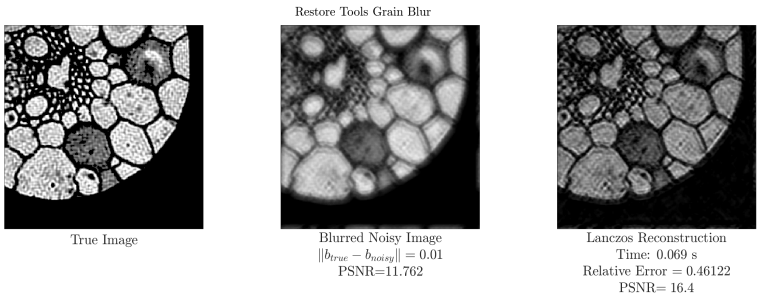

Fig. 5.3: Application of the Lanczos blind deconvolution algorithm applied to the grain problem with A_0 induced by a perturbed PSF from [2].

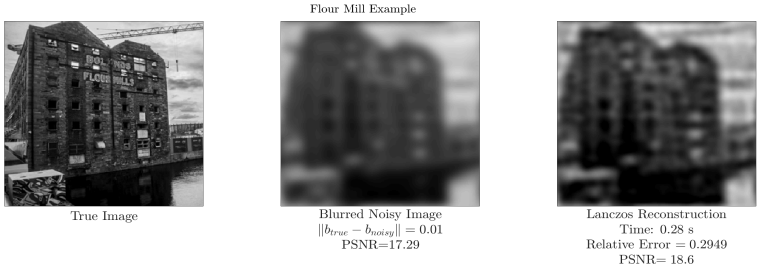

Fig. 5.4: Application of the Lanczos blind deconvolution algorithm applied to the Flourmill problem with A_0 induced by perturbed PSF from [2].

what we propose rather than precise timings. We also tested how well the Arnoldi–Tikhonov method [23] performs when generating a reconstruction using A_0. For the present example, this method was not able to deliver a reconstruction.

5.3. Flourmill problem. In the Flourmill example, displayed by Figure 5.4, we used the same perturbed PSF as was used for [2, Figures 5]. We observe that only the coarse details of the Flourmill were recovered by this method, and this required little time to execute. For such images, we would recommend our method as a fast way to construct a better initial approximation for another slower, more costly method, that may be able to determine a more accurate approximation of $\boldsymbol{x}_{true}$. This is meaningful because methods that produce accurate restorations, such as [2], are nonlinear. They can be sped up by providing an initial approximation of the restored image that is of higher quality than the image represented by $\boldsymbol{b}^{\delta}$.

5.4. Adaptive optics problem. The experiments of this section come from an astronomical imaging application with a large ground-based telescope, such as the planned European Extremely Large Telescope (E-ELT). One would like to view a part of the night sky in order to detect astronomical phenomena. A major cause of image distortion is that light from these phenomena must pass through the turbulent atmosphere of the earth to reach the telescope. To overcome this problem, adaptive optics (AO) systems have been developed. The purpose of these systems is to compensate for atmospheric distortions in real time; see, e.g., [9, 26] for discussions. An AO system uses indirect measurements from wavefront sensors (WFS) to adjust a de-

formable mirror (DM) so that it compensates for atmospheric turbulence. However, since the atmosphere above the telescope changes within 2 ms, the AO system does not perfectly correct for the turbulence as it suffers from time delay. As the acquisition time for a single image ranges from several seconds to a few minutes (depending on the brightness of the observed object), the image degradation due to the rapidly changing atmosphere depends on the average of noncompensated residual aberrations. This makes phenomena that are close difficult to identify and differentiate. This is illustrated by Figure 5.5 for the star cluster image from the Restore Tools software package [1, 25].

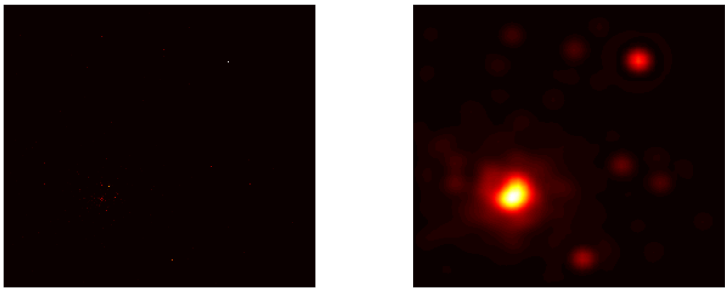

Fig. 5.5: The true star cluster image (left) and the star cluster image convolved with a simulated true PSF. The images are shown in log-scale to exaggerate the star brightness.

In this application we have a PSF that describes how a single point source of light is smeared out when light passes through the distorting medium (atmosphere) before arriving at the sensor. Here we have two unknown objects, the true image and the blurring operator. During an actual star observation with the telescope, the PSF is unknown. However, the PSF of the telescope exhibits the well-understood Airy pattern as the aperture of the telescope is of annular shape. This PSF is commonly referred to as the diffraction limited PSF and is of the form

$$PSF(x) = |\mathcal{F}(P(x))|^2, \tag{5.2}$$

where $\mathcal{F}$ is the Fourier transform and $P(x)$ is the *telescope pupil function*. Thus, $P(x) = 1$, if x is inside the aperture, and $P(x) = 0$ else.

A PSF associated with an observation through turbulent atmosphere with a telescope equipped with an AO system also has to take into account residual phase aberrations ϕ. This defines an instantaneous PSF, which is of the form

$$PSF_\phi(t) = |\mathcal{F}(P(x)e^{i\phi(x,t)})|^2.$$

An image I formed through telescope observation can mathematically be described as a convolution of the true object I_{true} and the PSF of the observing system, resulting in

$$I(t) = I_{true} * PSF_\phi(\cdot, t).$$

The acquisition of a single image takes longer than the PSF remains constant. We therefore have to consider a time average. Using the fact that the true observed object

does not change during the observation time, we obtain the equation

$$\langle I \rangle = I_{true} * \langle PSF_\phi(\cdot, t) \rangle,$$

where $\langle \cdot \rangle$ denotes the time average. In the following, we seek to estimate $PSF_{obs} := \langle PSF_\phi(\cdot, t) \rangle$.

The convolution of an image with the PSF induces a matrix of the same size as the image. From the measurements of the WFS, one can determine both how the DM should be adjusted, as well as the residual turbulent wavefront resulting from the time lag. The formulas above can be used to derive an estimate for PSF_{obs} from the residual wavefront turbulence. In the following, we will refer to the PSF determined from the WFS measurements as the reconstructed PSF. The first known successful attempt to determine a reconstructed PSF is described in [30] for a simple AO system. In recent years several methods for reconstructing the PSF have been developed; see [11, 13, 14, 19, 24].

In addition to the time lag, reconstructed PSFs also suffer from imperfections due to approximations in the modeling process and numerical errors. The latter stem both from discretization errors caused by the use of a fairly coarse grid for the WFS and from propagated round-off errors introduced, e.g., when computing the discrete Fourier transform.

Of course, we do not know the true PSF. However, using the official simulation tool of the European Southern Observatory, *Octopus* [22], we can obtain a high-quality approximation of the true PSF. We consider the latter PSF as the true PSF.

The use of an available approximate PSF makes experiments with our algorithm a bit more difficult to carry out than when an approximate blurring matrix is available. Since we do not know the blurring matrix associated with the available PSF, we cannot study its spectral factorization to understand how our algorithm behaves.

To gain some insight into the performance of our blind deconvolution method, we take advantage of the fact that the PSF can be convolved with images of different sizes. Thus, we can convolve the PSF with all canonical basis vectors from $\mathbb{R}^{1024}$, which span the space of 32×32 images. This can be performed in a reasonable amount of time and induces a blurring matrix that can be quickly decomposed and studied to help us understand the performance of our blind deconvolution method. Indeed, we noticed immediately that the PSFs from the AO application do not induce symmetric blurring matrices. This is due to the presence of small perturbations caused by noise in the measured PSF.

One can apply radial averaging to symmetrize the PSF. This is quite natural, as it is common to only consider the radial average of a PSF in real applications; see [24]. A justification of averaging is that in the AO system, all wavefront distortions are small and have expected value zero. This suggests that a PSF associated with a run of the AO system should be fairly close to the PSF of the telescope, which is symmetric.

The acquisition of an image takes a long time compared to measuring the wavefront distortions. Therefore, speckles occurring due to distortions in a short time frame and measurement errors tend to average out. The reconstructed PSF is in the direction of the telescope. Therefore, no hardware dependent asymmetric behavior of the system will show up at the center of our image. Figure 5.6 shows symmetrized versions of the PSFs. The distortion created by the action of each radially symmetrized PSF can be seen in the central image of Figure 5.10.

Our true and reconstructed radially symmetric PSFs allow us to determine associated blurring matrices of size 1024×1024 and study their properties. Plots of their

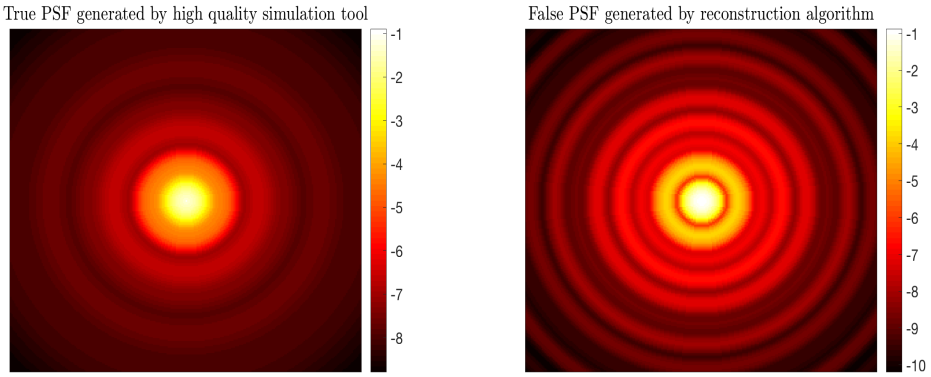

Fig. 5.6: Log plot of the absolute values of the true (left) and of the reconstructed radially symmetrized PSF (right).

entries, displayed in Figure 5.7, show the blurring matrices to be highly structured. Indeed, a cursory look at the entries suggests that the blurring matrices are close to block matrices in which each block is a Toeplitz matrix. It is well-known that block-Toeplitz-Toeplitz-block matrices are discrete convolution operators; see, e.g., [6, 18].

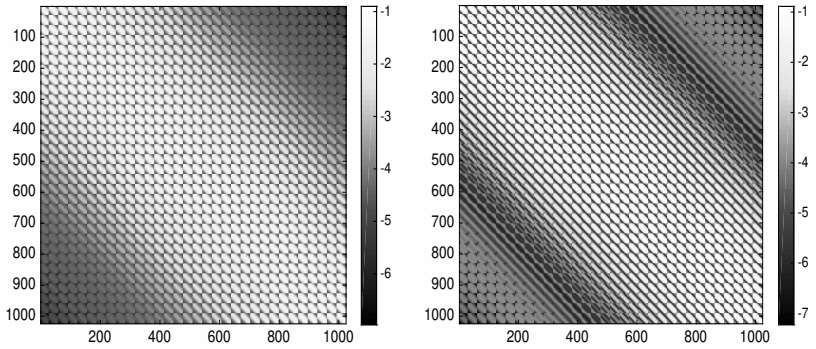

Fig. 5.7: Log plot of the absolute values of the matrices determined by radially symmetrized true and reconstructed PSFs.

Figure 5.8 displays the log-plot of the positive eigenvalues in descending order of the matrices displayed in Figure 5.7, and Figure 5.9 shows four unit eigenvectors associated with the largest eigenvalues of both matrices. One can appreciate that the dominant eigenvectors look fairly similar qualitatively and that the eigenvalues decay rapidly. We deduce that already a few iterations with the Lanczos process applied to this blurring matrix have the potential of giving accurate approximations of the dominant eigenvectors of the matrix induced by the true PSF. The method proposed in this paper can be used to determine suitable eigenvalues.

Judging the quality of the low-rank operator reconstruction is a bit more difficult. There is no explicitly available true operator with which to compare, and for an image of this size, the induced matrix representation of the PSF is prohibitively expensive

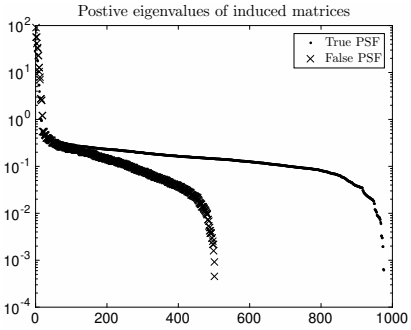

Fig. 5.8: Log plot of the positive eigenvalues of the matrices induced by applications of the true and false PSFs.

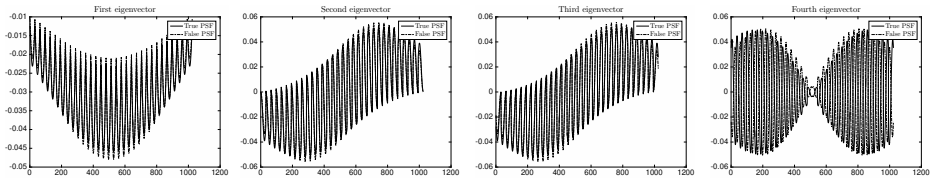

Fig. 5.9: Comparison of first four unit eigenvectors of matrices induced by the two PSFs. On close inspection, each pair differs in amplitude, but qualitatively has the same frequency.

to compute.

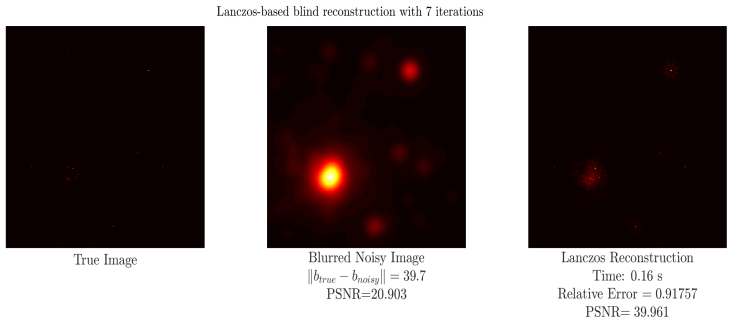

Fig. 5.10: Reconstruction of the symmetrized PSF from the AO application using the rougher approximate PSF.

For the same PSF, we carry out the same experiments for a less sparse image, namely the satellite image, also from Restore Tools [1, 25]. In Figure 5.11, we show an experiment with a different choice of t than is recommended in (4.1), as this yielded a better result.

6. Discussion and conclusion. We have developed a new blind deconvolution method which generates a Krylov subspace using a given, incorrect operator A_0 and a

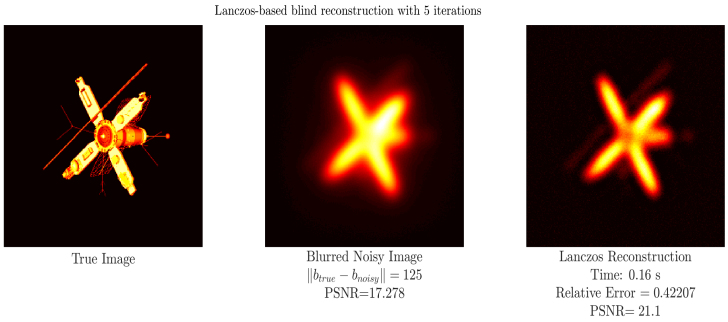

Fig. 5.11: Here we chose $t = 1$, as it yielded better performance than the default choice.

noise-contaminated vector $\boldsymbol{b}^{\delta}$. Using this subspace, we determine eigenvalue approximations, an approximation of the desired image, and a low-rank approximation of the true blurring matrix. This is accomplished through the minimization of a scalar functional that yields, for each approximate eigenvector, an eigenvalue of the low-rank matrix approximation.

We have shown that the method is able to deliver image and blurring matrix reconstructions. Differently from some other available blind deconvolution methods, the method presented requires no information about the blurring matrix and boundary conditions specific for the problem. The main computational burden is the evaluation of a few matrix-vector products with an available approximation of the actual symmetric blurring matrix. The method therefore is inexpensive to apply. Its performance depends on the relationship between the spectral information of the false available blurring matrix and the true unknown one. The difference between their eigenspaces is especially important since we use the approximate eigenvectors generated by the Lanczos process when applied to the false blurring matrix to construct an approximation of the true blurring matrix.

As the results for symmetric operator induced by radially symmetric PSFs are promising, the next step is to develop an extension of this method appropriate for non-symmetric problems. Furthermore, it is often the case that astronomers have specific information about a small part of the of the sky, such as where one should definitely see a star or where there should be no astronomical phenomena. It would be advantageous to use this information in the reconstruction of the image and the blurring matrix.

Acknowledgments. The authors would like to thank Alessandro Buccini for providing the blind deconvolution code associated to the paper [2] and for both providing the correct parameters for running experiments properly and further running additional experiments for us to use for comparison. The authors also would like to thank referees for comments. Research by LR was supported in part by NSF grants DMS-1720259 and DMS-1729509.

REFERENCES

[1] RestoreTools: An Object Oriented Matlab Package for Image Restoration, 2012. `http://www.mathcs.emory.edu/~nagy/RestoreTools`.

[2] A. Buccini, M. Donatelli, and R. Ramlau, *A semiblind regularization algorithm for inverse problems with application to image deblurring*, SIAM J. Sci. Comput., 40 (2018), pp. A452–A483.
[3] P. Campisi and K. Egiazarian, eds., Blind Deconvolution: Theory and Applications, CRC Press, Boca Raton, 2007.
[4] T. F. Chan and C. K. Wong, *Total variation blind deconvolution*, IEEE Trans. Image Process., 7 (1998), pp. 370–375.
[5] T. F. Chan and C. K. Wong, *Convergence of the alternating minimization algorithm for blind deconvolution*, Linear Algebra Appl., 316 (2000), pp. 259–285.
[6] J.-M. Conan, L. M. Mugnier, T. Fusco, V. Michau, and G. Rousset, *Myopic deconvolution of adaptive optics images by use of object and point-spread function power spectra*, Appl. Optics, 37 (1998), pp. 4614–4622.
[7] A. Cornelio, E. Loli Piccolomini, and J. G. Nagy, Constrained numerical optimization methods for blind deconvolution, Numer. Algorithms, 65 (2014), pp. 23–42.
[8] G. Desiderá, B. Anconelli, M. Bertero, P. Boccacci, and M. Carbillet, *Application of iterative blind deconvolution to the reconstruction of LBT LINC-NIRVANA images*, Astron. Astrophysics, 452 (2006), pp. 727–734.
[9] B. L. Ellerbroek and C. R. Vogel, *Inverse problems in astronomical adaptive optics*, Inverse Problems, 25 (2009), 063001 (37pp).
[10] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996
[11] J. Exposito, D. Gratadour, G. Rousset, Y. Clénet, E. Gendron, and L. Mugnier, *A new method for adaptive optics point spread function reconstruction*, Third AO4ELT Conference - Adaptive Optics for Extremely Large Telescopes, 2013. `http://ao4elt3.arcetri.astro.it/proceedings/fulltext_13328.pdf`, last accessed on 13 June, 2017.
[12] K. Gallivan, M. Heath, E. Ng, B. Peyton, R. Plemmons, J. Ortega, C. Romine, A. Sameh, and R. Voigt, Parallel Algorithms for Matrix Computations, SIAM, Philadelphia, 1990.
[13] E. Gendron, Y. Clénet, T. Fusco, and G. Rousset, *New algorithms for adaptive optics point-spread function reconstruction*, Astronomy and Astrophysics, 457 (2016), pp. 359–463.
[14] L. Gilles, C. Correia, J.-P. Véran, L. Wang, and B. Ellerbroek, *Simulation model based approach for long exposure atmospheric point spread function reconstruction for laser guide star multiconjugate adaptive optics*, Appl. Optics, 51 (2012), pp. 7443–7458.
[15] G. H. Golub and C. F. Van Loan, Matrix Computations, 4th ed., Johns Hopkins University Press, Baltimore, 2013.
[16] P. C. Hansen, J. G. Nagy, and D. P. O'Leary, *Deblurring Images: Matrices, Spectra, and Filtering*, SIAM, Philadelphia, 2006
[17] T. A. Hearn and L. Reichel, *Extensions of the Justen–Ramlau blind deconvolution method*, Adv. Comput. Math., 39 (2013), pp. 465–491.
[18] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, Englewood Cliffs, 1989.
[19] L. Jolissaint, C. Neyman, J. Christou, P. Wizinowich, and L. Mugnier, *First successful adaptive optics PSF reconstruction at W. M. Keck observatory*, Proc. SPIE 8447, Adaptive Optics Systems III (2012), 844728–844728–12.
[20] L. Justen and R. Ramlau, *A non-iterative regularization approach to blind deconvolution*, Inverse Problems, 22 (2006), pp. 771–800.
[21] L. Justen and R. Ramlau, *A general framework for soft-shrinkage with applications to blind deconvolution and wavelet denoising*, Appl. Comput. Harmon. Anal., 26 (2009), pp. 43–63.
[22] M. Le Louarn, C. Vérinaud, V. Korkiakoski, N. Hubin, and E. Marchetti, *Adaptive optics simulations for the European Extremely Large Telescope*, Proc. SPIE 6272, Advances in Adaptive Optics II, 2006, 627239–627239-p.
[23] B. Lewis and L. Reichel, *Arnoldi–Tikhonov regularization methods*, J. Comput. Appl. Math., 226 (2009), pp. 92–102.
[24] O. A. Martin, C. M. Correia, E. Gendron, G. Rousset, D. Gratadour, F. Vidal, T. J. Morris, A. G. Basden, R. M. Myers, B. Neichel, and T. Fusco, *PSF reconstruction validated using on-sky CANARY data in MOAO mode*, Proc. SPIE 9909, Adaptive Optics Systems V, 2016, pp. 091Q–13.
[25] J. G. Nagy, K. M. Palmer, and L. Perrone, *Iterative methods for image deblurring: A Matlab object oriented approach*, Numer, Algorithms, 36 (2004), pp. 73-–93.
[26] F. Roddier, *Adaptive Optics in Astronomy*, Cambridge, Cambridge University Press, 1999.
[27] M. Rudnaya, *Automated Focusing and Astigmatism Correction in Electronic Microscopy*, Ph.D. thesis, Department of Applied Mathematics, Technical University of Eindhoven, Eindhoven, The Netherlands, 2011.
[28] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, 2nd ed., SIAM, Philadelphia,

2011.

[29] G. Spaletta and L. Caucci, *Constrained iterations for blind deconvolution and convexity issues*, J. Comput. Appl. Math., 197 (2006), pp. 29–43.

[30] J.-P. Véran, F. Rigaut, H. Maître, and D. Rouan, *Estimation of the adaptive optics long exposure point spread function using control loop data*, J. Opt. Soc. Am. A, 14 (1997), pp. 3057–3069.