Fast Optimization Schemes for Phase Recovery in Bispectral Imaging

James L. Herring

Mathematics and Computer Science
Emory University
Atlanta, GA 30322, USA
jlherri@mathcs.emory.edu

James G. Nagy

Mathematics and Computer Science
Emory University
Atlanta, GA 30322, USA
nagy@mathcs.emory.edu

ABSTRACT

Speckle interferometry is a common method used to obtain astronomical images using ground-based telescopes to image through a turbulent atmosphere-telescope system. However, when imaging more complicated astronomical objects such as satellites, speckle interferometry methods necessitate the separate recovery of the object's Fourier phase to obtain more detailed images. Bispectral methods are one approach to solving this complementary problem of phase recovery in speckle interferometry. They retrieve an object's Fourier phase by matching it to the object's bispectrum, a collectable statistical quantity from the speckle data. Mathematically, phase retrieval from the bispectrum can be formulated as a large-scale, non-linear least-squares problem. We consider several optimization schemes from the literature for solving this phase retrieval problem. In particular, we focus on accelerating the speed and convergence of this optimization while maximizing the quality of the recovered image through efficient implementation, Hessian based optimization, and appropriate regularization.

1. INTRODUCTION

Astronomers have developed a number of techniques to solve the problem of obtaining high-resolution astronomical images in the visual spectrum from ground-based telescopes. Due to the faintness of many such astronomical objects, particular interest is paid to techniques that can produce high-resolution images from photon-limited data. One common method for obtaining high-resolution images under such circumstances is speckle interferometry, which was initiated by Labeyrie's observation in 1970 that high spatial frequency information could be recovered from low-light, short-exposure images [4]. His method uses two sets of short-exposure image frames, one set of the desired object and the other of a reference star, to obtain a single, composite reconstruction of an object's energy spectrum (or Fourier modulus.) However, while the object's recovered energy spectrum contains sufficient information about the object to produce images for many simple astronomical objects, it proves insufficient for obtaining high-resolution images for more complicated objects. In the case of a more complicated object, obtaining a high-resolution image also necessitates the reconstruction of the object's Fourier phase. Several methods have been proposed for this phase reconstruction, most utilizing an object's triple correlation, its bispectrum, or some subset of these high-order statistical correlation measures which are obtainable from the data [3, 9].

The goal of this paper is to review and expand on several of the algorithms presented in the literature for phase recovery using an object's bispectrum. In particular, we focus on the mathematical formulations of these phase retrieval algorithms as large-scale, non-linear least-squares problems and approach them using an optimization framework. Areas of interest include efficient implementation, the potential for

Hessian or approximate-Hessian based optimization, and appropriate regularization to improve the quality of the resulting images.

The paper is organized in the following sections. Section 2 provides a brief description of the speckle interferometry problem and the efficient collection of the object's bispectrum. Note that this paper only aims to cover this setup in sufficient depth to present the subsequent bispectrum recovery problem in a cogent manner. For a more extensive discussion of the speckle interferometry problem, we recommend the articles listed in the bibliography, particularly the paper of Negrete-Regagnon [7]. Section 3 introduces the problem of phase recovery from an object's bispectrum including the objective functions, gradients, and (approximate) Hessians used for the optimization schemes. Section 4 provides some numerical results for optimization, and the final sections offer some concluding remarks and acknowledgements.

2. PROBLEM SETUP

Calculating the Object Power Spectrum

Before one can consider Fourier phase recover using bispectral methods, one must approach the primary problem of speckle interferometry: recovery of the object's Fourier modulus or power spectrum. To do this, we start from the model for describing the blurring of an object $f(x, y)$ by an atmosphere-telescope system resulting in an observed image $i(x, y)$. For a single short-exposure observation, this blurring is expressed in terms of the convolution operator

$$i(x, y) = \iint_{-\infty}^{\infty} f(x', y')h(x' - x, y' - y)dx'dy' \tag{1}$$

where $h(x, y)$ is the point spread function (PSF) that models the blurring of a given point of the object as a function of the atmosphere-telescope system. In Fourier space, the convolution property of the Fourier transform allows the equation above to be expressed by

$$I(u, v) = F(u, v)H(u, v) \tag{2}$$

where the multiplication is taken component-wise in two dimensions. Here, $I(u, v)$, $F(u, v)$, and $H(u, v)$ denote the two dimensional Fourier transforms of their lower-case counterparts in Eq. 1. We refer to $H(u, v)$ as the optical transfer function (OTF), which can be expressed as a product of fixed aberrations due to telescope optics and random aberrations due to atmospheric turbulence. A more complete discussion of the properties and accurate simulation of $H(u, v)$ can be found in the paper by Negrete-Regagnon and the references therein [7].

The equations above describe a single instance of the blurring of an object, resulting in a single short-exposure image frame. In practice when dealing with photon-limited imaging, multiple frames are needed to provide sufficient data to recreate a single, high-resolution composite image. Following this lead, if we assume a set of N short-exposure images and denote the Fourier transform of the kth image by $I_k(x, y)$, then the ensemble average* of the N resulting realizations of Eq. 2 is given by

$$\langle I_k(u, v)\rangle = F(u, v)\langle H_k(u, v)\rangle \tag{3}$$

This approach is equivalent to producing the Fourier spectrum of a single long-exposure image, but has the drawback of suppressing desirable high-frequency information due to the "averaging out" effect of the ensemble-average operation. In order to avoid this drawback, Labeyrie proposed taking the ensemble average of the modulus squared of the Fourier transform of the data [4]

*Define ensemble average by $\langle I_k(u, v)\rangle = \frac{1}{N}\sum_{k=1}^{N} I_k(u, v)$

$$\langle |I_k(u,v)|^2 \rangle = |F(u,v)|^2 \langle |H_k(u,v)|^2 \rangle \tag{4}$$

Here, the modulus squared operation produces strictly non-negative values for the ensemble average which allows for the retention of high-frequency information about the Fourier spectrum of the object. It follows then that simple component-wise operations provide an expression for the recovery of the Fourier modulus of the object

$$|F(u,v)| = \left[\frac{\langle |I_k(u,v)|^2 \rangle}{\langle |H_k(u,v)|^2 \rangle + \epsilon} \right]^{\frac{1}{2}} \tag{5}$$

We make several remarks about the above expression. First, the ϵ in the denominator is a small parameter added to prevent division by zero. Also, we note that the modulus squared OTF, $|H(u,v)|^2$, is commonly called the modular transfer function (MTF) and in practice is unknown. Instead, it is replaced by the ensemble average of the Fourier modulus squared of an appropriate reference star, the second set of short-exposure image data collected for speckle interferometry. If we denote the kth frame of this star data by $s_k(x,y)$ and its Fourier transform by $S_k(u,v)$, we have

$$\langle |S_k(u,v)|^2 \rangle \approx \langle |H(u,v)|^2 \rangle \tag{6}$$

This substitution works because the reference star acts as a point source, and the data should be collected concurrently with the object data in order to ensure the same astronomical seeing conditions [7]. Effecting this substitution in Eq. 5, we can calculate the object's Fourier modulus, or power spectrum, of the object being imaged by

$$|F(u,v)| = \left[\frac{\langle |I_k(u,v)|^2 \rangle}{\langle |S_k(u,v)|^2 \rangle + \epsilon} \right]^{\frac{1}{2}} \tag{7}$$

Once the above expression has been calculated, a further windowing procedure must be applied in order to prevent the computed power spectrum from being overwhelmed by noise in the data. Details for this can be found in Section 7 of Negrete-Regagnon [7].

The above provides a short summary of both the impetus and the means to calculate an object's power spectrum. A few other remarks are worth noting. First, an object's power spectrum provides the most significant information relevant to accurate image recovery using speckle interferometry. It is the primary problem in the sense that without an accurate recovery of the object's power spectrum, recovering its Fourier phase will not allow for a good resultant image. Related to this, the ability to recover the object's power spectrum accurately is dependent on the data signal to noise ratio (SNR) which in turn depends on atmospheric conditions, light levels of the data (or photoevents per data frame), and optics setup among other things. For a more complete discussion of these factors, we again recommend Negrete-Regagnon and its references [7].

Accumulating the Object Bispectrum

Having outlined a method for obtaining an object's power spectrum using speckle interferometry, we now turn to the problem recovering the object's Fourier phase. Most techniques for this focus on the use of correlation techniques, with much of the literature focusing on the use of the object's triple correlation and its Fourier transform, the bispectrum [3, 9]. An object's triple correlation measures the correlation of the object against two independently shifted copies of itself in the spatial domain. In two dimensions, this is expressed by

$$f^{TC}(x_1,y_1,x_2,y_2) = \iint_{-\infty}^{\infty} f^*(x,y) f(x+x_1, y+y_1) f(x+x_2, y+y_2) dx dy \tag{8}$$

Taking a Fourier transform of the equation above and using the convolution property, we get the object's bispectrum

$$\begin{aligned} F^{(3)}(u_1, v_1, u_2, v_2) &= F(u_1, v_1)F(u_2, v_2)F^*(u_1 + u_2, v_1 + v_2) \\ &= |F^{(3)}(u_1, v_1, u_2, v_2)| \exp[i\beta(u_1, v_1, u_2, v_2)] \end{aligned} \tag{9}$$

Noting that we already have a method for obtaining the object's Fourier modulus from the previous section, we shift our focus to the exponential term of the bispectrum and its angle, which we call the phase, $\beta(u_1, v_1, u_2, v_2)$.

First, we want to establish the relationship between the phase of the bispectrum and the object's phase that we want to recover, ϕ. It has been shown that similarly to Eq. 4 we can relate the collected ensemble bispectrum of the data to the bispectrum of the object by

$$\langle I_k^{(3)}(u_1, v_1, u_2, v_2)\rangle = F^{(3)}(u_1, v_1, u_2, v_2)\langle H_k^{(3)}(u_1, v_1, u_2, v_2)\rangle \tag{10}$$

Further analysis by Lohmann *et al.* has shown that the bispectral transfer function, $\langle H_k^{(3)}(u_1, v_1, u_2, v_2)\rangle$ has an expected real value, meaning that its phase is effectively zero [5]. This then gives a direct relationship between the phase of the data's bispectrum, β, and the phase of the object that is to be recovered

$$\begin{aligned} \beta(u_1, v_1, u_2, v_2) &= \phi(u_1, v_1) + \phi(u_2, v_2) - \phi(u_1 + u_2, v_1 + v_2) \\ \beta(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}}) &= \phi(\overrightarrow{\boldsymbol{u}}) + \phi(\overrightarrow{\boldsymbol{v}}) - \phi(\overrightarrow{\boldsymbol{u} + \boldsymbol{v}}) \end{aligned} \tag{11}$$

Here, the second expression is identical to the first but substitutes in the vector notation $\overrightarrow{\boldsymbol{u}} = (u_1, v_1)$ and $\overrightarrow{\boldsymbol{v}} = (u_2, v_2)$. This convention will be used for the remainder of the paper to simplify indexing. This equation provides a direct relation between the elements of the phase of the data bispectrum and the phase of the object, and it serves as the starting point for all of the phase recovery algorithms in Section 3.

However, in order to use Eq. 11 to recover the object phase, one must first be able to efficiently accumulate the phase of the collected ensemble bispectrum of the data, $\beta(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}})$. To do this, we again look at the exponential term in Eq. 9. Denoting by ψ the phase of a single data frame $I_k(u, v)$, the exponential part of the equation can be written as

$$\exp[i\beta(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}})] = \exp[i\psi(\overrightarrow{\boldsymbol{u}})] \exp[i\psi(\overrightarrow{\boldsymbol{v}})] \exp[-i\psi(\overrightarrow{\boldsymbol{u} + \boldsymbol{v}})] \tag{12}$$

Taken over the set of N data frames, the phase of the bispectrum then becomes

$$\beta(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}}) = \text{angle}\left(\sum_{k=1}^{N} \exp[i\psi_k(\overrightarrow{\boldsymbol{u}})] \exp[i\psi_k(\overrightarrow{\boldsymbol{v}})] \exp[-i\psi_k(\overrightarrow{\boldsymbol{u} + \boldsymbol{v}})]\right) \tag{13}$$

From this, one can see that efficiently computing the object's bispectrum becomes a question of identifying the $(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}}, \overrightarrow{\boldsymbol{u} + \boldsymbol{v}})$ triplets necessary to evaluate the right hand side for each data frame (the triplets remain constant across all frames.) The most efficient method of doing this is by means of an indexing structure that saves the indices of these triplets and vectorizes computation of the bispectrum accumulation. Additional efficiency can be attained by exploiting the symmetries in the phase of real images in Fourier space. A description of the logic behind such an indexing structure can be found in work by Tyler and Schulze [8].

We note that even with the efficiency afforded by vectorization and symmetry available when using an indexing structure, the number of bispectrum elements becomes infeasible if $\overrightarrow{\boldsymbol{u}}$ and $\overrightarrow{\boldsymbol{v}}$ span over the whole image. In practice, we restrict $\overrightarrow{\boldsymbol{v}}$ to a set within some small radius and $\overrightarrow{\boldsymbol{u}}$ and $\overrightarrow{\boldsymbol{u}+\boldsymbol{v}}$ to within a larger radius. Fortunately, this choice is not just a computational compromise, but is driven by considerations inherent in the problem. Both radii are sensitive to both astronomical seeing conditions and telescope optics and should be chosen with care [8]. A good choice has the effect not only of ensuring the collected bispectrum has a good signal to noise ratio but also has the practical benefit of making the number of bispectrum elements computationally feasible for the phase recovery problem.

3. PHASE RECOVERY FROM THE BISPECTRUM

Objective Functions

Once the data bispectrum has been collected, we are prepared to approach the problem of recovering the object's phase. All the methods for the phase recovery are based on the relationship provided in Eq. 11. Considering this relationship across all bispectrum phase elements, we can reformulate the expression in matrix-vector form

$$\boldsymbol{\beta} = A\boldsymbol{\phi} \tag{14}$$

Here, $\boldsymbol{\beta}$ is an $m\times 1$ vector of all the accumulated bispectral elements corresponding to distinct $(\overrightarrow{\boldsymbol{u}}, \overrightarrow{\boldsymbol{v}}, \overrightarrow{\boldsymbol{u}+\boldsymbol{v}})$ triplets, $\boldsymbol{\phi}$ is the $n\times 1$ dimensional unknown object phase, and A is an $m\times n$ sparse matrix (with $m \gg n$) with three non-zeros entries per row, two 1's and one -1 corresponding the sign of the phase elements in Eq. 11.

From this expression, fitting the phase to the data is formulated as a large-scale least squares problem. However, linearity is lost due to the fact that the collected bispectrum is only known modulo-2π (or wrapped.) One idea then is to unwrap the phase to obtain a linear least-squares problem, as proposed by Marron *et al.* [6]. Another option is to solve the problem as a non-linear least-squares problem. Multiple schemes for solving the problem in its non-linear form have been proposed. We adopt this framework. The first scheme we look at was proposed by Haniff [2]

$$\min_{\boldsymbol{\phi}} E_1(\boldsymbol{\phi}) = \min_{\boldsymbol{\phi}} \frac{1}{2}\sum_{j=1}^{m} \left(\mathrm{mod}_{2\pi}(\beta_j - [A\boldsymbol{\phi}]_j)\right)^2 w_{jj} \tag{15}$$

where w_{jj} corresponds to the signal-to-noise ratio of the jth collected bispectrum phase element [7]. If we define W to be a diagonal matrix with entries w_{jj} and differentiate with respect to the phase, we then get the gradient and an approximate Hessian

$$\begin{aligned} \nabla_{\boldsymbol{\phi}} E_1 &= -A^T W[\mathrm{mod}_{2\pi}(\boldsymbol{\beta} - A\boldsymbol{\phi})] \\ \nabla^2_{\boldsymbol{\phi}} E_1 &= A^T W A \end{aligned} \tag{16}$$

We remark here that the introduction of the modulo-2π introduces a number of considerations for the problem. First, the modulo causes a loss of convexity as there is a periodic local minima for each phase element every 2π. This makes optimization methods more likely to be caught in one of these local minima and more sensitive to an intelligent starting guess. Furthermore, $E_1(\boldsymbol{\phi})$ is non-differentiable at the many periodic jump-continuities where the modulo-2π misfit wraps from 0 to 2π. In the derivatives above, we have simply ignored the modulus during differentiation, an idea which is used in the literature and which in practice has proven to be effective. However, it may introduce issues during the optimization.

To avoid the issue of non-differentiability, Haniff's paper also proposes an alternative non-linear least squares scheme that minimizes over the misfit in the object's phasor [2]†. The misfit function for this scheme is given by

$$\min_{\boldsymbol{\phi}} E_2(\boldsymbol{\phi}) = \min_{\boldsymbol{\phi}} \frac{1}{2} \sum_{j=1}^{m} \left(\left[\cos(\beta_j) - \cos([A\boldsymbol{\phi}]_j)\right]^2 + \left[\sin(\beta_j) - \sin([A\boldsymbol{\phi}]_j)\right]^2 \right) w_{jj} \tag{17}$$

where the weights w_{jj} are defined identically as above. Again, setting $W = \mathrm{diag}(w_{jj})$ and differentiating with respect to the phase, we derive expressions for the gradient and approximate Hessian

$$\begin{aligned} \nabla_{\boldsymbol{\phi}} E_2 &= A^T W \left[\cos(\boldsymbol{\beta}) \odot \sin(A\boldsymbol{\phi}) - \sin(\boldsymbol{\beta}) \odot \cos(A\boldsymbol{\phi})\right] \\ \nabla^2_{\boldsymbol{\phi}} E_2 &= A^T W D A \end{aligned} \tag{18}$$

Here, $\odot$ denotes the Hadamard product or component-wise multiplication of vectors and D is a diagonal matrix defined by $D = \mathrm{diag}(\cos(\boldsymbol{\beta}) \odot \cos(A\boldsymbol{\phi}) + \sin(\boldsymbol{\beta}) \odot \sin(A\boldsymbol{\phi}))$. Note that this formulation still is non-convex with periodic local minima every 2π, but it is differentiable everywhere.

In both formulations of the phase-matching problem above, the gradients and Hessians are calculated by differentiating with respect to the phase, $\boldsymbol{\phi}$. However, the final goal of the optimization is to combine the matched phase with the object's calculated power spectrum to obtain the best quality image possible from the data. Thus, it would be useful if the least-square optimization for matching the phase took into account the resulting image. To this end, Glindemann and Dainty proposed a different approach to Haniff's first objective function, $E_1(\boldsymbol{\phi})$. Rather than differentiating with respect to the phase, they proposed considering the phase retrieval as a function of the resultant image, $E_1(\boldsymbol{f})$ [1]‡. To do this one must also differentiate with respect to the resultant image. This can be done as an extension to Eq. 16 using the chain rule to differentiate the phase with respect to the image. The gradient and Hessian resulting from this approach are respectively,

$$\begin{aligned} \nabla_{\boldsymbol{f}} E_1 &= -\frac{\partial \phi^*}{\partial f} A^T W [\mathrm{mod}_{2\pi}(\boldsymbol{\beta} - A\boldsymbol{\phi})] \\ \nabla^2_{\boldsymbol{f}} E_1 &= \frac{\partial \phi^*}{\partial f} A^T W A \frac{\partial \phi}{\partial f} \end{aligned} \tag{19}$$

Here, the gradient is nearly the same as Eq. 16 with the exception of the additional $\frac{\partial \phi}{\partial f}$ operator. An expression for this operator and its adjoint, $\frac{\partial \phi^*}{\partial f}$, can be found in the Appendix.

Using this framework, we can also differentiate Haniff's second objective function, E_2, using Glindemann's idea. To our knowledge, this formulation for the minimization has not been explored in the literature and represents a new extension on the ideas above. The gradient and Hessian of $E_2(\boldsymbol{f})$ with respect to the resultant image are given by

$$\begin{aligned} \nabla_{\boldsymbol{f}} E_2 &= \frac{\partial \phi^*}{\partial f} A^T W \left[\cos(\boldsymbol{\beta}) \odot \sin(A\boldsymbol{\phi}) - \sin(\boldsymbol{\beta}) \odot \cos(A\boldsymbol{\phi})\right] \\ \nabla^2_{\boldsymbol{f}} E_2 &= \frac{\partial \phi^*}{\partial f} A^T W D A \frac{\partial \phi}{\partial f} \end{aligned} \tag{20}$$

†By phasor, we denote the normalized complex exponential of a phase element i.e. a phase ϕ corresponds to the phasor $\exp(i\phi) = \cos(\phi) + i\sin(\phi)$

‡The resultant image can be seen as $\boldsymbol{f} = \mathcal{F}^{-1}\left(|F(u,v)| \odot \exp[i\boldsymbol{\phi}]\right)$ where $\mathcal{F}^{-1}$ is an inverse 2D Fourier transform and $F(u,v)$ is the object's calculated Fourier modulus.

Differentiating both objective functions with respect to the resultant image has a number of possible advantages. First, we remarked above on the non-convexity of both least-squares formulations and the presence of many local minima. Differentiating with respect the resultant image has the potential to drive convergence to different minima, potentially resulting in improved image quality. Furthermore, differentiating with respect to the image rather than the phase presents the prospect for regularization with respect to the resultant image. This would allow for the enforcement of certain desirable image traits like non-negativity. With this in mind, we look at a potential regularizer.

Regularization

We look at a regularization scheme proposed by Glindemann and Dainty for enforcing non-negativity [1]. This regularization is used with $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ when differentiated with respect to the resultant image. The regularization introduces a penalty term to discourage the presence of negative pixel values in the resultant image and is written as

$$E_+(\boldsymbol{f}) = \frac{\alpha}{2} \sum_{\gamma} |\boldsymbol{f}(\gamma)|^2 \tag{21}$$

where γ is the set of indices corresponding to negative image values at the current iteration. The weighting parameter α should be chose so as to drive the solution to positive values while still allowing the optimization to effectively match the object's phase to the data bispectrum. Some discussion for the selection of the regularization parameter will be included in Sec. 4. Next, we must differentiate the regularizer; the element-wise first and second derivative are given then by

$$\begin{aligned} \frac{\partial E_+}{\partial f_j} &= \begin{cases} \alpha f_j & \text{if } j \in \gamma \\ 0 & \text{else} \end{cases} \\ \frac{\partial^2 E_+}{\partial f_j^2} &= \begin{cases} \alpha & \text{if } j \in \gamma \\ 0 & \text{else} \end{cases} \end{aligned} \tag{22}$$

These can then be put into vector and sparse, diagonal matrix form for the gradient and Hessian, respectively. Summing them with the gradient and Hessian from Eqs. 19 and 20 then follows.

4. NUMERICAL RESULTS

To test and compare the objective functions introduced in the previous section, we first simulated the necessary sets of speckle interferometry data. We generated 50 frames of short exposure data for both the object and the reference star with Fried parameter $D/r_0 = 30$, using different seeds for the random number generator to ensure independence of the randomness of the two data sets. The object data was scaled to include $3e6$ photo-events per frame and zero-mean Gaussian noise with standard deviation $\sigma_{rn} = 5$ was added. The star data was scaled to 5000 photo-events per frame.

From this data, we collected the object's power spectrum and phase of the data bispectrum. Before testing the various optimization schemes, we note once again that the objective functions $E_1(\boldsymbol{\phi})$, $E_2(\boldsymbol{\phi})$, $E_1(\boldsymbol{f})$, and $E_2(\boldsymbol{f})$ all have frequent local minima. As a consequence, the minimization of both objective functions is highly dependent on a good initial guess for the phase. To generate this guess, we used the recursive phase recovery scheme proposed by Tyler [8].

Using the initial phase collected via the recursive algorithm, we then tested gradient descent, nonlinear conjugate gradient (NCLG), and Gauss-Newton methods for minimizing each of the four schemes proposed in the previous section. Gradient descent is included as a benchmark while NLCG has been preferred in previous implementations due to reluctance to implement the large linear system solves necessary for Newton-based optimization [7]. Newton-based methods pose the challenge of solving a

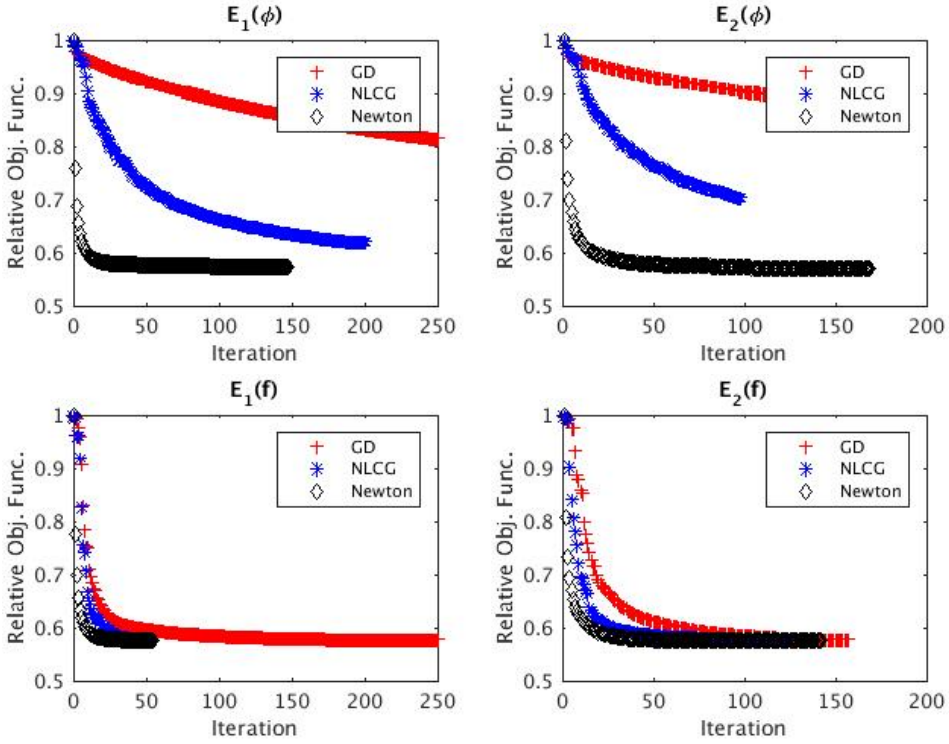

Figure 1: The subplots show the convergence of the proposed optimization strategies for matching the object phase $\boldsymbol{\phi}$ to the phase of the data bispectrum for 50 frames and $D/r_0 = 30$ with stopping tolerance $\text{tol} = 10^{-6}$. Subplots $(1,1)$ and $(1,2)$ show, respectively, the results for the objective functions $E_1(\boldsymbol{\phi})$ and $E_2(\boldsymbol{\phi})$ optimized with respect to the phase, while subplots $(2,1)$ and $(2,2)$ show the results for $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ when differentiated with respect to the resultant image. Note that all methods converge to approximately equal levels, but the Hessian-based methods reach their minima more quickly than the gradient-based methods. However, this effect is not as apparent when differentiating with respect the resultant image in subplots on the bottom row.

linear system with the Hessian when choosing a descent direction, but offer the promise of improved convergence. Thus, we consider whether the Hessian solve can be made efficient enough to offset the additional cost associated with the method.

Several choices were made for methods parameters. For each optimization method, we used a backtracking Armijo line search for the step length parameter. We note that this does not necessarily satisfy the strong Wolfe conditions necessary for a descent direction in NLCG, but in practice it has proved sufficient. The results of using each minimization scheme on each of our objective functions can be seen in Fig. 1. For stopping criteria, we used the relative change in the objective function per iteration. That is, if the objective value failed to decrease sufficiently by a given tolerance at each iteration, the method was stopped.

From Fig. 1, we can see that the methods perform as expected in terms of convergence: NLCG outperformed gradient descent and Gauss-Newton was superior to both in terms of convergence per iteration. These results were more pronounced in for $E_1(\boldsymbol{\phi})$ and $E_2(\boldsymbol{\phi})$ than in $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$, but the trend remains. However, the results from the figure do not give an indication of the total cost of the iterations in terms of time and computation. If the cost of the Hessian solve in the Gauss-Newton method is too costly, the improvement in convergence per iteration may be offset by the expense of each iteration itself. Thus, we must discuss the efficiency of the Hessian solve and per iteration cost of each

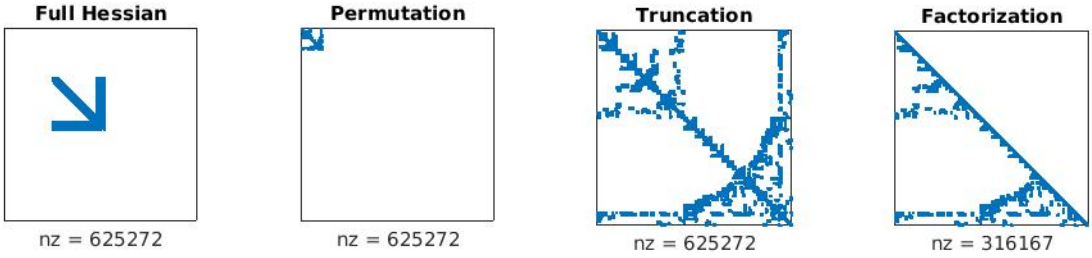

Figure 2: This shows the non-zero values for the Hessian $A^T W A$ and its treatment for the objective functions associated with the Hessians in Eqs. 16 & 18. The Fourier radius is 64 and second radius of 5 for a 256×256 image. These radii and considerations for symmetry limit the number of phase values to be recovered, making the matrix highly singular and sparse. The full Hessian with dimension $256^2 \times 256^2$ can be seen in the first subplot. A symmetric minimum degree permutation shifts the non-zeros toward the upper left of the matrix in subplot two. In the third subplot, we extract the non-singular sub-matrix in the upper left corner, extracting a truncated Hessian (dimension 7062×7062 for these radii.) Finally, we compute an incomplete Cholesky factorization of this truncated Hessian in subplot four, which preserves the sparsity pattern and limits fill in.

method before making any comment on the preferability of one method to another.

Handling the Hessian Solve and per Iteration Cost

To approach the Hessian solve for each iteration of Gauss-Newton using $E_1(\boldsymbol{\phi})$ and $E_2(\boldsymbol{\phi})$, we first note that the Hessian for $E_1(\boldsymbol{\phi})$ in Eq. 16 is independent of the $\boldsymbol{\phi}$ making it constant for all iterations of the optimization. As such, factorization for a direct, sparse solver is an option because any factorization can be computed once and stored for all iterations of the method. We also recall that due to signal to noise ratio considerations and the symmetries of real images in Fourier space, the number of phase elements $\tilde{n}^2$ to be recovered for an $n \times n$ image is significantly less than n^2 and exists within some radius on half of the Fourier plane. This means that the Hessian is highly rank deficient. Lastly, we note for $E_1(\boldsymbol{\phi})$, the Hessian is symmetric positive semi-definite because W has positive entries. Here, the semi-definiteness is a direct result of the rank deficiency.

Bearing these traits in mind, we implemented the following approach for the Hessian of $E_1(\boldsymbol{\phi})$. First, we compute and store a symmetric approximate minimum degree permutation to shift the non-zero rows and columns of the Hessian to the upper upper left-hand corner of the matrix. Next, we extract the resultant sub-matrix corresponding to the non-zero part of the Hessian. This sub-matrix has dimension $\tilde{n}^2 \times \tilde{n}^2$ and is much smaller than the full Hessian. It is also symmetric positive definite. As such, we can compute the Cholesky factorization or incomplete Cholesky factorization of the sub-matrix once and store the factors for all subsequent iterations of the method. This reduces the Hessian solve for each iteration from a potentially $\mathcal{O}(n^3)$ operation to a permutation followed by two smaller $\mathcal{O}(\tilde{n}^2)$ triangular systems, while the factorization is only done once. Additionally, it ensures that the non-zero components of the direction in the optimization correspond to the phases of elements in the Fourier plane which we wish to recover. In practice, we tend to use the incomplete Cholesky factorization because it minimizes fill in and maintains the sparsity pattern which speeds up solving the triangular system while delivering similar results in the optimization. Fig. 2 shows the transition for the permutation, truncation, and factorization of the Hessian matrix using MATLAB's `spy.m` function.

For the Hessian of $E_2(\boldsymbol{\phi})$, the strategy used for the Hessian of $E_1(\boldsymbol{\phi})$ presents a problem: the matrix D in the Hessian in Eq. 18 is dependent on $\boldsymbol{\phi}$. Also, D may contain non-positive entries causing the extracted permuted matrix to lose both positive definiteness and positive semi-definiteness, thus preventing the Cholesky factorization. To avoid these complications, we omit the matrix D in practice, making the Hessian for $E_2(\boldsymbol{\phi})$ equal to that of $E_1(\boldsymbol{\phi})$. This then allows us to follow the strategy used

$E_1(\boldsymbol{\phi})$			
	Gradient Descent	NLCG	Gauss-Newton
LS/Iter.	14.5	15.5	1.0
Sec./Iter.	0.58	0.62	0.12
$E_2(\boldsymbol{\phi})$			
	Gradient Descent	NLCG	Gauss-Newton
LS/Iter.	14.4	14.9	1.0
Sec./Iter.	0.86	0.89	0.16
$E_1(\boldsymbol{f})$			
	Gradient Descent	NLCG	Gauss-Newton
LS/Iter.	15.4	14.0	1.0
Sec./Iter.	0.84	0.75	11.06
$E_2(\boldsymbol{f})$			
	Gradient Descent	NLCG	Gauss-Newton
LS/Iter.	15.6	14.9	1.0
Sec./Iter.	1.17	1.12	6.75

Table 1: This table shows the average number of function calls in the line search per iteration and the average CPU time per iteration for gradient descent, non-linear conjugate gradient, and Gauss-Newton optimization for each of the four objective functions: $E_1(\boldsymbol{\phi})$, $E_2(\boldsymbol{\phi})$, $E_1(\boldsymbol{f})$, and $E_2(\boldsymbol{f})$.

above. In practice, this strategy has also delivered comparable convergence while avoiding updates to the factorization and loss of positive definiteness in the permuted sub-matrix (see Fig. 1.)

The Hessian solves for objective functions $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ are more complicated due to the $\frac{\partial \phi}{\partial \boldsymbol{f}}$ operator and its adjoint. These operators are dependent on the resultant image $\boldsymbol{f}$ and must be updated at each iteration. Additionally, the Fourier transforms present in these operators cause a loss of sparsity, meaning that the Hessians for both $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ are full matrices of dimension $n \times n$. This makes factorization and direct solvers infeasible. However, multiplication with the Hessian can be done efficiently, so instead we use MATLAB's `pcg` solver with a low tolerance ($\approx 10^{-1}$) to determine the search direction in the Hessian solve.

For $E_2(\boldsymbol{f})$ as for $E_2(\boldsymbol{\phi})$, we typically omitted D to ensure that the Hessian operator is symmetric positive definite, and we use the identical strategy to $E_1(\boldsymbol{f})$, with comparable results.

In Table 1, we can see a comparison of the average number of function calls in the line search and the average CPU time per iteration for three optimization methods on each of the objective functions. For $E_1(\boldsymbol{\phi})$ and $E_2(\boldsymbol{\phi})$, the Gauss-Newton scheme results in a significant speed up of the method. It converges in fewer iterations, and each iteration is faster than either gradient descent or NLCG due to a reduction function calls in the line search and the efficiency of the Hessian solve due to the strategy previously detailed. For $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$, the situation is less advantageous. The Gauss-Newton method converges in fewer iterations and displays the same advantages in the line search, but the cost of the iterative solver for the Hessian system is prohibitive. In order to improve the disparity in CPU times using Gauss-Newton in these cases, we are searching for a suitable preconditioner to reduce the number of `pcg` iterations. This represents future work.

Choosing a Regularization Parameter

Another important decision affecting the performance of $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ is the selection of a regularization parameter α for enforcing non-negativity when using $E_+(\boldsymbol{f})$. This is complicated by the non-linearity of our objective functions, which limits the applicability of many standard techniques for regularization parameter selection.

For our tests, we found using the average mean-squared error to be an effective method of parameter selection. If $\boldsymbol{f}_\alpha$ is the regularized solution for a given α parameter, we chose the α that minimized the two norm between the $\boldsymbol{f}_\alpha$ and the true solution $\boldsymbol{f}_{true}$. This was tested across a range of alpha and several different true images to test that the regularization parameter was a realistic choice. We found $\alpha = 100$ for both gradient-based methods and $\alpha = 1000$ for Gauss-Newton to be effective parameters for the methods when using $3e6$ photo-events per object data frame and 5000 photo-events per star data frame. We also note here that in general, α is sensitive to the number of photo-events in the object and star data and should be determined accordingly.

Using the average mean-squared error, we can ensure that the regularization parameter selected is affecting our solution and enforcing non-negativity. While using the true solution to select the regularization parameter is not a realistic expectation for real world data, it does verify that the regularization approach is effective. Finding a method of parameter selection which is independent of knowledge of the true solution represents future work.

Resultant Images

The final goal of the phase recovery is to combine it with the computed object power spectrum to obtain a good resultant image. Below, we display the resultant images from $E_1(\boldsymbol{\phi})$, $E_2(\boldsymbol{\phi})$, $E_1(\boldsymbol{f})$, and $E_2(\boldsymbol{f})$. For the first two objective functions, the phase is combined with the power spectrum after the iterative methods, where as for $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$, the power spectrum is included in the initial guess and the resultant images come directly from the iterative method. The resultant images for 50 data frames and $D/r_0 = 30$ can be seen below in Fig. 3.

From the figure, we can see various phases of image recovery. First we note that the power spectrum alone is insufficient for image recovery. Combining the power spectrum with the initial guess for the phase makes the object identifiable to the eye, and the subsequent optimization schemes offer varying improvement to the details of the object. Note that the regularization scheme pushes the negative image values upward toward zeros, but does not impose strict non-negativity. We have found that picking a regularization parameter that balances non-negativity and image quality yields the best results. Picking larger values for α tended to sacrifice resultant image quality for the sake of forcing background values to be non-negative.

5. CONCLUDING REMARKS

We conclude the paper with some final remarks on the utility of the methods implemented in this paper. The image quality obtained from the gradient-based methods and Gauss-Newton were mostly comparable, so the main utility of the Newton-based optimization lies in the improved convergence offered by the Hessian. In the case of $E_1(\boldsymbol{\phi})$ and $E_2(\boldsymbol{\phi})$, the Hessian offers not only a significant speed up in terms of convergence but also per iteration cost due to the one-time, storable factorization and effective line search. This makes it an attractive option, especially when a good solution is wanted quickly. An efficient implementation may make it a feasible method for obtaining an initial guess for more expensive methods such as multi-frame blind deconvolution if not as a stand alone method.

The results for $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ were also promising, although not from a cost per iteration perspective. The straightforwardness of implementing regularization for these methods results in improved images, and if a suitable preconditioner can be found for `pcg` in the Hessian solve, we believe these objective functions can be made comparable to the gradient-based alternatives. If so, the improved resultant images and prospect for intelligent regularization make this strategy more attractive as a source for detailed images.

Overall, the work demonstrates the utility of Newton-based optimization for problems in bispectral imaging. In particular, approaching the problem from a mathematical optimization and scientific computing standpoint introduced numerous areas where improvements were possible.

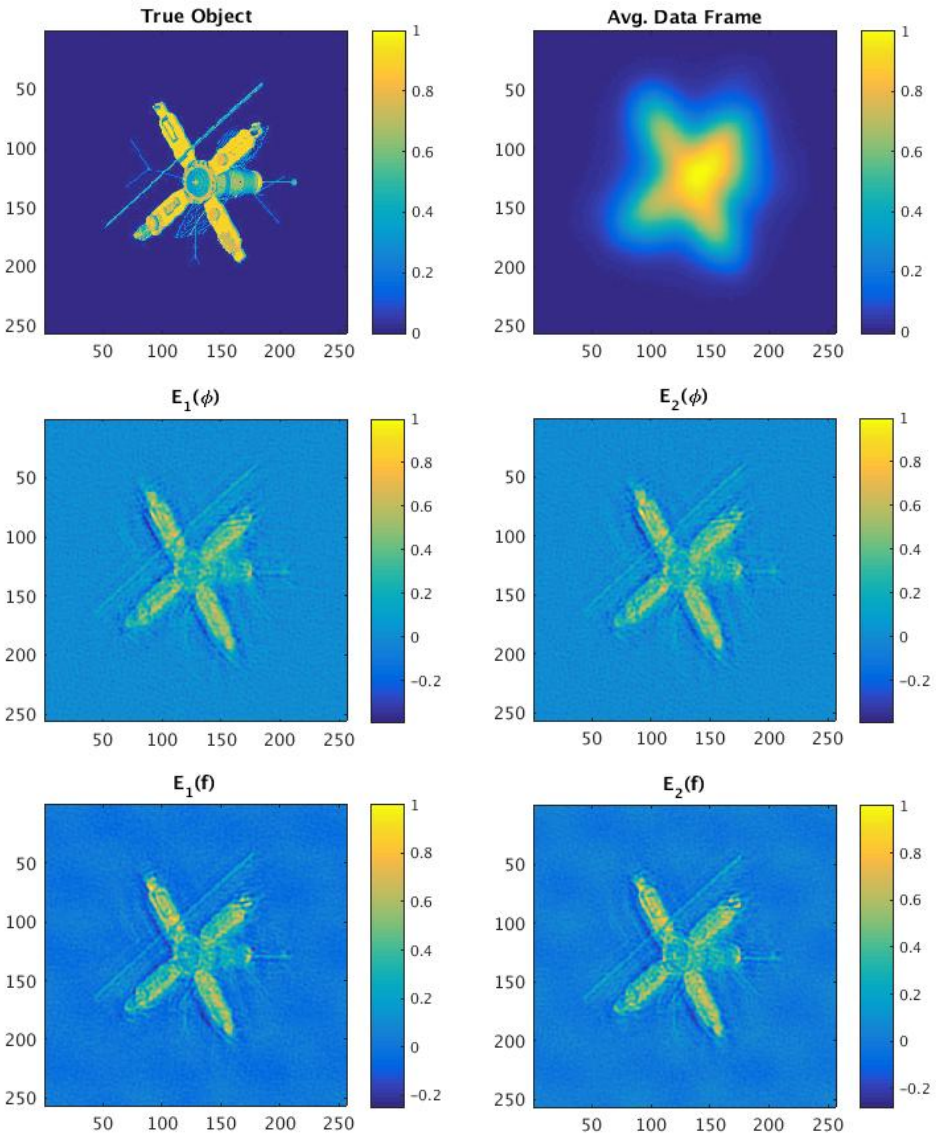

Figure 3: This figure shows the necessity of recovering the object's phase from the bispectrum and compares the resultant Newton solutions for each of the four proposed objective functions. Note that the regularization for $E_1(\boldsymbol{f})$ and $E_2(\boldsymbol{f})$ does not enforce complete non-negativity, but by penalizing some non-negativity succeeds in sharpening some of the edges and details of the satellite image. This represents a successful compromise in the regularization parameter selection.

6. APPENDIX

In order to implement the gradient and Hessian for Eqs. 19 and 20, it is necessary to differentiate the object's phase with respect to the image at the current iteration. To do this, we first note that the object's phase $\boldsymbol{\phi}$ can be expressed as a function of the current image $\boldsymbol{f}$ by

$$\boldsymbol{\phi} = \arctan\left(\frac{\mathrm{Im}(\mathcal{F}\boldsymbol{f})}{\mathrm{Re}(\mathcal{F}\boldsymbol{f})}\right) \tag{23}$$

From this, it is then possible to calculate the adjoint gradient operator $\frac{\partial \boldsymbol{\phi}^*}{\partial \boldsymbol{f}}$ via a combination of the chain rule and the quotient rule, as done in the appendix of Glindemann and Dainty [1]. The operator in the direction $\boldsymbol{v}$ is then given by

$$\begin{aligned}\frac{\partial \boldsymbol{\phi}^*}{\partial \boldsymbol{f}}(\boldsymbol{v}) &= \frac{\mathrm{Re}(\mathcal{F}\boldsymbol{f}) \odot \mathrm{Im}(\mathcal{F}\boldsymbol{v}) - \mathrm{Im}(\mathcal{F}\boldsymbol{f}) \odot \mathrm{Re}(\mathcal{F}\boldsymbol{v})}{|\mathcal{F}\boldsymbol{f}|^2} \\ &= \frac{\mathrm{Im}\left(\mathcal{F}\boldsymbol{v} \odot \overline{\mathcal{F}\boldsymbol{f}}\right)}{\mathcal{F}\boldsymbol{f} \odot \overline{\mathcal{F}\boldsymbol{f}}} \\ &= \mathrm{Im}\left(\frac{\mathcal{F}\boldsymbol{v}}{\mathcal{F}\boldsymbol{f}}\right)\end{aligned} \tag{24}$$

Here, the Hadamard product, the division, and the squared operation are all taken component-wise, while $\mathcal{F}$ again denotes a two dimensional Fourier transform. Some other considerations are important in computation. In order to avoid division by zero, the gradient at indices where the denominator is equal to zero should be set to zero. Also, the Fourier transforms need to be scaled properly according to the implementation used.

When computing the approximate Hessian for the Newton-based optimization, it is also necessary to compute the forward operator of Eq. 24 in the direction $\boldsymbol{v}$. This is most easily done and verified by an adjoint test, and is expressed as

$$\begin{aligned}\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{f}}(\boldsymbol{v}) &= \mathrm{Im}\left(\mathcal{F}\left[\frac{\mathrm{Re}(\mathcal{F}\boldsymbol{f}) \odot \boldsymbol{v}}{|\mathcal{F}\boldsymbol{f}|^2}\right]\right) - \mathrm{Re}\left(\mathcal{F}\left[\frac{\mathrm{Im}(\mathcal{F}\boldsymbol{f}) \odot \boldsymbol{v}}{|\mathcal{F}\boldsymbol{f}|^2}\right]\right) \\ &= \mathrm{Im}\left(\mathcal{F}\left[\frac{v \odot \overline{\mathcal{F}\boldsymbol{f}}}{\mathcal{F}\boldsymbol{f} \odot \overline{\mathcal{F}\boldsymbol{f}}}\right]\right) \\ &= \mathrm{Im}\left(\mathcal{F}\left[\frac{\boldsymbol{v}}{\mathcal{F}\boldsymbol{f}}\right]\right)\end{aligned} \tag{25}$$

Here again, division, squared operations, and Hadamard products are component-wise operations; indices corresponding to division by zero should be set to zero; and Fourier transforms should be scaled appropriately.

7. ACKNOWLEDGEMENTS

We would like to thank Brandoch Calef for his assistance and advice with this work. His expertise was invaluable first in introducing us to bispectral imaging and subsequently helping to resolve many of the problems and obstacles we came across during our research. This work was supported in part by grant No. DMS-1522760 from the US National Science Foundation.

REFERENCES

1. A. Glindemann and J. C. Dainty. Object fitting to the bispectral phase by using least squares. *JOSA A*, 10.5:1056–1063, 1993.
2. C. A. Haniff. Least-squares Fourier phase estimation from the modulo 2π bispectrum phase. *JOSA A*, 8.1:134–140, 1991.
3. K. T. Knox and B. J. Thompson. Recovery of images from atmospherically degraded short-exposure photographs. *Astrophys. J*, 193:L45–L48, 1974.
4. A. Labeyrie. Attainment of diffraction limited resolution in large telescopes by Fourier analyzing speckle patterns in star images. *Astron. Astrophys.*, 6:85–87, 1970.
5. A. W. Lohmann, G. Weigelt, and B. Wirnitzer. Speckle masking in astronomy: triple correlation theory and applications. *Applied Optics*, 22.24:4028–4037, 1983.
6. J. C. Marron, P. P. Sanches, and R. C. Sullivan. Unwrapping algorithm for least-squares phase recovery from the modulo 2π bispectrum phase. *JOSA A*, 7.1:14–20, 1990.
7. P. Negrete-Regagnon. Practical aspects of image recovery by means of the bispectrum. *JOSA A*, 13.7:1557–1576, 1996.
8. D. Tyler and K. Schulze. Fast phase spectrum estimation using the parallel part-bispectrum algorithm. *Pub. of the Astron. Soc. of the Pacific*, 116.815:65–76, 2004.
9. G. Weigelt. Modified speckle interferometry: speckle masking. *Opt. Commun.*, 21:55–59, 1977.