New algorithms for adaptive optics point-spread function reconstruction

Eric Gendron, Yann Clénet, Thierry Fusco, Gérard Rousset

Astronomy & Astrophysics manuscript no. gendron˙printform3.hyper29243
June 28, 2006

New algorithms for adaptive optics point-spread function reconstruction

E. Gendron[1], Y. Clénet[1], T. Fusco[2], and G. Rousset[1]

[1] Observatoire de Paris, LESIA, 5 place Jules Janssen, F-92195 Meudon Cedex, France
e-mail: eric.gendron@obspm.fr, yann.clenet@obspm.fr, gerard.rousset@obspm.fr
[2] ONERA, BP 52, 29 avenue de la Division Leclerc, 92320 Châtillon Cedex, France
e-mail: thierry.fusco@onera.fr

Received 6 March 2006; Accepted 20 June 2006

ABSTRACT

Context. The knowledge of the point-spread function compensated by adaptive optics is of prime importance in several image restoration techniques such as deconvolution and astrometric/photometric algorithms. Wavefront-related data from the adaptive optics real-time computer can be used to accurately estimate the point-spread function in adaptive optics observations. The only point-spread function reconstruction algorithm implemented on astronomical adaptive optics system makes use of particular functions, named U_{ij}. These U_{ij} functions are derived from the mirror modes, and their number is proportional to the square number of these mirror modes.

Aims. We present here two new algorithms for point-spread function reconstruction that aim at suppressing the use of these U_{ij} functions to avoid the storage of a large amount of data and to shorten the computation time of this PSF reconstruction.

Methods. Both algorithms take advantage of the eigen decomposition of the residual parallel phase covariance matrix. In the first algorithm, the use of a basis in which the latter matrix is diagonal reduces the number of U_{ij} functions to the number of mirror modes. In the second algorithm, this eigen decomposition is used to compute phase screens that follow the same statistics as the residual parallel phase covariance matrix, and thus suppress the need for these U_{ij} functions.

Results. Our algorithms dramatically reduce the number of U_{ij} functions to be computed for the point-spread function reconstruction. Adaptive optics simulations show the good accuracy of both algorithms to reconstruct the point-spread function.

Key words. Techniques: high angular resolution – Methods: numerical

1. Introduction

Since the advent of adaptive optics (AO), it has been possible to perform imaging with a spatial resolution very close to the diffraction limit. Despite this large improvement, the correction of the atmospheric turbulence by an AO system is only partial, and point source images are for example still affected by a halo that surrounds them.

Such effects of the partial AO correction can be corrected by image restoration techniques such as deconvolution algorithms. Except in the "blind" deconvolution case (Fusco et al. 1999; Christou et al. 1999), these algorithms need an estimation of the point-spread function (PSF), or its corresponding optical transfer function (OTF) derived from the PSF by a Fourier transform, to restore the image unaffected by the atmospheric turbulence. In addition, astrometric and photometric algorithms, e.g., StarFinder (Diolaiti et al. 2000) or DAOPHOT (Stetson 1987), usually also need an estimation of the PSF.

Wavefront-related data delivered by the AO real-time computer can help to accurately estimate the PSF. Véran et al. (1997) have been the first to develop such a PSF reconstruction algorithm. Implemented on the CFHT curvature sensing AO system PUEO (Arsenault et al. 1994), it has been routinely delivering reconstructed on-axis PSFs for more than 8 years now. A first attempt to transpose this algorithm to the Shack-Hartmann (SH) wavefront sensor has been undertaken by Harder & Chelli (2000).

Based on the Véran et al. (1997) algorithm, PSF reconstruction has been developed for three AO systems, equipped this time with SH wavefront sensors, and tested during a few runs of observations, leading to good results

- Weiss (2003) has written a piece of PSF reconstruction software for ALFA (Kasper et al. 2000), the SH AO system of the Calar Alto 3.5m telescope;
- Jolissaint et al. (2004) has written OPERA, a piece of PSF reconstruction software for Altair (Herriot et al. 2000), the 4-quadrant SH AO system of the Gemini North telescope; and

Send offprint requests to: Y. Clénet

– Fitzgerald et al. (2004) have written a piece of PSF reconstruction software for the SH AO system of the UCO/Lick Observatorys 3 m Shane Telescope (Bauman et al. 2002).

These current existing algorithms use particular functions, usually named U_{ij}, computed from the mirror modes. The number of U_{ij} functions is proportional to the square number of mirror modes, which leads to gigabytes of data to handle for systems with 150 to 200 actuators, and thus limits the efficiency of the PSF reconstruction process.

The goal of this article is to propose two new algorithms that avoid the use of these U_{ij} functions. The global scheme of the PSF reconstruction is not affected; we have simply replaced the steps involving the U_{ij} functions. We also provide, as a by-product, a way to estimate the PSF likelihood. The latter may be crucial for image deconvolution algorithms, which currently lack this kind of information.

We present the main points of the PSF reconstruction algorithm developed by Véran et al. (1997), as well as their different assumptions in Sect. 2. In Sect. 3, we describe the two algorithms we propose, and in Sect. 4, the tests from AO simulations. We discuss the choice and use of each algorithm in the last section.

2. The current U_{ij} algorithm

2.1. The long-exposure AO-corrected PSF expression

In the PSF reconstruction algorithm developed by Véran et al. (1997), assuming that the phase structure function defined in Eq. 4 below is stationary over the pupil, the AO-corrected monochromatic long-exposure OTF is decomposed as follows:

$$\left\langle OTF\left(\boldsymbol{\rho}/\lambda\right)\right\rangle = \left\langle OTF_{\phi_\epsilon}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle \times OTF_{\mathrm{tel}}\left(\boldsymbol{\rho}/\lambda\right), \quad (1)$$

where:

– $\left\langle OTF_{\phi_\epsilon}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle$ is the attenuation of the long-exposure OTF due to the partial correction of AO, and
– $OTF_{\mathrm{tel}}\left(\boldsymbol{\rho}/\lambda\right)$ is the perfect telescope OTF.

The phase ϕ_ϵ can be split into two parts: $\phi_{\epsilon_\parallel}$, which belongs to the vector space spanned by the mirror modes, and $\phi_{\epsilon_\perp}$, which is orthogonal to the former space.

$$\left\langle OTF\left(\boldsymbol{\rho}/\lambda\right)\right\rangle = \left\langle OTF_{\phi_{\epsilon_\parallel}}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle \times \left\langle OTF_{\phi_{\epsilon_\perp}}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle \times OTF_{\mathrm{tel}}\left(\boldsymbol{\rho}/\lambda\right) \quad (2)$$

and, this expression can be written

$$\left\langle OTF\left(\boldsymbol{\rho}/\lambda\right)\right\rangle \approx \exp\left(-\frac{1}{2}\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho})\right) \times \exp\left(-\frac{1}{2}\bar{D}_{\phi_{\epsilon_\perp}}(\boldsymbol{\rho})\right) \times OTF_{\mathrm{tel}}\left(\boldsymbol{\rho}/\lambda\right), \quad (3)$$

where:

– $\left\langle OTF_{\phi_{\epsilon_\parallel}}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle$ is the attenuation of the long-exposure OTF due the mirror component of the phase, i.e., the "residual parallel phase",
– $\left\langle OTF_{\phi_{\epsilon_\perp}}\left(\boldsymbol{\rho}/\lambda\right)\right\rangle$ is the attenuation of the long-exposure OTF due the component of the phase belonging to the space perpendicular to the mirror space, i.e., the "perpendicular phase",
– $\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho})$ is the mean structure function of the residual parallel phase,
– $\bar{D}_{\phi_{\epsilon_\perp}}(\boldsymbol{\rho})$ is the mean structure function of the perpendicular phase,
– $\boldsymbol{\rho}$ is a pupil plane coordinate vector, and
– λ is the wavelength of observation.

This decomposition is based on several assumptions (cf. Véran et al. 1997):

1. the complex field amplitude is uniform over the pupil (scintillation is neglected),
2. the residual parallel phase $\phi_{\epsilon_\parallel}(\boldsymbol{x})$ and the perpendicular phase $\phi_{\epsilon_\perp}(\boldsymbol{x})$ follow Gaussian statistics,
3. the phase structure function is stationary over the telescope pupil, and
4. the correlation between the mirror component and the perpendicular component of the phase is negligible.

From the expression of the structure function of the residual parallel phase:

$$D_{\phi_{\epsilon_\parallel}}(\boldsymbol{x},\boldsymbol{\rho}) = \left\langle \left(\phi_{\epsilon_\parallel}(\boldsymbol{x}) - \phi_{\epsilon_\parallel}(\boldsymbol{x}+\boldsymbol{\rho})\right)^2 \right\rangle \quad (4)$$

and the decomposition of the phase on the basis of the mirror modes $\{M_i(\boldsymbol{x})\}_{i=1\ldots N}$:

$$\phi_{\epsilon_\parallel}(\boldsymbol{x},t) = \sum_{i=1}^{N} \epsilon_{\parallel i}(t)\, M_i(\boldsymbol{x}), \quad (5)$$

one obtains:

$$D_{\phi_{\epsilon_\parallel}}(\boldsymbol{x},\boldsymbol{\rho}) = \sum_{i=1}^{N}\sum_{j=1}^{N} \langle \epsilon_{\parallel i}\epsilon_{\parallel j}\rangle \left(M_i(\boldsymbol{x}) - M_i(\boldsymbol{x}+\boldsymbol{\rho})\right)\left(M_j(\boldsymbol{x}) - M_j(\boldsymbol{x}+\boldsymbol{\rho})\right). \quad (6)$$

The mean structure function of the residual parallel phase $\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho})$ is the mean of $D_{\phi_{\epsilon_\parallel}}(\boldsymbol{x},\boldsymbol{\rho})$ over $\boldsymbol{x}$:

$$\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho}) = \frac{\int D_{\phi_{\epsilon_\parallel}}(\boldsymbol{x},\boldsymbol{\rho}) P(\boldsymbol{x}) P(\boldsymbol{x}+\boldsymbol{\rho}) \mathrm{d}\boldsymbol{x}}{\int P(\boldsymbol{x})\, P(\boldsymbol{x}+\boldsymbol{\rho}) \mathrm{d}\boldsymbol{x}}. \quad (7)$$

The expression of $\bar{D}_{\phi_{\epsilon_\perp}}(\boldsymbol{\rho})$ is similarly derived from $D_{\phi_{\epsilon_\perp}}(\boldsymbol{x},\boldsymbol{\rho})$.

The corresponding AO-corrected monochromatic long exposure PSF is derived as the Fourier transform of the OTF.

2.2. The $U_{ij}(\boldsymbol{\rho})$ functions

Equation 6 can be rewritten:

$$\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho}) = \sum_{i=1}^{N}\sum_{j=1}^{N}\langle \epsilon_{\parallel i}\epsilon_{\parallel j}\rangle\, U_{ij}(\boldsymbol{\rho}) \tag{8}$$

The $U_{ij}(\boldsymbol{\rho})$ functions are defined by:

$$\frac{\int \Big(M_i(\boldsymbol{x}) - M_i(\boldsymbol{x}+\boldsymbol{\rho})\Big)\Big(M_j(\boldsymbol{x}) - M_j(\boldsymbol{x}+\boldsymbol{\rho})\Big)P(\boldsymbol{x})P(\boldsymbol{x}+\boldsymbol{\rho})\mathrm{d}\boldsymbol{x}}{\int P(\boldsymbol{x})\,P(\boldsymbol{x}+\boldsymbol{\rho})\mathrm{d}\boldsymbol{x}}, \tag{9}$$

where $P(\boldsymbol{r})$ is the pupil function and $\boldsymbol{x}$ a coordinate vector in the pupil plane.

In practice, using the Fourier transform and the properties of the correlation function (Véran et al. 1997), the $U_{ij}(\boldsymbol{\rho})$ functions are computed as:

$$U_{ij}(\boldsymbol{\rho}) = \frac{\mathcal{F}^{-1}\Big(2\Re\left(\mathcal{F}(M_iM_jP)\mathcal{F}^*(P) - \mathcal{F}(M_iP)\mathcal{F}(M_jP)\right)\Big)}{\mathcal{F}^{-1}\left(|\,\mathcal{F}(P)\,|^2\right)}, \tag{10}$$

where $\mathcal{F}$ is the Fourier transform function and $\Re$ the complex number real part function.

Equation 8 is a key one for the experimental reconstruction of PSFs. The covariance matrix $\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle$ has to be measured experimentally on the AO system itself, by averaging the cross-products of wavefront measurements obtained during the time of the image exposure.

In the current PSF reconstruction algorithms, derived from Véran et al. (1997), the matrix $\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle$ is the basic entry point from which one can deduce successively the phase structure function, the OTF, and then the PSF. Additionally, one has to compute, store once for all, and also read the $U_{ij}(\boldsymbol{\rho})$ functions during the reconstruction process .

In Eq. 8, the i and j indices play a symmetric role, so that there are actually $N\times(N+1)/2$ "useful" $U_{ij}(\boldsymbol{\rho})$ functions. As an example, in the case of the VLT AO system NAOS (Rousset et al. 2000), the 159 compensated modes lead to 12720 "useful" $U_{ij}(\boldsymbol{\rho})$ functions. Today, the large number of $U_{ij}(\boldsymbol{\rho})$ hence represents, depending on the array size and data type, several gigabytes of data to compute, store, and read. Even if in practice, Eq. 8 can be efficiently implemented by Fourier transform (cf. Eq. 10), this leads to a heavy PSF reconstruction process, which will turn out to be impossible to handle in the future since the next AO systems are expected to have a largely increased number of modes: about 1370 actuators for the VLT-Planet Finder AO system (Fusco et al. 2005) and several tens of thousands for extremely large telescopes. In the following, we propose a way to achieve this computation, starting from the same covariance matrix $\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle$, without using the $U_{ij}(\boldsymbol{\rho})$.

3. Theory of the proposed algorithms

3.1. The V_{ii} algorithm

Let us consider the vector $\epsilon_\parallel(t)$, hereafter $\epsilon_\parallel$, made of the $\{\epsilon_{\parallel i}\}_{i=1\ldots N}$ coefficients, i.e., the vector representing $\phi_{\epsilon_\parallel}(\boldsymbol{x},t)$ in the basis of the mirror modes $M_i(\boldsymbol{x})$. The eigen decomposition of the residual parallel phase covariance matrix is:

$$\Lambda = B^t\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle B, \tag{11}$$

where Λ is a diagonal matrix that contains the $\{\lambda_i\}_{i=1\ldots N}$ eigenvalues and B is the matrix of eigenvectors: $B^tB = BB^t = Id$.

Equation 11 can be written:

$$\Lambda = \left\langle (B^t\epsilon_\parallel)(B^t\epsilon_\parallel)^t\right\rangle. \tag{12}$$

The vector η equal to $B^t\epsilon_\parallel$ represents $\phi_{\epsilon_\parallel}(\boldsymbol{x},t)$ in the basis that diagonalizes the residual parallel phase covariance matrix. Its coefficients are noted $\{\eta_i\}_{i=1\ldots N}$. From Eq. 12, the covariance matrix $\langle\eta\eta^t\rangle$ is diagonal, i.e., in this new basis, the residual parallel phase covariance matrix is diagonal, and the mean residual parallel phase structure function reduces to:

$$\bar{D}_{\phi_{\epsilon_\parallel}}(\boldsymbol{\rho}) = \sum_{i=1}^{N}\langle\eta_i\eta_i\rangle\, V_{ii}(\boldsymbol{\rho}) = \sum_{i=1}^{N}\lambda_i\, V_{ii}(\boldsymbol{\rho}), \tag{13}$$

where the $V_{ij}(\boldsymbol{\rho})$ functions are the equivalent in the new basis to the $U_{ij}(\boldsymbol{\rho})$ functions (Eq. 8). Similarly to Eq. 9, the $V_{ij}(\boldsymbol{\rho})$ functions are defined by

$$\frac{\int \Big(M'_i(\boldsymbol{x}) - M'_i(\boldsymbol{x}+\boldsymbol{\rho})\Big)\Big(M'_j(\boldsymbol{x}) - M'_j(\boldsymbol{x}+\boldsymbol{\rho})\Big)P(\boldsymbol{x})P(\boldsymbol{x}+\boldsymbol{\rho})\mathrm{d}\boldsymbol{x}}{\int P(\boldsymbol{x})\,P(\boldsymbol{x}+\boldsymbol{\rho})\mathrm{d}\boldsymbol{x}}, \tag{14}$$

such that $\mathcal{M}'$, the matrix made of the eigenvector modes $\{M'_i(\boldsymbol{x})\}_{i=1\ldots N}$ is given by $\mathcal{M}' = B^t\mathcal{M}$, $\mathcal{M}$ being the matrix made of the mirror modes $\{M_i(\boldsymbol{x})\}_{i=1\ldots N}$.

Using the V_{ii} algorithm, the computation of the residual parallel phase OTF only requires the computation of a number N of functions $V_{ii}(\boldsymbol{\rho})$. However, these $V_{ii}(\boldsymbol{\rho})$ functions have to be computed on the fly for each estimation of the mean residual parallel phase structure function.

3.2. The "instantaneous PSF" algorithm

The solution we propose here is similar to the algorithm presented by Roddier (1990) to simulate atmospherically distorted wavefronts, calculated from the covariance matrix of the coefficients of their expansion in Zernike modes. We extend this algorithm to any modal basis and covariance matrix, and use it to reconstruct AO-corrected PSFs.

Let us consider again the eigen decomposition of the residual parallel phase covariance matrix:

$$\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle = B\Lambda B^t, \tag{15}$$

If one generates a vector η whose coefficients are independent Gaussian random variables with zero mean and variance equal to the eigenvalue λ_i, i.e., $\langle\eta\eta^t\rangle = \Lambda$, then the vector $\beta = B\eta$ is a set of correlated random variables whose covariance matrix is $\langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle$:

$$\langle\beta\beta^t\rangle = \langle B\eta\eta^tB^t\rangle = B\Lambda B^t = \langle \epsilon_\parallel {\epsilon_\parallel}^t\rangle. \tag{16}$$

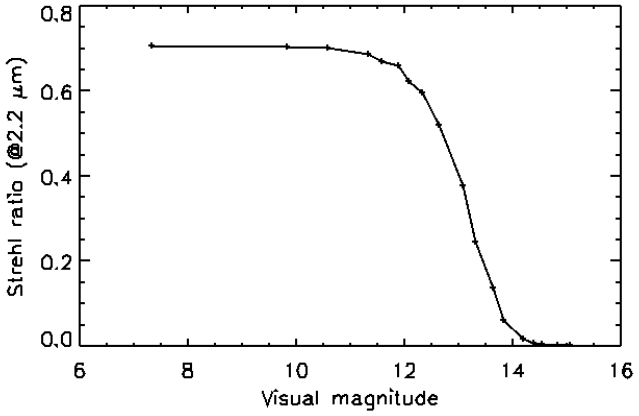

Fig. 1. Strehl ratio at 2.2 μm vs. guide star visual magnitude for a NAOS-like AO system.

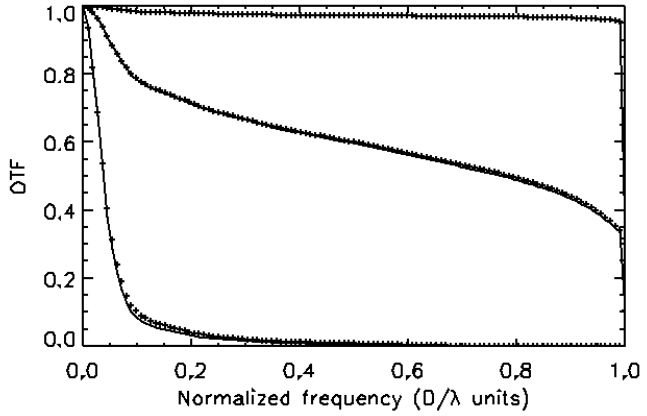

Fig. 2. Circular mean of the "atmospheric" OTFs for different conditions of correction: from top to bottom, the guide star magnitude is 7.3, 12.3, and 13.6. For each guide star magnitude, the solid line is obtained with the U_{ij}/V_{ii} and the crosses with the "instantaneous PSF" algorithm".

The phase represented by the vector β is:

$$\phi(\boldsymbol{x}, t) = \sum_{i=1}^{N} \beta_i(t)\, M_i(\boldsymbol{x}), \tag{17}$$

and the "instantaneous" PSF corresponding to that phase is:

$$PSF_{\|}(\boldsymbol{x}, t) = \left\| \mathcal{F}\Big(\exp(i\phi(\boldsymbol{x}, t)\Big) \right\|^2. \tag{18}$$

Then, by generating random η vectors such that $\langle \eta\eta^t \rangle = \Lambda$, we build instantaneous PSFs that, on average, converge to the long-exposure PSF of the mirror space. Note that the latter is not the "full PSF" that would be observed at the telescope since it does not include the uncorrected part of the phase (cf. Eq. 2). Finally:

$$\Big\langle OTF_{\phi_{\epsilon_{\|}}}\Big(\boldsymbol{x}/\lambda\Big)\Big\rangle \times OTF_{\rm tel}\Big(\boldsymbol{x}/\lambda\Big) = \mathcal{F}\left(\sum_t PSF_{\|}(\boldsymbol{x}, t) \right). \tag{19}$$

4. Test of the algorithms

4.1. Description of the simulation

To test our algorithms, we have used a Monte Carlo-based AO simulation software developed at ONERA. It is a complete end-to-end AO simulator divided into four main modules (calibration, propagation, closed-loop, and focal plane imaging modules) that make it as close as possible to an actual system. The complete algorithm is fully described in Conan R. et al. (2004) and has been used extensively for the design of AO systems, especially the future Planet Finder System for the VLT (Fusco et al. 2005), as well as for the tests and validations of existing AO systems (NAOS for instance).

We have tested our algorithms in the simple case of a NAOS-like AO system (Rousset et al. 2000): a 14×14 sub-pupil Shack-Hartmann wavefront sensor with 8×8 pixels per sub-aperture, a read-out noise of 3 e^- per pixel, and a sampling frequency of 500 Hz, 2048 loop iterations. The correction was performed with a 185 actuator piezostack deformable mirror plus a tip-tilt mirror. The wavefront sensing and observation wavelengths of the simulation were 0.65 and 2.2 μm, respectively. The seeing was 0.85″ at 0.5 μm.

Since we aimed at testing our algorithms with different conditions of correction, we ran the simulation with a guide star magnitude ranging from 7 to 15 so that the resulting Strehl ratio ranged from ∼70% down to ∼0.2% (Fig. 1). For a given guide star magnitude, we stored all the values $\epsilon_{\|}(t)$ obtained from the simulation to compute the covariance matrix $\langle \epsilon_{\|}\epsilon_{\|}^t \rangle$ and ran the U_{ij}, V_{ii}, and "instantaneous PSF" algorithms (code detailed in Appendix A) from this covariance matrix to derive the corresponding OTFs for comparison.

4.2. Results of the simulations

Figure 2 represents the circular mean of the residual "atmospheric" OTFs (i.e., not multiplied by the telescope OTF) produced by the different algorithms in three cases: good, moderate and poor correction (7.3^{th}, 12.3^{th}, and 13.6^{th} magnitude guide stars, respectively). Within the numerical uncertainties, the U_{ij} and V_{ii} algorithms produce exactly the same OTFs, which is not unexpected since they mathematically do the same computations, but in a different modal basis. Their corresponding PSF profiles in Fig. 3 are consequently also the same.

This is of course not the case for the "instantaneous PSF" algorithm, which needs a large enough number of iterations to converge. However, even in the poor correction case, the OTF profile obtained with the "instantaneous PSF" algorithm well reproduces the U_{ij} profile (Fig. 2) with a maximum error of a few 10^{-2} around 0.05 D/λ. This is also confirmed with the PSF profiles that are almost superimposed, even in the poor correction case (Fig. 3).

5. Discussion

The U_{ij} and V_{ii} algorithms mathematically produce the same OTFs. In practice, the U_{ij} algorithm requires reading each of the $N \times (N+1)/2$ U_{ij} functions (where N is the number

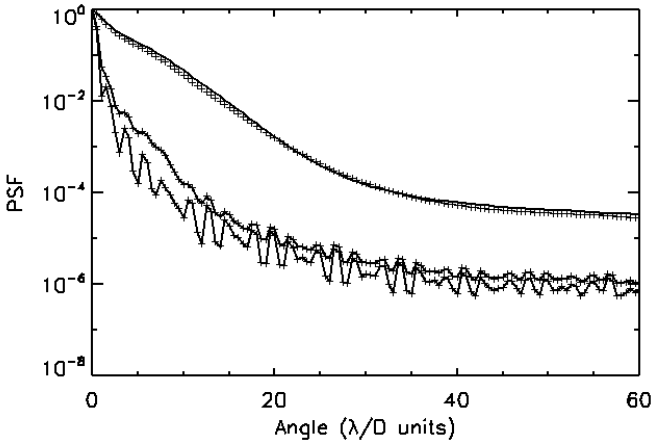

Fig. 3. Circular mean of the PSFs for different conditions of correction: from top to bottom, the guide star magnitude is 13.6, 12.3, and 7.3. For each guide star magnitude, the solid line is obtained with the U_{ij}/V_{ii} and the crosses with the "instantaneous PSF" algorithm". Note that the PSFs considered here only correspond to the residuals and do not include the perpendicular part of the phase.

of modes), which have been computed and stored previously, once. In comparison, the V_{ii} algorithm requires to diagonalize the covariance matrix, to compute the new modes $\{M'_i(\boldsymbol{x})\}_{i=1...N}$, and to compute each V_{ii} function. This latter step is composed of calls to these functions (cf. Appendix A):

- 2 FFTs,
- 1 square function,
- 1 real part of a complex number, and
- 1 modulus of a complex number.

As an example, the V_{ii} algorithm took ∼8 seconds in our simulation: ∼2 seconds to diagonalize the covariance matrix and compute the modes (the former taking a negligible time), and ∼6 seconds to compute the N V_{ii} functions. This is ∼25 times faster than the U_{ij} algorithm (∼191 seconds). In addition to this huge gain in computation time, a large amount of disk space is saved. This causes the V_{ii} algorithm to always be preferred to the U_{ij} one.

As noticed by Conan J.-M. (1994), averaging short-exposure OTFs, as we do in the "instantaneous PSF" algorithm, is a process that converges very slowly, especially at large D/r_0 or a low correction level. In addition, it does not lead to the infinitely long exposure OTF since a given number of short exposures are averaged. Besides, Conan J.-M. (1994) has shown that in the poor correction case, the error in computing the long-exposure OTF of such an algorithm is larger than for the U_{ij} algorithm, and then the V_{ii} algorithm as well.

Though, we emphasize that, in addition to the OTF computation itself, the "instantaneous PSF" algorithm can provide an estimation about the variability of the OTF, which can help a lot in some deconvolution algorithms. The estimation of the infinitely long exposure OTF that results from the convergence of our "instantaneous PSF" algorithm and that corresponds to a given covariance matrix $\langle \epsilon_\parallel \epsilon_\parallel{}^t \rangle$ is unique: let us call it $OTF_\infty(\boldsymbol{\rho}/\lambda)$. The dispersion in the random generation of OTFs can be computed as

$$\sigma^2(\boldsymbol{\rho}/\lambda) = \left\langle \left\| OTF_\infty(\boldsymbol{\rho}/\lambda) - OTF_i(\boldsymbol{\rho}/\lambda) \right\|^2 \right\rangle_i, \quad (20)$$

where OTF_i is the i^{th} draw of a randomly-generated OTF. If we call $OTF_{\rm obs}(\boldsymbol{\rho}/\lambda)$ the OTF actually observed on the instrument during the given, non infinite, observing time $T_{\rm int}$, when the given covariance matrix $\langle \epsilon_\parallel \epsilon_\parallel{}^t \rangle$ was measured, we can evaluate how far our estimation OTF_∞ is from $OTF_{\rm obs}$ by writing:

$$\left\langle \left\| OTF_\infty(\boldsymbol{\rho}/\lambda) - OTF_{\rm obs}(\boldsymbol{\rho}/\lambda) \right\|^2 \right\rangle = \frac{\sigma^2(\boldsymbol{\rho}/\lambda)}{n}, \quad (21)$$

where n is the equivalent number of independent realisations of PSFs, whose sum has resulted in the final PSF observed by the instrument during the given, non infinite integration time $T_{\rm int}$. An estimation of n could be obtained for example from a full simulation of the AO system under the same atmospheric conditions as during the observation. It is also be reasonable to consider that the impact of the correction by the AO system is to shorten the image correlation time compared to the image correlation time $\tau_0(\lambda)$ of the atmospheric seeing (Rigaut et al. 1991). We can then find a lower bound given by $n > T_{\rm int}/\tau_0(\lambda)$.

References

Arsenault, R., Salmon, D., & Kerr, J., et al. 1994, Proc. SPIE, 2201, 833

Bauman, B., Gavel, D., & Waltjen, K. 2002, Proc. SPIE, 4494, 19

Christou, J., Bonnacini, D., Ageorges, N., & Marchis, F. 1999, ESO Messenger, 97, 14

Conan, R., Fusco, T., Rousset, G., et al. 2004, Proc. SPIE, 5490, 602

Conan, J.-M., 1994, PhD Thesis, Université Paris XI Orsay, France

Diolaiti, E., Bendinelli, O., Bonaccini, D., et al. 2000, A&AS, 147, 335

Fitzgerald M. 2004, in The 1st Victoria Workshop on AO-PSF reconstruction, ed. L. Jollissaint, J.-P. Véran, J. Christou, & T. Rimmele, http://cfao.ucolick.org/meetings/psf reconstruction

Fusco, T., Rousset, G., Beuzit, J.-L. et al. 2005, Proc. SPIE, 5903, 148

Fusco, T., Véran, J.-P., Conan, J.-M., & Mugnier, L. 1999, A&AS, 134,1

Harder, S., & Chelli, A. 2000, A&AS, 142, 119

Herriot, G., Morris, S., Anthony, A., et al. 2000, Proc. SPIE, 4007, 115

Jolissaint, L., Véran, J.-P., & Marino, J. 2004, Proc. SPIE, 5490, 151

Kasper, M., Looze D., Hippler S., et al. 2000, Experimental Astronomy, 10, 49

Mugnier, L., Fusco, T., & Conan, J.-M. 2003, JOSA A, 21, 1841

Rigaut, F., Rousset, G., Kern, P., et al. 1991, A&A, 250, 280

Roddier, N. 1990, Proc. SPIE, 1237, 668

Rousset, G., Lacombe, F., Puget, P., et al. 2000, Proc. SPIE, 4007, 72
Stetson, P. 1987, PASP, 99, 191
Véran, J.-P., Rigaut, F., Maître H., & Rouan, D. 1997, Journal of the Optical Society of America A, 14, 3057
Weiss, R. 2003, PhD Thesis, Naturwissenschaftlich-Mathematische Gesamtfakultät der Universität Heidelberg, Germany

Appendix A: IDL codes

A.1. Common parts

```
    cbmes = readfits('cbmes.fits')
    s2m = readfits('slopes2modes.fits')
    modes = readfits('modalbasis.fits')
    nmodes=(size(modes))(3)
    lambda_im=2.2d-6
    lambda_aso=0.65d-6
    ratio_lambda=lambda_aso/lambda_im
; Telescope OTF computation and corresponding mask
    apert=readfits('pupille.fits')
    dim=(size(apert))(1)
    dim2=2*dim
    pup=dblarr(dim2,dim2)
    pup(0,0)=apert
    pupfft=fft(double(pup))
    conjpupfft=conj(pupfft)
    otftel = real_part(fft(pupfft*conjpupfft,/inverse))
    den=1/(otftel)
    den(where(finite(den) eq 0))=0
    mask=fltarr(dim2,dim2)+1
    mask(where(otftel lt 1e-5))=0
    otftel=otftel/max(otftel)
;Computation of the covariance matrix over the modes
    cbmode = cbmes##transpose(s2m)
    nbcb = (size(cbmode))(2)
    covmode = (cbmode#transpose(cbmode))/nbcb
```

A.2. The U_{ij} algorithm

```
; Computation and storage of the Uij functions
    modei = dblarr(dim2,dim2)
    modej = dblarr(dim2,dim2)
    for i=0,nmodes-1 do begin
        modei(0,0) = modes(*,*,i)*apert
        for j=i,nmodes-1 do begin
            modej(0,0) = modes(*,*,j)*apert
            term1 = real_part(fft(modei*modej)*conjpupfft)
            term2 = real_part(fft(modei)*conj(fft(modej)))
            uij = real_part(fft(2*(term1-term2),/inverse))*den*mask
            writefits,'Uij/u_'+string(i,format='(i2.2)')+'_'+$
                              string(j,format='(i2.2)')+'.fits',uij
        endfor
    endfor
; Computation of the OTF with the Uij
    dph1 = dblarr(dim2,dim2)
    for i=0,nmodes-1 do begin
        for j=i,nmodes-1 do begin
            uij = readfits('Uij/u_'+string(i,format='(i2.2)')+'_'+$
                    string(j,format='(i2.2)')+'.fits',/silent)
            fac = double((i ne j)+1)
            dph1 = dph1+(fac*covmode(i,j)*uij)
        endfor
    endfor
    otf1 = exp(-0.5*dph1*ratio_lambda^2)*mask
    otf1 = otf1/max(otf1)
```

A.3. The V_{ii} algorithm

```
; New modes that diagonalize the covariance matrix
    l = (eigenql(covmode,eigenvectors=b))>0
    s = b#diag_matrix(sqrt(l))
    newmodes = reform((reform(modes,dim^2,nmodes))#b,$
                                              dim,dim,nmodes)
; Computation of the OTF with the Vii
    newmodei = dblarr(dim2,dim2)
    temp = dblarr(dim2,dim2)
    for i=0,nmodes-1 do begin
        newmodei(0,0) = newmodes(*,*,i)*apert
        term1 = real_part(fft(newmodei^2)*conjpupfft)
        term2 = (abs(fft(newmodei)))^2
        temp = temp+((term1-term2)*l(i))
    endfor
    otf2 = exp(-0.5*real_part(fft(2*temp,/inverse))*den$
                                  *ratio_lambda^2)*mask
    otf2 = otf2/max(otf2)
```

A.4. The "instantaneous PSF" algorithm

```
    psf = dblarr(dim2,dim2)
    tmp = ratio_lambda*reform(modes,dim^2,nmodes)
    for i=0,nbcb-1 do begin
        phi(0,0) = reform((s#randomn(seed,nmodes))##tmp,dim,dim)
        psf = psf+(abs(fft(pup*exp(dcomplex(0,phi)))))^2
    endfor
    psf = psf/nbcb
    otf3 = real_part(fft(psf))
    otf3 = otf3/max(otf3)/otftel*mask
    otf3(where(finite(otf3) eq 0))=0
```