ÖAW AUSTRIAN ACADEMY OF SCIENCES

Johann Radon Institute for
Computational and Applied Mathematics
Austrian Academy of Sciences (ÖAW)

RICAM
JOHANN · RADON · INSTITUTE
FOR COMPUTATIONAL AND APPLIED MATHEMATICS

A finite element-wavelet hybrid algorithm for atmospheric tomography

M. Yudytskiy, T. Helin, R. Ramlau

RICAM-Report 2013-14

A finite element-wavelet hybrid algorithm for atmospheric tomography

Mykhaylo Yudytskiy,[1,*] Tapio Helin,[2] and Ronny Ramlau[3,1]

[1] *Johann Radon Institute for Computational and Applied Mathematics (RICAM), Altenbergerstrasse 69, 4040 Linz, Austria*

[2] *University of Helsinki, Gustaf Hallströmin katu 2b, FI-00014 Helsinki, Finland*

[3] *Johannes Kepler University Linz, Altenbergerstrasse 69, 4040 Linz, Austria*

compiled: September 18, 2013

Reconstruction of the refractive index fluctuations in the atmosphere, or atmospheric tomography, is an underlying problem of many next generation adaptive optics (AO) systems, such as the multi conjugate adaptive optics (MCAO) or multi object adaptive optics (MOAO). The dimension of the problem for the extremely large telescopes, such as the European Extremely Large Telescope (E-ELT), suggests the use of iterative schemes as an alternative to the matrix-vector multiply (MVM) methods. Recently, a novel algorithm based on the wavelet representation of the turbulence has been introduced in [11] by the authors to solve the atmospheric tomography using the conjugate gradient iteration. An efficient frequency-dependent preconditioner has been found for the wavelet method in [27]. In this paper we study the computational aspects of the wavelet algorithm. We introduce three new techniques, the dual domain discretization strategy, a scale-dependent preconditioner and a ground layer multi-scale (GLMS) method to derive a method that is globally $\mathcal{O}(n)$, parallelizable and compact with respect to memory. We present the computational cost estimates and compare the theoretical numerical performance of the method with the MVM. The quality of the method is evaluated in terms of an MOAO simulation for the E-ELT on the European Southern Observatory (ESO) end-to-end simulation system OCTOPUS. The method is compared to the ESO version of the FrIM [23] in terms of quality.

OCIS codes: (000.3860) Mathematical methods in physics; (010.7350) Wavefront sensing; (110.1080) Active or adaptive optics; (350.1260) Astronomical optics.

http://dx.doi.org/10.1364/XX.99.099999

1. Introduction

The adaptive optics (AO) systems in the next generation of ground-based telescopes aim to produce good correction in a large field of view. To achieve this, several wavefront sensors (WFSs) are employed in combination with multiple deformable mirrors (DMs). Many of these systems, such as the multi conjugate adaptive optics (MCAO), laser tomography adaptive optics (LTAO) and multi object adaptive optics (MOAO) require reconstruction of the turbulence profile in the atmosphere. The optimal mirror commands are derived based on this reconstruction.

The problem of reconstructing the turbulence profile is commonly called atmospheric tomography and it has been widely studied during the last decade, see, e.g., [6, 24, 25]. Mathematically the problem is ill-posed, i.e., there is an unstable relation between measurement data and the solution. In consequence, regularization techniques must be applied. A common procedure in the literature is to formulate the problem in the Bayesian framework, where statistical information regarding the turbulence model and sensor noise can be utilized. Since both prior and noise models are typically Gaussian, the maximum a-posteriori (MAP) estimate provides an optimal point estimate for the solution.

Currently, only the standard matrix-vector multiplication (MVM) algorithm has been implemented in a working MCAO system [14]. In MVM the precomputed inverse of the regularized system matrix is multiplied with the data. The computational cost of such an algorithm scales as $\mathcal{O}(n^2)$, where n is the dimension of the AO system. This dimension is expected to increase rapidly during next generations of telescopes. For the extremely large telescopes, such as the proposed European Extremely Large Telescope (E-ELT), the number of degrees of freedom ranges between 10^4 and 10^5, making the MVM with current hardware computationally demanding in the millisecond time frame even with heavy parallelization.

A viable alternative to the direct solvers, such as the MVM, are the iterative solvers, which have been studied by the AO community during recent years. Unlike the inverse, the forward system matrix associated to the problem tends to be sparse. Iterative solvers benefit from this property, which reduces both computational cost and memory storage. Their disadvantage is that several iterations must be applied until a solution with

* Corresponding author: mykhaylo.yudytskiy@ricam.oeaw.ac.at

sufficient precision is obtained.

The regularized problem of atmospheric tomography is symmetric and positive definite. This allows the most prominent iterative method for such kind of problems, the conjugate gradient (CG) algorithm, to be used. One of the best ways to reduce the number of iterations in this scheme is by preconditioning. Several preconditioned conjugate gradient (PCG) methods have been introduced for atmospheric tomography.

The multigrid preconditioners (MG-PCG) have been proposed in [8] where the complexity of the problem was of the order $\mathcal{O}(n^{3/2})$. Later, in [26] the Fourier-domain preconditioning techniques (FD-PCG) led to methods which scale according to $\mathcal{O}(n \log n)$. A very promising method, called the Fractal Iterative Method (FrIM), has been developed by Tallon and others in [23], where an efficient factorization of the inverse prior matrix yields a method scaling in $\mathcal{O}(n)$ operations.

Notice that iterative methods have been developed also outside the Bayesian framework. An algorithm based on the Kaczmarz iteration was proposed in [19, 21]. The method splits the atmospheric tomography problem into two sub-problems of reconstructing the wavefronts from WFS measurements and consequently reconstructing the atmospheric layers using the Kaczmarz iteration. The problem is regularized by an early termination of the iteration. Combined with a low iteration count due to the split approach, the method scales with $\mathcal{O}(n)$ even without preconditioning.

Here, we concentrate on a wavelet-based iterative method introduced in [11], where the MAP estimate was solved using a CG-based algorithm. One of the key features of this method is to utilize the fast decay of compactly supported orthogonal wavelets in the frequency domain. This allows to approximate the inverse covariance with completely diagonal representation. Numerical experiments on a simulated MCAO system have demonstrated that the method performs well in low imaging conditions. Later, a frequency dependent preconditioning technique was introduced in [27] for the wavelet method to reduce the number of iterations.

Let us mention that wavelets have been applied in single conjugated adaptive optics in [9, 10]. To the knowledge of the authors wavelets have not been considered before in the context of atmospheric tomography prior to [11]. For a general introduction to wavelets, we refer to [3].

In this work we study the real time aspects of the wavelet algorithm. The performance of the algorithm depends on two factors: the numerical effort of applying the forward operator and the number of iterations. Below we demonstrate that the wavelet method is globally $\mathcal{O}(n)$ and is highly parallelizable. This is achieved by combining several techniques, such as the frequency dependent preconditioning in [27], with novel ideas of dual-domain strategy and ground layer multi-scale method.

The numerical effort is directly related to how the forward operator can be factorized and to the sparsity of related operators. Due to the approximations mentioned above, the wavelet discretization of the penalty term is represented by a completely diagonal matrix. Also, the fitting term has an asymptotically sparse structure with respect to the wavelet basis. In practise, however, this is not the case due to the limited amount of scales. At lower scales, the support of the basis functions in space is not well-localized and a sparse structure is not achieved. This is the case especially for Daubechies wavelets with a high index.

In a finite element basis the situation is somewhat reversed. Due to the localized support of the basis functions, the forward operator is very sparse. However, the inverse prior covariance has a dense representation. Some sparse approximations have been introduced in this setting [4, 23].

In the dual-domain strategy we aim to use the best from both worlds. The regularization and fitting terms are discretized in the wavelet and piecewise bilinear domain, respectively. This leads to a computationally efficient representation of both terms. A fast transition between the bases is achieved by applying the discrete wavelet transform (DWT). The DWT is a transform of linear complexity and is parallelizable.

In addition to the frequency dependent preconditioner, we improve convergence and stabilize the reconstructor by utilizing the multi-scalar structure of wavelets. Namely, we solve a sub-problem defined for the coarse wavelet scales of the ground layer to improve the initial guess of the CG iteration. We call this technique ground layer multi-scale (GLMS) method.

We demonstrate the performance of the algorithm for an MOAO system using OCTOPUS, the end-to-end simulation tool of the European Southern Observatory. We break down the simulations into several studies concentrating on the performance with respect to noise level and different choices of cut-off scale in the GLMS and the frequency-dependent preconditioner. Furthermore, convergence speed is considered. Most importantly, a comprehensive analysis of the theoretical computational cost of the method is given. The results are very promising and the computational cost is in line with the theoretical expectation of the method being $\mathcal{O}(n)$.

This paper is structured as follows. In Section 2 we give a short introduction to the atmospheric tomography problem. Wavelets and the discrete wavelet transform are discussed in Section 2.B after which we introduce the approximative prior models obtained in [11]. The Section 3 is devoted to the description of the algorithm. We discuss the topics of dual-domain strategy, preconditioner choice and the GLMS separately. The numerical simulations are presented in Section 4. Here, the computational cost is analysed and we give results regarding the qualitative performance. Finally, conclusions and an outlook are given in Section 5.

2. Preliminaries

2.A. Atmospheric tomography problem

We assume that the AO system utilizes M guide stars consisting of M_{lgs} laser guide stars (LGS) and $M - M_{\text{lgs}}$ natural guide stars (NGS). Further, we assume a layered atmospheric turbulence model [20], where the refractive index is described by thin horizontal layers assigned to different altitudes. We denote the refractive index fluctuations in the atmosphere by a vector $\boldsymbol{\phi} = (\phi_1, ..., \phi_L)$, where ϕ_ℓ, $\ell = 1, ..., L$ stands for a two-dimensional layer. The sub-problems for different guide star directions are given by

$$\mathbf{s}_m = \Gamma_m P_m^{\text{LGS}} \boldsymbol{\phi} \quad \text{and} \quad \mathbf{s}_{m'} = \Gamma_{m'} P_{m'}^{\text{NGS}} \boldsymbol{\phi} \tag{1}$$

for $1 \le m \le M_{\text{lgs}}$ and $M_{\text{lgs}} < m' \le M$. Here, P_m^{LGS} and $P_{m'}^{\text{NGS}}$ are geometric propagation operators in directions of the guide stars associated to an LGS with cone effect and an NGS, respectively. Moreover, Γ_m is the Shack-Hartmann operator. The propagation operators are given by

$$(P_m^{\text{LGS}} \boldsymbol{\phi})(\mathbf{r}) = \sum_{\ell=1}^{L} \phi_\ell \left(\left(1 - \frac{h_\ell}{H} \right) \mathbf{r} + \boldsymbol{\theta}_m h_\ell \right)$$

and

$$(P_{m'}^{\text{NGS}} \boldsymbol{\phi})(\mathbf{r}) = \sum_{\ell=1}^{L} \phi_\ell (\mathbf{r} + \boldsymbol{\theta}_{m'} h_\ell),$$

where $\mathbf{r} = (x, y, 0)$ is the location at the aperture, $\boldsymbol{\theta}_m = (\theta_x, \theta_y, 1)$ is the direction of the guide star m, h_ℓ is the layer altitude and H is the altitude where the laser scatters. The LGS propagation operator takes the cone effect into account via the scaling factor $1 - h_\ell / H$.

Furthermore, we write

$$\mathbf{s} = (\mathbf{s}_m)_{m=1}^{M} = \mathbf{A}\boldsymbol{\phi}, \tag{2}$$

where $\mathbf{A}$ is the concatenation of operators $\Gamma_m P_m^{\text{LGS}}$ and $\Gamma_{m'} P_{m'}^{\text{NGS}}$. Estimating $\boldsymbol{\phi}$ in (2) from a given $\mathbf{s}$ is called the atmospheric tomography problem.

The underlying mathematical problem in (2) is ill-posed and regularization is necessary to obtain a stable solution from noisy measurements. To this end, the problem is often formulated in the Bayesian framework, where the statistics of turbulence and noise can be utilized in the model. The optimal solution in this setting is the maximum a-posteriori (MAP) estimate, which is obtained from

$$\text{argmin}_{\boldsymbol{\phi}} \left(\|\mathbf{C}_\Phi^{-1/2} \boldsymbol{\phi}\|_2^2 + \|\mathbf{C}_\mathcal{E}^{-1/2} (\mathbf{s} - \mathbf{A}\boldsymbol{\phi})\|_2^2 \right). \tag{3}$$

Here, $\mathbf{C}_\Phi$ and $\mathbf{C}_\mathcal{E}$ stand for the turbulence and measurement noise covariance matrices, respectively.

The prior covariance matrix $\mathbf{C}_\Phi = \text{diag}(C_1, ..., C_L)$ has a block–diagonal structure, where C_ℓ is the covariance matrix on layer ℓ. The turbulence statistics is assumed to obey the von Karman power law. Following [11] we show that the penalty term in (3) has an efficient approximation in the wavelet domain, which we describe in detail in Section 2.C.

The noise covariance matrix $\mathbf{C}_\mathcal{E} = \text{diag}\left(\widehat{C}_1, ..., \widehat{C}_{M_{\text{lgs}}}, \widehat{C}_{M_{\text{lgs}}+1}, ..., \widehat{C}_M \right)$ also has a block-diagonal structure. The model for $\widehat{C}_m$, $1 \le m \le M_{\text{lgs}}$ corresponding to the spot elongated LGS measurements follows [1]. Notice that each of the sub-matrices have a 2×2 block diagonal representation. The noise covariances $\widehat{C}_{m'}$, $M_{\text{lgs}} < m' \le M$ corresponding to the NGS measurements are modeled by an identity matrix scaled with the noise variance.

It is well-known that the LGS measurements contain an erroneous tip-tilt component. In our method, the tip-tilt component in the LGS measurements is removed by modifying (1) to $T\mathbf{s}_m = T\Gamma_m P_m^{LGS} \boldsymbol{\phi}$ for $1 \le m \le M_{\text{LGS}}$. Here, T is the orthogonal projection operator to the complement of the tip-tilt measurements space. For the regularized problem (3), such noise modeling can be interpreted as using an updated inverse measurement noise covariance matrix

$$\mathbf{C}_\mathcal{E}^{-1} = \text{diag}\left((T\widehat{C}_m^{-1} T)_{m=1}^{M_{\text{lgs}}}, \quad (\widehat{C}_{m'}^{-1})_{m'=M_{\text{lgs}}+1}^{M} \right) \tag{4}$$

in combination with the original equations in (1).

2.B. Wavelets and the DWT

Let φ be a scaling function of a multiresolution analysis (MRA) of $L^2(\mathbb{R})$ with compact support. Further, we assume that shifts of φ form an orthonormal basis of a subspace of the MRA. Then, there exists a wavelet function ψ associated to φ. Together, φ and ψ form a wavelet family.

In two dimensions, the scaling function is defined by

$$\varphi(x) = \varphi(x_1)\varphi(x_2), \quad x = (x_1, x_2), \tag{5}$$

and

$$\begin{aligned} \psi^1(x) &= \varphi(x_1)\psi(x_2), \\ \psi^2(x) &= \psi(x_1)\varphi(x_2), \\ \psi^3(x) &= \psi(x_1)\psi(x_2), \end{aligned}$$

are the wavelet functions of three types. The dilated and translated scaling and wavelet functions are denoted by

$$\begin{aligned} \varphi_{jk}(x) &= 2^{j/2} \varphi(2^j x - k) \quad \text{and} \\ \psi_{jk}^t(x) &= 2^{j/2} \psi^t(2^j x - k). \end{aligned}$$

Above, indices $j \in \mathbb{Z}$ and t stand for the scale index and the wavelet type, $t = 1, 2, 3$, respectively. The index $k = (k_1, k_2) \in \mathbb{Z}^2$ enumerates the location in space.

Let $f \in \text{span}\{\varphi_{Jk} : k \in \mathbb{Z}^2\}$ and $a_{Jk} = \langle f, \varphi_{Jk} \rangle$, $k \in \mathbb{Z}^2$, denote the approximation coefficients at the scale J. Due to the MRA properties, f can be approximated at scale 0 with coefficients $a_{0k} = \langle f, \varphi_{0k} \rangle$, such that the complementary detail coefficients $d_{jk}^t = \langle f, \psi_{jk}^t \rangle$ at scales

$j = 0, ..., J-1$ allow a lossless decomposition of f,

$$f(x) = \sum_{k\in\mathbb{Z}^2} a_{0k}\varphi_{0k}(x) + \sum_{j=0}^{J-1}\sum_{k\in\mathbb{Z}^2}\sum_{t=1}^{3} d_{jk}^t \psi_{jk}^t(x). \quad (6)$$

Here, $\langle\cdot,\cdot\rangle$ denotes the scalar product in $L^2(\mathbb{R}^2)$.

Decomposition (6) is referred to as the wavelet decomposition of f. The DWT is an algorithm, which transforms approximation coefficients $\{a_{Jk}\}$ to wavelet coefficients $\{a_{0k}, d_{jk}^t\}$. For a wavelet family with compact support, the DWT algorithm corresponds to a hierarchical convolution with a finite filter and downsampling. The inverse DWT maps wavelet coefficients back to approximation coefficients.

We restrict the discussion of the DWT to square signals of finite dimension $N = 2^{2J}$ for some $J \in \mathbb{N}$. At every scale $j = 0, ..., J$ there are 2^{2j} approximation coefficients $a_j = (a_{jk})_{k_1=0,k_2=0}^{2^j-1}$. Additionally, at every scale $j = 0, ..., J-1$ there are three types of 2^{2j} detail coefficients $d_j^t = (d_{jk}^t)_{k_1=0,k_2=0}^{2^j-1}$ for $t = 1, 2, 3$. In our approach, we treat signals of finite dimension as functions on a bounded domain. We impose periodic boundary conditions for the numerical implementation.

To each wavelet family φ and ψ, we can associate low and high pass filters l and $h \in \ell^2$. Compactly supported wavelets have filters of finite length. For instance, the Daubechies n wavelet family used in this work have filters of length $2n$.

Then, the convolution operation with filters l and h at scale j can be represented by a $2^{j+1} \times 2^{j+1}$ matrix W_j with non-zero entries

$$(W_j)_{i,(2i+p \bmod 2^j)} = \sum_{\substack{0\le q<|l| \\ (q \bmod 2^j)=p}} l_q,$$

$$(W_j)_{2^j+i,(2i+p \bmod 2^j)} = \sum_{\substack{0\le q<|l| \\ (q \bmod 2^j)=p}} h_q,$$

for $i = 0, ..., 2^j - 1$ and $p = 0, ..., |l| - 1$. This wavelet transforms handles the signal at the boundary by periodic extension. A schematic representation of W_j is presented in Figure 1.

The wavelet transform at scale j in two dimensions corresponds to two multiplications by the wavelet transform matrix,

$$(W_j(W_j a_{j+1})^T)^T = W_j a_{j+1} W_j^T = \begin{pmatrix} a_j & d_j^1 \\ d_j^2 & d_j^3 \end{pmatrix}. \quad (7)$$

Similarly, one level of the inverse DWT in two dimensions is realized through the transposed multiplication,

$$W_j^T \begin{pmatrix} a_j & d_j^1 \\ d_j^2 & d_j^3 \end{pmatrix} W_j = a_{j+1}. \quad (8)$$

The DWT algorithm corresponds to applying (7) sequentially to decompose the approximation coefficients for $j = J-1, ..., 0$. Inverse DWT corresponds to reconstructing the approximation coefficients using the information of the coarser scale via (8) for $j = 0, ..., J-1$.

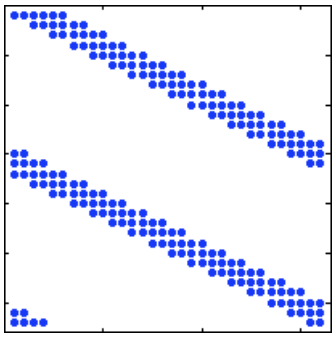

Fig. 1. Schematic representation of a DWT matrix W_4 of Daubechies 3 wavelets. Non-zero entries of the matrix are marked. Daubechies 3 filters l and h have six elements. Low-pass and high-pass filters are repeated in rows 0 to 2^4-1 and 2^4 to 2^5-1, respectively. A horizontal-shift of two elements in each row relates to the downsampling of the signal. Periodic extension of the wavelets on the boundary is represented by wrapping around the filter coefficients.

Finally, the approximation coefficients of the finest scale a_{Jk} and a piecewise bilinear function f are related by a scaling constant. Let $f : [0, \delta(2^J - 1)]^2 \to \mathbb{R}$ be a piecewise bilinear function with equidistant spacing δ over a $2^J \times 2^J$ square grid of points. Then the approximation coefficients of the finest scale are related to the nodal values of f via the approximation $a_{J(k_1k_2)} \approx \delta f(\delta k_1, \delta k_2)$. The scaling parameter δ ensures that the representation is norm-preserving.

2.C. Turbulence statistics

The statistical models for turbulence have received wide attention since the classical works by Kolmogorov and the topic is often studied via the structure function. From this perspective, wavelets and their approximative properties have been studied before in [17]. However, the typical minimum variance solution methods in adaptive optics utilize Bayesian inference where the inverse covariance of the turbulence model is needed. This is not always easily available from structure function approximations. We are motivated by finding an efficient numerical approximation to the inverse covariance in the wavelet scheme.

In this work we assume that the spectral density of the turbulence field satisfies the von Karman model

$$m(\kappa) = c_\rho(h)\left(\frac{1}{K_{out}^2} + |\kappa|^2\right)^{-11/6},$$

where K_{out} denotes the outer scale of the inertial range. Above, c_ρ describes the measure of the optical turbulence strength depending on the altitude h.

A stochastic process with spectral density m does not have a straightforward random variable representation in Sobolev spaces since its realizations are not compactly

supported and do not decay at infinity. Here, we avoid these technical considerations and assume that the prior induces the widely used penalty term

$$\|C_\phi^{-1/2} f\|_{L^2}^2 = \frac{1}{c_\rho(h)} \|(K_{out}^{-2} + |\kappa|^2)^{\frac{11}{12}} \mathcal{F} f\|_{L^2}^2 \simeq \frac{1}{c_\rho(h)} \left(K_{out}^{-\frac{11}{3}} \|f\|_{L^2}^2 + \|(-\Delta)^{\frac{11}{12}} f\|_{L^2}^2 \right) \quad (9)$$

for any $f \in C_0^\infty(\mathbb{R}^2)$. Here and in what follows, we write $p \simeq q$ if the two pseudo-norms p and q are equivalent.

Assume for the moment that the wavelets studied here are 2-regular, i.e., have 2 vanishing moments and are 2 times continuously differentiable. Then the last term in (9) is equivalent to the expression

$$\|(-\Delta)^{\frac{11}{12}} f\|_{L^2}^2 \simeq \sum_{j \in \mathbb{Z}} \sum_{k \in \mathbb{Z}^2} 2^{2 \cdot \frac{11}{6} j} |\langle f, \psi_{jk} \rangle|^2, \quad (10)$$

where j is the wavelet scale of the wavelet. The equivalence above is known as the Bernstein-Jackson inequalities [15].

In the numerical simulations we utilize the Daubechies wavelet basis. They are an orthogonal wavelet family with compact support [3]. For a sufficiently large n, the Daubechies n wavelets are 2–regular. In order to keep the basis functions well-localized in the spatial domain, we have chosen to use $n = 3$. It is well–known that the Daubechies 3 wavelets belong to the Hölder space $C^{1+\delta}(\mathbb{R}^2)$ with $\delta \approx 0.0878$ [3]. Even though they are not 2–regular, they form a Riesz basis in $H^2(\mathbb{R}^2)$ [2]. Based on our numerical tests we believe that this is sufficient in practice.

In the discretized problem, the function f in (9) at altitude h_ℓ is represented at a finite number of wavelet scales. In that case, an equivalent representation for the regularizing term in (9) can be produced by a diagonal matrix D_ℓ that satisfies

$$D_\ell = \operatorname{diag}\left(\frac{1}{c_\rho(h_\ell)} \left(K_{out}^{-\frac{11}{3}} + 2^{\frac{11}{3} j} \right) \right)_{j,k_1,k_2},$$

for wavelet scale index $j = 0, ..., J-1$ and shift indices $0 \le k_1, k_2 \le 2^j - 1$. Moreover, we denote $\mathbf{D} = \operatorname{diag}(D_1, ..., D_L)$. Finally, by approximating the prior covariance of the discretized problem by the ideal model we get

$$\|C_\Phi^{-1/2} \boldsymbol{\phi}\|_{(L^2)^L}^2 = \sum_{\ell=1}^L \|C_\ell^{-1/2} \phi_\ell\|_{L^2}^2 \simeq \sum_{\ell=1}^L (D_\ell \mathbf{c}_\ell, \mathbf{c}_\ell)_2 = (\mathbf{D}\mathbf{c}, \mathbf{c})_2 \quad (11)$$

for $\boldsymbol{\phi} = (\phi_\ell)_{\ell=1}^L$ and the vector of wavelet coefficients $\mathbf{c} = (\mathbf{c}_\ell)_{\ell=1}^L$, where $\mathbf{c}_\ell$ is the vector of wavelet coefficients associated to the wavelet decomposition (6) of ϕ_ℓ.

We point out that similar approximation techniques can be applied to more general classes of power laws. This is valuable since experiments suggest that deviations from the Kolmogorov model may appear [16]. More generalized random processes have been considered, e.g., in [18].

3. Algorithm

3.A. Dual-domain discretization

Each layer $\ell = 1, ..., L$ is discretized on an equidistant square grid of $n_\ell \times n_\ell$ points with spacing δ_ℓ. The points must be chosen such that $n_\ell = 2^{J_\ell}$ for some $J_\ell \in \mathbb{N}$ and the regions observed by all propagation operators P_m^{LGS} and $P_{m'}^{\mathrm{NGS}}$ are contained within the convex hull of the discretization grid. The first restriction is necessary to define the wavelet transform on the grid.

The layer grid induces $(n_\ell - 1)^2$ square subdomains with equal area. A continuous piecewise bilinear function, i.e., a continuous function on the whole grid that is bilinear on each subdomain, is defined on the grid. Such a function is uniquely determined by n_ℓ^2 function values at the grid points. We denote the vector of these values by b_ℓ.

Further, to each layer grid we associate a wavelet transform operator W_ℓ. The operator is defined as a sequential application of operators (7) for $j = J_\ell - 1, ..., 0$. The corresponding inverse wavelet transform operator W_ℓ^{-1} is defined as a sequential application of operators (8) for $j = 0, ..., J_\ell - 1$. Thus, the relation between the bilinear coefficients and the wavelet coefficients c_ℓ is given by $c_\ell = \delta_\ell W_\ell b_\ell$. We use the notation

$$\mathbf{W} = \operatorname{diag}(\delta_1 W_1, ..., \delta_L W_L) \quad \text{and} \quad \mathbf{W}^{-1} = \operatorname{diag}(\delta_1^{-1} W_1^{-1}, ..., \delta_L^{-1} W_L^{-1}) \quad (12)$$

for the direct and inverse wavelet transform of all layers. Observe that the transpose of the inverse transform coincides with the direct transform operator up to the grid spacing coefficients, $(\mathbf{W}^{-1})^T = \operatorname{diag}(\delta_1^{-1} W_1, ..., \delta_L^{-1} W_L)$.

Next, we discretize the incoming wavefronts $m = 1, ..., M$ in each guide star direction on an equidistant square grid of $n_m \times n_m$ points. The wavefront grid models the Fried geometry corresponding to the corner points of the WFS subapertures. On this grid we define continuous piecewise bilinear functions of incoming wavefronts.

The geometric propagation operator P_m in the bilinear setting interacts between the layer coefficients and the wavefront coefficients. It is defined by a concatenation of interpolation operators,

$$P_m = \begin{pmatrix} I_{m1} & \dots & I_{mL} \end{pmatrix}. \quad (13)$$

The operator $I_{m\ell} : \mathbb{R}^{n_\ell^2} \to \mathbb{R}^{n_m^2}$ interpolates the layer ℓ onto the wavefront component of the incoming wavefront m at layer ℓ. The wavefront grid is aligned to the layer grid according to the direction of the guide star, the altitude of the layer and the cone effect scaling if $1 \le m \le M_{\mathrm{lgs}}$.

In Figure 2 we demonstrate the action of the operator $I_{m\ell}$ graphically for a high altitude layer: an interpolation of piecewise bilinear domain is a concatenation of two linear interpolation steps in the x- and y-directions. Notice how due to the cone effect, the spacing of the projected wavefront grid is smaller than the spacing of the layer grid.

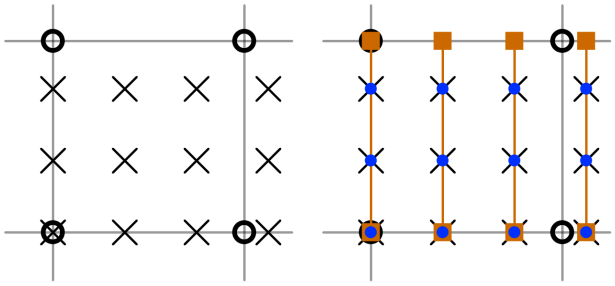

Fig. 2. Left: The operator $I_{m\ell}$ produces a bilinear interpolation of the layer grid (circles) to the incoming wavefront grid (crosses). Right: The operation is factorized into a linear interpolation in the x-direction (orange squares) followed by a linear interpolation in the y-direction (blue circles).

Finally, the Shack-Hartmann operator in the bilinear setting is defined as for the Fried geometry.

Altogether, the minimization problem (3) formulated for wavelet coefficients of turbulence layers reduces to solving a system of linear equations

$$\underbrace{((\mathbf{W}^{-1})^T\mathbf{A}^T\mathbf{C}_{\mathcal{E}}^{-1}\mathbf{A}\mathbf{W}^{-1}+\alpha\mathbf{D})}_{\mathbf{M}}\mathbf{c}=(\mathbf{W}^{-1})^T\mathbf{A}^T\mathbf{C}_{\mathcal{E}}^{-1}\mathbf{s}. \tag{14}$$

Here, $\mathbf{A}$ is the concatenation of Shack-Hartmann and interpolation operators (13) in the piecewise bilinear domain, $\mathbf{W}^{-1}$ is the inverse DWT, $\mathbf{C}_{\mathcal{E}}^{-1}$ is the inverse noise covariance (4) and $\mathbf{D}$ is the diagonal operator from the approximation (11). The operator on the left-hand side is abbreviated by $\mathbf{M}$ for convenience.

As we only have an equivalence in (11) instead of equality, a scalar parameter α is introduced for fine-tuning the balance between the fitting and the regularizing terms. The effect of the parameter has already been studied in [11].

3.B. Preconditioning

In this work we use the frequency-dependent preconditioning strategy developed for the wavelet method in [27]. There, a modified Jacobi preconditioner is used to weigh the low and high frequency components differently. The benefit of such an approach is an increased robustness and overall stability of the method. Naturally, the number of iterations is reduced as well. The preconditioner is formulated as a diagonal matrix adding very low computational cost.

Below, we introduce a modification to the technique used in [27]. Instead of using a global thresholding parameter for all layers, the thresholding on each layer is performed separately according to the wavelet scale. We define a set of preconditioners

$$\mathbf{J}_j=\mathrm{diag}((\mathbf{W}^{-1})^T\mathbf{A}^T\mathbf{C}_{\mathcal{E}}^{-1}\mathbf{A}\mathbf{W}^{-1})+\alpha\mathbf{D}_j, \tag{15}$$

where

$$\mathbf{D}_j=\mathrm{diag}(\max(\tau_{1,j}I,D_1),...,\max(\tau_{L,j}I,D_L)),$$

and the parameters $\tau_{\ell,j}$ are given by the value of $\mathbf{D}_\ell$ at scale j, i.e., by

$$\tau_{\ell,j}=c_\rho(h_\ell)^{-1}\left(K_{out}^{-\frac{11}{3}}+2^{\frac{11}{3}j}\right).$$

Above, the maximum is taken component-wise. Notice that $\mathbf{J}$ depends on the scale where thresholding is performed. In the following we study how this choice affects the convergence.

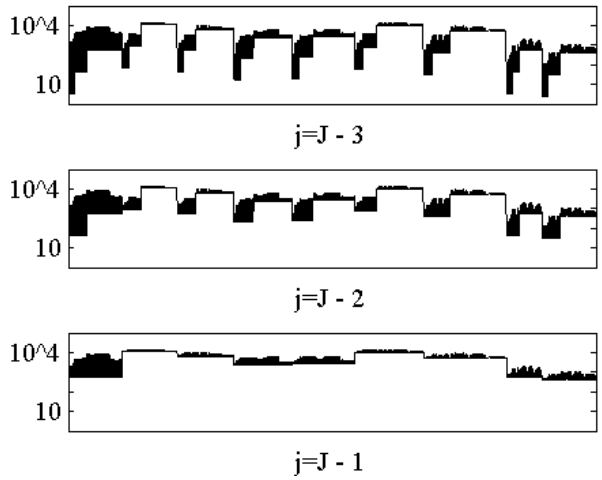

Fig. 3. The effect of thresholding at scale j to the preconditioner is illustrated. The values on the diagonal of $\mathbf{J}_j$ are plotted for three thresholding scales $j=J_\ell-3, J_\ell-2$ and $J_\ell-1$. The y-axis corresponds to the values on a logarithmic scale. The x-axis stands for the global index of the wavelet coefficients. These are ordered from left to right starting from the ground layer up to the layer with highest altitude; for each layer the values are plotted from left to right starting with the coarsest scale.

In Figure 3 we plot the diagonal values of the preconditioner for $j=J_\ell-3, J_\ell-2$ and $J_\ell-1$, where $J_\ell=7$ scales.

3.C. Ground layer multi-scale method

Convergence of any iterative scheme can be improved by finding a better initial value. Iterative methods in AO benefit from the possibility of the warm restart technique, i.e., using the solution generated in the previous closed or open loop iteration. In our algorithm we implement an additional step, where the warm restart initial value is improved at the coarse wavelet scales.

Due to the multi scale structure of wavelets the minimization problem (14) can be posed at the coarse scales by truncating the wavelet decomposition of the unknown. To keep the computational cost low, we define this intermediate problem only for the ground layer, where the highest turbulence strength is located.

The resulting ground layer multi-scale method is described in the following algorithm.

Algorithm 3.1 (Ground layer multi-scale method)**.**

In: wavelet coefficients $\mathbf{c}_0$, residual $\mathbf{r}_0 = \mathbf{b} - \mathbf{M}\mathbf{c}_0$

Out: wavelet coefficients $\mathbf{c}_1$, residual $\mathbf{r}_1 = \mathbf{b} - \mathbf{M}\mathbf{c}_1$

1. $\tilde{\mathbf{r}} = \mathbf{R}\mathbf{r}_0$ (restrict residual to coarse scale)
2. $\tilde{\mathbf{v}} = \widetilde{\mathbf{M}}^{-1}\tilde{\mathbf{r}}$ (compute solution on coarse scale)
3. $\mathbf{v} = \mathbf{R}^T\tilde{\mathbf{v}}$ (prolongate solution)
4. $\mathbf{c}_1 = \mathbf{c}_0 + \mathbf{v}$ (update coefficients)
5. $\mathbf{r}_1 = \mathbf{b} - \mathbf{M}\mathbf{c}_1 = \mathbf{r}_0 - \mathbf{M}\mathbf{v}$ (update residual)

In the wavelet domain, the restriction matrix $\mathbf{R} = \begin{pmatrix} I & 0 \end{pmatrix}$ selects only the wavelet coefficients of the lower scales. The prolongation matrix $\mathbf{R}^T$ pads the coarse scale wavelet coefficients with zeros at high scales. The matrix $\widetilde{\mathbf{M}} = \mathbf{R}^T\mathbf{M}\mathbf{R}$ is a coarse scale approximation of $\mathbf{M}$ defined in equation (14). Symmetry and positive definiteness properties of $\mathbf{M}$ are inherited by $\widetilde{\mathbf{M}}$.

For a small dimensional $\widetilde{\mathbf{M}}$, it is computationally reasonable to directly invert the operator. In this case, the second step is a matrix-vector multiplication. Moreover, in step 5 the zero-padding of vector $\mathbf{v}$ keeps the computational cost low. This is analysed in more detail in Section 4.B.

The GLMS method has certain similarities to a multi-grid method. Like a multi-grid method, the problem is defined on several grids. However, whereas the goal of a multi-grid method is to eliminate the high-frequency error components using smoothing iterations, such as Jacobi or Gauss-Seidel iterations, on each grid, the goal of our multi-scale adaptation is to improve the low-frequency solution before the PCG is applied to the full problem. This is a well-known strategy for ill-posed problems solved by the CG algorithm on multiple grids, see, e.g., [12]. A multi-grid method has been studied in terms of AO in [8].

3.D. Algorithm and control

The full reconstuctor for an open loop system, such as the MOAO, is described in the following algorithm.

Algorithm 3.2 (Open loop reconstructor)**.**

In: data vector $\mathbf{s}$, mirror shapes $\mathbf{a}_0$, wavelet coefficients $\mathbf{c}_0$, wavelet residual $\mathbf{r}_0 = \mathbf{b}_0 - \mathbf{M}\mathbf{c}_0$, right-hand side $\mathbf{b}_0$

Out: mirror shapes $\mathbf{a}_1$, wavelet coefficients $\mathbf{c}_1$, wavelet residual $\mathbf{r}_1 = \mathbf{b}_1 - \mathbf{M}\mathbf{c}_1$, right-hand side $\mathbf{b}_1$

1. $\mathbf{b}_1 = (\mathbf{W}^{-1})^T\mathbf{A}^T\mathbf{C}_{\mathcal{E}}^{-1}\mathbf{s}$ (compute new RHS)
2. $\mathbf{r}_{0,1} = \mathbf{b}_1 - \mathbf{M}\mathbf{c}_0 = \mathbf{r}_0 + (\mathbf{b}_1 - \mathbf{b}_0)$ (update residual)
3. $(\mathbf{c}_{0,2}, \mathbf{r}_{0,2}) = \mathrm{GLMS}(\mathbf{c}_0, \mathbf{r}_{0,1})$ (update initial guess)
4. $(\mathbf{c}_1, \mathbf{r}_1) = \mathrm{PCG}(\mathbf{c}_{0,2}, \mathbf{r}_{0,2})$ (solve eq. (14))
5. $\mathbf{a} = \mathbf{P}\mathbf{W}^{-1}\mathbf{c}_1$ (compute mirror shape)
6. $\mathbf{a}_1 = (1 - \mathrm{gain}) \cdot \mathbf{a}_0 + \mathrm{gain} \cdot \mathbf{a}$ (control)

Above, $\mathbf{P}$ is the concatenation of bilinear projection operators (13) in the directions of interest. In the third step we utilize Algorithm 3.1. Further, the fourth step refers to the standard PCG algorithm, see e.g., [22], with a precomputed residual. The preconditioner is given by (15). Observe that the forward operator $\mathbf{M}$ is not precomputed as single matrix, but, instead, is applied according to the factorization in (14). We discuss the computational cost estimates of applying the operator in Section 4.B.

The open loop algorithm can be easily extended to a closed loop setting by using the pseudo-open loop control (POLC) [5]. In this case, an additional step for computing the pseudo-open loop data is included.

4. Numerical simulation results

4.A. System description

For the simulations, we use the proposed MOAO configuration for the E-ELT. The telescope diameter is 42 m, of which roughly 28 percent are obstructed. There are six LGSs positioned in a circle with a diameter of 7.5 arcmins at 90 km altitude with 84×84 SH-WFS. The LGSs suffer from tip-tilt uncertainty and spot elongation. Four laser side-launch positions are used. Additionally, there are three NGSs positioned in a circle with a diameter of 10 arcmins with 84×84 SH-WFS. LGS and NGS flux varies between 5 and 500 photons per subaperture and frame. The DM is modeled by a piecewise bilinear function with a total number of 5402 active actuators.

We simulate one second of the nine layer ESO median seeing atmosphere on OCTOPUS [13], i.e., 500 steps in open loop. The measurements suffer from photon noise. A two-step delay is observed. As the quality evaluation criteria, we use the long exposure (LE) Strehl ratio in K band (for a wavelength of 2200 nm). The direction of interest is the zenith.

For the reconstruction we use Algorithm 3.2. The method is configured to reconstruct nine layers of the simulated atmosphere. Variable parameters of the method are the number of PCG iterations, the number of GLMS scales, the regularization parameter α, a spot elongation tuning parameter, the cut-off scale index j of the preconditioner and the gain.

We compare our method to the MVM in terms of speed and to FrIM in terms of quality. The FrIM reference results are provided by the ESO. In this configuration the FrIM provides a quality benchmark by using up to 100 CG iterations.

4.B. Computational cost estimates

Numerical efficiency of a method depends on several factors. The number of floating point operations (FLOP) is one of them. The FLOP determines the number of additions and multiplications of real numbers.

For a modern system, however, FLOP is not the only indicator of numerical efficiency. Current CPUs use a multi-core architecture with a shared memory cache. Thus, properties, such as parallelization and small memory consumption, play a vital role as well.

In this work we refrain from confining to a specific processor architecture. Instead we focus on the theoretical capabilities of the method in terms of computational estimates. We show that the method has linear complexity and derive the complexity constants for the specific MOAO system. Apart from that, we analyze the memory consumption and parallizability of the algorithm. Further, we compare the performance of the method with the MVM in terms of FLOP and memory and point out the compactness of the reconstructor when multiple mirrors are utilized.

The computational performance of Algorithm 3.2 can be analyzed by investigating the components of the left-hand side operator $\mathbf{M}$, i.e., the operators $(\mathbf{W}^{-1})^T, \mathbf{A}^T, \mathbf{C}_{\mathcal{E}}^{-1}, \mathbf{A}, \mathbf{W}^{-1}$ and $\mathbf{D}$. Each of these operators has a block decomposition associated either to layers or guide star directions. The analysis of the blocks gives a better understanding of the numerical cost of the method.

The block representation of the operators has three main advantages. First, the computational cost of applying each block independently is significantly lower than applying the operator in the form of a precomputed matrix.

Second, each block can be represented as a routine instead of a sparse matrix. This reduces both the amount of memory needed to store the operator and the computational cost of applying the operator.

Finally, the block representation induces a natural parallelization of the operators. The operators $(\mathbf{W}^{-1})^T, \mathbf{W}^{-1}$ and $\mathbf{D}$ can be applied in parallel with respect to layers. The operators $\mathbf{C}_{\mathcal{E}}^{-1}$ and $\mathbf{A}$ can be applied in parallel with respect to guide stars. In $\mathbf{A}^T$ the transposed Shack-Hartmann operator can first be applied in parallel with respect to guide stars, after which the transposed interpolation operations can be applied for each layer separately.

We express the numerical cost in terms of the layer grid size and the number of subapertures of the WFS. As in Section 3.A, the layers are discretized on grids with $n_\ell \times n_\ell$ points and the incoming wavefront grids are defined by $n_m \times n_m$ points. For convenience we use the notation, $N_\ell = n_\ell^2$, $N_m = n_m^2$ and $N_s = (n_m - 1)^2$ to denote the total number of points on a layer and a total number of wavefront discretization points and subapertures associated to a WFS, respectively. Moreover, we define $n_s = n_m - 1$.

In Table 1 we present the FLOP and the memory consumption of the blocks. We distinguish between the memory used to store an operator and the temporary memory used by the operator during the computation.

Observe how the Shack-Hartmann operator is more efficient in terms of memory and numerical cost as a routine, rather than as a sparse matrix. The operator in a sparse matrix form consumes $8N_s$ entries in memory and requires $14N_s$ floating point operations to be applied.

Table 1. Computational cost and memory usage estimates of the operator blocks. In the case of asymptotically constant memory consumption, we write 0.

Operator	FLOP	Memory	Temporary memory
Γ_m	$6N_s + 2n_s$	0	$N_s + n_m$
Γ_m^T	$6N_s + 2n_s$	0	0
$T'\widehat{C}_m^{-1}T'^{(1)}$	$14N_s$	$3N_s^{(3)} + n_s^{(4)}$	0
$\widehat{C}_m^{-1(2)}$	$2N_s$	$n_s^{(4)}$	0
$I_{m\ell}{}^{(5)}$	$[5.2N_m, 7N_m]^{(6)}$	$4n_m$	$[0.4N_m, N_m]^{(6)}$
$I_{m\ell}^T{}^{(5)}$	$[7N_m, 10N_m]^{(6)}$	$4n_m$	$[0.4N_m, N_m]^{(6)}$
αD_ℓ	N_ℓ	0	0
W_ℓ, W_ℓ^{-1}	$117.3N_\ell{}^{(7)}$	0	N_ℓ

(1) – for $1 \le m \le M_{\text{lgs}}$
(2) – for $M_{\text{lgs}} < m \le M$
(3) – per laser launch position
(4) – mask information, stored once for all WFS
(5) – for $1 < \ell \le L$; layer 1 has no computational cost
(6) – varies due to the cone effect, exact coefficients depend on the system configuration
(7) – Daubechies 3 wavelets have filter length 6; FLOP for filter length N: $8(2N-1)(4N_\ell - 1)/3$

The spot elongation covariance operator $\widehat{C}_m^{-1}$ is stored as three vectors of size N_s, corresponding to the x-x, x-y and y-y noise correlations. These vectors need to be stored only once per laser launch position. The interpolation operators $I_{m\ell}$ vary in computational cost and temporary memory usage due to the cone effect. For higher altitude layers, the observed region is smaller and the interpolation step is cheaper.

Using the cost estimates in Table 1 it is easy to derive the costs of the components of $\mathbf{M}$. We now focus to the specific MOAO system with $M = 9$ guide stars, where we have $M_{\text{lgs}} = 6$ LGSs and $L = 9$ layers. In our implementation the algorithm is parallelized on $\max(L, M) = 9$ cores. For all layers, N_ℓ is constant.

The corresponding computational costs of the component operators are given in Table 2. In addition to the

total number of floating point operations we also state the maximum FLOP on one processor core for the parallelized operators.

Observe that an efficient parallelization of the combined operator $\mathbf{M}$ is only possible on a shared memory architecture. For instance, the inverse DWT is performed on each layer in parallel. Then, the operator $\mathbf{A}$ writes to measurement vectors and reads from all layers simultaneously. On a shared memory architecture it is possible to avoid data communication between these two operations. Layer vectors can be read by different processors during the operation $\mathbf{A}$ without the need to copy the layer vectors to individual processors. Moving vectors between processors is prohibitively expensive for a real-time system.

Table 2. Computational cost estimates of components of $\mathbf{M}$.

Operator	FLOP	FLOP one core	Parallel
$\mathbf{W}^{-1}, (\mathbf{W}^{-1})^T$	$1065N_\ell$	$118N_\ell$	layers
$\mathbf{A}$	$587N_s$	$70N_s$	guide stars
$\mathbf{A}^T$	$774N_s$	$86N_s$	guide stars and layers
$\mathbf{C}_\mathcal{E}^{-1}$	$90N_s$	$14N_s$	guide stars
$\alpha\mathbf{D}$	$9N_\ell$	N_ℓ	layers

The total cost of one PCG iteration is given by the cost of applying $\mathbf{M}$, the cost of applying the diagonal preconditioner and nine additional vector operations. Each operation can be carried out in parallel with respect to layers or guide stars for the wavelet method. Thus, a PCG iteration is parallelizable.

The computational cost of the GLMS Algorithm 3.1 depends on the number of scales that are chosen in the coarse scale approximation, $\tilde{J}_1$. Let $\tilde{N}_1 = 2^{2\tilde{J}_1}$ be the grid dimension of the coarse scale. Then, the computational cost of GLMS is $\tilde{N}_1^2 + 1311N_\ell + 918N_s$ without parallelization and at most $\tilde{N}_1^2/9 + 239N_\ell + 106N_s$ on one core. Note that the method still scales with linear complexity if $\tilde{N}_1$ is constant with respect to N_ℓ.

In Table 3 we present the computational complexity of a GLMS step compared to one PCG iteration. To illustrate the performance of GLMS we use fixed grids of $N_\ell = 128^2$ points and $N_s = 84^2$ subapertures and a varying coarse scale grid $\tilde{N}_1$. The cost is given in terms of N_ℓ. Memory consumption and parallelizability are taken into consideration.

It can be observed that in terms of FLOP, five scales of GLMS is still cheaper than one PCG iteration. However, the memory consumption increases rapidly with the number of scales, due to the storage of an inverted matrix. At four scales the increase in memory consumption is still negligible.

In Section 4.C.3 we show how much performance can be gained by using more scales in GLMS. In fact, it is possible to reduce the number of PCG iterations by using more scales. The trade-off of the reduced computational cost of the overall method is the increased memory consumption. The exact balance between these two factors depends on the computational architecture. We concentrate on small memory consumption and therefore choose four scales for the further discussion.

Table 3. Computational cost and memory usage estimates of a GLMS step.

No. of scales $\tilde{J}_1$	grid size $\tilde{N}_1$	FLOP	FLOP one core	Memory$^{(1)}$
3	8^2	$1707N_\ell$	$284N_\ell$	$144N_\ell$
4	16^2	$1711N_\ell$	$285N_\ell$	$148N_\ell$
5	32^2	$1771N_\ell$	$291N_\ell$	$208N_\ell$
6	64^2	$2731N_\ell$	$398N_\ell$	$1168N_\ell$
7	128^2	$18091N_\ell$	$2105N_\ell$	$16528N_\ell$
One PCG iteartion		$2854N_\ell$	$322N_\ell$	

$^{(1)}$ – memory consumption of the full algorithm

To summarize, we present in Table 4 the computational costs of the main steps of the Algorithm 3.2. The total cost and the memory consumption of the full reconstructor are also stated.

Table 4. Computational cost estimates.

Operation	FLOP	FLOP one core
Compute RHS	$1065N_\ell + 864N_s$	$118N_\ell + 100N_s$
Apply GLMS	$1311N_\ell + 918N_s$	$239N_\ell + 106N_s$
Apply $\mathbf{M}$	$2139N_\ell + 1451N_s$	$239N_\ell + 170N_s$
Apply preconditioner	$9N_\ell$	N_ℓ
One PCG iteration	$2229N_\ell + 1451N_s$	$249N_\ell + 170N_s$
Compute DM shape	$1065N_\ell + 63N_s$	$118N_\ell + 63N_s$
Total FLOP	$3441N_\ell + 1845N_s$ $+(2229N_\ell + 1451N_s)N_{\text{iter}}$	
Total FLOP one core	$625N_\ell + 269N_s$ $+(249N_\ell + 170N_s)N_{\text{iter}}$	
Total memory used	$90N_\ell + 125N_s$	

Observe that the total cost of the method depends on the number of iterations N_{iter}. In the following section we show that for the underlying MOAO system, eight iterations are sufficient to obtain a good reconstruction.

Using the estimates in Table 4 we can compare the computational performance of the method with the MVM. The underlying MOAO system uses a grid of $N_m = 85^2$ actuators and $N_s = 84^2$ subapertures. By taking the mask into account, the MVM size for one direction of interest is 5402×99900. This implies a computational cost of $1.079 \cdot 10^9$ operations. The memory consumption for storing the matrix is $5.397 \cdot 10^8$.

In contrast, the cost of the wavelet method depends on the layer grid size. We choose $N_\ell = 128^2$ as the number of layer discretization points. Then, the overall cost of the wavelet method for one direction of interest with eight iterations is $4.435 \cdot 10^8$. The corresponding

memory consumption is $2.357 \cdot 10^6$.

Thus, the wavelet method uses 2.4 times less FLOP than the MVM for the MOAO system with one DM. More significantly, the MVM consumes 230 times more memory than the wavelet method. In terms of a single precision memory format of four bytes, this corresponds to 9.25 megabytes for the wavelet method compared to 2 gigabytes for MVM. Thus, the wavelet method has the advantage of completely fitting into the cache of a modern CPU.

The advantage of the wavelet algorithm over the MVM is further highlighted by the scalability of the method with respect to the number of DMs. The wavelet algorithm can compute multiple DM shapes on a single system. The atmospheric reconstruction is computationally the heaviest part of the wavelet algorithm. After the atmospheric reconstruction has been performed, one CPU core is needed to compute a mirror shape (the projection operation in step 5 of Algorithm 3.2). Thus, several mirrors can be computed simultaneously without any impact on the computational performance of the algorithm. In comparison, the MVM requires a full computing system for each additional mirror.

Apart from the CPU, an interesting architecture for the wavelet method is the GPU. The main operations of the wavelet method are bilinear interpolation, vector operations and the DWT. These operations are particularly efficient on a GPU. For a study on scalability of the DWT on a GPU we refer to, e.g., [7]. We leave the discussion on comparison of computational architectures for the wavelet method for future work.

4.C. Quality tests

In this section we demonstrate the qualitative performance of the wavelet method. In all test cases, all layers are discretized over $N_\ell = 128^2$ points. This corresponds to a wavelet decomposition of $J_\ell = 7$ scales. Grid spacing of layers $\ell = 1, ..., 7$ is $\delta_\ell = 0.5$ m. Due to the wide angle of separation of the guide stars, the two uppermost layers have the grid spacing of $\delta_\ell = 1$ m over the 128^2 points. The number of subapertures is defined by the specifications of the AO system, i.e., $N_s = 84^2$.

4.C.1. Performance with noisy data

We demonstrate the performance of the method with respect to the change in flux. The LGS and NGS flux varies between 5 and 500 photons per subaperture and frame.

The reference method for this example is the FrIM [23] and the results were provided by the ESO. Here, the FrIM is applied without a preconditioner using the regular CG algorithm. The method uses up to 100 iterations.

The wavelet method uses eight PCG iterations independent of the flux. We use the scale dependent preconditioner (15), where $j = J_\ell - 2$ for all layers. The regularization parameter α, spot elongation tuning parameter α_η and the gain are adjusted for each flux level separately. These parameters are outlined in Table 5. In GLMS, the coarse grid of the ground layer is set to $\tilde{J}_1 = 4$ scales, i.e., a grid of $\tilde{N}_1 = 256$ points is used.

flux	500	100	20	10	5
α	24	4	1	1	1
α_η	0.2	0.4	0.8	0.8	0.8
gain	0.5	0.4	0.2	0.2	0.2

Table 5. Parameter settings of the wavelet method for varying LGS and NGS flux.

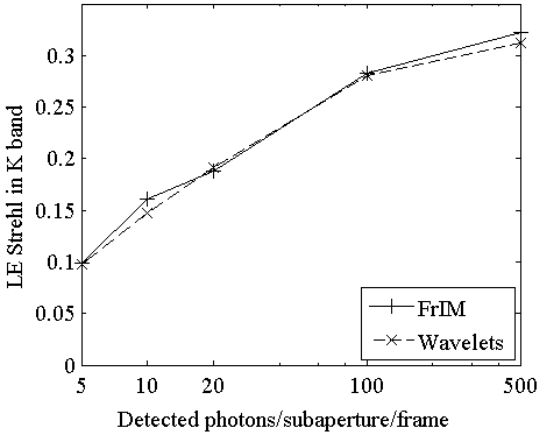

Fig. 4. LE Strehl vs. detected number of photons per subaperture and frame.

In Figure 4, a comparison in quality of the reconstruction is plotted for the wavelet method and the FrIM. The wavelet method performs similar to the FrIM in all but two cases, 10 and 500 photon flux, where it is slightly worse. In the 500 photon case, the benchmark set by the FrIM with up to 100 CG iterations can be attained by the wavelet method with more iterations of the PCG.

4.C.2. Convergence of the method

In this test we demonstrate the convergence properties of the method. The LGS and NGS flux is fixed to 100 photons per subaperture and frame.

The standard parameters for the 100 photons case in Table 5 are used. The number of iterations varies between 1 and 10. As before, $\tilde{J}_1 = 4$ scales are used to represent the coarse grid of the ground layer. The scale-dependent preconditioner is used with $j = J_\ell - 2$.

In Figure 5 we plot the LE Strehl of the reconstructor using a different number of PCG iterations. The wavelet method reaches a high performance with seven iterations. Increasing the number of iterations above eight brings only a minor improvement in the result.

4.C.3. Performance of GLMS

In this example we compare the performance of the reconstructor with different number of scales for the GLMS method. As in the previous test, the LGS and

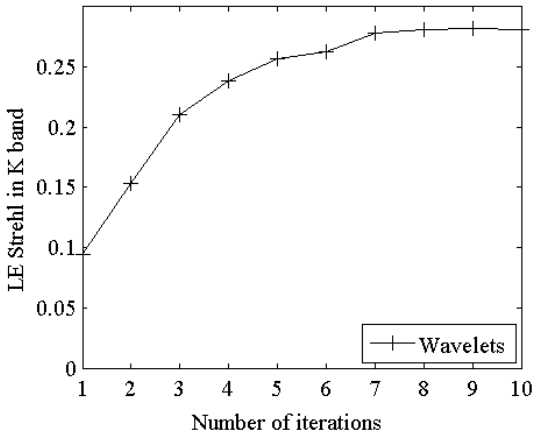

Fig. 5. Performance of the algorithm with different number of iterations.

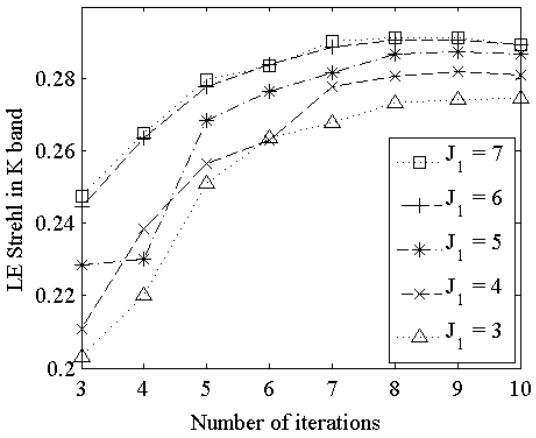

Fig. 6. Sensitivity of the method with different GLMS scales.

NGS fluxes are fixed to 100 photons per subaperture and frame.

Parameters for the 100 photon case in Table 5 are used. The scale-dependent preconditioner is used with $j = J_\ell - 2$. The number of scales $\tilde{J}_1$ in the GLMS varies between three and seven.

In Figure 6 the performance of the method is shown for different numbers of GLMS scales. The corresponding FLOP and memory comparisons are found in Table 3.

Generally, using more scales in the GLMS implies a better qualitative performance of the method. For instance, a similar result with $\tilde{J}_1 = 4$ scales and eight iterations is attained with five iterations and $\tilde{J}_1 = 6$ scales. However, at six scales, the memory consumption of the overall method is roughly eight times larger. The actual trade-off between computational cost and the memory depends on the architecture of the computational system.

No significant improvement is gained by increasing the number of scales from six to seven.

4.C.4. Influence of the preconditioner scales

The choice of the thresholding scale j of the preconditioner (15) has a significant influence on the behavior of the overall method. When the value is set too high, the method shows a slow, but stable convergence. When the value is set too low, an instability during first 5 iterations can be observed. At the optimal level, the method shows a fast and stable convergence.

We illustrate this phenomenon in the 100 photon case example. In Figure 7 we plot the LE Strehl for PCG with varying number of iterations for different values of j. The optimal choice for the thresholding scale is $j = J_\ell - 2$. With $j = J_\ell - 1$ the performance is stable, but convergence is slow. Using $j = J_\ell - 3$, the performance of the method is unstable. The diagonal values of the three preconditioners are plotted in Figure 3.

Similar behavior has already been observed for the

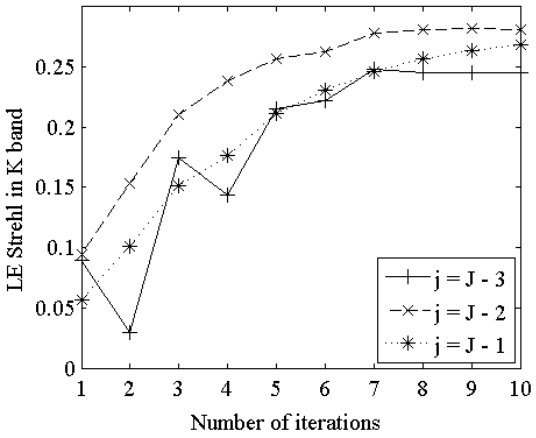

Fig. 7. Sensitivity of the method with respect to the preconditioner thresholding scale j. Thresholding is at three last scales (solid curve), two last scales (dashed curve) and last scale (dotted curve).

MCAO reconstructor with the frequency dependent preconditioner in [27].

5. Conclusions

Here we have introduced a novel dual-domain wavelet algorithm which is based on our previous work described in [11, 27]. It has been demonstrated by numerical simulations that the method is robust with respect to the noise and practical phenomena like the tip-tilt uncertainty and the spot elongation. Moreover, it is shown that the convergence can be substantially accelerated by the techniques we call GLMS and the frequency-dependent preconditioner [27]. The resulting method is faster than the original method introduced in [11]. We established that the method needs eight iterations in the open loop for the proposed MOAO system for the E-ELT. Most importantly, we have provided a detailed computational cost analysis in Section 4.B and shown that the method is

linear with respect to the number of layer discretization points. Moreover, an MOAO system with multiple mirrors can be implemented in a more compact way using the method than with an MVM.

In view of the results, the dual-domain wavelet algorithm is a promising method when solving the atmospheric tomography problem. However, we would like to continue our work on reducing the number of iterations with more elaborate schemes. In this regard, the GLMS could be extended by including the coarse scales on other layers into the problem.

Finally, the choice of computational architecture can have a substantial impact on results when comparing any two algorithms. The future work includes to assess how the wavelet method performs and how well it can be parallelized on different computational platforms like the GPUs mentioned in Section 4.

Acknowledgements

This work was done in the framework of the project *Mathematical Algorithms and Software for E-ELT Adaptive Optics*. The project is in cooperation with the ESO and is funded by the Austrian Federal Ministry of Science and Research. The authors are grateful to the Austrian Adaptive Optics team for support. Special thanks go to Curtis R. Vogel and Kirk Soodhalter for valuable suggestions. Helin was partly funded by the project ERC-2010 Advanced Grant, 267700.

References

[1] C. Béchet, M. Le Louarn, R. Clare, M. Tallon, I. Tallon-Bosc, and É. Thiébaut. Closed-loop ground layer adaptive optics simulations with elongated spots: impact of modeling noise correlations. In *AO4ELT Proceedings*, page 03004, 2010.

[2] W. Dahmen. Multiscale analysis, approximation, and interpolation spaces. In *Approximation theory VIII, Vol. 2 (College Station, TX, 1995)*, volume 6 of *Ser. Approx. Decompos.* World Sci. Publ., River Edge, NJ, 1995.

[3] I. Daubechies. *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, Pa., 1992.

[4] B. Ellerbroek. Efficient computation of minimum-variance wave-front reconstructors with sparse matrix techniques. *J. Opt. Soc. Am.*, 19(9):1803, 2002.

[5] B. Ellerbroek and C. R. Vogel. Simulations of closed-loop wavefront reconstruction for multiconjugate adaptive optics on giant telescopes. *Proc. SPIE*, 5169:206–217, 2003.

[6] B. Ellerbroek and C. R. Vogel. Inverse problems in astronomical adaptive optics. *Inverse Problems*, 25:1–37, 2009.

[7] J. Franco, G. Bernabe, J. Fernandez, and M.E. Acacio. A parallel implementation of the 2d wavelet transform using cuda. In *Parallel, Distributed and Network-based Processing, 2009 17th Euromicro International Conference on*, pages 111–118, 2009.

[8] L. Gilles, B. Ellerbroek, and C. R. Vogel. Preconditioned conjugate gradient wave-front reconstructors for multiconjugate adaptive optics. *Appl. Opt.*, 42(26):5233, 2003.

[9] P. J. Hampton, P. Agathoklis, R. Conan, and C. Bradley. Closed-loop control of a woofer–tweeter adaptive optics system using wavelet-based phase reconstruction. *J. Opt. Soc. Am. A*, 27(11):A145–A156, Nov. 2010.

[10] P. J. Hampton and C. Bradley. A new wave-front reconstruction method for adaptive optics systems using wavelets. *Selected Topics in Signal Processing, IEEE Journal of*, 2(5):781 –792, Oct. 2008.

[11] T. Helin and M. Yudytskiy. Wavelet methods in multi-conjugate adaptive optics. *Inverse Problems*, 29(8):085003, 2013.

[12] E. Klann, R. Ramlau, and L. Reichel. Wavelet-based multilevel methods for linear ill-posed problems. *BIT Numerical Mathematics*, 2011.

[13] M. Le Louarn, C. Vérinaud, V. Korkiakoski, N. Hubin, and E. Marchetti. Adaptive optics simulations for the European Extremely Large Telescope. In *Proc. SPIE 6272, Advances in Adaptive Optics II*, 2006.

[14] E. Marchetti, N. Hubin, E. Fedrigo, et al. MAD the ESO multi-conjugate adaptive optics demonstrator. In *Proc. SPIE 4839, Adaptive Optical System Technologies II*, pages 317–328, 2003.

[15] Y. Meyer and D. H. Salinger. *Wavelets and Operators*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1992.

[16] T. W. Nicholls, G. D. Boreman, and J. C. Dainty. Use of a Shack–Hartmann wave-front sensor to measure deviations from a Kolmogorov phase spectrum. *Opt. Lett.*, 20(24):2460–2462, Dec 1995.

[17] G.C. Papanicolaou and K. Sølna. Wavelet based estimation of local Kolmogorov turbulence. In *Theory and applications of long-range dependence*, pages 473–505. Birkhäuser Boston, Boston, MA, 2003.

[18] D.G. Pérez and L. Zunino. Generalized wavefront phase for non-kolmogorov turbulence. *Opt. Lett.*, 33(6):572–574, Mar 2008.

[19] R. Ramlau and M. Rosensteiner. An efficient solution to the atmospheric turbulence tomography problem using Kaczmarz iteration. *Inverse Problems*, 28(9):095004, 2012.

[20] M. Roggeman and B. Welsh. *Imaging Through Turbulence*. Cambridge University Press, Cambridge, 1996.

[21] M. Rosensteiner and R. Ramlau. Kaczmarz algorithm for multiconjugated adaptive optics with laser guide stars. *J. Opt. Soc. Am. A*, 30(8):1680–1686, Aug 2013.

[22] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, second edition, April 2003.

[23] M. Tallon, I. Tallon-Bosc, C. Béchet, F. Momey, M. Fradin, and É. Thiébaut. Fractal iterative method for fast atmospheric tomography on extremely large telescopes. In *Proc. SPIE 7736, AO Systems II*, 2010.

[24] A. Tokovinin, M. Le Louarn, and M. Sarazin. Isoplanatism in a multiconjugate adaptive optics system. *J. Opt. Soc. Am.*, 17(10):1819, 2002.

[25] A. Tokovinin and E. Viard. Limiting precision of tomographic phase estimation. *J. Opt. Soc. Am.*, 18(4):873, 2001.

[26] Q. Yang, C. R. Vogel, and B. Ellerbroek. Fourier domain preconditioned conjugate gradient algorithm for atmospheric tomography. *Appl. Opt.*, 45(21):5281, 2006.

[27] M. Yudytskiy, T. Helin, and R. Ramlau. A frequency dependent preconditioned wavelet method for atmospheric tomography. In *preprint*, 2013.