Warm-started wavefront reconstruction for adaptive optics

Laurent Lessard,[1],* Matthew West,[2] Douglas MacMynowski,[3] and Sanjay Lall[1]

[1]*Department of Aeronautics and Astronautics, Stanford University, Durand Building, 496 Lomita Mall, Stanford, California 94305, USA*
[2]*Department of Mechanical Science and Engineering, University of Illinois at Urbana–Champaign, 1206 W. Green Street, Urbana, Illinois 61801, USA*
[3]*Department of Control and Dynamical Systems, California Institute of Technology, 1200 E. California Boulevard, Pasadena, California 91125, USA*
*Corresponding author: lessard@stanford.edu

Future extreme adaptive optics (ExAO) systems have been suggested with up to 10^5 sensors and actuators. We analyze the computational speed of iterative reconstruction algorithms for such large systems. We compare a total of 15 different scalable methods, including multigrid, preconditioned conjugate-gradient, and several new variants of these. Simulations on a 128×128 square sensor/actuator geometry using Taylor frozen-flow dynamics are carried out using both open-loop and closed-loop measurements, and algorithms are compared on a basis of the mean squared error and floating-point multiplications required. We also investigate the use of warm starting, where the most recent estimate is used to initialize the iterative scheme. In open-loop estimation or pseudo-open-loop control, warm starting provides a significant computational speedup; almost every algorithm tested converges in one iteration. In a standard closed-loop implementation, using a single iteration per time step, most algorithms give the minimum error even in cold start, and every algorithm gives the minimum error if warm started. The best algorithm is therefore the one with the smallest computational cost per iteration, not necessarily the one with the best quasi-static performance. © 2008 Optical Society of America

OCIS codes: 010.1080, 010.1330, 010.7350.

1. INTRODUCTION

Adaptive optics (AO) compensates for atmospheric turbulence in real time. There are two steps in the control algorithm. In the estimation step, a sensor array gathers noisy measurements and the phase aberration is reconstructed. In the control step, the deformable mirror (DM) is adjusted to cancel out these offsets. Small AO systems (10^3 sensors) have been built and used with great success, but future systems will be much larger (10^4–10^5 sensors). The estimation or reconstruction step is computationally expensive and typically scales poorly with the number of sensors. The optimal reconstruction matrix is dense, and thus it is expensive to store it and to perform a full matrix–vector multiplication at every time step.

Since the original paper on AO reconstruction [1], there has been much effort made to accelerate the estimation task. Sparse matrix factorization methods [2] have been used with a conjugate-gradient iterative scheme paired with either a multigrid (MG) [3,4] or a Fourier [5] preconditioner. This provides convergence only in a small number of iterations. Other methods of acceleration include a Fourier-domain (FD) reconstruction [6], a blended FD/preconditioned conjugate-gradient (FD-PCG) method [7], and a local control approach [8].

These methods are typically simulated in open loop using a *quasistatic* assumption: A single phase screen is used to generate a measurement, construct an estimate, and compute the error. In other words, an independent estimation problem is solved for every measurement. In this paper we incorporate a one-step delay in all open-loop simulations so that all errors are computed with respect to the next phase screen, rather than the current one. The closed-loop system is usually simulated using a simple integral gain, which is what we adopt here.

In this paper, we compare the computational performance of 15 iterative reconstructors by running numerical simulations in both open loop and closed loop of a large single-conjugate AO system (SCAO). The sensor sampling rate is chosen by examining the trade-off between sampling rate and minimum achievable estimation error.

We also examine the benefits of warm starting, where the most recent estimate is stored and used to initialize the subsequent iteration. The alternate approach is to cold start, where the iterative schemes are initialized at zero. We show that although the iterative reconstructors may converge at different rates in open loop with a cold start, they all require a single iteration per time step in closed loop. The best methods are the ones with the cheapest cost per iteration.

Similar solution techniques apply to the multiconjugate (MCAO) case, but the results are different. In MCAO, minimum-variance reconstructors (MVR) or some other type of regularization must be implemented to achieve an acceptable performance. In SCAO, there is no benefit to using MVR. A simple least-squares reconstructor has a virtually identical performance.

Standard closed-loop techniques give rise to stability

problems in MCAO [9,10]. We tried three different closed-loop architectures in SCAO and found them all to be stable for every method tested. We also show that the FD-PCG algorithm [5,11] performs well in the SCAO case.

It is worth noting that temporal atmospheric models do exist. A popular model is the Taylor frozen-flow approximation, which assumes the atmosphere is composed of stacked translating layers. State–space representations of this model have been proposed, which have led to formulations using the theory of optimal control [12–14]. However, these methods require that the layer wind velocities be either estimated or known *a priori*.

Another possibility is to project the phase onto a different basis, such as Zernike polynomials, and model each mode separately [15,16]. While the spatiotemporal statistics produced are correct, the associated cost of solving discrete algebraic Riccati equations and storing large dense covariance matrices is very high. Recent work by Poyneer *et al.* [14] avoids this problem by using a modal decomposition to decouple the Riccati equation, thereby greatly reducing the cost.

In Section 2, we describe the system model (geometry, sensors, and noise). In Section 3, we discuss least-squares and minimum-variance reconstructions. In Section 4, we discuss iterative schemes used to accelerate reconstruction. Finally, in Section 5, we analyze computational performance through simulation using a Taylor frozen-flow model and a square 128×128 sensor/actuator array.

2. SYSTEM MODEL

A. Sensor and Actuator Geometry
In optical telescopes the aperture is typically annular, so the DM and sensor array also share this shape. The algorithms we will discuss use sparse matrix operations and do not rely on a particular choice of geometry. Thus, for simplicity, we will assume a square sensor and actuator grid arranged in a SCAO setup with a single centered natural guide star. The actuators lie at the vertices of an $N \times N$ grid, and the sensors are aligned with the centers of the faces (Fried geometry); see Fig. 1. The *influence*
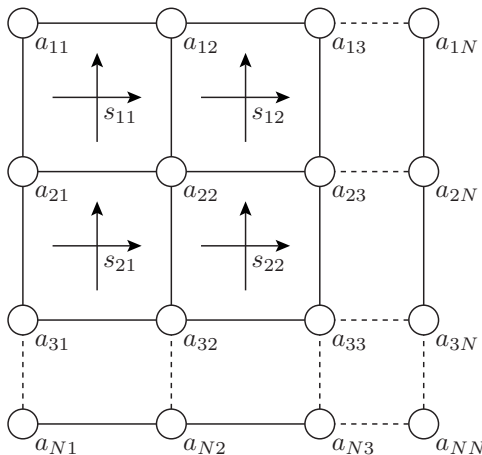

Fig. 1.   Sensor (s_{ij}) and actuator (a_{ij}) arrangement for a Fried geometry. We measure a noisy gradient of the phase at s_{ij}, and the goal is to estimate the phase at a_{ij}.

function is the identity, and there is no fitting step in the reconstruction, only an estimation step.

B. Measurement Equation
The most commonly used wavefront sensors measure either the local gradient or the local curvature in the incident wavefront. We will assume a Shack–Hartmann sensor array, which produces gradient measurements in both transverse directions. The measurements can be written as a linear function of the phases at each of the four nearest actuator locations plus Gaussian sensor noise [17]. For example, referring to Fig. 1,

$$y(s_{12}) = \frac{1}{2}\begin{bmatrix} \phi(a_{13}) - \phi(a_{12}) + \phi(a_{23}) - \phi(a_{22}) \\ \phi(a_{13}) + \phi(a_{12}) - \phi(a_{23}) - \phi(a_{22}) \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}.$$

If we collect all the phases into a vector ϕ arranged in a column-major ordering, $\phi_k = \phi(a_{ij})$, where $k = i + (j-1)N$, and we do the same for y and v, we can write the following linear equation relating the phase offsets to the measurements:

$$y = G\phi + v,$$

where $\phi \in \mathbb{R}^{N^2}$, $y \in \mathbb{R}^{2(N-1)^2}$, and v is a vector of zero-mean independent identically distributed Gaussian random variables with cov $v = \sigma^2 I$. Note that G is a sparse matrix (four nonzero entries per row). From now on, let $n = N^2$ denote the length of the vector ϕ.

The sensor measurements y depend only on relative phase measurements. To take advantage of this, we assume each set of phases is translated such that the average value is zero. In other words, define $x = (I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\phi$, where $\mathbf{1}$ is the $n \times 1$ vector of ones. This is equivalent to removing the *piston mode*, further discussed in Section 3. The same linear equations hold:

$$y = Gx + v. \tag{1}$$

In the next section, we will discuss how to model the noise variance σ^2.

C. Noise Model
What follows is a summary of the Shack–Hartmann sensor noise model developed in [[17], Section 5.C]. The sensor noise v is Gaussian to a good approximation, but its variance σ^2 depends on a variety of factors, including the guide star brightness and the noise in the CCD detectors:

$$\sigma^2 = \left(\frac{3\pi^2 K_g}{8}\right)^2 \frac{n_{\text{ph}} + n_{\text{bg}} + N_D \sigma_e^2}{n_{\text{ph}}^2}, \tag{2}$$

where n_{ph} and n_{bg} are the expected number of signal and background photoelectrons, respectively, hitting a single sensor per sampling interval; σ_e is the root-mean-square (RMS) read-out error in each of the N_D CCD detector pixels forming each sensor; and K_g is a correction factor that accounts for the small gaps between the sensors. Equation (2) has this particular form because the number of photons hitting a sensor is distributed as a Poisson process. The number of photoelectrons is related to the sample rate via

$$n_{\mathrm{ph}} = \frac{\eta \mu_{\mathrm{ph}} A}{b}, \qquad n_{\mathrm{bg}} = \frac{\eta \mu_{\mathrm{bg}} A}{b}, \qquad (3)$$

where b is the sample rate in hertz, μ_{ph} and μ_{bg} are photon fluxes in photons per square meters per second, A is the portion of the area of the main mirror that projects onto a single sensor subaperture (in square meters), and η is the product of the quantum efficiency of the CCD and the optical efficiency of the various mirrors and filters. Combining Eqs. (2) and (3) yields

$$\sigma^2 = \left(\frac{3\pi^2 K_g}{8} \right)^2 \left(\frac{\eta(\mu_{\mathrm{ph}} + \mu_{\mathrm{bg}})A}{b} + N_D \sigma_e^2 \right) \left(\frac{b}{\eta \mu_{\mathrm{ph}} A} \right)^2. \quad (4)$$

This equation relates the sensor noise variance σ^2 to the sample rate b and the photon arrival rates μ_{ph} and μ_{bg}, which are a function of the guide star brightness. Note that increasing the sampling rate makes the sensors noisier, and using a brighter guide star makes the sensors more accurate.

3. WAVEFRONT RECONSTRUCTION

A. Least-Squares Reconstruction
The objective is to estimate the phase at each of the actuator locations so that we may send this information to the DM and cancel the aberration. Only the relative phase is meaningful, so we can estimate x instead of ϕ. From Eq. (1),

$$y = Gx + v.$$

Since the system is overdetermined (roughly twice as many sensor measurements as actuators), one can minimize the norm of the residual. This is known as least-squares reconstruction. However, a regularization must be performed because G is not full rank: The piston (constant) and waffle (checkerboard) modes are in the null space of G and hence are unobservable. We construct phase estimates with a zero waffle mode because the waffle mode is small in practice, and we also make the piston mode zero because only relative phase offsets matter. If we let V denote the $n \times 2$ matrix whose columns are the normalized piston and waffle modes, the problem becomes finding $\hat{x}$ in order to

$$\text{minimize} \qquad \|y - G\hat{x}\|^2$$

$$\text{subject to} \qquad V^T \hat{x} = 0.$$

The solution to this problem is

$$\hat{x} = (G^T G + VV^T)^{-1} G^T y.$$

We can compute $\hat{x}$ by solving the linear system

$$(G^T G + VV^T)\hat{x} = G^T y. \qquad (5)$$

B. Minimum-Variance Reconstruction
In order to better estimate x, one must know something about its prior distribution. The first complete summary of the theory of propagation through atmospheric turbulence was by Tatarskii [18], based on the assumption of a Kolmogorov power spectral density (PSD) for spatial phase distribution $\Phi(k) \propto k^{-11/3}$. The model is widely accepted because its predictions agree well with experimental evidence.

The phase ϕ has infinite variance because the PSD is unbounded at zero. However, the piston-removed phase x is normally distributed with zero mean and finite covariance C. The inverse of this covariance matrix can be approximated by a product of sparse matrices:

$$C^{-1} \approx LL^T, \qquad (6)$$

where L is proportional to a discretization of the Laplacian operator. This procedure was originally applied to a Hudgin geometry [2] but works just as well for a Fried geometry [19], as long as we modify the discretized Laplacian accordingly. In both cases, the correct choice is that L be proportional to $G^T G$. The piston and waffle modes are therefore still unobservable. We can estimate x by minimizing the conditional mean square error (MSE). We must find $\hat{x}$ in order to

$$\text{minimize} \qquad \mathrm{E}(\|x - \hat{x}\|^2 \,|\, y)$$

$$\text{subject to} \qquad V^T \hat{x} = 0,$$

where x is a zero-mean random variable with covariance matrix C. The solution to this problem is

$$\hat{x} = (G^T G + \sigma^2 C^{-1} + VV^T)^{-1} G^T y.$$

This is known as minimum-variance reconstruction. We can compute $\hat{x}$ by using the approximation in Eq. (6) and solving the linear system

$$(G^T G + \sigma^2 LL^T + VV^T)\hat{x} = G^T y. \qquad (7)$$

This is very similar to the least-squares solution. Indeed, the solutions are identical in the case of zero sensor noise.

Note that the original formulation of minimum-variance reconstruction [2] takes into account both the fitting and estimation steps. Since we neglect the fitting step and all statistics are Gaussian, the minimum-variance estimate is the same as a maximum *a posteriori* (MAP) estimate [[5], p. 5282].

4. ITERATIVE METHODS

A. MG and MG-PCG Methods
What follows is a brief review of MG and PCG methods. Equations (5) and (7) are of the form $A\hat{x} = b$, where Au can be computed in $\mathcal{O}(n)$ floating-point multiplications for arbitrary $u \in \mathbb{R}^n$. This follows because

- G and L are sparse so $G^T Gu$ and $LL^T u$ cost $\mathcal{O}(n)$, and

- $VV^T u = V(V^T u)$ costs $\mathcal{O}(n)$ because $V \in \mathbb{R}^{n \times 2}$.

Such systems can be solved efficiently by using MG methods, which use a smoother, cheap linear iterative method that rapidly removes high-frequency content in the error. The residual is projected onto a coarser grid using a restriction operator, and the smoother is applied again. The general idea is that low-frequency content in the residual becomes high-frequency content when projected onto a coarser grid. This process continues, and the various coarse-level corrections are interpolated back

onto the fine grid using a prolongation operator. The corrections are then added to the original estimate to improve it. When done repeatedly, this is known as a MG iterative method. If we alternate between a MG iteration and a conjugate-gradient iteration, this is known as a MG-PCG.

Both MG [19] and MG-PCG [3,4] methods provide $\mathcal{O}(n)$ convergence for the least-squares reconstructions and MVR. The important parameters are these:

1. The type of smoother used, typically a weighted-Jacobi (J) or Gauss–Seidel (GS) iteration.

2. The number of smoothing iterations to run on each level before (ν_1) and after (ν_2) applying the coarse-level correction.

3. The choice of restriction and prolongation operator. In this paper we use full weighting for restriction and bilinear interpolation for prolongation [20, Section 2.C].

4. The cycle pattern, describing how the various levels are visited. In this paper, we use V-cycles (each level is visited twice per iteration), unless otherwise indicated.

For a comprehensive look at MG methods, we refer the reader to [20]. Once we have selected the specific MG method, the general procedure is as follows:

1. Measurement arrives.

2. Run a predetermined fixed number of iterations of the chosen method, which ensures the estimate has converged.

3. Send the estimate to the controller, which passes the appropriate actuator signals to the DM.

4. Go to step 1.

We call this procedure a cold-start configuration because every time a new measurement arrives, the iterative process is restarted with an initial guess of $\hat{x}_0 = 0$, the prior mean of the distribution of x.

B. Warm-Start Configuration
In the warm-start configuration, the most recent phase estimate is used as a guess value for the first iteration whenever a new measurement arrives. This technique is commonly used in numerical linear algebra and has also been used in the context of AO [11].

Atmospheric phase offset is strongly correlated in time, so we can expect the most recent estimate to be a good guess for the current phase. As we will see, iterative MG reconstruction schemes converge much faster when used in a warm-start configuration.

C. Computational Cost
We will evaluate various iterative schemes on the basis of computational cost. In this paper, we chose to count the number of floating-point multiplications. This cost includes smoother iterations, computation of residuals, restriction and prolongation operations required to pass corrections up and down the hierarchy of levels, and conjugate-gradient iterations if applicable. All costs were computed analytically to ensure a fair comparison.

In the case of Fourier-based methods, we associated a cost of $\frac{1}{2}n \log_2(n)$ multiplications to perform a fast Fourier transform (FFT) on a vector of length n. This is consistent

with the cost of a Radix-2 implementation (n is a power of 2). It is worth noting that FFTs can be implemented very efficiently in hardware, so floating-point operations per second may not be an accurate representation of true performance.

There are other possible choices, such as the number of total floating-point operations (multiplications and additions), or memory considerations.

5. SIMULATION
In this section we present our simulation results. Using a Taylor frozen-flow temporal dynamics model, we simulated the open-loop cold-start and warm-start cases, as well as the closed-loop case.

A. Parameters
The photon flux from the guide star in the visible is

$$\mu = 0.9405 \times 10^{10-0.4M}, \tag{8}$$

where M is the stellar magnitude and μ is the photon flux measured in photons per square meters per second. We assumed a 30 m $\times$ 30 m square aperture and a 128×128 array of sensors arranged using a Fried geometry. There are therefore 129^2 actuators.

We assumed Shack–Hartmann quad-cell ($N_D = 4$) sensors with a RMS read error of $\sigma_e = 7$ electrons. The gap correction factor was chosen to be $K_g = 1.2$, which is typical of this type of sensor [17]. We also chose a quantum efficiency of 0.8 and an optical efficiency of 0.5, for a total efficiency of $\eta = 0.4$. For the background photons, we chose a value of 20 magnitudes/arc sec^2, which translates to $n_{\text{bg}} \approx 0.01 n_{\text{ph}}$ in this case.

Phase screens with the proper spatiotemporal correlations were generated using Arroyo [21,22], a C++ library for the simulation of electromagnetic wave propagation through turbulence. We chose seeing conditions consistent with Ellerbroek's Cerro Pachon layered atmospheric model [23] and assumed Kolmogorov statistics for each layer (Taylor frozen-flow hypothesis).

Using Arroyo, we generated a data set consisting of 1000 independent runs for each of 25 logarithmically spaced sample rates between 10 and 10^4 Hz. Each run consists of 10 time-correlated phase screens. Each simulation was generated from this data set, and all results were averaged over the 1000 independent runs.

Least-squares reconstructors and MVR were implemented. For the minimum-variance case, we used $L = \gamma G^T G$, where $\gamma^2 = 0.2$ was calculated via Monte Carlo simulation to minimize the piston-removed MSE.

B. Open-Loop Cold-Start Results
Regardless of the estimation method, there is a trade-off for choosing the optimal sample rate, which depends on the seeing conditions and guide star brightness [[17], p. 74].

The sensors take time-averaged measurements, the iterative scheme takes time to converge, and the control algorithm takes time in responding to the changes in the estimate. These delays result in estimation error, because even if the DM is ideal, it will assume the shape of the measured wavefront, not the current wavefront. As we in-
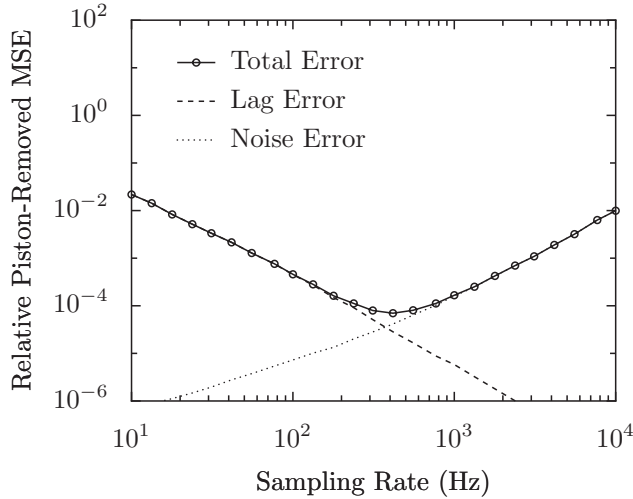
Fig. 2. Error measures for least-squares reconstruction using a guide star of magnitude $M=8$ and the seeing conditions from Subsection 5.A. The exact inverse was computed and averaged over 1000 independent runs. In these conditions, the optimal sensor sampling rate is about 417 Hz, and the minimum expected relative error is just under 10^{-4}.

crease the sample rate of the sensors, this delay error is reduced. However, as seen in Subsection 2.C, increasing the sample rate makes the sensors noisier, resulting in increased estimation error. With this trade-off in mind, we can select the sample rate that minimizes the minimum achievable MSE.

For each of the $M=1000$ runs indexed by i, we took two temporally correlated phase screens $x_1^{(i)}$, $x_2^{(i)}$. The first screen is used to generate a noisy measurement and estimate:

$$y_1^{(i)} = Gx_1^{(i)} + v_1^{(i)},$$

$$\hat{x}_1^{(i)} = A^{-1}G^T y_1^{(i)}.$$

Note that the noise strength $\mathrm{cov}\, v_1^{(i)} = \sigma^2 I$ depends on the sample rate as in Eq. (4). The coefficient matrix to be used is either $A = G^T G + VV^T$ (least squares), or $A = G^T G + \sigma^2 LL^T + VV^T$ (minimum variance). The estimate is com-
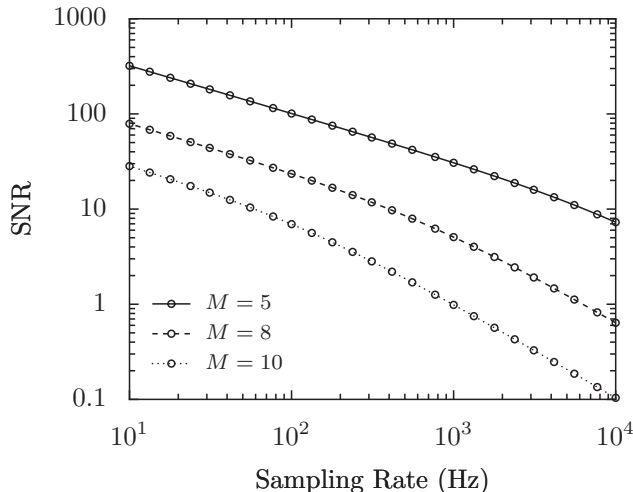

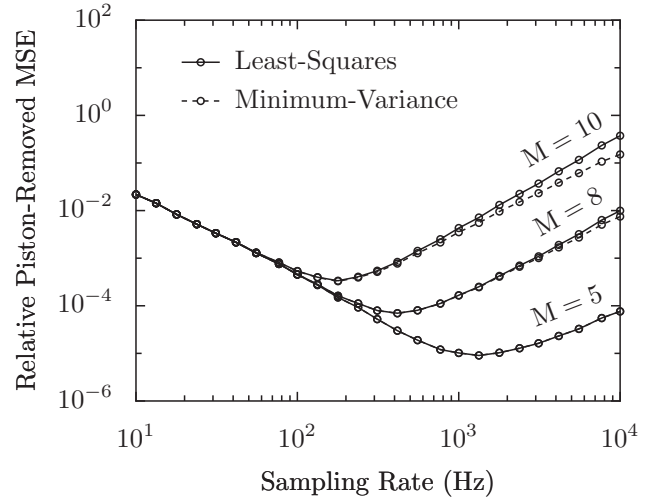Fig. 3. SNR variation for various guide star magnitudes M as a function of the sensor sampling rate.

Fig. 4. Bandwidth-error trade-off curve for various star magnitude values (M) using least-squares and minimum-variance reconstructions. The SNR at the minimum points are 26.2 for $M=5$, 9.7 for $M=8$, and 4.5 for $M=10$. We averaged 1000 independent runs.

pared with the second screen to compute various normalized MSE measures:

1. Lag error: $\dfrac{1}{M}\sum_{i=1}^{M}\left\|x_1^{(i)} - x_2^{(i)}\right\|^2 \Big/ \dfrac{1}{M}\sum_{i=1}^{M}\left\|x_2^{(i)}\right\|^2,$

2. Noise error: $\dfrac{1}{M}\sum_{i=1}^{M}\left\|\hat{x}_1^{(i)} - x_1^{(i)}\right\|^2 \Big/ \dfrac{1}{M}\sum_{i=1}^{M}\left\|x_2^{(i)}\right\|^2,$

3. Total error: $\dfrac{1}{M}\sum_{i=1}^{M}\left\|\hat{x}_1^{(i)} - x_2^{(i)}\right\|^2 \Big/ \dfrac{1}{M}\sum_{i=1}^{M}\left\|x_2^{(i)}\right\|^2.$

Using the seeing conditions chosen in Subsection 5.A, together with a guide star magnitude of 8, we plotted the three error measures (Fig. 2). The total error coincides with the lag error at low sample rates and with the noise error at high sample rates. The optimal sample rate is about 417 Hz for these conditions. As previously noted, we are dealing with open-loop estimation. In a closed-loop

Table 1. List of Iterative Schemes[a]

Number	Type	Smoother	Cycle	Cost per Iteration
1	MG	GS(1,0)	V	3.406×10^5
2	MG	GS(1,1)	V	4.723×10^5
3	MG	GS(2,2)	V	7.358×10^5
4	MG	J(1,0)	V	3.406×10^5
5	MG	J(1,1)	V	4.723×10^5
6	MG	J(2,2)	V	7.358×10^5
7	MG	GS(1,0)	W	5.050×10^5
8	MG	GS(1,1)	W	7.003×10^5
9	MG-PCG	SGS(1,1)	V	6.377×10^5
10	MG-PCG	SGS(2,2)	V	9.012×10^5
11	MG-PCG	J(1,1)	V	6.377×10^5
12	MG-PCG	J(2,2)	V	9.012×10^5
13	MG-PCG	J(1,1)	W	8.657×10^5
14	MG-PCG	SGS(1,1)	W	8.657×10^5
15	FD-PCG	—	—	12.984×10^5

[a]The cost per iteration is measured in floating-point multiplications. For all methods except FD-PCG, this cost is proportional to the number of actuators. SGS, symmetric Gauss–Seidel (see text).
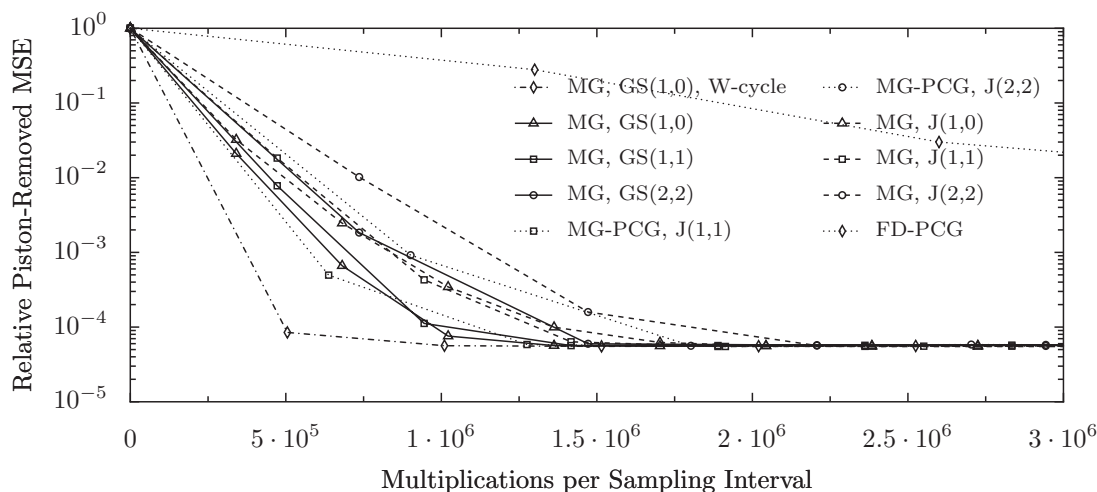
Fig. 5.   Convergence plots comparing simple MG, with conjugate-gradient methods using either a MG-PCG or FD-PCG preconditioner. MG methods use Gauss–Seidel (GS) or Jacobi (J) smoothers. The pair (ν_1, ν_2) is the number of presmoothing and postsmoothing steps. Most methods converge in a few iterations with comparable computational effort. We averaged 1000 independent runs.

configuration, the plot would look similar but would have a different optimal frequency due to the additional errors incurred by the controller dynamics.

One way to quantify the noise level is to use the notion of *signal-to-noise ratio* (SNR). For consistency with existing literature, we will use the quantity from [3,5,7,16,19]:

$$\text{SNR} = \left(\frac{\text{E}\|Gx\|^2}{\text{E}\|v\|^2} \right)^{1/2}.$$

Choosing a brighter star magnitude $M=5$ and a fainter star magnitude $M=10$, we can see how SNR varies with sample rate in Fig. 3. In the literature, SNR values ranging from 1 to 100 are typically assumed, which is consistent with the range obtained in this figure.

Using these three different star brightness values as a way of characterizing different noise levels, we obtained different trade-off curves and corresponding optimal sample rates (Fig. 4).

The MVR outperforms least squares only when we are noise dominated (either a faint guide star or an excessively high sample rate). When using the optimal sample

Table 2. Cost Comparison (Iterating to Convergence)[a]

Iterative Scheme	Iteration Cost (Multiplications)	Iterations to Convergence	Total Cost (Multiplications)
FD-PCG	1.30×10^6	9	1.17×10^7
MG, J(1,0)	3.41×10^5	6	2.04×10^6
MG-PCG, J(2,2)	9.01×10^5	2	1.80×10^6
MG, GS(2,2)	7.36×10^5	2	1.47×10^6
MG, GS(1,1)	4.72×10^5	3	1.42×10^6
MG, GS(1,0)	3.41×10^5	4	1.36×10^6
MG-PCG, J(1,1)	6.38×10^5	2	1.27×10^6
MG, GS(1,0), W-cycle	5.05×10^5	2	1.01×10^6

[a]The iterations to convergence were extracted from Fig. 5. The cheapest method using this metric is GS(1,0), using a W-cycle.

rate for a large SCAO reconstruction, there is no advantage to using minimum-variance reconstruction. This is consistent with the observation by Ellerbroek that conventional least-squares reconstruction is near optimal for future large AO systems [2].

Next we compared several existing iterative methods on the basis of computational cost (as described in Subsection 4.C). See Table 1 for a complete list. We used both Jacobi (J) and Gauss–Seidel (GS) smoothers, with and without PCG, and we varied the number of presmoothing and postsmoothing steps per iteration. When we used PCG, GS was replaced with a symmetric Gauss–Seidel (SGS) smoother to obtain a proper symmetric preconditioner. We added some new methods that to our knowledge have not been published specifically for AO: a W-cycle MG scheme (V-cycles were used for all other methods) and asymmetric MG schemes that perform only one smoothing step per V-cycle, such as GS(1,0). Also, the recently proposed FD-PCG method has been simulated for MCAO systems [5,11]; here we demonstrate its performance on an SCAO system.

See Fig. 5 for the results. The simulation parameters are the same as those used in Fig. 2, running at the optimal sample rate of 417 Hz. Note that with the exception of FD-PCG, the various methods converge to the minimum error from Fig. 2 in a few iterations with comparable computational effort. These plots are similar to the ones produced in [3,19], except that the x axis counts multiplications required rather than iterations.

We can compare the various cold-start and warm-start schemes by evaluating the total cost in multiplications. Table 2 shows that the fastest convergence was obtained when we used GS with a W-cycle and the smallest number of smoothing steps possible.

C. Open-Loop Warm-Start Results
We now generate the analogous plot to Fig. 5 but using warm start. We used the same simulation parameters running at the same optimal rate of 417 Hz. Note that it is not *a priori* obvious that this is the right choice of sample rate. In a warm-start configuration, if multiple it-
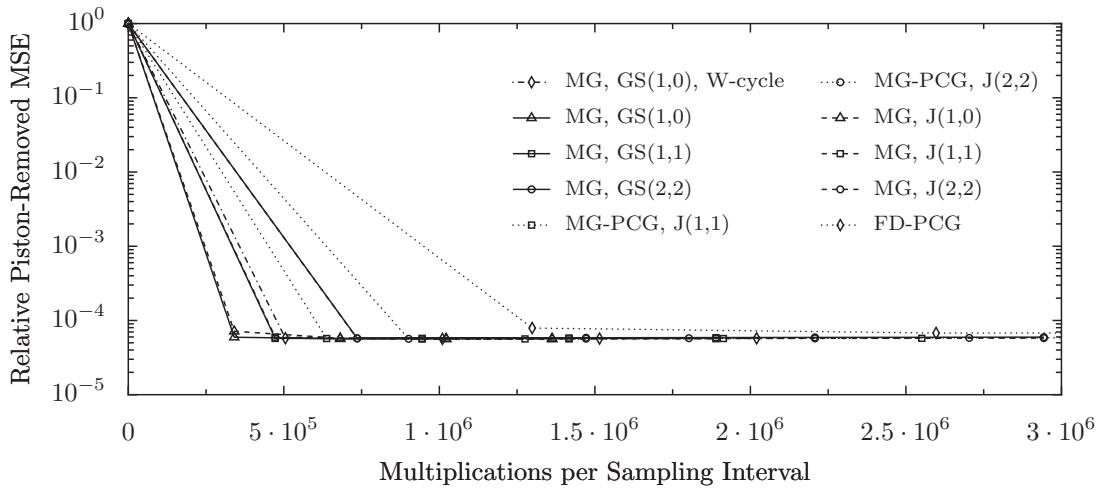
Fig. 6.   Plots comparing converged values of various methods using the open-loop warm-start technique. For every method except MG-J(1,0) and FD-PCG, only one iteration per measurement is required for minimum error. The best method to choose is simply the one that has the smallest iteration cost. We averaged 1000 independent runs.

erations are required to reach convergence, then it may be better to sample more frequently so that newer information is being used in the iteration. In other words, it may be better to *not* iterate to convergence before taking in the next measurement. However, we found that almost every iterative algorithm we tried converged in a single iteration for the parameters used herein. Thus the separation of the choice of algorithm and choice of sample rate remains as in the cold-start case.

We used $\hat{x}_0 = 0$ to start the first iteration when the first measurement arrived. Whenever a new measurement arrived, we used the most recent estimate as an initial guess for the subsequent iteration. We ran each test for 20 sampling intervals to ensure the iterative scheme was operating in steady state (the transient behavior typically disappeared after 3–6 measurements had been processed). The converged results were then used to compute the relative piston-removed MSE.

For each iterative scheme, we varied the number of iterations executed during each sampling interval and plotted the resulting average relative error; see Fig. 6. Only

one iteration per time step was required for every MG method we tested with the exception of MG-J(1,0) and FD-PCG.

It is worth noting that by warm starting, we get no reduction in the expected steady-state error, just faster convergence to that error. Thus, the trade-offs from Figs. 2 and 4 still hold when we warm start.

See Table 3, where we compare the various iterative schemes once again. This table tells us how much computation would be required to implement the algorithms in a warm-start configuration rather than iterating to convergence every time a new measurement arrives.

D. Closed-Loop Results
In a closed-loop setting, the DM corrects the incident phase *before* the sensors take measurements.

In this simulation, we began with a sequence of time-correlated piston-removed phase screens generated using Arroyo: $\{x_1, x_2, \ldots\}$. We closed the loop in two different ways: a standard loop closure (left) and a pseudo-open-loop (POLC) [10] implementation (right):

$$y_t = G(x_t - u_{t-1}) + v_t, \qquad y_t = G(x_t - u_{t-1}) + v_t,$$

$$\hat{e}_t = Ky_t, \qquad\qquad \hat{x}_t = K(y_t + Gu_{t-1}),$$

$$u_{t+1} = u_t + \beta\hat{e}_t, \qquad u_{t+1} = u_t + \beta(\hat{x}_t - u_{t-1}).$$

In the standard case, the iterative scheme is used to find the estimated error, whereas in POLC, the previous input is used to compute an equivalent open-loop measurement, and the iterative scheme is used to find an estimate of the actual phase screen x_t. In both cases, K is *one iteration* of the least-squares iterative scheme of our choice. In practice, a wide range of control gains lead to stable systems with the minimum error. We chose $\beta = 0.5$ for all our simulations.

Both models assume two time steps of delay: a one-step delay to process the measurements since y_t depends on the past input u_{t-1}, and a one-step delay to compute the estimate since the future input u_{t+1} depends on y_t. The relative error at time step t is computed using the formula

Table 3. Cost Comparison (Fewest Possible Iterations)[a]

Iterative Scheme	Iteration Cost (Multiplications)	Iterations per Time Step	Total Cost (Multiplications)
FD-PCG	1.30×10^6	2	2.60×10^6
MG-PCG, J(2,2)	9.01×10^5	1	9.01×10^5
MG, GS(2,2)	7.36×10^5	1	7.36×10^5
MG, J(1,0)	3.41×10^5	2	6.81×10^5
MG-PCG, J(1,1)	6.38×10^5	1	6.38×10^5
MG, GS(1,0), W-cycle	5.05×10^5	1	5.05×10^5
MG, GS(1,1)	4.72×10^5	1	4.72×10^5
MG, GS(1,0)	3.41×10^5	1	3.41×10^5

[a]The iterations to convergence were extracted from Fig. 6. In a warm-start configuration, we do not need to iterate to convergence at every time step in order to achieve the minimum error. Note that the methods are ordered differently here than in Table 2. The cheapest method here is GS(1,0).

$\|x_t - u_{t-1}\|^2/\|x_t\|^2$. This is analogous to the way we computed the error for the cold-start and warm-start open-loop simulations. Note that the average error in closed loop will be higher than the average error in open loop, because our closed-loop implementations have a two-step delay.

We simulated the standard case using both cold start and warm start, and we simulated the POLC case using warm start. In cold-start, POLC requires at least a few iterations per time step to converge to the minimum error, as in Fig. 5. We found that all three loop closure schemes are stable for every method we tested and that every one can be implemented with one time step per iteration.

See Fig. 7 for the results. For the standard closed-loop case using cold start, every iterative scheme with the exception of MG-J(1,0) and FD-PCG converged to a minimum error floor. These are the only two methods we tested that require more than one iteration per time step to achieve the minimum error. Note that the iterative scheme is estimating the *error*, and not the actual phase.

Since we expect the error to be small, an initial guess value of 0 (cold starting) produces fast convergence in only one iteration.

We can also use warm start with the standard case. After iterating, we store the error vector and use it to warm start our iteration at the next time step. With this small change, all the methods we tested converged to the minimum error floor with virtually identical error.

Finally, we tested the POLC case. In this case, we store our previously applied input and use it to convert our closed-loop measurement into an open-loop measurement. The iterative scheme is then used to estimate the actual phase. This method benefits greatly from warm starting, as one might expect from the open-loop results in Fig. 6. Once again, all methods converged to the minimum error. We also tried this method with cold start and found that three or four iterations per time step were typically required, as in Fig. 5.

This is the same conclusion we drew from the open-loop warm-start plot, which means that we can compute the
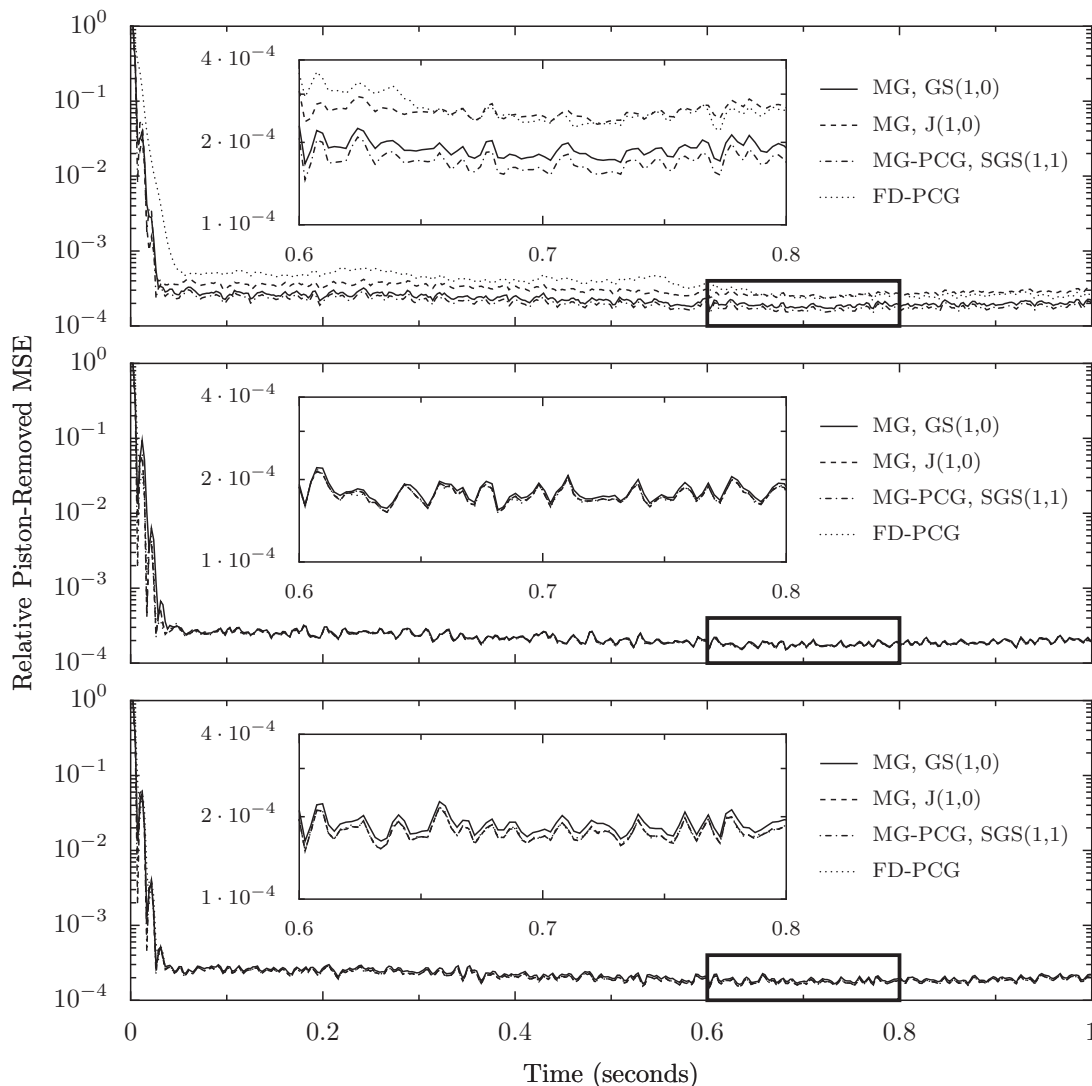

Fig. 7. Closed-loop time series in cold start (top), warm start (middle), and warm-started POLC (bottom); 500 time steps were simulated, performing one iteration per time step. We tested 15 different iterative schemes (only 4 are plotted). All were stable and performed equally well, with the exception of MG-J(1,0) and FD-PCG in the cold-start standard case.

cost required for closed-loop implementation directly from Table 3. To find the required computer speed, we multiply the total cost by the sample rate.

6. CONCLUSION

We have explored the effectiveness of using warm-started iterative methods for AO reconstruction. In open-loop or POLC, warm start provides a significant benefit. When applied to most iterative methods, convergence is achieved in only one iteration, reducing the number of multiplications required by a factor of about 3.

For a standard closed-loop implementation, the advantage of warm starting is less significant because most iterative algorithms already achieve optimal performance with a single cold-started iteration per time step. The control removes much of the (large-amplitude) correlated information between time steps so that the residual error is both smaller and less correlated. Nevertheless, when we warm start in this case, every algorithm we tested provides the same minimal error, and convergence in one iteration per time step. This shows that iteration cost is the most meaningful performance metric.

In principle, any iterative scheme can be warm started and should always yield a computational speedup. For methods such as Fourier-based reconstruction, which do not afford an iterative implementation, it is not clear how to take advantage of the warm-start approach.

REFERENCES

1. E. P. Wallner, "Optimal wave-front correction using slope measurements," J. Opt. Soc. Am. **73**, 1771–1776 (1983).
2. B. L. Ellerbroek, "Efficient computation of minimum-variance wave-front reconstructors with sparse matrix techniques," J. Opt. Soc. Am. A **19**, 1803–1816 (2002).
3. L. Gilles, C. R. Vogel, and B. L. Ellerbroek, "Multigrid preconditioned conjugate-gradient method for large-scale wave-front reconstruction," J. Opt. Soc. Am. A **19**, 1817–1822 (2002).
4. L. Gilles, "Order-N sparse minimum-variance open-loop reconstructor for extreme adaptive optics," Opt. Lett. **28**, 1927–1929 (2003).
5. Q. Yang, C. R. Vogel, and B. L. Ellerbroek, "Fourier domain preconditioned conjugate gradient algorithm for atmospheric tomography," Appl. Opt. **45**, 5281–5293 (2006).
6. L. A. Poyneer, D. T. Gavel, and J. M. Brase, "Fast wave-front reconstruction in large adaptive optics systems with use of the Fourier transform," J. Opt. Soc. Am. A **19**, 2100–2111 (2002).
7. H. Ren, R. Dekany, and M. Britton, "Large-scale wave-front reconstruction for adaptive optics systems by use of a recursive filtering algorithm," Appl. Opt. **44**, 2626–2637 (2005).
8. D. G. MacMartin, "Local, hierarchic, and iterative reconstructors for adaptive optics," J. Opt. Soc. Am. A **20**, 1084–1093 (2003).
9. L. Gilles, "Closed-loop stability and performance analysis of least-squares and minimum-variance control algorithms for multiconjugate adaptive optics," Appl. Opt. **44**, 993–1002 (2005).
10. B. L. Ellerbroek and C. R. Vogel, "Simulations of closed-loop wavefront reconstruction for multiconjugate adaptive optics on giant telescopes," Proc. SPIE **5169**, 206–217 (2003).
11. L. Gilles, B. Ellerbroek, and C. Vogel, "A comparison of multigrid V-cycle versus Fourier domain preconditioning for laser guide star atmospheric tomography," in *Adaptive Optics: Analysis and Methods/Computational Optical Sensing and Imaging/Information Photonics/Signal Recovery and Synthesis Topical Meetings on CD-ROM*, OSA Technical Digest (CD) (Optical Society of America, 2007), paper JTuA1.
12. D. T. Gavel and D. Wiberg, "Toward Strehl-optimizing adaptive optics controllers," Proc. SPIE **4839**, 890–901 (2003).
13. P. Piatrou and M. Roggemann, "Performance analysis of Kalman filter and minimum variance controllers for multiconjugate adaptive optics," Proc. SPIE **5894**, 288–296 (2005).
14. L. A. Poyneer, B. A. Macintosh, and J.-P. Véran, "Fourier transform wavefront control with adaptive prediction of the atmosphere," J. Opt. Soc. Am. A **24**, 2645–2660 (2007).
15. R. N. Paschall and D. J. Anderson, "Linear quadratic Gaussian control of a deformable mirror adaptive optics system with time-delayed measurements," Appl. Opt. **32**, 6347–6358 (1993).
16. B. Le Roux, J.-M. Conan, C. Kulcsár, H.-F. Raynaud, L. M. Mugnier, and T. Fusco, "Optimal control law for classical and multiconjugate adaptive optics," J. Opt. Soc. Am. A **21**, 1261–1276 (2004).
17. J. W. Hardy, *Adaptive Optics for Astronomical Telescopes* (Oxford U. Press, 1998).
18. V. I. Tatarskii, *Wave Propagation in a Turbulent Medium* (McGraw-Hill, 1961).
19. C. R. Vogel and Q. Yang, "Multigrid algorithm for least-squares wavefront reconstruction," Appl. Opt. **45**, 705–715 (2006).
20. U. Trottenberg, A. Schüller, and C. W. Oosterlee, *Multigrid Methods* (Academic, 2000).
21. M. C. Britton, "Arroyo C++ library: object oriented class libraries for the simulation of electromagnetic wave propagation through turbulence," http://eraserhead.caltech.edu/arroyo/arroyo.html (April 8, 2008).
22. M. C. Britton, "Arroyo," Proc. SPIE **5497**, 290–300 (2004).
23. B. L. Ellerbroek and F. J. Rigaut, "Scaling multiconjugate adaptive optics performance estimates to extremely large telescopes," Proc. SPIE **4007**, 1088–1099 (2000).