Evaluation of statistical turbulence models for use in the Distributed Kalman Filter

Jesse Cranney[1,2,*], Piotr Piatrou[2,3], Jose De Dona[1], Francois Rigaut[2,3] and Visa Korkiakoski[2,3]

[1]University of Newcastle, Callaghan, Australia
[2]Space Environment Research Centre, Canberra, Australia
[3]Australian National University, Canberra, Australia
[*]Corresponding author email: jesse.cranney@uon.edu.au

Abstract

The Kalman Filter (KF) application to wavefront control in Adaptive Optics (AO) shows great promise with respect to achieving the best atmospheric turbulence error rejection, provided an adequate model for the turbulence temporal dynamics exists. The advent of the computationally efficient Distributed Kalman Filter (DKF) algorithm makes the KF approach especially attractive for the future of high DOF AO systems (As is the case in the Extremely Large Telescope class). In this work we try to address a significant drawback of the existing DKF state space model, which is too simplistic (single parameter scalar diagonal state matrix) to describe the atmosphere turbulence dynamics. We investigate a broader family of Block-Toeplitz with Toeplitz Blocks (BTTB) state matrices, which, on one hand, are able to better grasp statistical properties of the turbulence to provide significantly better prediction power and, on the other hand, preserve the shift invariance property, a corner stone of the DKF algorithm. We demonstrate the capabilities of the new model in the end-to-end simulations of the DKF-driven Single Conjugate AO system.

1 Introduction

The future AO systems for Extremely Large Telescopes (ELTs) requires new, low complexity, control algorithms that would beat the typical $O(N^2)$ scaling law (N is the number of AO system degrees of freedom, DOF) of classical Matrix-Vector Multiplication (MVM) based algorithms, which appear to be inadequate for the ELT's multi-thousand DOF AO systems. A recently proposed [1] Distributed Kalman Filter with $O(N \log N)$ complexity looks especially attractive for use in the ELT AO systems because it gives an elegant solution to the long-standing problem of low-complexity application of the KF to AO systems allowing to achieve a near-optimal performance in comparison to the Minimum Variance controllers [2] and also provide a method of system state prediction for efficient temporal lag compensation [3].

In this paper we revisit the DKF theory initially presented in the seminal paper [1] to address some of its drawbacks. In particular, we re-derive the DKF equations strictly within pure matrix algebra showing that the DKF essentially results from the circulant approximation of the system matrices involved in the classical KF equations while such an approximation is the consequence of the shift-invariance inherent in most AO systems. Another improvement to the seminal DKF is the replacement of the overly simplistic single-parameter diagonal state transition matrix used in [1] with a matrix of the block-circulant class, which does not violate the DKF assumptions but contains significantly more free parameters. As an illustration of our approach we design a DKF for a case of high-complexity 50-meter class telescope Single-Conjugate Adaptive Optics (SCAO) system and validate it via end-to-end simulations.

2 Linear System Review

2.1 Discrete Time State Space Dynamic System

The standard state space description of a Linear Time-Invariant (LTI) dynamic system:

$$\begin{array}{rcl} x_{t+1} & = & Ax_t + Bu_t \\ y_t & = & Cx_t + Du_t \end{array} \tag{1}$$

is used to represent an AO system. In (1), the parameters and variables are defined as:

- x_t, the state vector of turbulence phase values at time t;
- y_t, the measurement vector of the sensor readouts at time t;
- u_t, the concatenated noise vectors at time t, since the filtering problem has no control input. $u_t = \begin{bmatrix} p_t \\ n_t \end{bmatrix}$, where p_t is the process noise due to turbulence at time t, and n_t is the measurement noise due to the wave-front sensor (WFS) process at time t.
- A is the state matrix, which approximates the state vector at time $t+1$ given the state vector at time t.
- B is set to $[I\ 0]$ in order to map the process noise, p_t, onto the state vector at time $t+1$. p_t is temporally white noise with a covariance matrix found from the Lyapunov Equation [4]:

  $$\langle pp^T \rangle = \langle xx^T \rangle - A\langle xx^T \rangle A^T \tag{2}$$

  where $\langle xx^T \rangle$ is the turbulence covariance matrix.
- C is the turbulence-to-WFS poke matrix. This maps the turbulence phase to the WFS measurements.
- D is set to $[0\ I]$ in order to map the measurement noise, n_t, onto the measurements at time t.

These definitions allow the use of a linear model with known noise covariance to model the dynamic evolution of atmospheric turbulence as it would be seen by the pupil.

2.2 Kalman Filter

The recursive equations [3] for the Kalman Filter are the most convenient in this demonstration. For time $t = 0,\ 1,\ \ldots$

$$\begin{array}{rcl} \hat{x}_{t+1|t} & = & A\hat{x}_{t|t} \\ \hat{x}_{0|0} & = & 0 \\ \hat{y}_{t+1|t} & = & C\hat{x}_{t+1|t} \\ \hat{x}_{t+1|t+1} & = & \hat{x}_{t+1|t} + K_t\big(y_{t+1} - \hat{y}_{t+1|t}\big) \\ \Sigma_{0|0} & = & \langle pp^T \rangle \\ \Sigma_{t+1|t} & = & A\Sigma_{t|t}A^T + \langle pp^T \rangle \\ K_{t+1} & = & \Sigma_{t+1|t}C^T\Big(C\Sigma_{t+1|t}C^T + \langle nn^T \rangle\Big)^{-1} \\ \Sigma_{t+1|t+1} & = & \Sigma_{t+1|t} - K_{t+1}C\Sigma_{t+1|t} \end{array} \tag{3}$$

Here, $\hat{x}$ denotes the estimate of the true value of the variable x; K_t is the Kalman gain at time t; $\Sigma = \big\langle (x - \hat{x})(x - \hat{x})^T \big\rangle$ is the estimation error covariance matrix. The steady-state value of K_t, i.e., K_∞ is used in the filter implementation.

It is required that for $\lim_{t\to\infty} K_t$ to converge, the $(A,\ C)$ pair should be observable and the $(A,\ \langle pp^T \rangle)$ pair should be controllable.

Problems with K_t convergence may arise in the case of piston and/or tip/tilt removal.

2.3 Similarity Transformation

Take an invertible matrix, T, and perform a change of variables by the substitution of: $x \to T^{-1}x'$

After this substitution, (1) becomes:

$$\begin{aligned} x'_{t+1} &= TAT^{-1}x'_t + TBu_t \\ y_t &= CT^{-1}x'_t + Du_t \end{aligned} \tag{4}$$

The state space system in (4) is equivalent to (1) but in different co-ordinates and with different system matrices. The motivation behind such a transformation is to diagonalise the system matrices [1]

2.4 Shift Invariance

Any linear transformation in finite-dimensional space can be represented as a matrix-vector multiplication (MVM). Index-wise, $y = Ax$ is identical to:

$$y_i = \sum_{j=1}^{N} A_{ij} x_j$$

Extending this to infinite-dimensional space yields the integral:

$$y(r) = \int A(r, r')x(r')dr'$$

where A_{ij} becomes the transformation kernel $A(r, r')$. If the kernel is shift-invariant, it depends only on $(r - r')$. This is then equivalent to the convolution:

$$y(r) = \int A(r - r')x(r')dr'$$

Returning to finite-dimensional space, the corresponding convolution matrix A_{ij} has a special structure. In one dimension, this structure is known as the Toeplitz Matrix.

2.5 Convolution as a Matrix

Simply by observation, one can see that the one-dimensional discrete convolution:

$$y_i = \sum_{j=1}^{N} h_{i-j} x_j$$

is representable as a MVM with a Toeplitz Matrix, defined element wise by:

$$A_{ij} = A_{i+1\,j+1} = h_{i-j}$$

Two-dimensional discrete convolution:

$$y_{ij} = \sum_{k=1}^{N} \sum_{\ell=1}^{N} h_{i-k,j-\ell} x_{k\ell}$$

is similarly representable by a Block Toeplitz matrix with Toeplitz Blocks (BTTB):

$$A_{ij,k\ell} = h_{i-k,\,j-\ell}$$

where the MVM of $Ax = y$ is computed via multi-index:

$$y_{ij} = \sum_{k=1}^{N} \sum_{\ell=1}^{N} A_{ij,k\ell} x_{k\ell}$$

The set of Toeplitz (or BTTB) matrices of finite size possesses an unpleasant property: it is closed with respect to addition but not multiplication (Toeplitz times Toeplitz is not generally Toeplitz).

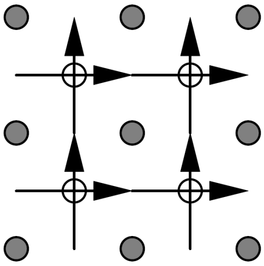


Figure 1: A 3 by 3 phase array in Fried Geometry, with 2 by 2 lots of slopes within them.

This is because Toeplitz (or BTTB) matrices of finite size represent convolution within finite space and therefore, on the border, the convolution is truncated.

This boundary effect can be overcome by extending the matrix to infinity, or applying a Periodic Boundary Condition (PBC). The PBC leads to the circulant matrix, which is a square matrix defined element wise as:

$$C_{ij} = a_{i-j+1},\ i,j = 1, \ldots, n$$

and all $i-j+1$ are computed modulo n. The Block Circulant matrix with Circulant Blocks (BCCB) is defined analogously to the BTTB matrix.

2.6 Sensor Measurement Matrix

For the case of Fried Geometry, the measurement matrix mapping the 2-D phase grid into 2-D phase slope grid is:

$$C_x = \frac{1}{2d}\begin{bmatrix} -1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 \end{bmatrix}$$

$$C_y = \frac{1}{2d}\begin{bmatrix} 1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 \end{bmatrix} \tag{8}$$

where d is the phase grid spacing (lenslet size). Note that the measurement matrix is BTTB, but non-square because the four phase points make only one slope (See Figure 1).

To make the BCCB approximation of C_x and C_y there are two steps:

- Augment the slope point grid to have the same number of points as the phase grid (and, therefore, result in square matrices).
- Apply PBC to create *virtual* phase points necessary to compute slopes in the augmented slope grid positions as shown in Figure 2.

The resulting BCCB measurement matrices take the form:

$$C_x = \frac{1}{2d}\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & -1 \\ -1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -1 & 1 & 0 & 1 \end{bmatrix} \tag{9x}$$

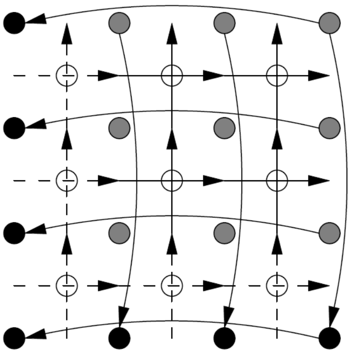


Figure 2: Illustration of the PBC application to the 3×3 phase grid and the Fried WGS geometry. White dots correspond to the slope grid, grey dots are the phase grid, and black dots are the virtual points of the augmented phase grid. Arrows show how the phase values are copied according to PBC. Dashed arrows are the virtual slopes that are created by the PBC application.

$$C_y = \frac{1}{2d} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ -1 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & 1 \\ 1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 1 & -1 & 0 & 0 & 0 \\ -1 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 & -1 \\ 0 & 0 & 0 & -1 & 0 & 1 & -1 & 0 & 1 \end{bmatrix} \tag{9y}$$

which are perfectly BCCB.

2.7 Discrete Fourier Transform as a Matrix

A circulant matrix has the convenient property: it is diagonalised by the Discrete Fourier Transform (DFT). The 1-D DFT $\tilde{f}$ of a vector f is:

$$\tilde{f}_n = \sum_{m=0}^{M-1} e^{-2\pi i m n/M} f_m, \; n = 0, 1, \ldots, M-1$$

The corresponding DFT matrix is (with $\omega = e^{-2\pi i/M}$):

$$F_M = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & \omega^1 & \ldots & \omega^{1(M-1)} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{M-1} & \ldots & \omega^{(M-1)(M-1)} \end{bmatrix}$$

such that $\tilde{f} = F_M f$. The 2-D DFT matrix of an $N \times M$ image is $F_M \otimes F_N$ where $\otimes$ is the Kronecker Matrix product.

Theorem ([5]). *Let C be a Circulant Matrix defined by its first column c. C is diagonalised by the DFT matrix:*

$$C = F_M^{-1} \Delta F_M, \; \Delta = \mathrm{diag}(F_M c) \tag{5}$$

Corollary. *Using (5), it is clear that Circulant matrices are closed under multiplication (circulant times circulant is circulant). Moreover, they commute.*

Returning to the state space model in (1), and applying the DFT similarity transforms, the system becomes:

$$\begin{aligned} x_{t+1} &= F^{-1} \Delta_A F x_t + p_t \\ y_t &= F^{-1} \Delta_C F x_t + n_t \end{aligned}$$

Left-multiplying by F and re-assigning $\tilde{x}_t = Fx_t$, $\tilde{y}_t = Fy_t$, $\tilde{p}_t = Fp_t$, $\tilde{n}_t = Fn_t$ we arrive at:

$$\begin{aligned} \tilde{x}_{t+1} &= \Delta_A \tilde{x}_t + \tilde{p}_t \\ \tilde{y}_t &= \Delta_C \tilde{x}_t + \tilde{n}_t \end{aligned} \tag{6}$$

where Δ_A and Δ_C are diagonal. Analogously, assuming that A, C, $\langle pp^T \rangle$, $\langle nn^T \rangle$ are Circulant (or BCCB) it is clear that the Kalman Filter Equations (3) are diagonalised too.

2.8 Circulant Approximation of Phase Covariance

Thus, from Section 2.7, it is very desirable to have all state space matrices to be or to be approximated by BCCB matrices. Since the AO system, at least atmospheric turbulence and the Shack Hartmann WFS are shift-invariant, the AO matrices are BTTB.

To approximate the BTTB matrix with a BCCB one, a PBC is applied (see Figure 3). In order to minimise the effect of this periodicity, the matrices are *padded* with virtual states and measurements which are purely for structure.

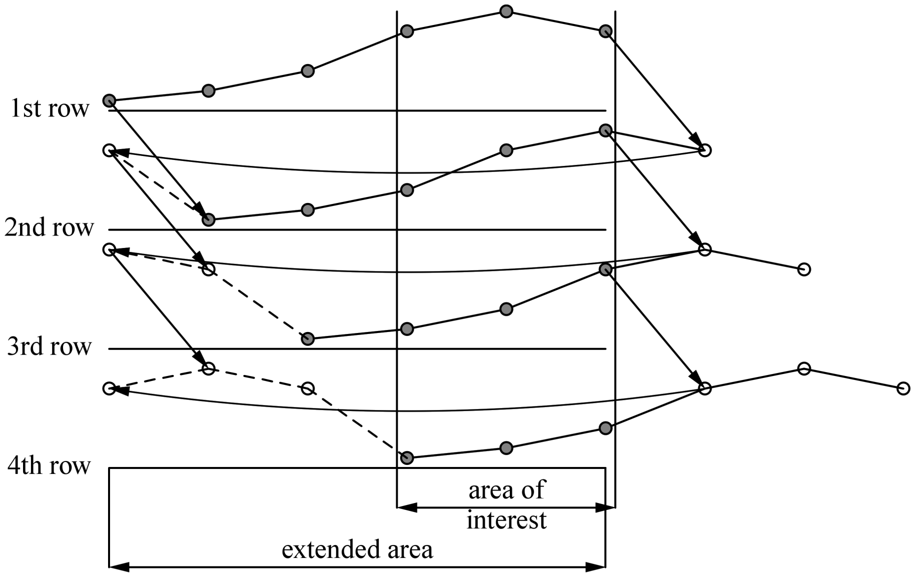


Figure 3: Illustration of the Periodic Boundary Condition application in the case of 1D Toeplitz convolution matrix. The PBC returns the lost values to the beginning of the next row. N.B. No values are altered by PBC within the area of interest if we embed it into a twice as large domain.

The shift-invariant turbulence phase covariance has the form:

$$\Sigma_{ij} = \langle \phi \phi^T \rangle_{ij} = \Sigma\big(|x_i - x_j|\big)$$

where $\{x\}_{i=1}^{N^2}$ are 2-D phase point position vectors from an $N \times N$ map, column-wise concatenated into an $N^2 \times 1$ column. $\Sigma(r)$ is the Von Karman turbulence phase covariance function [6]:

$$\Sigma(r) = \frac{0.033 \cdot 0.185^{\frac{5}{3}} \cdot 4\pi^2}{2^{\frac{5}{6}} \Gamma(\frac{11}{6})} \left(r_0 \frac{2\pi}{L_0}\right)^{-\frac{5}{3}} \left(r \frac{2\pi}{L_0}\right)^{\frac{5}{6}} K_{\frac{5}{6}} \left(r_0 \frac{2\pi}{L_0}\right) \tag{7}$$

where r_0 is the Fried Parameter; L_0 is the outer scale; $K_{\frac{5}{6}}$ is the Bessel function of second kind, with order $\frac{5}{6}$; $\Gamma(\cdot)$ is the Gamma-function; and r is the distance between a pair of phase points to be correlated.

Note that the Σ matrix is BTTB due to shift-invariance of the $\Sigma(r)$ function. With the goal of circulant approximation, the BTTB matrix is embedded into a region twice as large, and the PBC is applied. This is illustrated below for the 1D case (see also Figure 3). For a Toeplitz matrix:

$$\begin{bmatrix} a_0 & a_1 & a_2 \\ a_{-1} & a_0 & a_1 \\ a_{-2} & a_{-1} & a_0 \end{bmatrix}$$

with compact representation $\begin{bmatrix} a_{-2} & a_{-1} & a_0 & a_1 & a_2 \end{bmatrix}^T$, the circulant embedding has compact representation $\begin{bmatrix} a_0 & a_{-1} & a_{-2} & a_2 & a_1 \end{bmatrix}^T$. The corresponding full circulant matrix after such an embedding looks like:

$$\begin{bmatrix} a_0 & a_1 & a_2 & a_{-2} & a_{-1} \\ a_{-1} & a_0 & a_1 & a_2 & a_{-2} \\ a_{-2} & a_{-1} & a_0 & a_1 & a_2 \\ a_2 & a_{-2} & a_{-1} & a_0 & a_1 \\ a_1 & a_2 & a_{-2} & a_{-1} & a_0 \end{bmatrix}$$

The correct covariance matrix is within any framed part of the circulant covariance matrix approximation meaning that within any $N \times N$ sub-region inside the extended $2N \times 2N$ domain the phase will have the correct covariance matrix, but not for the entire extended domain.

2.9 Turbulence State Matrix

In the paper of Massioni et al [1], the state matrix is chosen to be the identity with a decay factor, $|\alpha| < 1$, such that $A = \alpha \mathrm{I}$. Being diagonal, this matrix is BCCB. This matrix ignores the dynamic evolution of the turbulence, and as such, motivates the search for a BCCB matrix capable of including this information. The most natural state matrix to consider would be the BCCB approximation of the "near-Markov" matrix [7]:

$$A = \langle \phi_{t+1} \phi_t^T \rangle \langle \phi_t \phi_t^T \rangle^{-1} \tag{10}$$

which is the minimum variance linear statistical estimate of the phase at time $t+1$ using the phase data at time t and the phase cross, $\langle \phi_{t+1} \phi_t^T \rangle$, and auto, $\langle \phi_t \phi_t^T \rangle$ temporal covariances.

Under the following assumptions, the A matrix in (10) can be computed analytically:

- The turbulence parameters (r_0 and L_0) are known.
- The temporal evolution of the turbulence is describable by the Taylor Hypothesis [8] (Frozen Flow) and the layer velocities associated with this assumption are known.

Since the different turbulence layers are statistically independent of each other, the A matrix takes the form:

$$A = \Big(\sum_{\ell=1}^{L} \langle \phi_{t+1} \phi_t^T \rangle_\ell \Big) \Big(\sum_{\ell=1}^{L} \langle \phi_t \phi_t^T \rangle_\ell \Big)^{-1}$$

and, with the Taylor Hypothesis:

$$A_{ij} = \Big(\sum_{\ell=1}^{L} \Sigma_\ell (x_i - x_j - v_\ell dt) \Big) \Big(\sum_{\ell=1}^{L} \Sigma_\ell (x_i - x_j) \Big)^{-1}$$

where v_ℓ are the wind velocity vectors for each layer, dt is the frame exposure time, $\Sigma_\ell(\cdot)$ are the layer covariance functions. The cross-covariance of a single layer is the shift of the layer's auto-covariance. Moreover, from the Von Karman equation (7), it is seen that a difference in r_0 manifests only in a multiplicative constant change. If it is assumed that L_0 are equal for all layers, the latter equation in the Fourier domain reduces to:

$$\tilde{A}(f) = \frac{R \sum_{\ell=1}^{L} w_\ell \mathrm{XY}_\ell(f) \tilde{\Sigma}(L_0, f)}{R \sum_{\ell=1}^{L} w_\ell \tilde{\Sigma}(L_0, f)} = \sum_{\ell=1}^{L} w_\ell \mathrm{XY}_\ell(f) \tag{12}$$

where a *tilde* refers to the Fourier Transformed functions; $\Sigma(L_0, f)$ is the Von Karman phase covariance for fixed L_0 and $r_0 = 1$; $R = \left(\frac{2\pi r_0}{L_0} \right)^{-\frac{3}{5}}$; $\mathrm{XY}_\ell(f) = e^{-2\pi\, i\, dt\, v_\ell\, f}$ is the wind shift term in the Fourier Domain; and $\{w_\ell\}_{\ell=1}^{L}$, $\sum w_\ell = 1$, are the relative turbulence layer weights. Finally, it can be noted that the state matrix based only on wind shifts does not depend on the turbulence statistics.

3 Distributed Kalman Filter Equations

The process of finding the DKF equations from the standard Kalman Filter equations in (3) is demonstrated here. The diagonalisation is complicated due to the C matrix and $\langle nn^T \rangle$ matrix not being BCCB, but instead consisting of BCCB blocks (we deliberately avoid extending the acronym). This, however, still can be cast into a diagonal form by the DFT. Denote:

$$\Sigma_{k|p} = F^{-1} \Delta_{k|p} F \tag{13a}$$

$$C = \begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} F^{-1} & 0 \\ 0 & F^{-1} \end{bmatrix} \begin{bmatrix} \Delta_x \\ \Delta_y \end{bmatrix} F \tag{13b}$$

$$\langle nn^T \rangle = \begin{bmatrix} n_{xx} & n_{xy} \\ n_{yx} & n_{yy} \end{bmatrix} = \begin{bmatrix} F^{-1} & 0 \\ F^{-1} & 0 \end{bmatrix} \begin{bmatrix} \Delta_{xx} & \Delta_{xy} \\ \Delta_{yx} & \Delta_{yy} \end{bmatrix} \begin{bmatrix} F & 0 \\ F & 0 \end{bmatrix} \tag{13c}$$

where F is the Fourier Transform matrix, Δ's are the diagonal matrices of the circulant matrix eigenvalues coinciding with the Fourier Transform of the corresponding compact[1] BCCB representations.

Letting:

$$G = \begin{bmatrix} \Delta_x \Delta_{t+1|t} \Delta_x^* + \Delta_{xx} & \Delta_x \Delta_{t+1|t} \Delta_y^* + \Delta_{xy} \\ \Delta_y \Delta_{t+1|t} \Delta_x^* + \Delta_{yx} & \Delta_y \Delta_{t+1|t} \Delta_y^* + \Delta_{yy} \end{bmatrix}^{-1}$$

and substituting (13) into (3), for Kalman Gain and residual phase covariance, we get:

$$K_{t+1} = F^{-1} \Delta_{t+1|t} \left[\Delta_x^* \ \Delta_y^* \right] G \begin{bmatrix} F & 0 \\ 0 & F \end{bmatrix}$$

$$\Sigma_{t+1|t+1} = F^{-1} \left(\Delta_{t+1|t} - \Delta_{t+1|t} \left[\Delta_x^* \ \Delta_y^* \right] G \begin{bmatrix} \Delta_x \\ \Delta_y \end{bmatrix} \Delta_{t+1|t} \right) F$$

where $(\cdot)^*$ denotes the complex conjugate. Note that the matrix to invert contains 2×2 diagonal blocks. The inverse will have the same structure and is easy to find by rearranging the matrix into block-diagonal form with 2×2 blocks. There is, however, a simpler way of writing the same equations taking advantage of the Matrix Inversion Lemma [4][2]:

$$\Sigma_{t+1|t} C^T \left(C \Sigma_{t+1|t} C^T + \langle nn^T \rangle \right)^{-1} = \left(C^T \langle nn^T \rangle^{-1} C + \Sigma_{t+1|t}^{-1} \right)^{-1} C^T \langle nn^T \rangle^{-1}$$

The $\langle nn^T \rangle$ and $\Sigma_{t+1|t}$ inverse are easy to find; the former is a matrix with 2×2 diagonal blocks with equal elements on the diagonals, so it is enough to invert one 2×2 matrix to get $\langle nn^T \rangle^{-1}$, the $\Sigma_{t+1|t}$ inverse is just a reciprocal of $\Delta_{t+1|t}$. Using the Inversion Lemma we get the alternative Kalman gain and residual covariance equations:

$$K_{t+1} = F^{-1} \left(\Delta_n + \Delta_{t+1|t}^{-1} \right)^{-1} \left[\Delta_{nx} \ \Delta_{ny} \right] \begin{bmatrix} F & 0 \\ 0 & F \end{bmatrix}$$

$$\Sigma_{t+1|t+1} = F^{-1} \left(\Delta_{t+1|t} - \left(\Delta_n + \Delta_{t+1|t}^{-1} \right)^{-1} \Delta_n \Delta_{t+1|t} \right) F$$

where:

$$C_n = C_x^T i_{xx} C_x + C_x^T i_{xy} C_y + C_y^T i_{yx} C_x + C_y^T i_{yy} C_y = F^{-1} \Delta_n F$$

$$\left[N_x \ N_y \right] = \left[C_x^T i_{xx} + C_y^T i_{yx} \ \ C_x^T i_{xy} + C_y^T i_{yy} \right] = F^{-1} \left[\Delta_{nx} \ \Delta_{ny} \right] \begin{bmatrix} F & 0 \\ 0 & F \end{bmatrix}$$

$$\langle nn^T \rangle^{-1} = \begin{bmatrix} i_{xx} & i_{yx} \\ i_{xy} & i_{yy} \end{bmatrix}$$

An important note should be made about the $\left(C \Sigma_{t+1|t} C^T + \langle nn^T \rangle \right)$ or $\left(C^T \Sigma_{t+1|t}^{-1} C + \Sigma_{t+1|t}^{-1} \right)$ matrix inversion. Since C as well as $\Sigma_{t+1|t}$ have piston in their null space, the aforementioned matrices are not invertible. To fix this, one needs to inject a piston mode [2]:

$$C^T \Sigma_{t+1|t}^{-1} C + \Sigma_{t+1|t}^{-1} \rightarrow C^T \Sigma_{t+1|t}^{-1} C + \Sigma_{t+1|t}^{-1} + ZZ^T$$

[1] The compact form of a BCCB matrix is defined to be the kernel of the corresponding convolution (since an MVM with a BCCB matrix is equivalent to the circular convolution with a given kernel. An interested reader should look to [5] for the full derivations.

where Z is the piston vector consisting of ones. Note that ZZ^T is BCCB. This means the inverse exists and this operation will add an arbitrary amount of piston into the phase estimate, which needs to be removed afterwards. In the case of working in the Fourier space, the piston injection/removal is simply adding to the first element of the C and $\Sigma_{t+1|t}$ matrices to make the DC components non-zero, then set the DC component of K_t to be zero to remove the piston from the estimate.

4 Results

The prediction power of the “wind-shift” BCCB A matrices is shown in Figure 4. As can be seen, the prediction power is moderate (though significantly better than that of the non-predictive diagonal A-matrix). It should be pointed out, however, that we are considering the sum of the turbulence layers as the only observable entity in the case of the SCAO. The prediction power of the Near-Markov approximation in the case of a single turbulence layer is much better [3].

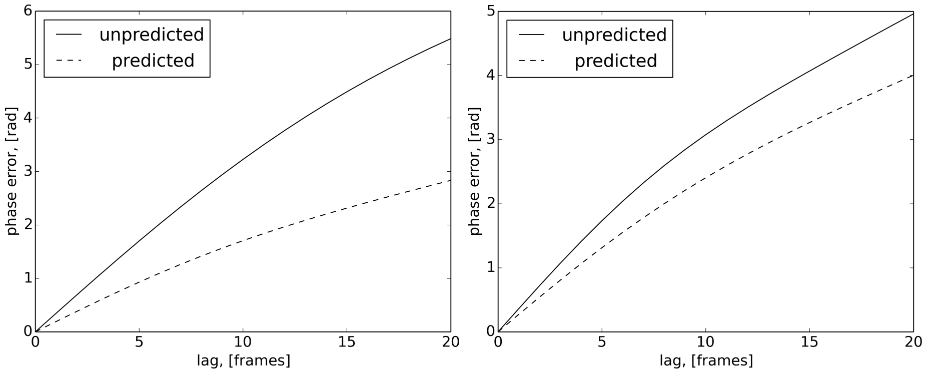
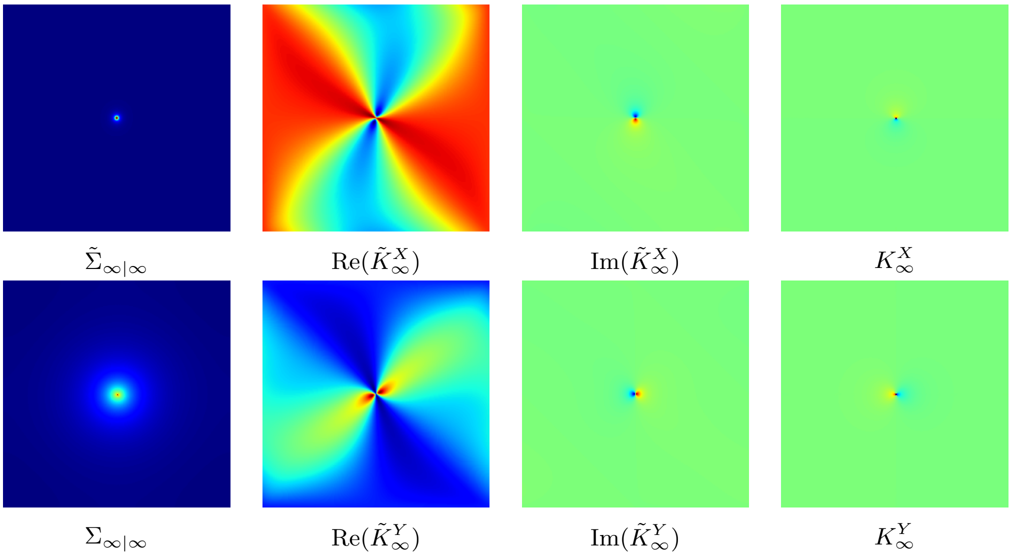


Figure 4: Phase error vs. time lag for Mauna Kea [9] (left) and Cerro Tololo [10] (right) turbulence profiles. Phase prediction is done for the sum of turbulence layers via the circulant approximation to the Near-Markov A-matrix.

Presented on Figure 5 are the images of the compact BCCB forms of the K and Σ matrices for the Mauna Kea turbulence profile.



$\tilde{\Sigma}_{\infty|\infty}$ Re$(\tilde{K}^X_\infty)$ Im$(\tilde{K}^X_\infty)$ K^X_∞

$\Sigma_{\infty|\infty}$ Re$(\tilde{K}^Y_\infty)$ Im$(\tilde{K}^Y_\infty)$ K^Y_∞

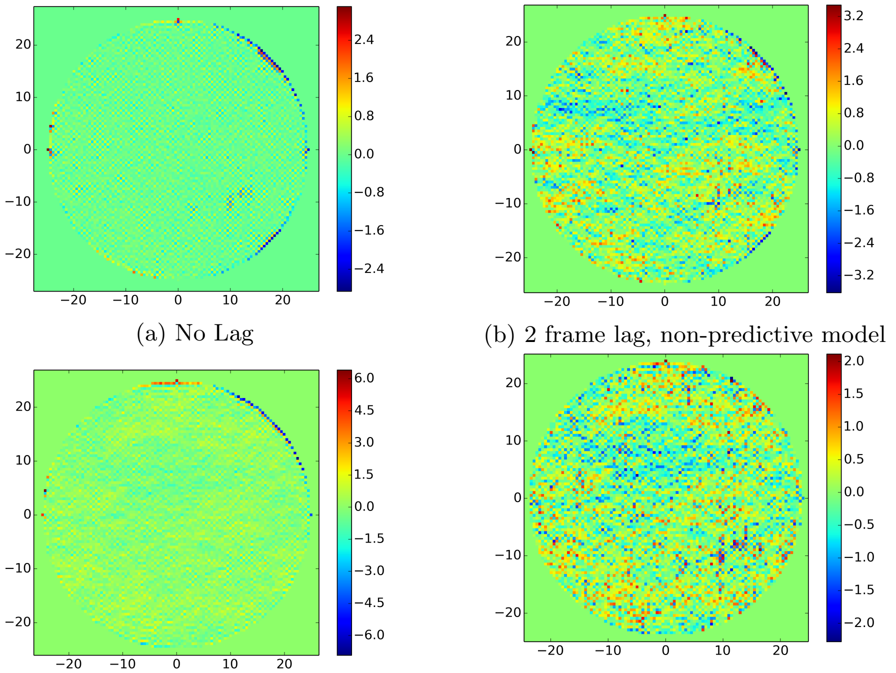
Figure 5: Density maps of the Kalman gain and residual phase covariance matrices in BCCB representation on the augmented 256 × 256 phase grid.

End-to-end simulations were made to evaluate the DKF performance. The following assumptions are in consideration:

- A 100 × 100 lenslet SCAO system for a generic 50-meter telescope was modeled.

- 6-layer Mauna Kea [9] and 7-layer Cerro Tololo [10] turbulence profiles were considered.
- Turbulence phase screens were generated via Fractal Method [11].
- Layer wind shift was done via Fourier domain shift operators.
- Simple Fried geometry model was used for slope generation from phase points.
- No measurement noise, but the $\langle nn^T \rangle$ for DKF was computed for noise with $SNR = 10$.
- Only 0.5-meter Fried geometry phase grid was used. Thus no fitting error, no aliasing, just the pure estimation residual.

The end-to-end simulation results are presented on Figures 6 and 7. The most pronounced consequence of the circulant approximation, as seen from Figure 6, is the large phase residual at the pupil edge resulting from limiting the phase and slope values to the pupil area. This error will always be seen in a pure DKF algorithm due to the shift-invariance assertion, though modifications of this algorithm could see it reduced.



(a) No Lag

(b) 2 frame lag, non-predictive model

(c) 2 frame lag, predictive model

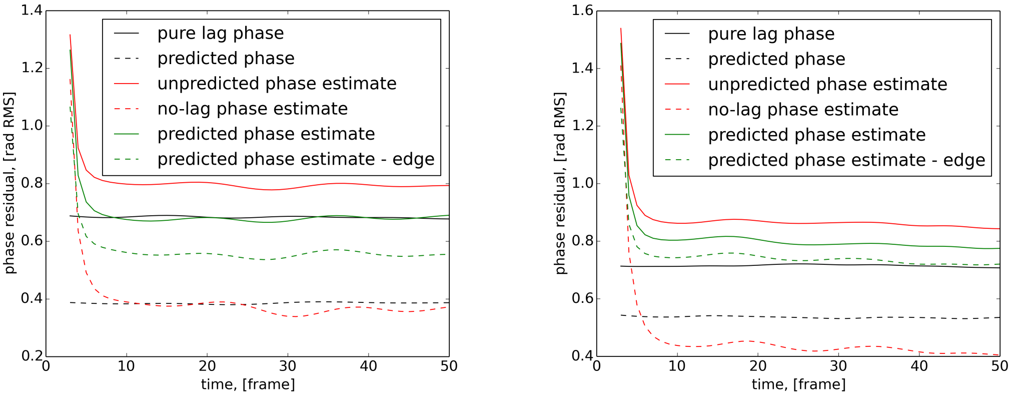
(d) 2 frame lag, 2-pixel edge error suppressed (for demonstration only)

Figure 6: Residual phase maps (note the edge effect)

4.1 Data Extraction for Model

In order to construct the state transition matrix A as described by Equation (12), it is necessary to know the wind vectors (and weights) for the layers. This data can be extracted from WFS slope sequences via a simplified version of the SLODAR algorithm [12]. The simplification comes from the fact that in SCAO, turbulence layer altitudes are not necessary and therefore there is no need for double referenced stars.

The estimation is based on recording the sample cross-covariance matrix between pairs of slope vectors taken with time lag. Owing to the BTTB structure of the cross-covariance, its compact representation is actually recorded by collapsing all equivalent lenslet pairs into one pixel. The resulting 2D image has a nice property; the cross-covariance maximum for each layer occurs at $[V_x,\, V_y]\frac{dt}{dx}$ position, where $[V_x,\, V_y]$ is the wind speed vector of the layer, dx, dt are the lenslet size and time lag, respectively. The turbulence estimation algorithm can be split into two parts:



(a) Mauna Kea Turbulence

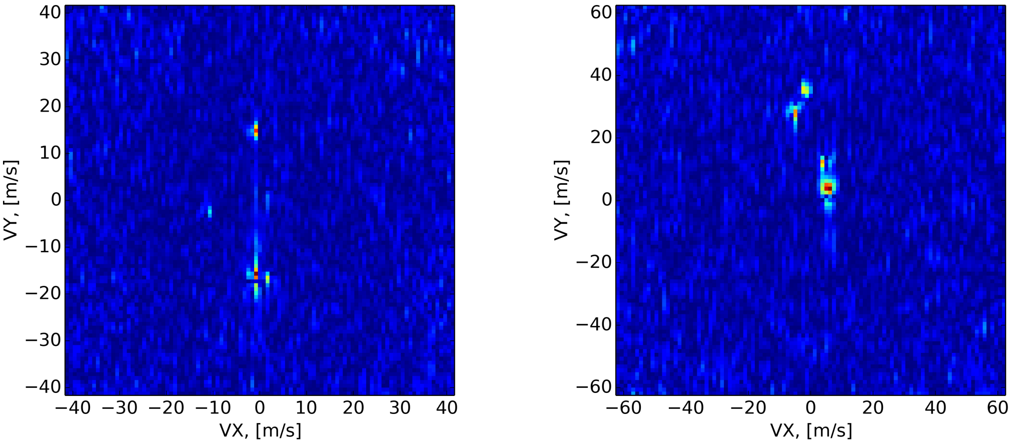
(b) Cerro Tololo Turbulence

Figure 7: Transient response of residual phase error for DKF

1. The wind velocity vectors are found from the cross-covariance BTTB imagine via a peak detection algorithm. In order to reliably distinguish the peaks, it is advisable to set ample time lag between the slope samples[2].

2. Given the peak number and positions, the relative weights are estimated as the relative peak heights.

It is found that the best results are obtained on the closed-loop slopes, that is, the residuals after DM correction as their covariance peaks are the narrowest. The Kalman Filter phase reconstructor performance with theoretical and sample-based A matrix are nearly indistinguishable. The typical simulated slope covariance maps are shown in Figure 8



(a) Mauna Kea Turbulence

(b) Cerro Tololo Turbulence

Figure 8: Estimated wind speed density maps to be used for layer identification. 300-frame lag at 500Hz, 400 samples.

5 Conclusion

In this work we revisited the theory of the Distributed Kalman Filter and showed that it can be derived as a circulant approximation of the classical Kalman Filter for AO systems possessing shift-invariance. This provides certain flexibility in the AO system modeling, in particular, there is no need to use a WFS model based on the Fried Geometry, since any model that is shift-invariant is valid for the DKF. Additionally, we have shown a natural way to construct the shift-invariant state matrix with non-trivial

[2] The number of peaks is not necessarily equal to the number of turbulence layers, if the angular velocities of those layers are sufficiently close. Fortunately, this does not affect the reconstructor performance, since there is fundamentally no difference in the turbulence as seen at the pupil.

prediction power. While our application of DKF is restricted only to the SCAO case, the topic for the future work could be a generalisation of this approach to tomographic AO systems.

6 Acknowledgments

The authors would like to acknowledge the support of the Cooperative Research Centre for Space Environment Management (SERC Limited) through the Australian Government's Cooperative Research Centre Programme.

References

[1] P. Massioni, C. Kulcsar, H. F. Raynaud, and J. M. Conan, *Fast computation of an optimal controller for large-scale adaptive optics*, J. Opt. Soc. Am. A 28, 2298-2309 (2011).

[2] B. Ellerbroek, *Efficient computation of minimum-variance wave-front reconstructors with sparse matrix techniques*, J. Opt. Soc. Am. A 19, 1803-1816 (2002).

[3] P. Piatrou and M. Roggemann, *Performance study of Kalman filter controller for multiconjugate adaptive optics*, Appl. Opt. 46, 1446 - 1455 (2007).

[4] J.B. Burl, *Linear Optimal Control: H_2 and H_∞ Methods*, (Addison-Wesley, 1998)

[5] C. Vogel, *Computational Methods for Inverse Problems*, (SIAM, 2002).

[6] M. Roggemann, B. Welsh, *Imaging Through Turbulence*, (CRC Press, 1996).

[7] D. T. Gavel and D. M. Wiberg, *Towards strehl-optimizing adaptive optics controllers*, in Proc. SPIE 4839, 972-982 (2002).

[8] L. Poyneer, M. van Dam, and J. P. Veran, *Experimental verification of the frozen flow atmospheric turbulence assumption with use of astronomical adaptive optics telemetry* J. Opt. Soc. Am. A. Vol. 26, No. 4, (2009).

[9] P. Massioni, L. Gilles, B. Ellerbroek, *Adaptive distributed Kalman filtering with wind estimation for astronomical adaptive optics*, J. Opt. Soc. Am. A 32 2353-2364 (2015).

[10] S. Els, M. Schok, E. Bustos, J. Seguel, J. Vasquez. D. Walker, R. Riddle, W. Skidmore, T. Travoulion, K. Vogiatzis, *Four Years of Optical Turbulence Monitoring at the Cerro Tololo Inter-American Observatory (CTIO)*, Publications of the astronomical society of the Pacific, 121:922-934 (2009).

[11] E. Thiebaut, M. Tallon, *Fast minimum variance wavefront reconstruction for extremely large telescopes*, J. Opt. Soc. Am. A 27 1046-1059 (2010).

[12] R. W. Wilson, *SLODAR: measuring optical turbulence altitude with a Shack-Hartmann wavefront sensor*, Mon. Not. R. Astron. Soc. 337, 103-108 (2002).