Tensor-based predictive control for large-scale AO

Baptiste Sinquin [1] *and Michel Verhaegen* [2]
Delft Center for Systems and Control
Technische Universiteit Delft, the Netherlands

October, 22nd 2018

[1] b.sinquin@tudelft.nl
[2] m.verhaegen@tudelft.nl

The model assumptions for single-conjugate AO

The sensor is a Shack-Hartmann of size $N \times N$:

$$\mathbf{s}(k) = \mathbf{G}\phi(k) + \mathbf{e}(k) \tag{1}$$

where $\mathbf{e}(k) \sim \mathcal{N}(0, \sigma_e^2 \mathbf{I})$.
The mirror is a static device with a one time step delay:

$$\begin{bmatrix} \mathbf{s}_m(k) \\ \phi_m(k) \end{bmatrix} = \begin{bmatrix} \mathbf{B} \\ \mathbf{H} \end{bmatrix} \mathbf{u}(k-1) \tag{2}$$

The sensor measures the contribution of both the disturbance and of the corrections applied by the mirror:

$$\begin{bmatrix} \mathbf{s}(k) \\ \phi(k) \end{bmatrix} = \begin{bmatrix} \mathbf{s}_m(k) \\ \phi_m(k) \end{bmatrix} + \begin{bmatrix} \mathbf{s}_t(k) \\ \phi_t(k) \end{bmatrix} + \begin{bmatrix} \mathbf{e}(k) \\ 0 \end{bmatrix} \tag{3}$$

The LQG criteria in AO boils down to a linear-quadratic estimation of $\phi(k+1)$ and a deterministic control problem for deriving the control inputs written as:

$$\min_{\mathbf{u}(k)} \quad \|\widehat{\phi}(k+1|k)\|_2^2 + \mathbf{u}(k)^T \mathbf{Q} \mathbf{u}(k) \tag{4}$$

for $\mathbf{Q}$ semi-positive definite.

The model assumptions for single-conjugate AO

The sensor is a Shack-Hartmann of size $N \times N$:

$$\mathbf{s}(k) = \mathbf{G}\phi(k) + \mathbf{e}(k) \tag{1}$$

where $\mathbf{e}(k) \sim \mathcal{N}(0, \sigma_e^2 \mathbf{I})$.
The mirror is a static device with a one time step delay:

$$\begin{bmatrix} \mathbf{s}_m(k) \\ \phi_m(k) \end{bmatrix} = \begin{bmatrix} \mathbf{B} \\ \mathbf{H} \end{bmatrix} \mathbf{u}(k-1) \tag{2}$$

The sensor measures the contribution of both the disturbance and of the corrections applied by the mirror:

$$\begin{bmatrix} \mathbf{s}(k) \\ \phi(k) \end{bmatrix} = \begin{bmatrix} \mathbf{s}_m(k) \\ \phi_m(k) \end{bmatrix} + \begin{bmatrix} \mathbf{s}_t(k) \\ \phi_t(k) \end{bmatrix} + \begin{bmatrix} \mathbf{e}(k) \\ 0 \end{bmatrix} \tag{3}$$

The LQG criteria in AO boils down to a linear-quadratic estimation of $\phi(k+1)$ and a deterministic control problem for deriving the control inputs written as:

$$\min_{\mathbf{u}(k)} \quad \|\widehat{\phi_t}(k+1|k) + \mathbf{H}\mathbf{u}(k)\|_2^2 + \mathbf{u}(k)^T \mathbf{Q}\mathbf{u}(k) \tag{4}$$

for $\mathbf{Q}$ semi-positive definite.

Deriving an unbiased minimum-variance estimate of $\phi_t(k+1)$

The temporal dynamics of the sensor signals are modelled in general with a state-space model:

$$\begin{cases} \mathbf{x}(k+1) & = & \mathbf{A}\mathbf{x}(k) + \mathbf{K}\mathbf{v}(k) \\ \mathbf{s}_t(k) & = & \mathbf{C}\mathbf{x}(k) + \mathbf{v}(k) \end{cases} \tag{5}$$

An estimate of $\mathbf{s}_t(k+1)$ is then available using the innovation form:

$$\begin{cases} \widehat{\mathbf{x}}(k+1|k) & = & (\mathbf{A} - \mathbf{K}\mathbf{C})\widehat{\mathbf{x}}(k|k) + \mathbf{K}\mathbf{s}_t(k) \\ \widehat{\mathbf{s}}_t(k|k) & = & \mathbf{C}\widehat{\mathbf{x}}(k|k) \end{cases} \tag{6}$$

Data-driven methods - not scalable:

- Subspace identification, Hinnen and Verhaegen (2007)
- AutoRegressive modeling, Guyon and Males (2017)

When setting $\mathbf{C} = \mathbf{G}, \mathbf{x}(k) = \phi_t(k)$ and $\mathbf{A} = a\mathbf{I}$, solve a **Riccati equation**:

- Exploit sparsity, Correia et al. (2010)
- Distributed controller using FFT operations, Massioni et al. (2011)

Research questions

An alternative cost function for the LQR problem is:

$$\min_{\mathbf{u}(k)} \quad \|\widehat{\mathbf{s}}_t(k+1|k) + \mathbf{B}\mathbf{u}(k)\|_2^2 + \mathbf{u}(k)^T\mathbf{Q}\mathbf{u}(k) \tag{7}$$

The state-space model in innovation form is approximated by a VAR model[3] with temporal order p such that the prediction is:

$$\widehat{\mathbf{s}}_t(k+1|k) \approx \sum_{i=0}^{p-1} \mathbf{M}_i \mathbf{s}_t(k-i) \tag{8}$$

1. *What is a dense though data-sparse representation for identifying from data and in a scalable manner the spatial and temporal dynamics of the turbulence?*
2. *To what extent the data-driven approach proposed handles the balance between computational complexity and data storage, and minimizing the temporal error?*

[3] We assume the driving noise of this VAR model zero mean white Gaussian with identity covariance matrix.

Tensor auto-regressive models

The Kronecker product

Definition 1. *For two matrices* $\mathbf{A}, \mathbf{B}$ *in* $\mathbb{R}^{N\times N}$*, the Kronecker product* $\mathbf{A} \otimes \mathbf{B}$ *in* $\mathbb{R}^{N^2\times N^2}$ *is defined with:*

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \dots & a_{1N}\mathbf{B} \\ \vdots & & \vdots \\ a_{N1}\mathbf{B} & & a_{NN}\mathbf{B} \end{bmatrix}$$

Proposition 1. For matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$ of compatible sizes,

$$\text{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B})$$

Fibers, matricization of a tensor and the n-mode tensor product

Definition 2. *A n-mode fiber of a d-th order tensor* $\mathcal{X}$ *is a vector* $\mathcal{X}(i_1, \ldots, i_{n-1}, :, i_{n+1}, \ldots, i_d)$.

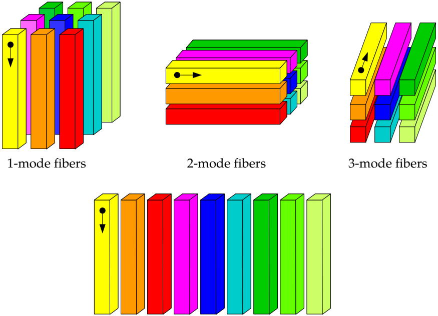


Figure 1: The n-mode matricization is formed by reshuffling the n-mode fibers to be the columns of the matrix $\mathbf{X}_{(n)}$.

Proposition 2. *Let* $(\mathcal{X}, \mathcal{Y}) \in \mathbb{R}^{J_1 \times \ldots \times J_d} \times \mathbb{R}^{I_1 \times \ldots \times I_d}$. *If* $\mathbf{M} \in \mathbb{R}^{I_n \times J_n}$, *then* $\mathcal{Y} = \mathcal{X} \times_n \mathbf{M}$ *is equivalently written with* $\mathbf{Y}_{(n)} = \mathbf{M}\mathbf{X}_{(n)}$.

The spatial dynamics are embedded into the coefficient matrices of VAR models.

A static input-output map between vectorized 2D signals has a two-level structure: the matrices are block-matrices.

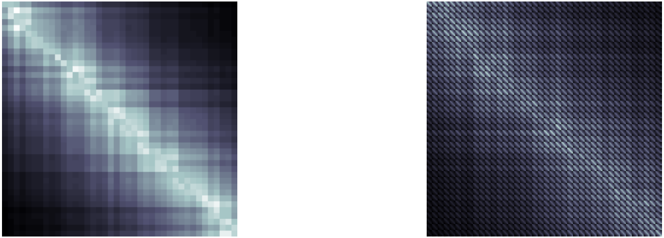


Figure 2: One-dimensional (left) and two-dimensional (right) array of sensor.

When the underlying function from $\mathbb{R}^2$ to $\mathbb{R}$ is separable in its coordinates, the matrix is written with a single Kronecker product.

The class of low Kronecker rank matrices

Proposition 3. *Any matrix* $\mathbf{M}_i \in \mathbb{R}^{2N^2 \times 2N^2}$ *can be decomposed with* $\sum_{j=1}^{r} \mathbf{M}_{i,j,2} \otimes \mathbf{M}_{i,j,1}$ *where* $(\mathbf{M}_{i,j,1}, \mathbf{M}_{i,j,2}) \in \mathbb{R}^{2N \times 2N} \times \mathbb{R}^{N \times N}$ *.*

The integer r is called the Kronecker rank.

Definition 3. $\mathbf{M}_i$ *is said to be low Kronecker rank when* $r \ll N$*.*

This parametrization is such that:

- it is not affine in the parameters.
- it is a data-sparse representation: rN^2 parameters to store compared to N^4 when unstructured.

A matrix-AR model

The VAR model is rewritten into a matrix-AR model:

$$\mathbf{S}_t(k+1) = \sum_{i=0}^{p-1} \sum_{j=1}^{r} \mathbf{M}_{i,j,1} \mathbf{S}_t(k-i) \mathbf{M}_{i,j,2}^T + \mathbf{V}(k) \tag{9}$$

where $\mathbf{S}_t(k) \in \mathbb{R}^{2N \times N}$ is such that:

$$\mathbf{S}_t(k) = \begin{bmatrix} \mathbf{s}_{t_{1,1}}(k) & \mathbf{s}_{t_{1,2}}(k) & \dots & \mathbf{s}_{t_{1,N}}(k) \\ \vdots & & & \vdots \\ \mathbf{s}_{t_{N,1}}(k) & \mathbf{s}_{t_{N,2}}(k) & \dots & \mathbf{s}_{t_{N,N}}(k) \end{bmatrix} \tag{10}$$

Computing $(\mathbf{M}_{i,j,2} \otimes \mathbf{M}_{i,j,1})\mathbf{s}_t(k)$ costs $\mathcal{O}(N^4)$ compared to $\mathcal{O}(N^3)$ reshuffling into $\mathbf{M}_{i,j,1}\mathbf{S}_t(k)\mathbf{M}_{i,j,2}^T$.

A matrix-AR model

The VAR model is rewritten into a matrix-AR model:

$$\mathbf{S}_t(k+1) = \sum_{i=0}^{p-1} \sum_{j=1}^{r} \mathbf{S}_t(k-i) \times_1 \mathbf{M}_{i,j,1} \times_2 \mathbf{M}_{i,j,2} + \mathbf{V}(k) \tag{9}$$

where $\mathbf{S}_t(k) \in \mathbb{R}^{2N \times N}$ is such that:

$$\mathbf{S}_t(k) = \begin{bmatrix} \mathbf{s}_{t_{1,1}}(k) & \mathbf{s}_{t_{1,2}}(k) & \dots & \mathbf{s}_{t_{1,N}}(k) \\ \vdots & & & \vdots \\ \mathbf{s}_{t_{N,1}}(k) & \mathbf{s}_{t_{N,2}}(k) & \dots & \mathbf{s}_{t_{N,N}}(k) \end{bmatrix} \tag{10}$$

Computing $(\mathbf{M}_{i,j,2} \otimes \mathbf{M}_{i,j,1})\mathbf{s}_t(k)$ costs $\mathcal{O}(N^4)$ compared to $\mathcal{O}(N^3)$ reshuffling into $\mathbf{M}_{i,j,1}\mathbf{S}_t(k)\mathbf{M}_{i,j,2}^T$.

Toward tensor AR models

Let $d \in \mathbb{N}$ and $(J_1, \ldots, J_d)$ integers such that $\prod_{i=1}^{d} J_i = 2N^2$. We parametrize $\mathbf{M}_i$ with:

$$\mathbf{M}_i = \sum_{j=1}^{r} \mathbf{M}_{i,j,d} \otimes \ldots \otimes \mathbf{M}_{i,j,1} \tag{11}$$

where $\mathbf{M}_i \in \mathbb{R}^{J_i \times J_i}$.

It can be shown that the VAR model can be transformed into a tensor AR model:

$$\mathcal{S}_t(k) = \sum_{i=0}^{p-1} \sum_{j=1}^{r} \mathcal{S}_t(k-i) \times_1 \mathbf{M}_{i,j,1} \times_2 \ldots \times_d \mathbf{M}_{i,i,d} + \mathcal{V}(k) \tag{12}$$

We define next the tensor $\mathcal{S}_t(k)$.

Tensorizing the sensor data

Tensorizing the sensor data corresponds to partitioning the 2D sensor array. The vector $\mathbf{s}_t(k)$ is reshaped into a tensor denoted as $\mathcal{S}_t(k) \in \mathbb{R}^{J_1 \times \dots \times J_d}$. Each sensor signal at node i, j is re-indexed with a tuple of size d rather than with two position indices.

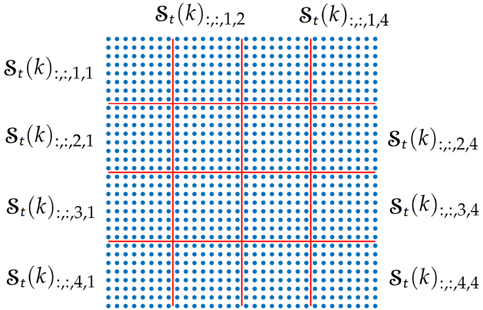


Figure 3: Partitioning a 2D array of sensor data with 32×32 nodes (in blue) with a 4th order tensor $\mathcal{S}_t(k) \in \mathbb{R}^{8 \times 8 \times 4 \times 4}$. The red lines indicate the partition into blocks of 8×8 matrices.

The computational advantages

Tensor-based predictive control: the closed-loop

Algorithm 1: Control algorithm minimizing the residual sensor measurement with a tensor-based wavefront prediction

1: $\mathbf{s}_t(k) = \mathbf{s}(k) - \mathbf{B}\mathbf{u}(k-1)$
2: Reshuffle $\mathbf{s}_t(k)$ into $\mathcal{S}_t(k)$
3: Compute a prediction $\widehat{\mathcal{S}}_{\boldsymbol{t}}(k+1|k) = \sum_{i=0}^{p-1} \sum_{j=1}^{r} \mathcal{S}_t(k-i) \times_1 \mathbf{M}_{i,j,1} \times_2 \ldots \times_d \mathbf{M}_{i,j,d}$
4: Reshuffle $\widehat{\mathcal{S}}_{\boldsymbol{t}}(k+1|k)$ into $\widehat{\mathbf{s}_t}(k+1|k)$
5: Solve the sparse least-squares to get $\mathbf{u}(k)$

Efficient online prediction for dense data-sparse models

There is no over-parametrization as when using Kronecker products: the entries of a tensor are only reshuffled.

Complexity: $\mathcal{O}(N^{2(d+1)/d})$

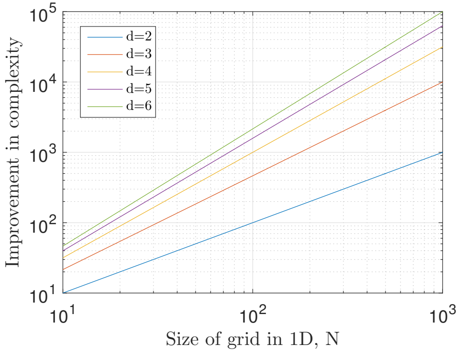


Figure 4: Ratio $\frac{N^4}{N^{2(d+1)/d}}$ which reflects the improvement in the computational complexity w.r.t the unstructured case for computing online a prediction as a function of the size of the array.

The identification problem

We collect N_t temporal samples in open-loop.

If $\mathbf{M}_{i,j,\bar{n}}$ is known for all $(i, j, \bar{n}) \in \{1, ..., p\} \times \{1, ..., r\} \times \{1, ..., n-1, n+1, ..., d\}$, then we identify the remaining ones from the (now convex) cost function:

$$\min_{\mathbf{M}_{i,j,n}} \quad \sum_{k=p+1}^{N_t} \|\mathbf{s}_t(k) - \sum_{i=1}^{p}\sum_{j=1}^{r}(\mathbf{M}_{i,j,d} \otimes \ldots \otimes \mathbf{M}_{i,j,1})\mathbf{s}_t(k-i)\|_2^2 \qquad (13)$$

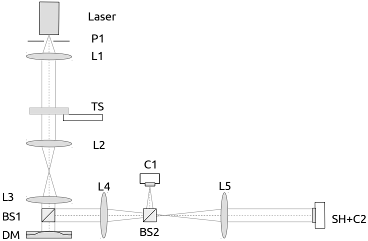
which is rewritten into:

$$\min_{\mathbf{M}_{i,j,n}} \sum_{k=p+1}^{N_t} \|\mathbf{S}_{t_{(n)}}(k) - \sum_{i=1}^{p}\sum_{j=1}^{r}\mathbf{M}_{i,j,n}\mathbf{S}_{t_{(n)}}(k-i)$$
$$\left(\mathbf{M}_{i,j,d} \otimes \ldots \otimes \mathbf{M}_{i,j,n+1} \otimes \mathbf{M}_{i,j,n-1} \otimes \ldots \otimes \mathbf{M}_{i,j,1}\right)^T\|_F^2 \qquad (14)$$

Starting from random initial guesses, this least-squares is solved sequentially for all $n \in \{1, \ldots, d\}$, and this is repeated until convergence to a stationary point.

Laboratory experiments

The optical testbed



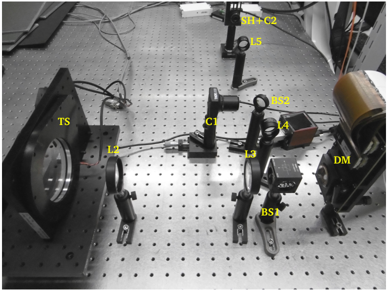


Figure 5: Views of the laboratory testbed. P1 is a pin-hole, L1 till L5 are lenses, TS is a rotating disk for simulating the turbulence, BS1 and BS2 are beam splitters, DM is the kilo-DM, C1 is the Point-Spread-Function camera, SH+C2 is the wavefront sensor.

Laser wavelength λ	635nm
Beam size	9mm
Fried parameter r_0	from 1.2 to 1.8mm
Active lenslets	689 (array of 30×30)
Active actuators	706

The experiment

Objective. *Analyze the prediction error when parametrizing the coefficient matrices with a sum of Kronecker.*

The experiment. We vary the rotation speed of the disk to vary the Greenwood per sample frequency ratio:

$$\bar{f} = \frac{f_G}{f_S} := 0.427 \frac{v}{r_0} \frac{1}{f_S} \tag{15}$$

We collect open-loop data, identify a model, check its accuracy on a different data batch, and close the loop.

Table 1: Partitions in the SH sensor associated with the parametrization

Tensor order, d	**Size of factor matrices**, J
2	$(60, 30)$
3	$(12, 6, 25)$
4	$(12, 5, 6, 5)$

Open-loop: validation dataset

$$\text{MSE:} \qquad \sigma^2 = \frac{1}{N_{val}} \sum_{k=0}^{N_{val}-1} \left(\widehat{\phi_t}(k+1|k) - \phi_t(k+1)\right)^2 \tag{16}$$

where N_{val} is the number of temporal samples in the validation dataset.

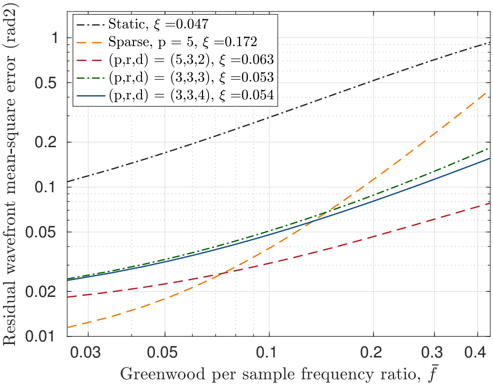


Figure 6: MSE on validation data as a function of the Greenwood per sample frequency ratio. ξ is the relative RMSE between the interpolation with a second order polynomial and the experimental points.

Influence of the parameters p and r

Table 2: Relative improvement on σ^2 when increasing either the temporal order or the Kronecker rank while $d = 2$. $(r_a, p_a) \to (r_b, p_b) := \frac{|\sigma^2_{(p,r)=(p_a,r_a)} - \sigma^2_{(p,r)=(p_b,r_b)}|}{\sigma^2_{(p,r)=(p_a,r_a)}}$

$(r_a, p_a) \to (r_b, p_b)$	$\bar{f} \in [0.026, 0.061]$	$\bar{f} \in [0.069, 0.10]$	$\bar{f} \in [0.11, 0.15]$	$\bar{f} \in [0.15, 0.22]$	$\bar{f} \in [0.24, 0.31]$	$\bar{f} \in [0.33, 0.40]$
$(3,1) \to (3,3)$	0.23	0.22	0.29	0.27	0.29	0.28
$(3,3) \to (3,5)$	0.067	0.10	0.11	0.12	0.13	0.14
$(1,3) \to (3,3)$	0.48	0.30	0.35	0.30	0.28	0.24
$(3,3) \to (5,3)$	0.16	0.14	0.13	0.12	0.10	0.11

Closed-loop: Strehl ratio

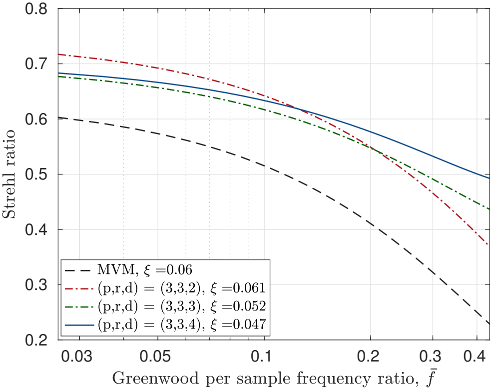


Figure 8: Strehl ratio as a function of the Greenwood per sample frequency ratio. ξ is the relative RMSE between the interpolation with a second order polynomial and the experimental points.

Closed-loop: influence of the temporal order p

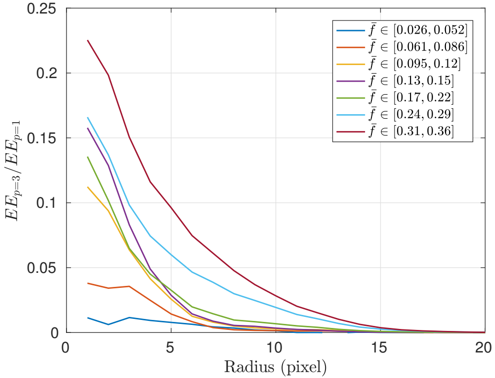


Figure 9: Encircled energy as a function of the Greenwood per sample frequency ratio. The relative improvement brought by the case $(p, r, d) = (3, 3, 4)$ over $(p, r, d) = (1, 3, 4)$ is shown.

Conclusion

Conclusion

The main points:

- A large-scale problem with unknown matrix structure is parametrized with a sum of Kronecker products.
- Trade-off between data-sparsity of the model representation and the bias between the true and approximated model structure.
- Especially relevant for large-scale sensors and AO systems operating in large Greenwood per sample frequency ratio.

Current/future work:

- Further tests of the algorithm under more various atmospheric settings
- How to efficiently solve Lyapunov and Riccati equations when all state-space matrices are sums-of-Kronecker?

Further references

AO-related papers:

- B. Sinquin and M. Verhaegen, "Tensor-based predictive control for extremely large-scale single conjugate adaptive optics," in *J. Opt. Soc. Am. A* 35, 1612-1626 (2018)
- G. Monchen, B. Sinquin, M. Verhaegen, "Recursive Kronecker-Based Vector Autoregressive Identification for Large-Scale Adaptive Optics", in *IEEE Control on Systems Technology*, 2018.

Matlab toolbox T4SID: `https://bitbucket.org/csi-dcsc/t4sid/`

Numerical experiments with OOMAO: settings

Number of lenslets	16×16
Diameter	4.8m
Fried parameter r_0 (meter)	0.15
Outer scale (meter)	30
Number of actuators	15×15
Number of temporal samples in identification batch	10^4

Atmosphere with 3 layers at altitude $\{0, 4, 10\} \times 10^3 m$ with speed $\{V, 10, 25\}$ in the wind directions $\{0, \pi/4, \pi\}$

Numerical experiments with OOMAO: results

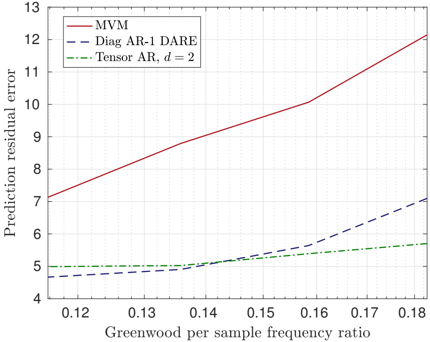


Figure 10: Residual wavefront in closed-loop for the MVM, Kalman filtering with $\mathbf{A} = a\mathbf{I}$, and a tensor autoregressive model with $d = 2$.

Dealing with a circular aperture

1. Pad with 0
2. Solve a low-rank matrix completion problem:

$$\min_{m_t(k)(i,j)_{(i,j)\in\mathcal{E}\setminus\mathcal{A}}} \quad \|\boldsymbol{M}_t(k)\|_\star \tag{17}$$

$$\text{s.t} \quad \forall(i,j)\in\mathcal{A}, m_t(k)(i,j) = s_t(i,j) \tag{18}$$

3. Recast the sensor data into a third-order tensor and estimate the missing data assuming a low-rank Canonical Polyadic Decomposition (CPD), i.e sum of few rank-one terms:

$$\mathfrak{X} = \sum_{i=1}^{r} \mathbf{a}_{j,1} \circ \ldots \circ \mathbf{a}_{j,d}$$