Delft University of Technology
sc42025 Filtering & Identification
Practical Assignment 2018-2019

# Turbulence modeling for adaptive optics

This assignment is part of the course *Filtering and Identification* (SC42025). It is intended to be solved by a *group of two* students. The goal is to use the methods learned throughout the course to solve control problems that arise in adaptive optics (AO) system. It covers weighted least squares, Kalman filtering, Vector- Auto-Regressive modeling and stochastic subspace identification. In solving this assignment you should use the following material:

- Course book: *Filtering and System Identification: A Least Squares Approach* [1]. Each question in this assignment will refer to the corresponding chapters of the books that are of interest.
- 1 Matlab file entitled: *AOloop-nocontrol.m.*
- 2 datasets:
  - *systemMatrices.mat* contains the matrices $G$, $H$ and measurement noise parameter $\sigma_e$.
  - *turbulenceData.mat* contains turbulence data collected in open-loop for 20 realizations.

It is *neither* allowed to use the *System Identification* toolbox from Matlab nor the *LTI Toolbox* from DCSC.

## Desired output

One report must be written by each group. A few recommendations follow. Please *read them carefully* to avoid unecessary decrease of your grade!

- The cover page of the report must contain: your names, student numbers, the date and a title of the document.
- Clearly indicate which question number is answered where in the report.
- Figures should be labeled and commented such that it is clear what to conclude from it.
- The Matlab scripts *written by yourself* shall be included in an Appendix, and indexed by question number.
- Read all questions carefully and make sure you have answered them completely. Try to be as complete as possible by clearly explaining all the steps you take in your report. Missing steps and explanations in the report will be penalized even if the final answer is correct.
- Although discussing the questions with your colleagues is encouraged, your should never copy text, figures or codes from other groups. Reports that are (partially) copied will not be graded!

## Handing in your solutions

Please leave a *hard copy* of your solutions in the box that will be placed in front of the DCSC secretariat. The deadline of the assignment is on 18 January 2019 at 5pm.

## Grading

You will receive a grade between 1 and 10. To pass the course *Filtering and Identification*, your grade for this assignment must be at least a 6.

## Support & using Piazza

If you have any questions or remarks, don't hesitate to post them on Piazza. In this way, both the instructors and your fellow students can help you out. Also, Piazza will be the main tool for instructors to communicate any clarifications of the assignment, so make sure you check it *regularly* to avoid missing any updates. Besides Piazza, the lecture on 19 December will be an introduction to the assignment and a couple of instruction sessions wild be held after the Christmas break (dates and locations will be announced soon).

# Introduction to Adaptive Optics

High resolution imaging in the visible spectrum from ground-based telescopes is seriously hampered by atmospheric turbulence. *Adaptive optics* (AO) is the system that corrects in real-time atmospheric aberrations as displayed in Figure 1. Incoming light is split in two beams and directed towards a wavefront sensor (WFS) and the science camera. The information provided by the WFS is processed to estimate a future wavefront and converted into voltage for the actuators located under a deformable mirror (DM). The DM flattens the wavefront in order to retrieve the original image quality as if there was no atmosphere.

## An adaptive optics model

While modeling real-life systems, a first direction is to use the knowledge available on the system and write corresponding equations. This approach is said to be based on *first principles.* In this part we introduce the AO equations based on first principles. It introduces the key elements in an AO system and helps to understand the AO Matlab-based simulator. First we start with describing the wavefront sensor and then the deformable mirror.

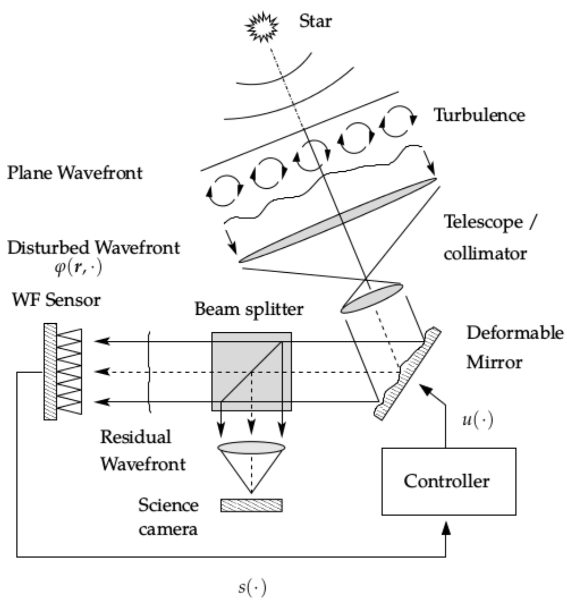



Figure 1: Schematic of an adaptive optics system. Courtesy: [2].

### The Shack-Hartmann sensor

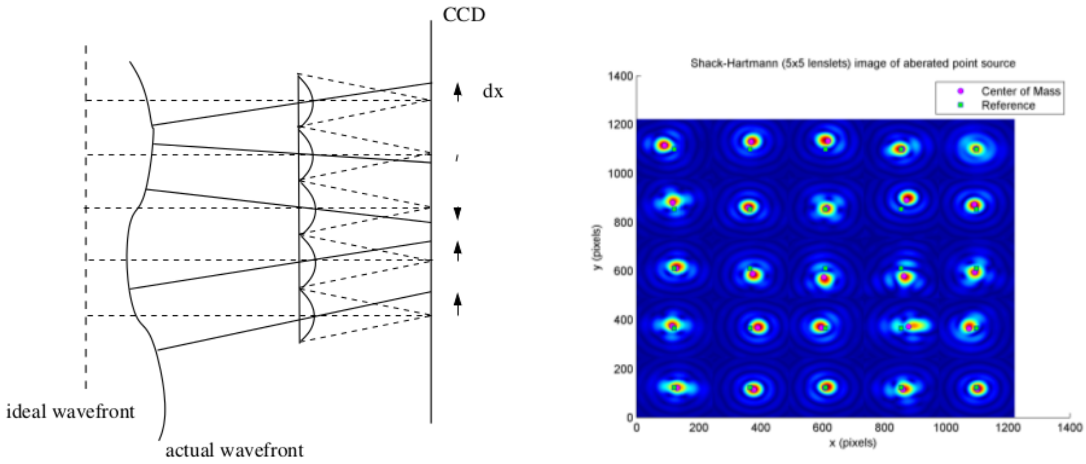
A Shack-Hartmann sensor is a sensor that measures the first derivative of the wavefront. It consists of a 2D-array of lenses, each of which project the wavefront on the camera located parallel to the lenses, at their focal distance. See Figure 2a for an illustration. A flat wavefront results in an image on the camera with dots well aligned one with another. However local tilts of the wavefront deviate the point where lights rays focus as displayed in Figure 2b. These local tilts are measured and are called the slopes. These slopes are put into a vector which is denoted with $s(k)$.

To clarify the ideas, we consider a square array of $p \times p$ lenslets. The wavefront is evaluated (sampled) at the corners of the subapertures (i.e. lenslets) of the Shack-Hartmann sensor. In other words, it is spatially sampled with $(p+1) \times (p+1)$ points:

$$\Phi(k) = \begin{bmatrix} \phi_{1,1}(k) & \dots & \phi_{1,(p+1)}(k) \\ \vdots & & \vdots \\ \phi_{(p+1),1}(k) & & \phi_{(p+1),(p+1)}(k) \end{bmatrix}, \qquad \phi(k) = \mathrm{vec}\Big(\Phi(k)\Big) \tag{1}$$

The slopes are related to the vectorized wavefront at time instant $k$, $\phi(k)$, with the linear measurement equation:

$$s_o(k) = G\phi(k) + e(k) \tag{2}$$




(a) (b)

Figure 2: (a) Schematic of an Shack-Hartmann array of lenses to measure the slopes of the wavefront aberration distributed in space. (b) Simulated reading of the Shack-Hartmann sensor for an aberrated wavefront. Courtesy: [2].

where $G \in \mathbb{R}^{2p^2 \times (p+1)^2}$ and $e(k)$ is zero-mean white Gaussian noise and with covariance matrix $\sigma_e^2 I$. The matrix $G$ corresponds to a finite-difference operation. The subscript in $s_o(k)$ indicates that these are *open-loop* measurements, i.e. without actuating the deformable mirror.

### The deformable mirror

We consider a deformable mirror with a square array of $m \times m$ actuators underneath its refractive surface that are able to change the surface's shape. The temporal dynamics of the mirror are neglected since the control frequency is much smaller than the first resonance frequency. The influence function of each actuator is modeled by a 2D Gaussian influence function. By setting the actuators to some values, a wavefront is induced by the mirror and is denoted as $\phi_{DM} \in \mathbb{R}^{(p+1)^2}$. A one-step delay is assumed between the time at which we apply the control inputs and the time at which the wavefront is induced, i.e the relationship between the control inputs and the induced wavefront is:

$$\phi_{DM}(k) = Hu(k-1) \tag{3}$$

where $H \in \mathbb{R}^{(p+1)^2 \times m^2}$ is the DM *influence matrix*. In the following we consider the special case where $H$ is a square matrix, i.e $p+1 = m$. The actuator signal $u(k-1)$ is the lifted vector of the 2D input data, with the same reordering as in (1). It means the actuators are located at the same position as the phase points $\phi$. This particular actuator-sensor geometry is called the *Fried geometry*.

### Closing the loop

Finally, the loop is closed. The residual wavefront $\epsilon(k)$ is defined by:

$$\epsilon(k) = \phi(k) - \phi_{DM}(k) = \phi(k) - Hu(k-1) \tag{4}$$

As also illustrated in Figure 1, the Shack-Hartmann measurements in closed loop correspond to the residual wavefront, i.e.

$$s(k) = G\epsilon(k) + e(k) \tag{5}$$

In adaptive optics, we are interested in flattening the residual wavefront to obtain the highest image resolution as possible. Because of the dynamic nature of the turbulence and the assumed one-step delay in the system, we will consider the control law to be

$$\min_{u(k)} \|\hat{\epsilon}(k+1|k)\|_2^2 \tag{6}$$

where

$$\hat{\epsilon}(k+1|k) = \hat{\phi}(k+1|k) - Hu(k) \tag{7}$$

is denoting the optimal one-step-ahead predictor given certain assumptions on the dynamical model. Computing $\hat{\epsilon}(k+1|k)$, or $\hat{\phi}(k+1|k)$, will be the main problem in this assignment.

*NB: If you want to learn more about AO systems, you can attend the course SC42030: Control for High Resolution Imaging and SC42065: Adaptive Optics Design Project.*

## Other important information

### Simulating an AO system

To help you get started with the assignments, all necessary information to the simulate the AO system is given in the m-file *AOloop-nocontrol.m* and the dataset *systemMatrices.mat*. *AOloop-nocontrol.m* gives an example code to simulate an open loop AO system and *systemMatrices.mat* contains the matrices $H$ and $G$ and the measurement noise parameter $\sigma_e$. For simulations, you should use the data called *phiSim*, such that you don't use the same data as for the identification.

### Available data for identification

Often, we have theoretical knowledge of the phase covariance matrix $C_\phi(0) = E[\phi(k)\phi^T(k)]$, but general knowledge on $C_\phi(\tau) = E[\phi(k)\phi^T(k-\tau)]$ for $\tau > 0$ is usually not available. However, in this assignment you can assume that you have access to the real turbulence data in *turbulenceData.mat* for the identification, called *phiIdent*. In practice, we only have the slopes $s(k)$, which are noisy and even contain unobservable modes.

Since we consider to have the data $\phi(k)$, we can use it to approximate a covariance matrix. For $N$ time samples of the wavefront $\phi(k),\ k = 1,\dots,N$, we get:

$$C_\phi(\tau) \approx \frac{1}{N-\tau}\sum_{i=\tau+1}^{N} \phi(i)\phi(i-\tau)^T \tag{8}$$

Of course, in practice we won't have this access and we would need other methods to approximate $C_\phi(\tau)$. For the interested students, a more realistic identification and control approach for AO is described in [2].

### Removing the mean

Because we can only measure local gradients with the sensor, the mean wavefront $\mathrm{mean}(\epsilon(k))$ cannot be measured. In order to have a fair comparison, it is important to remove the spatial mean of the residual wavefront (i.e. make sure that for each $k$, $\mathrm{mean}(\epsilon(k)) = 0$) before computing its variance).

## Data-driven turbulence modeling

Modeling the atmospheric turbulence dynamics is of key importance for predicting the near-future of the wavefront, and hence achieve optimal control performances. We study in this assignment the crucial role of having an accurate model in order to improve the control performances. You will implement three different methods, each one focusing on a different assumption:

1. Method 1: the spatial and temporal dynamics are decoupled by assuming that the wavefront has evolved as a random walk process during the computation time. The wavefront is first reconstructed and is then mapped onto the mirror.

2. Method 2: spatio-temporal dynamics of the wavefront are represented by a Vector- Auto-Regressive (VAR) model of order 1. A Kalman filter is derived to obtain the optimal prediction.

3. Method 3: spatio-temporal dynamics of the wavefront are represented by a stochastic state-space model which we identify from open-loop sensor data using stochastic subspace identification.

# References


[1] M. Verhaegen, V. Verdult. "Filtering and System Identification: a least-squares approach", Cambridge University Press, 2007.

[2] M. Verhaegen, G. Vdovin, O. Soloviev. "Control for High Resolution Imaging", Lecture notes sc4045, TU Delft, 2016.

# 1 Random walk model

First, you will perform *wavefront reconstruction*, i.e. finding an unbiased minimum variance estimate of $\phi(k)$ based on the sensor data $s(k)$ and, when available, prior statistical information.

1. [**Chapter 2-4.**] We first collect wavefront sensor data $\{s_o(k) \mid k = 1, \ldots, N_t\}$ in open-loop. We would like to reconstruct the wavefront. Determine an expression for the estimate of the wavefront $\phi(k)$ using the measurement $s_o(k)$ with the given matrix $G$ in *systemMatrices.mat*. Assume there is *no* prior information on $\phi(k)$ available.

2. [**Chapter 2-4.**] We now assume we have prior knowledge of the statistical properties of $\phi(k)$. Determine an unbiased minimum variance estimate, denoted by $\hat{\phi}(k|k)$, of the turbulence wavefront $\phi(k)$ using the open-loop measurement $s_o(k)$, the noise variance $\sigma_e^2$ and the prior wavefront information: $E[\phi(k)] = 0$ and $E\left[\phi(k)\phi(k)^T\right] = C_\phi(0) > 0$. *Hint: use* (8) *to compute an approximation of* $C_\phi(0)$.

Next, we consider a *random-walk* model to describe the turbulence dynamics, i.e the estimate of the atmospheric wavefront at time $k+1$ is such that:

$$\phi(k+1) = \phi(k) + \eta(k) \tag{9}$$

where $\eta(k)$ is a zero-mean white Gaussian noise with identity covariance matrix. We are closing the loop as defined in (4). *In this section only*, to formulate an estimate of $\epsilon(k)$, you can assume $E[\epsilon(k)] = 0$ and $E[\epsilon(k)\epsilon(k)^T] = C_\phi(0)$.

3. [**Chapter 2-4.**] Considering the closed loop system, derive an unbiased minimum variance estimate, denoted by $\hat{\epsilon}(k|k)$, of the residual wavefront $\epsilon(k)$ using the measurement $s(k)$ as in (5), the noise variance $\sigma_e^2$ and the prior wavefront information: $E[\epsilon(k)] = 0$ and $E\left[\epsilon(k)\epsilon(k)^T\right] = C_\phi(0) > 0$.

4. [**Chapter 2-4.**] Based on the random walk model, what is the optimal one step ahead prediction of $\epsilon(k)$, denoted by $\hat{\epsilon}(k+1|k)$ in terms the current optimal estimate $\hat{\epsilon}(k|k)$, control inputs $u(k)$ and $u(k-1)$ and matrix $H$.

5. [**Chapter 2-4.**] Denote $\delta u(k) := u(k) - u(k-1)$. Determine the optimal increment of the input commands $\delta u(k)$ that minimize the control law (6) as a function of the current output $s(k)$ and the system matrices.

Next, you have to implement the control law you derived above and discuss the performance of this control method.

6. [**Matlab.**] Write the Matlab routines to close the loop as depicted in the schematic in Figure 1. It is asked to insert the code in a function entitled:

$$[var_{eps}] = \text{AOloopRW}(G, H, C_\phi(0), \sigma_e, \phi_{sim})$$

where $\phi_{sim} \in \mathbb{R}^{(p+1)^2 \times N_t}$ is the simulation data from *turbulenceData.mat* and $var_{eps}$ is the variance of the residual wavefront when taking $N_t$ time points within the closed-loop operation.

7. [**Matlab.**] Compare the performances of this method with the case when no control is applied. What is the variance accounted for (see Eq. 10.22 in [1]) of the estimate $\hat{\epsilon}(k+1|k)$?

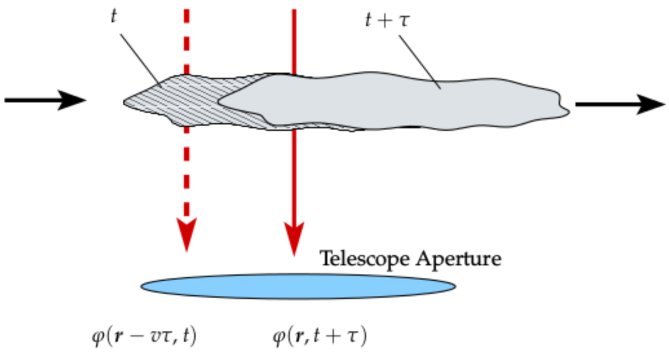



Figure 3: A layer of frozen turbulence is passing over the telescope aperture. $\varphi(\mathbf{r}, t)$ in the figure corresponds to the wavefront at position $\mathbf{r}$ and time $t$. The frozen flow assumption states that for a certain wind speed $v$, $\varphi(\mathbf{r} - v\tau, t) = \varphi(\mathbf{r}, t + \tau)$. Courtesy: [2].

## 2 Vector Auto-Regressive model of Order 1

The method studied in Section 1 ignores the spatio-temporal correlations. It is more reasonable to assume that the turbulence flows over the telescope aperture with a given pattern, see Figure 4 for an illustration. The time it takes for the turbulence to cross the line of sight of the telescope is way longer than a sampling period. Therefore, in this section we study a Vector Auto-Regressive (VAR) model as first approximation:

$$\phi(k+1) = A\phi(k) + w(k) \tag{10}$$

where $A \in \mathbb{R}^{(p+1)^2 \times (p+1)^2}$, and $w(k) \sim \mathcal{N}(0, C_w)$. Moreover, the signal $w(k)$ is uncorrelated with both the measurement noise $e(k)$ and the turbulent wavefront $\phi(k)$:

$$E\left[w(k)e(k)^T\right] = 0, \qquad E\left[w(k)\phi(k)^T\right] = 0 \tag{11}$$

1. [**Misc.**] Denote $C_\phi(0) = E\left[\phi(k)\phi(k)^T\right]$ and $C_\phi(1) = E\left[\phi(k+1)\phi(k)^T\right]$. Relate $C_\phi(0), C_\phi(1)$ and $A$ from the VAR1 model (10). Determine a closed-form expression for $A$ assuming the covariance information is known.

2. [**Misc.**] We assume the wavefront $\phi(k)$ is a wide-sense stationary signal. Derive a relationship between $C_\phi(0), C_w$ and $A$ in order to estimate the covariance matrix $C_w$.

3. [**Misc.**] From the equations (4), (5) and (10), formulate a state-space model in which the state is equal to the closed-loop residual wavefront $\epsilon(k)$ and output equal to $s(k)$.

As in the previous section, we want to find the optimal one step ahead predictor $\hat{\epsilon}(k+1|k)$. Since we now have a more realistic dynamical model, we expect the prediction to be more accurate. In the next part, you will derive a Kalman filter associated with the above state-space model.

4. [**Chapter 5.**] Write an expression for the Kalman filter in observer form corresponding to the state-space model. Explain how you can compute the Kalman gain $K$.

5. [**Chapter 5.**] Based on this model, give an expression for the optimal one step ahead prediction $\hat{\epsilon}(k+1|k)$ of $\epsilon(k)$.

6. [**Chapter 2-4.**] Using the control law (6) with the estimate $\hat{\epsilon}(k+1|k)$ from the Kalman filter, give an expression of the optimal control action $u(k)$.

7. [**Matlab.**] Write the Matlab routines to compute $A, C_w$ and the Kalman gain $K$ and embed it into:

$$[A, C_w, K] = \text{computeKalmanAR}(C_\phi(0), C_\phi(1), G, \sigma_e)$$

Write the Matlab code to close the loop and embed the code in the function:

$$[var_{eps}] = \text{AOloopAR}(G, H, C_\phi(0), \sigma_e, A, C_w, K, \phi_{sim})$$

8. [**Matlab.**] Compare the performance of this method with the controller based on the random-walk assumption. What is the VAF for the estimate $\hat{\epsilon}(k+1)$?

## 3 Subspace identification

Finally, we consider that the turbulence is modeled by a more general stochastic state-space model. Including the output equation (2), we can formulate a total open-loop system of the form:

$$x(k+1) = A_s x(k) + K_s v(k) \tag{12}$$

$$s(k) = C_s x(k) + v(k) \tag{13}$$

where $v(k)$ is the innovation sequence and $x(k)$ is a general state that does *not* necessarily equal $\epsilon(k)$. You will identify the matrices $A_s$, $C_s$ and $K_s$ from the turbulence data given in *turbulenceData.mat* using subspace identification. In order to obtain identification dataset $\{s_{id}(k), k = 1,, N\}$, you have to generate it yourself using $\phi_{id}(k)$ from *turbulenceData.mat* (i.e. *phiIdent*) and the measurement equation (2).

1. [**Chapter 9.**] Write the data equation and explain how to estimate the system matrices $A_s$, $C_s$ and the state sequence $x(k)$ using the subspace identification. *Hint: Make sure that $A_s$, $C_s$ and state sequence are retrieved up to the same similarity transformation. One way of achieving this is by following the N4SID method described in Section 9.6.2 of the book.*

2. [**Chapter 9.**] Explain how we can find an estimate of the covariance matrices and the Kalman gain $K_s$ using the estimated state matrices $A_s$, $C_s$, the estimated state sequence and the identification data. *Hint: you can use the method described in Section 9.6.3 of the book.*

Now we have obtained our model, we want to formulate the optimal one step ahead predictor $\hat{\epsilon}(k+1|k)$ and implement our controller.

3. [**Chapter 5.**] Based on this model, give an expression for a one step ahead prediction $\hat{\epsilon}(k+1|k)$ of $\epsilon(k)$. *Hint: first derive an optimal one-step ahead predictor $\hat{x}(k+1|k)$ and use it to find $\hat{\epsilon}(k+1|k)$. You can use that $\hat{\phi}(k+1|k) = \Gamma\hat{s}(k+1|k)$, where $\Gamma$ describes the relation you found in question 1.1.*

4. [**Chapter 2.**] Give an expression for the control action $u(k)$ minimizing (6) using the expression in the previous question.

5. [**Matlab.**] Write the Matlab routines to identify the matrices $A_s, C_s, K_s$ from open-loop wavefront sensor data with the method you derive above. Embed the code in the file:

$$[A_s, C_s, K_s] = \text{SubId}(s_{id}, N_{id}, N_{val}, s, n)$$

where $s_{id}$ are the simulated measurements given the turbulence data in *turbulenceData.mat*, $N_{id}$ and $N_{val}$ are the number of points used respectively for identification and validation. $s$ is the upper bound on the order $n$. Write the Matlab routines for the latter algorithm and embed the code in the file:

$$[var_{eps}] = \text{AOloopSID}(G, H, A_s, C_s, K_s, \sigma_e, \phi_{sim})$$

where $\sigma_e^2$ is the measurement noise variance. *Hint: since we have a MIMO model, you can take $s < n$ as long as $2p^2 s > n$.*

6. [**Matlab.**] What is the VAF of the estimate $\hat{\epsilon}(k+1|k)$? How does this compare to the other two models studied in this assignment? Make sure you clearly discuss how you have chosen the input parameters $s$, $n$, $N_{id}$ and $N_{val}$. Also compare the closed-loop performances of this approach to the other two methods.

# 4 Critical thinking

Finally, we will look back at the methods we have used and make some critical remarks on their performance, scalability and physical interpretation.

1. [**Misc./Matlab**] Throughout the exercise, we have derived methods for minimizing the 2-norm of the predicted wavefront that is coming from the atmosphere. We now wonder whether it is necessary to reconstruct the wavefront. Derive a method inspired on a random walk model for minimizing the residual slopes $s(k+1)$ rather than the residual wavefront $\epsilon(k+1)$. Compare it numerically with the results obtained in Section 1 and comment.

2. [**Misc.**] How do the methods perform compared to each other? Is this what you would expect? Discuss for all the four methods: random walk (Section 1), VAR1 (Section 2), state-space (Section 3) and random walk minimizing the slopes (question 4.1).

3. [**Misc.**] The Shack-Hartmann sensor has two unobservable modes. Since it can only measure the local slopes, the *piston* ($\phi(k) = \mathbf{1}$, i.e. an offset in the phase) cannot be measured. How does the second mode look like? You can use *imagesc* in matlab to visualize it on the square grid as in (1). How could you make sure that these modes do not appear in $\phi_{DM}$?

4. [**Misc./Matlab**] In practice, the $\sigma_e$ is not always known and becomes a user defined tuning parameter. What role does it play for reconstructing the wavefront as you have done in question 1.2? How do you tune it to increase the performance of the control method?
   *Hint: you can formulate the wavefront reconstruction problem as a minimization problem of the following type:*
   $$\min_{\phi(k)} \quad \|f(\phi(k))\|_2^2 + \sigma_e \|g(\phi(k))\|_2^2 \tag{14}$$
   *where $f(\phi(k))$ contains the data-fitting term and $g(\phi(k))$ the prior information on $\phi(k)$.*

5. [**Misc.**] For large-scale systems, it is important that the number of elementary operations $(+, \times)$ to be performed online doesn't sharply increase when the number of actuators $m^2$ or the number of sensors measurements $2p^2$ increases. What is the operation that has to be computed *online* for the control law derived in Section 1? How many of these elementary operations does this contain? Conclude on the scalability of the method.
   *Hint: make a distinction between online and offline computations, i.e. do not include operations that can be computed beforehand.*