



European Organisation
for Astronomical
Research in the
Southern Hemisphere

Organisation Européenne
pour des Recherches
Astronomiques
dans l'Hémisphère Austral

Europäische Organisation
für astronomische
Forschung in der
südlichen Hemisphäre

VERY LARGE TELESCOPE

ADAPTIVE OPTICS FACILITY

Pseudo-synthetic Interaction Matrix simulations

Doc. No.: VLT-TRE-ESO-22000-5114

Issue: 1

Date: 03.12.2012

Function		Name	Date	Signature
Author		J. Kolb	03/12/12	
Job Manager				
AOF	SE	P.-Y. Madec	03/12/12	
	PM	R. Arsenault	3/12/13	
Releaser		N. Hubin	3/12/12	

☒ This document is under configuration control

CO-AUTHORS

Co-Authors	Affiliation, Division

REVIEWERS

Reviewers	Affiliation, Division
Clémentine Béchet	ESO, AOD
Emmanuel Aller Carpentier	ESO, VIF
Stefan Stroebele	ESO, AOD
Pierre-Yves Madec	ESO, AOD

CHANGE RECORD

ISSUE	DATE	SECTION/PARA. AFFECTED	REASON/INITIATION DOCUMENTS/REMARKS
1	03.12.2012	ALL	First Issue

TABLE OF CONTENTS

1	Scope	6
2	List of Abbreviations & Acronyms	6
3	List of Applicable and Referenced Documents	7
3.1	Applicable documents	7
3.2	Reference documents	7
4	Introduction.....	9
5	Generalities	10
5.1	General structure of the code	10
5.2	Coordinates definition.....	12
6	SHWFS model.....	12
6.1	Input parameters	12
6.2	Main loop.....	13
6.3	Geometric model.....	16
6.3.1	Introduction	16
6.3.2	Z-tilt method (projection on Zernike Tip and Tilt modes)	17
6.3.3	G-tilt method (Angle of Arrivals)	18
6.3.4	Comparison Z-tilt vs. G-tilt.....	19
6.3.4.1	Introduction.....	19
6.3.4.2	Sensitivity to thresholding (no noise case)	20
6.3.4.3	Sensitivity to photon noise.....	21
6.3.4.4	Sensitivity to combined photon noise and thresholding.....	22
6.3.4.5	AOF case	23
6.3.4.6	Conclusion.....	23
6.3.5	Photon noise in geometric model.....	24
6.3.6	Other application: seeing estimation from NAOS real-time data	24
6.4	Sub-apertures definition (diffractive model).....	26
6.5	Wavelength	27
6.6	PSF computation (diffractive model)	28
6.6.1	Description of the code	28
6.6.2	Other application: long-exposure PSF	29
6.7	Image sensor simulation (diffractive model).....	30
6.7.1	Description of the code	30
6.7.1.1	General.....	30
6.7.1.2	Detector PSF	31
6.7.1.3	Excess noise	33
6.7.1.4	Photon Transfer Curve	34
6.7.2	Other application: the EELT WFS detector	34
6.8	Slopes Computation (diffractive model).....	35
6.8.1	Description of the code	35
6.8.2	Response verification.....	36
6.8.3	Other application: slopes computation from existing images.....	38
6.9	Diffractive model.....	40
6.9.1	Other application – WFS Linearity.....	40
6.9.1.1	MAD WFS	41
6.9.1.2	Source size.....	41

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 5 of 74</p>
--	--	--

6.9.1.3	Detector PSF	42
6.9.1.4	Photon and Read-Out noises	43
6.9.1.5	Measurement improvement on MAD	44
6.9.1.6	Detector PSF on MAD	44
6.9.1.7	Pixel scale on MAD	45
6.9.1.8	Centroiding method	46
6.9.2	Other application - Long-exposure VLT Active Optics images	49
7	Simulation of Interaction Matrix recording	51
7.1	Introduction	51
7.2	Pupil shape	51
7.3	AOF Influence Functions	52
7.3.1	Handling of FEA IFs	52
7.3.2	Other application: shell surface error	54
7.3.3	Other application: offload to the VLT M1 elastic modes	55
7.4	Interaction Matrices	55
7.5	Verification on NAOS	57
7.6	Mis-registration parameters estimation from comparison of IMs	59
7.7	Code efficiency	61
8	Closed loop	63
8.1	IM inversion	63
8.2	Phase Screens	64
8.3	Control	64
8.4	Closed loop	65
9	Future implementations	66
9.1	Elongated spot	66
9.2	Rotation lenslet / detector	66
9.3	Field stop	67
9.4	Multi-lambda	67
9.5	Lenslet aberrations	67
9.6	Pixel scale per sub-aperture	67
9.7	Higher order pupil aberrations	67
9.8	Different amplitude per actuator	68
9.9	DSM shape display / Forces display / saturations	68
9.10	Choice finite/infinite DM	68
9.11	Reduce extension of Influence Functions	68
9.12	Regularisation of IM inversion	69
10	Conclusion	70
11	Appendices: How to run the code?	71
11.1	Input file for the function 'SHWFS_PSIM'	71
11.2	PSF computation	72
11.3	Detector read-out simulation	72
11.4	WF sensing	73
11.5	Slopes measurement	73
11.6	IM recording	74

1 Scope

This document presents the code developed to build Pseudo-Synthetic Interaction Matrices for the AOF.

2 List of Abbreviations & Acronyms

This document employs several abbreviations and acronyms to refer concisely to an item, after it has been introduced. The following list is aimed to help the reader in recalling the extended meaning of each short expression:

AA	Angle of Arrival
AD	Applicable Document
ADC	Analog to Digital Converter
ADU	Analog-to-Digital Units
AO	Adaptive Optics
AOF	Adaptive Optics Facility
CCD	Charge-Coupled Device
CL	Closed Loop
CM	Control Matrix
CoG	Center of Gravity
DSM	Deformable Secondary Mirror
FEA	Finite Elements Analysis
GALACSI	Ground Adaptive LAYer Corrector for Spectroscopic Imaging
GLAO	Ground Layer AO
GRAAL	GRound layer Adaptive optics Assisted by Lasers
HO	High Order
HW	Hardware
IFs	Influence Functions
IM	Interaction Matrix
KL	Karhunen–Loève
LGS	Laser Guide Star
LO	Low Order
LTAO	Laser Tomography AO
MCM	Maintenance and Commissioning Mode (of GRAAL)
N/A	Not Applicable
NFM	Narrow Field Mode (of GALACSI)
NGS	Natural Guide Star
OL	Open Loop
PSF	Point Spread Function
PSIM	Pseudo-Synthetic Interaction Matrix
RD	Reference Document
SCAO	Single Conjugate AO
SHWFS	Shack-Hartmann WFS
SVD	Single Value Decomposition

	Pseudo-synthetic Interaction Matrix simulations	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 7 of 74
--	--	---

SW	Software
TBC	To Be Confirmed
TBD	To Be Defined
TSVD	Truncated SVD
TT	Tip-Tilt
VLT	Very Large Telescope
WFE	Wave Front Error
WFM	Wide Field Mode (of GALACSI)
WF	Wave Front
WFS	Wave Front Sensor

3 List of Applicable and Referenced Documents

3.1 Applicable documents

#	Title	Document number	Issue	Date
AD1.				

3.2 Reference documents

#	Title	Document number	Issue	Date
RD1.	A. Tokovinin: <i>From Differential Image Motion to seeing</i>	PASP 114:1156–1166		October 2002
RD2.	P. Martinez et al.: <i>Atmospheric image blur with finite outer scale or partial adaptive correction</i>	A&A Volume 516, id.A90		June 2010
RD3.	P. Martinez et al.: <i>On the difference between seeing and image quality</i>	ESO Messenger No 141		September 2010
RD4.	J. Janesick: <i>Photon Transfer $\lambda \rightarrow DN$</i>	SPIE Press, PM 170, Bellingham, WA		2007
RD5.	P. Martinez et al.: <i>How to use Active Optics Shack-Hartmann data to measure the seeing at the focal plane of a telescope</i>	Proceedings of the conference “Astronomical site testing data in Chile”		To be published in 2011
RD6.	P. Martinez et al.: <i>Active optics Shack-Hartmann sensor: using spot sizes to measure the seeing at the focal plane of a telescope</i>	Mon. Not. R. Astron. Soc.		To be published in 2011
RD7.	J. Kolb, <i>DSM fitting error analysis</i>	VLT-TRE-ESO-22000-5070	1	22.09.2010
RD8.	P. Dierickx, <i>Polynomial</i>	VLT-SPE-ESO-11110-0006	3	13.10.1992

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 8 of 74
--	--	---

	<i>expansion of the sixteen first natural modes of the VLT primary mirrors</i>			
RD9.	C. Béchet, <i>AO Interaction Matrix update in closed-loop and on-sky: method and simulation results</i>	VLT-TRE-ESO-22000-5312	1	17.05.2011
RD10.	E. Aller Carpentier, <i>Extreme Adaptive Optics system optimization for High Contrast Imaging with the High Order Test bench</i>	PhD thesis		June 2011
RD11.	J. Kolb, <i>Calibration and test tools for MCAO systems: Application to the ESO Demonstrator MAD</i>	PhD thesis		December 2005
RD12.	S. Oberti, <i>Calibration strategies overview for large DM AO system</i>	VLT-TRE-ESO-11250-3761	1	06.09.2005
RD13.	M. Le Louarn, <i>AO simulation methods</i>	VLT-TRE-ESO-14850-4204	2	21.12.2007
RD14.	Joana Santos, <i>Linearity and plate scale of the Multi-Shack Hartmann WFS</i>	MAD-TRE-0010	1	01.07.2005
RD15.	Jerome Paufigue, <i>Interaction matrix representation for AO systems</i>	Internal Memorandum	1	02.11.2010
RD16.	Francois Roddier, <i>Adaptive Optics in Astronomy</i>	ISBN 0 521 55375		First published 1999

4 Introduction

In an Adaptive Optics (AO) system, the Interaction Matrix (IM) gathers the response of the Wave-Front Sensor(s) (WFS) to the actuation of the Deformable Mirror (DM). The baseline for the calibration of the IMs of the Adaptive Optics Facility (AOF) project is to generate (by the mean of computer simulation) Pseudo-Synthetic Interaction Matrices (PSIM). The main reason is that unlike in a post-focal AO system, there is no possibility to place a fibre source at a focal plane before the DM to record an IM. Those are Synthetic because they use a model of the AO system, but are only Pseudo-Synthetic because they incorporate some measured properties of the system to fine-tune the IM.

The ASSIST test bench will allow comparing the quality of the PSIM with ones measured on a fibre, and once the AOF installed on the telescope there will still be the possibility to go through the long procedure of recording an IM on the sky for comparison with the PSIM.

The first step towards generating a PSIM is to build a model of the system that is ready to incorporate measured system parameters. This model, written in Matlab, is presented in this document, including some measured properties. Its main component is the simulation of a Shack-Hartmann Wave-Front Sensor (SHWFS), but it comprises also a model of DM and of the optical aberrations that will create mis-registration between the two.

The general structure of the code is first introduced (§5) before the SHWFS model is described in detail (§6). More than described, each functionality of the model has been tested to make sure that it is performing as expected and the results of those tests are also reported. The code was written in a modular way so that parts of it can be used for other applications, and when it is the case the report mentions it.

The model of the DM and of the system mis-registration are presented in the §7. In this same §7 are shown the first PSIMs and how the code has been optimized for speed, §8 focuses on the application of the PSIM code to the simulation of a full AO loop, §9 lists possible future improvement, just before the conclusion (§10). As appendices (§11) are given some examples on how to use the various pieces of the PSIM code.

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 10 of 74
--	--	--

5 Generalities

5.1 General structure of the code

Figure 1 shows the general structure of the code for simulation of Pseudo-Synthetic Interaction Matrix (PSIM). All the routines that include “PSIM” in their name have been created especially for this application, or improved from existing pieces of code.

The core of the code is the simulation of the SHWFS. This is done by the routine “SHWFS_PSIM” that loads the WFS parameters from an input file “SHWFS_input[...]”, initiates variables, and saves them before performing the loop on the sub-apertures with the routine “SHWFS_PSIM_Mainloop”. This one slices the input WF into sub-apertures and outputs the slopes measured using a geometric or diffractive model, according to settings in the input file “SHWFS_input[...]”. In the second case, PSFs are generated from the WF in the sub-apertures (routine “makepsf_PSIM”), the resulting image goes through a detector simulation (“ImSensor_PSIM”) and the computation of the slopes (“Slopes_PSIM”) via the centroiding routine “Centroid_PSIM” that can use different methods (Center of Gravity, weighted CoG, Correlation).

In order to generate an IM, Deformable Mirror modes have been previously generated and saved in the mat-files “[Modes]” and are successively analysed by the SHWFS model (the loop is assured by the routine “makeIM_PSIM”).

Finally, the control matrix can be created by SVD (“invIM”) and a SCAO closed loop is simulated by “Closed_Loop_PSIM” using turbulence Phase Screens generated by “createPS_PSIM”.

The rest of the routines shown on Figure 1 are either minor or part of a Matlab toolbox:

- “readfits” and “write_fits” are obvious
- “createZ_PSIM” generates Zernike polynomials (using “FindNM” and “ZPolynomeQuick”)
- “nansum” and “nanmean” are coming from the Matlab Statistics Toolbox
- “bin_PSIM” bins an image by a given factor
- “Crop_PSIM” crops a part of an image
- “Corr_PSIM” computes the criterion to minimize in the case of centroiding by correlation (least square fit with a Gaussian psf)
- “poissrnd” (Statistics Toolbox) and “imnoise” (Image Processing Toolbox) add respectively Poisson and random noise to an image.

The most important routines will be described in the following paragraphs, together with the results of tests performed to validate the routines and their performance.

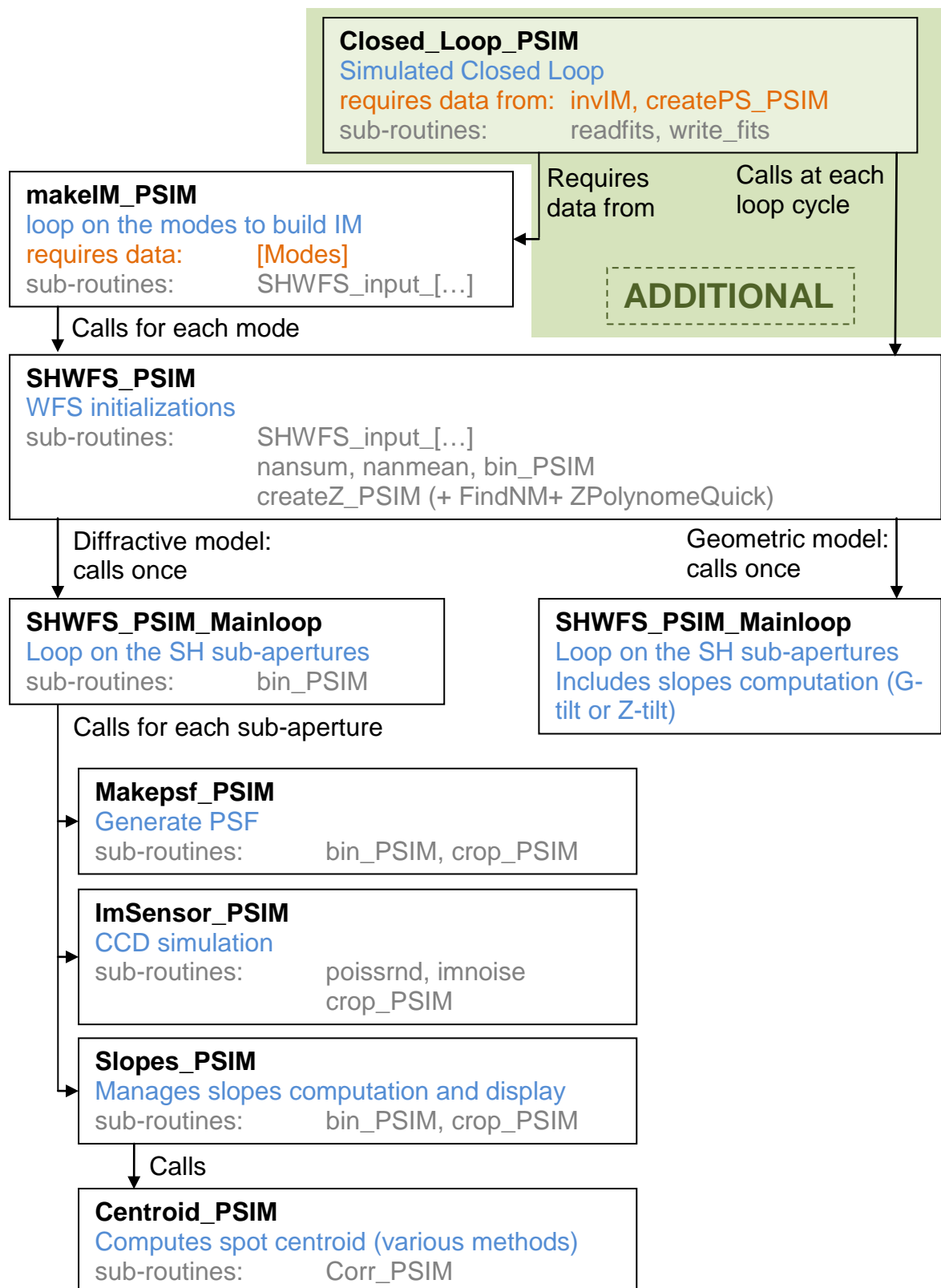


Figure 1: general structure of the code for simulating a Pseudo-Synthetic Interaction Matrix. The additional code in green is not required for PSIM generation but only for Closed Loop simulations.

5.2 Coordinates definition

In the whole code we have used the same definition of coordinates for wavefronts, pixels... This definition is illustrated by the left drawing on the Figure 2 and referred to as “Average” approach. In that case, if we want to sample an image of size d by points separated by $d/3$, we use 3 points at coordinates $d/6$, $d/2$, $5d/6$. Each point is then supposed to represent the image averaged over an area $d/3 \times d/3$. It is what physically happens when an image sensor (detector) pixel intercepts a portion of image and averages all the photons in it to give only one value, and this one the main reason for choosing this approach.

By contrast, the other approach (“Sampling” approach illustrated on Figure 2, right) uses 4 points at coordinates 0 , $d/3$, $2d/3$, d . In that case the borders of the image seem to be better defined. Such sampling is what occurs when the image is numerically computed (by FFT for example) on those exact points.

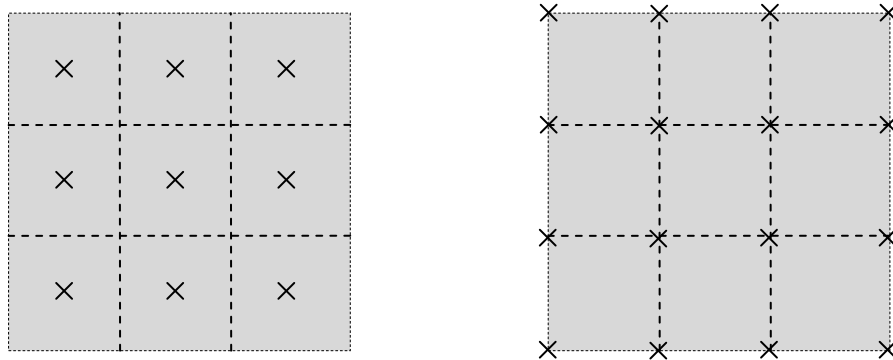


Figure 2: 2 possible ways of sampling an image (grey area). Left: “Average” approach. Right: “Sampling” approach.

6 SHWFS model

6.1 Input parameters

The parameters describing the SHWFS are stored in the Matlab m-file “SHWFS_input[...]”:

- Model (geometric or diffractive)
- Centroiding method (when model is diffractive) + a parameter (correlation map, weighting...)
- Telescope diameter
- Number of sub-apertures across the diameter
- Number of pixels per sub-aperture
- Pixel scale (in arc seconds)
- Sensing wavelength
- Source size (for extended sources)
- Pixel physical size

	<p>Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 13 of 74
--	---	--

- Inner and outer pupil diameters defined by the pupil mask on the lenslet array
- Sub-aperture illumination threshold (see §7.2 about management of truncated sub-apertures)
- Possible over-sampling of the generated image before sampling by the detector's pixels

And in the case noise is present:

- Number of photons (for photon noise)
- Exposure time (for Dark current)
- Quantum Efficiency (not a noise but a characteristics of the detector)
- PSF (charge diffusion factor)
- Dark current
- Detector saturation level
- Detector conversion factor (microvolts per electron)
- Detector Bias voltage
- Excess noise (for Electron Multiplying CCDs)
- Read-Out Noise
- ADC gain (electrons per ADU)

From those parameters, the routine "SHWFS_PSIM" computes other parameters, saves them or loads them for time saving in case the routine is used in a loop (like when computing an IM):

- Coordinates of the valid and of the fully illuminated sub-apertures
- Pupil mask
- Parameters of "fake" Gaussian spots in geometric model (see second paragraph of §6.2)
- Tilt reference in geometric model
- Tilt to move the spot to the crossing of 4 pixels in diffractive model (FFT produces a spot centred on one pixel)
- Detector PSF image

6.2 Main loop

The routine "SHWFS_PSIM_Mainloop" slices the WF into sub-apertures, calls for sub-routines for PSF generation and/or detector simulations and/or slopes computation, and displays some information (see example on Figure 3):

- Input WF
- Slopes represented by arrows
- X and Y slopes on 2D maps of values
- Detector image with super-imposition of SHWFS grid and sub-apertures centre
- Detector image as analyzed by the centroiding routine (after thresholding...)

In the case a geometric model is used, "fake" Gaussian spots are generated from the slopes measurements and displayed as "reconstructed" WFS image.

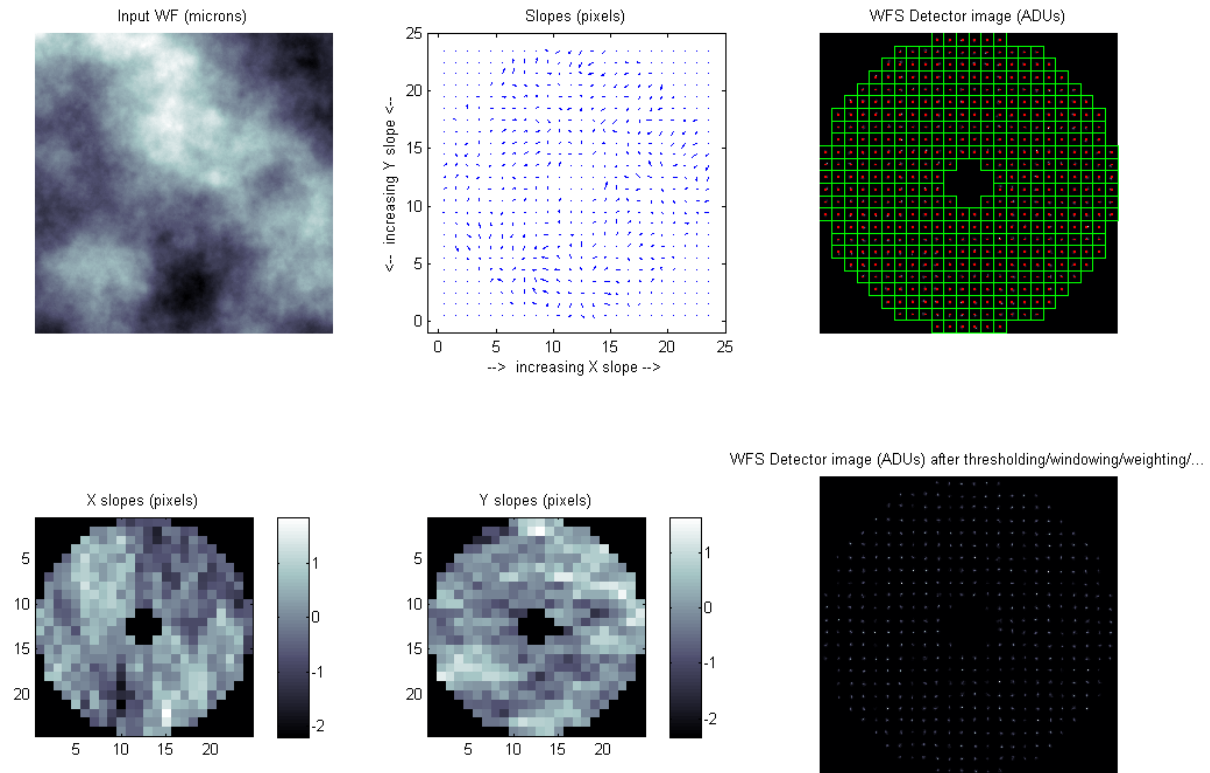


Figure 3: Typical display generated by the routine “SHWFS_PSIM_Mainloop”

To verify that the code can be used to simulate any WFS configuration and not only the AOF one for which it was created, several set of parameters were used to simulate various existing WFS:

- VLT Active Optics 24 x 24 (Figure 3)
- AOF 40 X 40 (Figure 4, top)
- MAD 8 X 8 (Figure 4, bottom-left)
- TT SENSOR 1 X 1 with noise (Figure 4, bottom-right)

In the following we will focus on the AOF High-Order 40 x 40 WFS model.

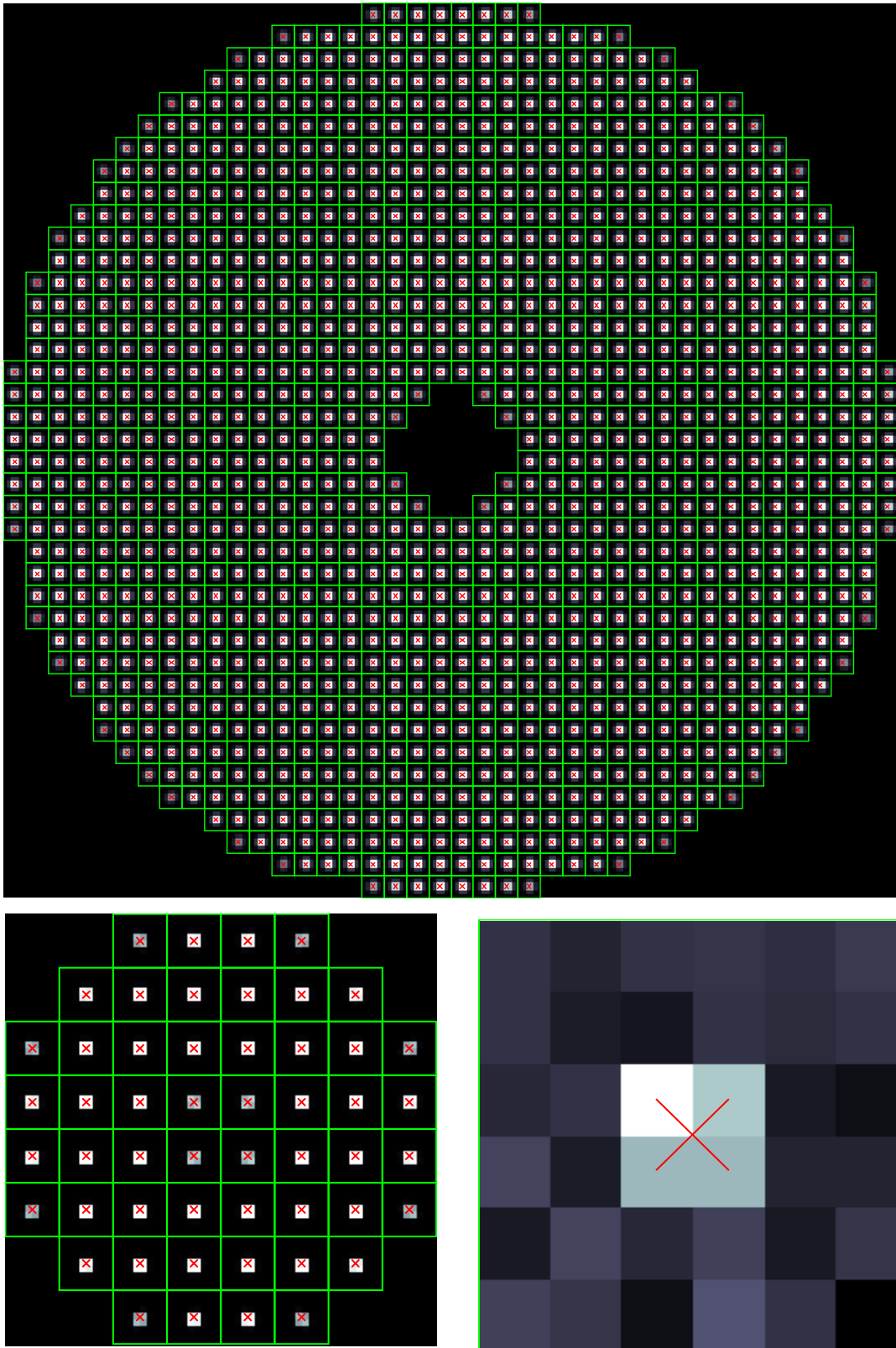


Figure 4: validation of the SHWFS simulation in different configurations: AOF (top), MAD (bottom-left), TT sensor (1 sub-aperture of 6x6 pixels) with noise (bottom-right).

	<p style="text-align: center;">Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue: 1 Date: 03.12.2012 Page: 16 of 74</p>
---	---	--

6.3 Geometric model

6.3.1 Introduction

The principle of a SHWFS consists in measuring the angle of arrival of the light in each sub-aperture (\mathbf{a}_x along one axis, \mathbf{a}_y along the other), which is proportional to the gradient of the phase ϕ in that sub-aperture (of area \mathbf{A}_{sa}):

$$a_x = \frac{\lambda}{2\pi A_{sa}} \int_{\text{sub-aperture}} \frac{\partial \phi}{\partial x} dx dy \quad (1)$$

In practice one measures the CoG (\mathbf{c}_x and \mathbf{c}_y) of the spots at the focal plane of the lenslet array:

$$c_x = \frac{\sum_{i,j} x_{i,j} I_{i,j}}{\sum_{i,j} I_{i,j}} \text{ and } c_y = \frac{\sum_{i,j} y_{i,j} I_{i,j}}{\sum_{i,j} I_{i,j}} \quad (2)$$

where $I_{i,j}$ and $(x_{i,j}, y_{i,j})$ are the signal and the position coordinates of the CCD pixel (i,j) . In RD16 is stated that it is possible to show that by replacing the discrete sum by a continuous integral and neglecting the scintillation, eq. (2) exactly determines the average WF slopes over the sub-aperture of area \mathbf{A}_{sa} , i.e. the angles of arrival \mathbf{a}_x and \mathbf{a}_y (on the sky):

$$a_x = \frac{c_x}{fM} \text{ and } a_y = \frac{c_y}{fM} \quad (3)$$

where f is the lenslet focal length and M the magnification between the lenslet plane and the telescope entrance plane.

A geometric model aiming at reproducing at best the behaviour of an ideal SHWFS without computing the PSFs should then be based on the computation of the derivative of the phase. It is equivalent to computing the average phase difference between the two sides of the sub-aperture orthogonal to the slope direction.

This method of **Angles of Arrival** is also called **G-tilt** method, where the G stands for centre of Gravity, because it is rigorously equivalent to the CoG measurement on the PSF in ideal conditions (no noise, infinite sampling, no clipping and no threshold). It has been implemented in the PSIM code (see 6.3.3).

Another method has also been implemented (see 6.3.2) because in real conditions which are never ideal, it held the promise of being faster to compute while giving results as good as the G-tilt method, in spite of having no formal link to the CoG measurement. This is the **Z-tilt** method, where Z stands for Zernike, because it is based on the projection of the WF on Zernike Tip and Tilt modes.

In the following chapters I will draw a comparison of both methods based on several criteria, and conclude on the necessity to use the formally correct G-tilt method, or on the possibility to use the approximate Z-tilt method (faster).

Note that whatever the input number of pixel of the detector, the geometric model can be used with as little as 2x2 pixels per sub-aperture (but the quality of the slopes measurement will then be poorer).

6.3.2 Z-tilt method (projection on Zernike Tip and Tilt modes)

In this model, the WF in a sub-aperture is projected on reference Tip and Tilt and the coefficients of this projection are the X and Y slopes (Z-tilts).

Special care has to be taken in the definition of the reference Tip and Tilt (see §5.2), especially in the non-fully illuminated sub-apertures.

On Figure 5 is shown a portion of the reference Tip map in the case of the AOF SHWFS, with 240 pixels across the pupil (6 pixels per sub-aperture). One can see the identical map in all full-illuminated sub-apertures, and different maps in the non-fully illuminated ones: it is not only a truncation of the first ones.

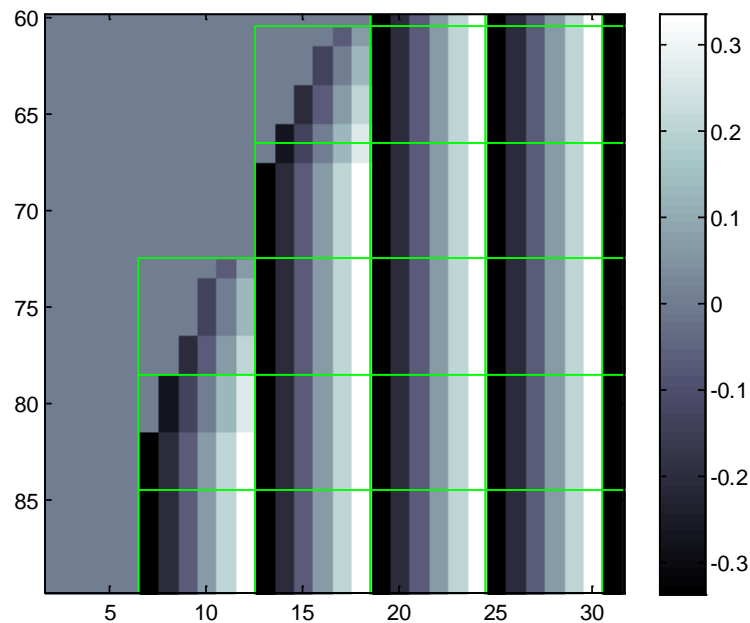


Figure 5: Portion of the reference Tip map. The grey zone in the top right is outside of the pupil.

We have verified that the response to a flat WF is equal to zero, as shown on the example of Figure 6: the largest slopes are of the order of 10^{-15} pixel (numerical precision).

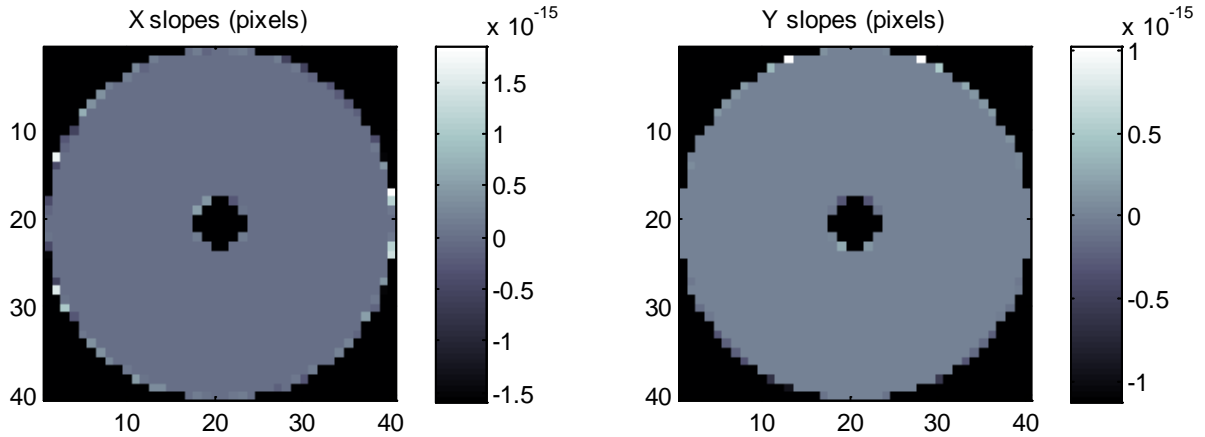


Figure 6: Response of the SHWFS geometric model (Z-tilt method) to a flat WF.

The correct projection of reference Tip and Tilt also ensures the linear response of the SHWFS. This is tested by measuring the response of the model to a Tilt of an angle equal to the pixel scale. This response is shown on the Figure 7. We see that the response on the Y axis to a Tilt in X is increased compared to the flat WF case (maximum 10^{-14}), but still in the numerical noise. The response in X is very close to 1 in all sub-apertures. A second scale has been added to the left to show the difference of the X slopes with the reference (equal to 1). Those differences are between -5.37×10^{-12} and -5.43×10^{-12} , in the numerical noise.

The behaviour as response to a Tilt in Y is the same but with inverting the axes.

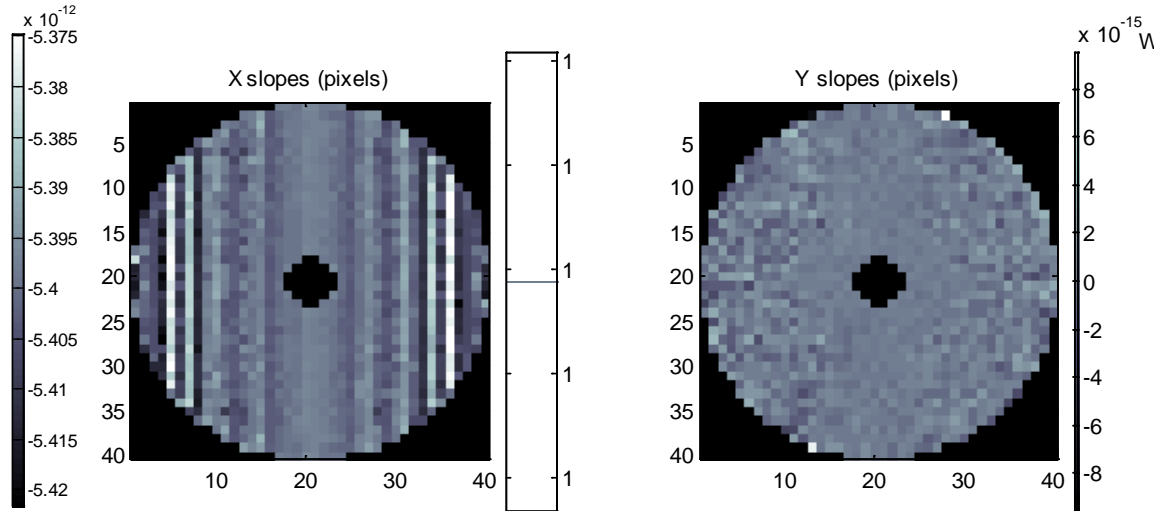


Figure 7: Response of the SHWFS geometric model (Z-tilt method) to a Tilt. The bar on the left is the scale for the difference between the X slopes and the reference (equal to 1).

6.3.3 G-tilt method (Angle of Arrivals)

The AA in X (resp. Y) is computed by subtracting the WF at the right (top) border of sub-aperture and the one at the left (bottom), then dividing by the sub-aperture size, without caring about the WF in the middle. As for the Z-tilt, special care has to be taken when simulating the partially-illuminated sub-apertures.

Figure 8 shows the simulation pixels numbering in one of the partially illuminated sub-apertures already shown on Figure 5. Equations 4 and 5 detail the computation of the X and Y Angles of Arrival for this sub-aperture (units of simulation pixel, to be renormalized for the actual sub-aperture size).

				1	2
			3	4	5
			6	7	8
			9	10	11
		12	13	14	15
		16	17	18	19

Figure 8: Numbering of the simulation pixels in a partially illuminated sub-aperture.

$$c_x = \frac{\varphi_2 - \varphi_1}{2} + \frac{\varphi_5 + \varphi_8 + \varphi_{11} - \varphi_3 - \varphi_6 - \varphi_9}{3} + \frac{\varphi_{15} + \varphi_{19} - \varphi_{12} - \varphi_{16}}{4} \quad (4)$$

$$c_y = \frac{\varphi_{12} - \varphi_{16}}{2} + \frac{\varphi_3 + \varphi_{17}}{5} + \frac{\varphi_1 + \varphi_2 - \varphi_{18} - \varphi_{19}}{6} \quad (5)$$

The routine “Centroid_PSIM” offers the possibility to compute the G-tilt as well as the Z-tilt, and it was verified that the response to a flat WF and to a pure tilt are the same as for the Z-tilt (no noise case, similar to Figure 6 and Figure 7).

6.3.4 Comparison Z-tilt vs. G-tilt

6.3.4.1 Introduction

The Z-tilt method (§6.3.2) measures exactly the tilt (to the Zernike sense) in the sub-apertures, but the physical behaviour of a SHWFS in ideal conditions is correctly described by the G-tilt (AA). This difference can be well illustrated by the example of the Coma aberration within a sub-aperture, which is orthogonal to Tip and Tilt as Zernike modes, but does produce a displacement of the sub-aperture’s PSF CoG.

The difference between Z-tilt and G-tilt is studied in RD1, from which Figure 9 was copied. One can see that in presence of Coma, Z-tilt (not represented) always gives zero, while G-tilt (= Angle of Arrival) gives a linear response. RD1 shows that things get more complicated when one adds thresholding or windowing to CoG, or makes a Gaussian fit (a method between weighted CoG and correlation). In those cases a diffractive SHWFS seems to behave more like it measured Z-tilt, but on the other hands the example of Figure 8 uses very aggressive windowing (at first Airy ring) or thresholding (20% of the image maximum) parameters.

I have tried to reproduce and extend the results of Figure 9, as one can see on Figure 10 to Figure 13. I have created a square sub-aperture with the following characteristics: diameter = 0.2 m, 48x48 pixels of 0.2 arcsec, and studied its response to a Coma (defined on a circular support within the square sub-aperture).

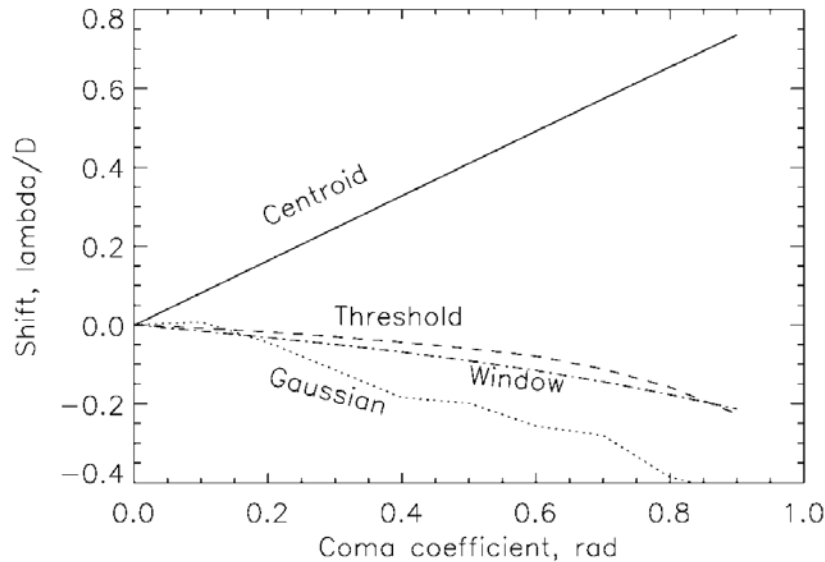


Figure 9: From RD1: Displacement of the image centre computed by different methods (without noise) as a function of the coma aberration. Solid line: true centroid; dashed line: thresholding at 20% of image max.; dash-dotted line: windowing at first Airy ring; dotted line: Gaussian fit.

6.3.4.2 Sensitivity to thresholding (no noise case)

One can see on Figure 9:

- as on Figure 8, the Z-tilt method doesn't see coma while G-tilt method does.
- CoG (diffractive model) sees something very close to G-tilt, the difference being explained by the limited FoV of the sub-aperture (clipping at the edges). This is valid for an image with no noise only.
- Adding an increasing threshold to the CoG changes completely the behaviour of the sub-aperture, which response to coma is smaller and smaller (seeming then closer to Z-tilt) until even inverting for a threshold value larger than 10% of the maximum sub-aperture value. This is due to the fact that the coma PSF is strongly asymmetric.

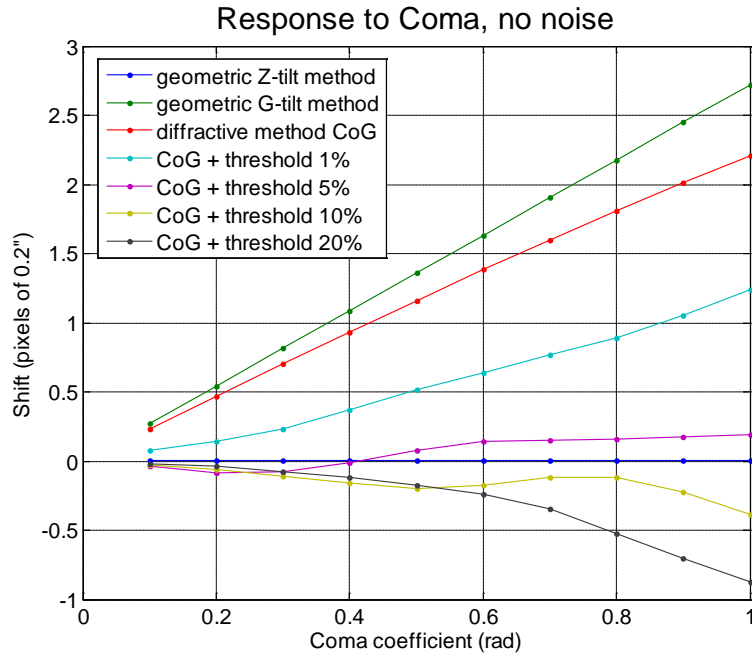


Figure 10: Displacement of the image centre computed by different methods (without noise) as a function of the coma aberration.

6.3.4.3 Sensitivity to photon noise

Then I have added photon noise and looked at the centroid measurement by CoG (no threshold), averaged over 200 realizations of the photon noise. Figure 11 shows that the response to a Coma decreases with the flux. Note that the lower flux studied is 300 photons while the AOF expects only 80 photons per sub-aperture, because we wanted to see the trend at high flux. Also the AOF pixels are much larger, but here we wanted to study the effect of noise only and not clipping or sampling. The AOF case is studied in §6.3.4.5.

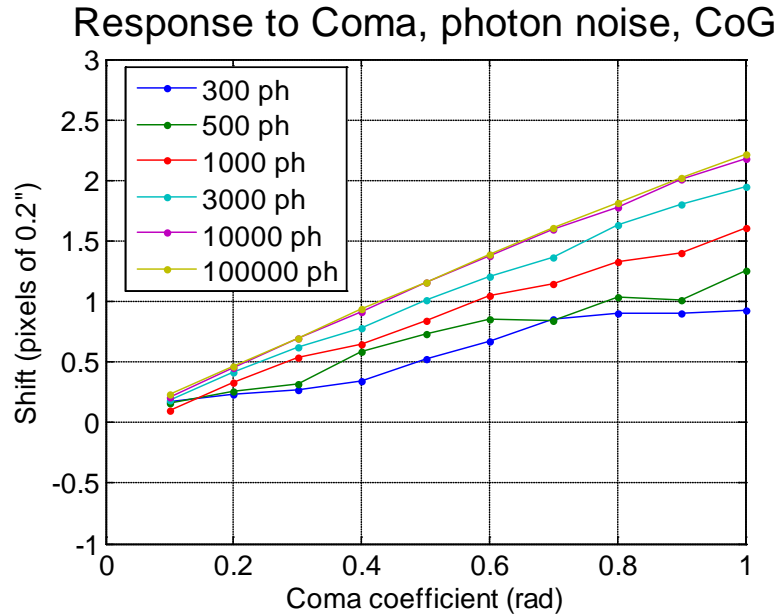


Figure 11: Displacement of the image centre computed by CoG (diffractive model) and no thresholding, at different flux level (with photon noise) as a function of the coma aberration.

6.3.4.4 Sensitivity to combined photon noise and thresholding

The next logical step is to continue including photon noise and study the response of a diffractive model with CoG and thresholding (set to 20 % of the maximum sub-aperture flux as in RD1), averaged over 200 realization of the photon noise. The result on Figure 12 shows that using thresholding in addition to CoG changes the response, mainly inverting it.

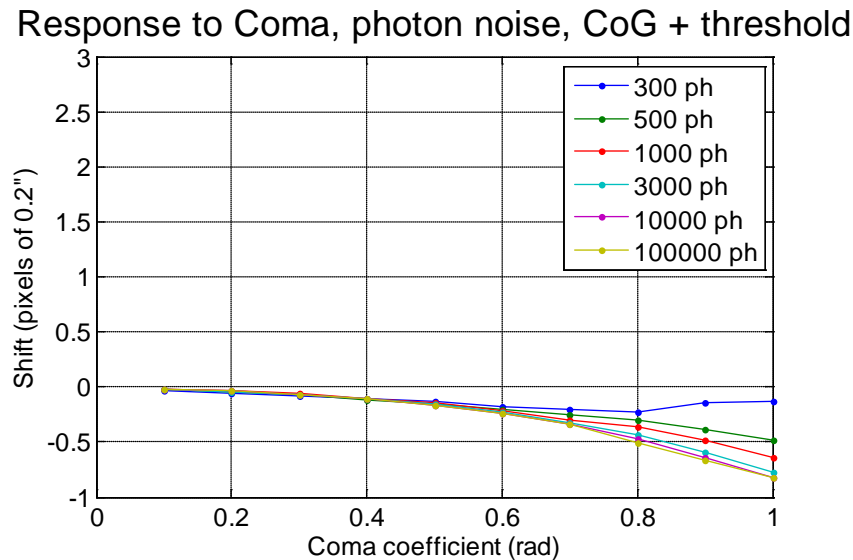


Figure 12: Displacement of the image centre computed by CoG (diffractive model) and thresholding, at different flux level (with photon noise) as a function of the coma aberration.

6.3.4.5 AOF case

Finally I wanted to simulate the response of one sub-aperture of the AOF WFS. Thus the parameters were set to diameter = 0.2 m, 6x6 pixels of 0.83 arcsec, source size = 1.3", 80 photons per sub-aperture and per frame, excess noise of 1.4 (see §6.7.1.2 for definition) and Read-Out Noise of 0.5 e-/px/frame. The response displayed on Figure 13 shows the usual difference between Z-tilt and G-tilt, and also that the AOF sub-apertures' response to a Coma saturates quite rapidly due to their small FoV. It is difficult to conclude on the response in presence of noise (averaged over 1000 realizations), as it seems to be between the no noise case and the Z-tilt method.

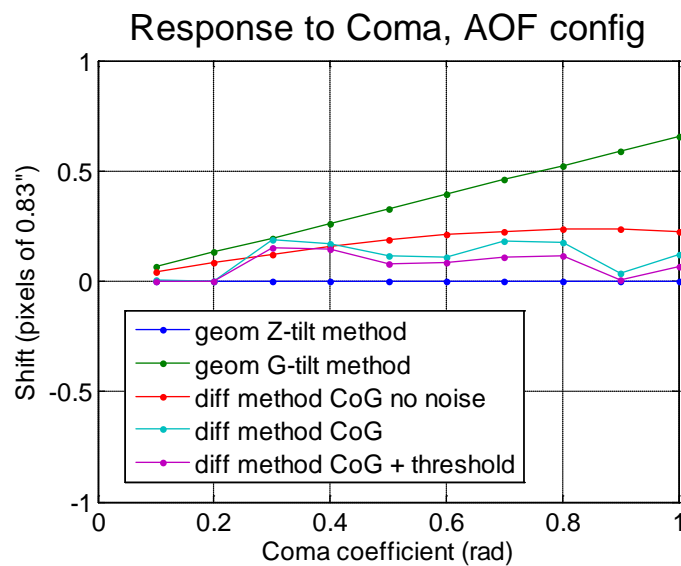


Figure 13: Displacement of the image centre computed by different methods in the AOF configuration: geometric (geom) SHWFS model with Z- or G-tilt, and diffractive (diff) model with CoG and no noise, noise or noise + threshold.

At this point, it is interesting to know the following simulation results: a seeing of 1 arcsec produces 0.14 radian of coma rms (i.e. <0.42 radian 99.7% of the time) in the sub-apertures of the AOF in Open Loop. It will be the same in Closed Loop as those spatial frequencies are not corrected by the AO. And during calibration (I have tried both zonal and modal), the DM displacements to produce an IM with enough signal (2 pixels PTV) produce 0.1 radian of coma rms.

6.3.4.6 Conclusion

For computing efficiency, one would like to use a geometric model of SHWFS to generate the Pseudo-Synthetic Interaction Matrices of the AOF. To the question "Should one use the G-tilt method or the approximate Z-tilt one?", preliminary answers are:

- An ideal SHWFS using a CoG algorithm measures Angles of Arrival (G-tilt), so if one wants to simulate an Interaction Matrix recorded in the best of cases, the G-tilt model would be better

	<p style="text-align: center;">Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue: 1 Date: 03.12.2012 Page: 24 of 74</p>
--	---	--

- In operating conditions, the AOF SHWFS will use weighted CoG on noisy images, with finite sampling and FoV. The SHWFS response is then far from the ideal one. On the example of Coma the response is actually closer to a Z-tilt than G-tilt. But this is an extreme case, and in operating conditions, the spot will not be as asymmetric. A fair conclusion would be to say that the combination of real conditions (sampling, noise, clipping) bring the SHWFS response closer to a Z-tilt than a G-tilt.
- The computation of the formally incorrect Z-tilt didn't turn out to be much faster than the G-tilt one, only 10% (see §7.7).
- The recommendation would then be to rather use the method of Angle of Arrivals (G-tilt), formally correct in the ideal case, and to possibly calibrate it for deviations to the ideal case (especially the effect of clipping that biases the measurement) if one wants to get an absolute measurement of the WF. This might not be necessary in Closed Loop operations, but might be when one wishes to estimate absolute turbulence parameters from Closed Loop data, or when the PSIM is to be compared with a measured one.

6.3.5 Photon noise in geometric model

Finally, I have implemented in the code the possibility to simulate the effect of photon noise, although no image is ever computed. To this aim, I add to the X and Y slopes a random noise with a Gaussian distribution and an amplitude computed using the following approximation:

$$\sigma_{noise}^2 \approx \frac{FWHM^2}{8\ln(2) \times N_{ph} \times \alpha^2} \quad (6)$$

where σ_{noise} is the measurement noise level (on the slopes in a sub-aperture, in pixels), $FWHM$ is the full-width at half-maximum of the spot in the sub-aperture (in arcsec), N_{ph} the number of detected photons per sub-aperture per frame and α the pixel scale (in arcsec/pixel).

In the case of the AOF for example, $FWHM$ is expected to be around 1.0 arcsec and $N_{ph} = 80$ photons, which leads to $\sigma_{noise} = 0.06$ pixel (considering a pixel scale of 0.83 arcsec/pixel). When $FWHM = 1.3$ arcsec and we assume the CCD220 excess noise is equivalent to halving the QE ($N_{ph} = 40$ photons), then $\sigma_{noise} = 0.11$ pixel.

6.3.6 Other application: seeing estimation from NAOS real-time data

The SHWFS of an AO system can be used to estimate the atmospheric seeing. This is simply done by measuring on a guide star the WFS slopes through turbulence for about 30 seconds, projecting those slopes on Zernike modes, taking their variance over time and fitting the best r_0 and L_0 parameters of the Von Kármán spectrum. The projection of slopes to Zernike modes requires a model of the WFS such as the one developed for the AOF PSIM generation: 100 Zernike modes are applied to the WFS model and their response recorded. We have then a modal interaction matrix of the system (Zernike modes to slopes, see Figure 14). This matrix is inverted to obtain the projection matrix (slopes to Zernike modes, piston filtered).

Such data have been recorded using the NAOS AO system at the VLT. 4096 consecutive vectors of slopes were recorded at a loop frequency of 240 Hz, and the procedure described above was used to retrieve a seeing value. Preliminary results of this study (several measurement series) are shown on the Figure 15 and compared with the seeing values provided by the DIMM. We see a rough agreement. NB1: the same study can be performed using closed loop data; in that case the Deformable Mirror voltages have to be recorded as well, and projected into the WFS space thanks to the measured IM of the system before being converted to Zernike coefficients. The advantage of this method is that the measured slopes are smaller and thus their measurement is more linear.

NB2: Another technique would consist in projecting all voltages and slopes into the DM space and use its influence functions to compute the Zernike decomposition. Unfortunately the Influence Functions, even when measured once precisely, can be variable with time, especially if the DM undergoes transport, maintenance, changes of temperature, or simply the passing of time. Modelling the WFS seems to be a more stable approach.

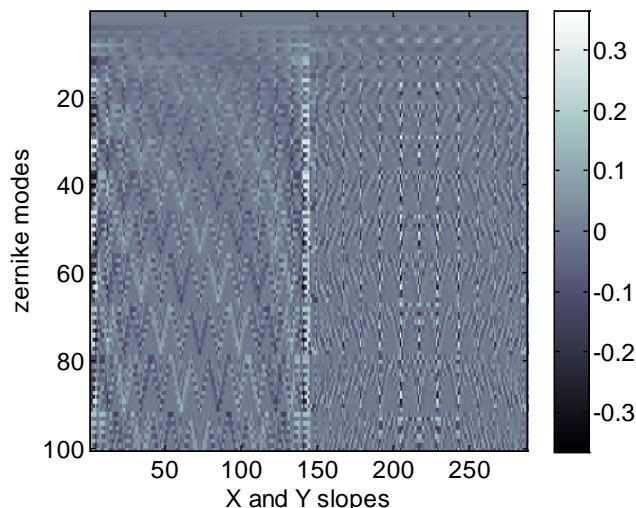


Figure 14: Matrice of Zernike modes to slopes on the example of NAOS.

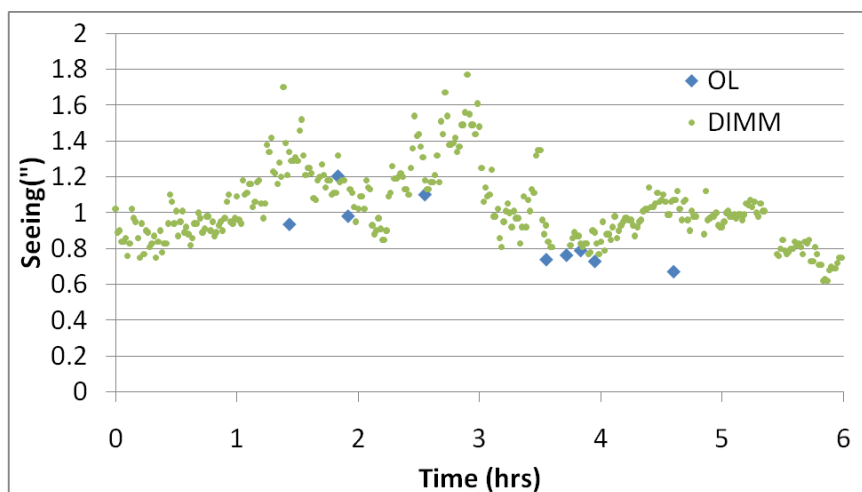


Figure 15: Seeing estimation from NAOS real-time data (Open Loop as blue dots) compared to the DIMM site monitor values (green dots).

6.4 Sub-apertures definition (diffractive model)

The routine “SHWFS_PSIM_Mainloop” slices the WF into sub-apertures, and calls for sub-routines to compute the individual PSFs, assemble them to reconstruct the full image seen by the WFS detector, and measure the slopes. This construction is equivalent to assuming that the practical WFS is equipped with a perfect field stop having a square FoV equal to the FoV of a sub-aperture.

That is to say there is no optical cross-talk between the sub-apertures, except when detector PSF (charge diffusion) is used (electrical cross-talk then, see §6.7.1.2).

The Figure 16 is extracted from RD11 and shows the impact of a field stop on a sub-aperture’s response. The data points were measured on MAD before its WFSs were equipped with field stops. The simulation curves (from a first version of the PSIM SHWFS code written in 2004) show that:

- the linearity on the response is poor when a diffraction limited source is used,
- the linearity is better with an extended source (same as shown in §6.9.1)
- In both cases there is cross-talk between the sub-apertures (i.e. a field stop like in the PSIM code) and we notice a saturation of the signal for large tilt values (spot truncation), before it starts to decrease again (spot outside of the sub-aperture’s FoV)
- When the field stop is not used the signal values saturate earlier, the following decrease is much sharper (spot of the adjacent sub-aperture entering the sub-aperture of interest) and the overall curve shape well matches the measurements.

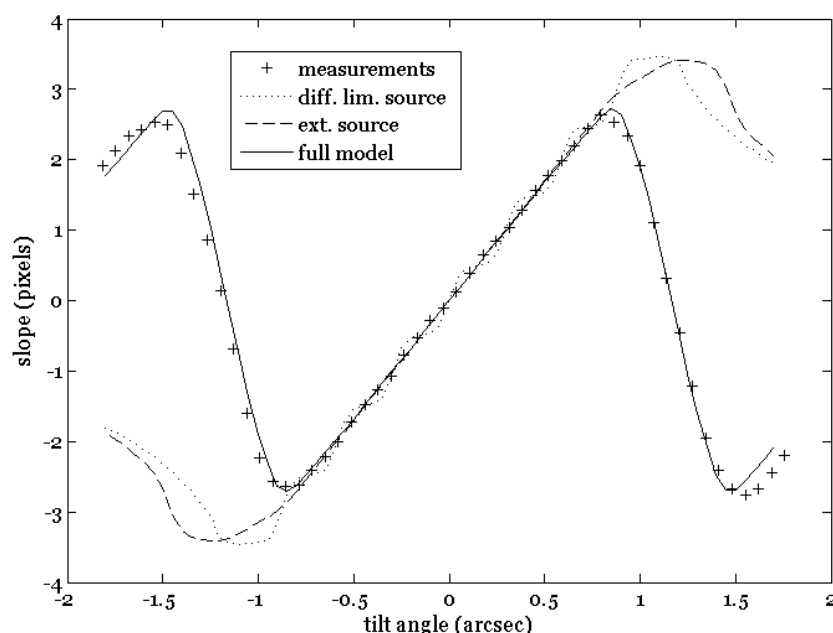


Figure 16: Comparison of the MAD SHWFS measurements with no field stop installed (crosses) with different models: diffraction limited source with field stop (dotted line), extended source with field stop (dashed line), full model taking into account the first neighbouring sub-apertures, i.e. no field stop (solid line).

In summary, we see that the response of a SHWFS without field stop could be well simulated, and that the strong cross-talk between sub-apertures can be avoided by the use of a field stop. The linearity range is then larger, and the saturation is due only to spot truncation at the edge of the FoV.

6.5 Wavelength

The SHWFS simulation is monochromatic, at the selected wavelength, but in a real system (NGS) the WFS is sensitive to a whole spectrum of wavelengths (combination of source spectrum + instrument throughput + detector QE). Only in the case of a LGS is the monochromatic assumption correct.

This assumption has two consequences on the polychromatic spot:

- The definition of the spot width as λ/D doesn't have a meaning
- The diffraction pattern (Airy rings) will be blurred

A simple simulation has been performed to see how the spot is affected. It used as chromatic response the predicted Quantum Efficiency curve from the CCD220 detectors to be used in the AOF SHWFS (Figure 17), i.e. doesn't consider the source spectrum or instrument throughput. The mean wavelength of this curve, weighted by the QE, is 673 nm.

On Figure 18 are shown the monochromatic PSF at 673 nm and the polychromatic PSF. We see that their FWHM are identical, but as expected the Airy rings are blurred in the second one.

Note that those PSFs are not seen in such detail at the focal plane of a WFS, but are sampled by the detector pixels. This is illustrated on Figure 19 with the example of the AOF where the FWHM of the diffraction spot is about 0.7" and the pixel angular size is 0.83" (Visible WFS of the Maintenance and Commissioning Mode). We see on the difference of the PSFs that the relative error can reach 30% but only in pixels with low flux. In the ones that matter for centroiding, it is smaller than 5%.

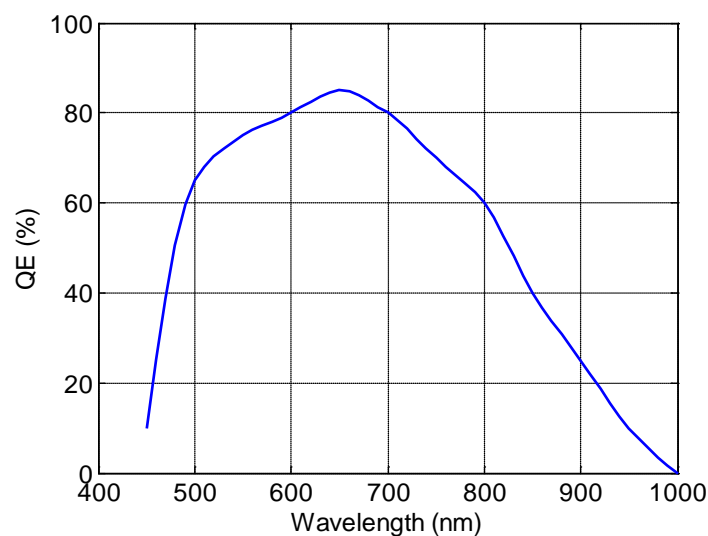


Figure 17: QE curve of the CCD220.

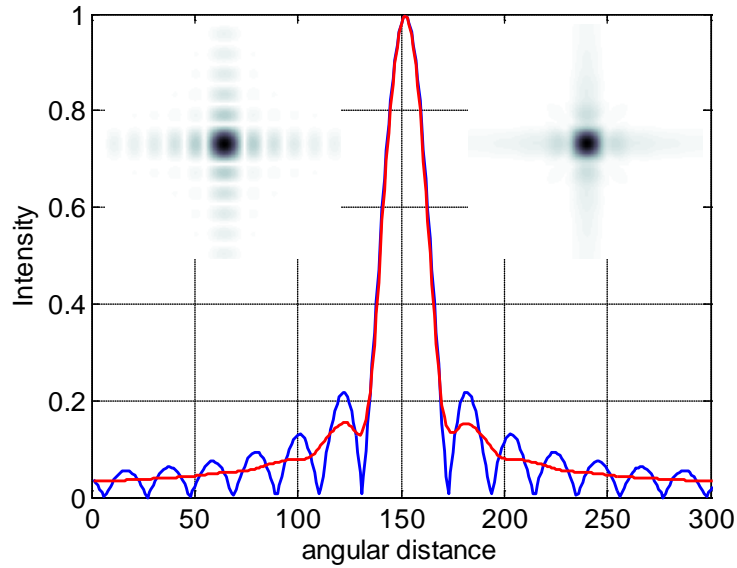


Figure 18: Superimposition of monochromatic (top-left, cut in blue) and polychromatic (top-right, cut in red) PSFs.

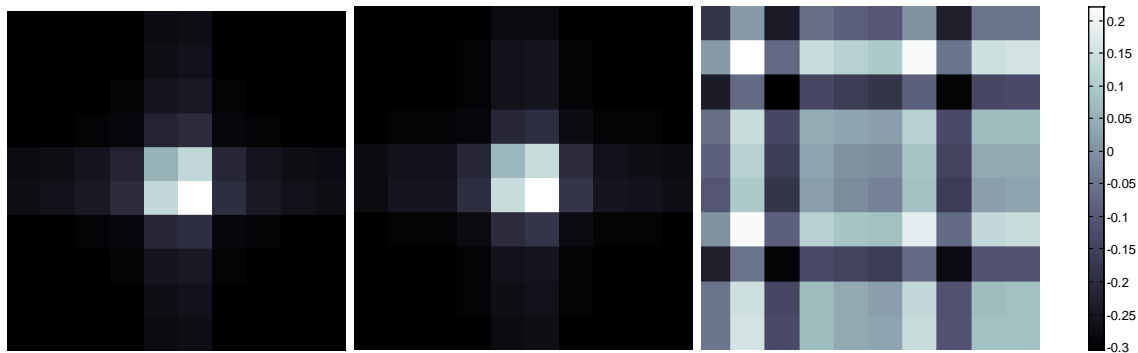


Figure 19: Left: Monochromatic PSF after detector sampling. Centre: Polychromatic PSF. Right (with scale): relative difference.

6.6 PSF computation (diffractive model)

6.6.1 Description of the code

The routine “makepsf_PSIM” allows generating (by FFT) a PSF from any input WF (and pupil mask) at the desired wavelength, given the pupil physical diameter, number of pixels, pixel scale and source size. The two properties of the image to reproduce are:

- **Spot size**
- **Response to a tilt**

An error message is generated if the required FoV cannot be simulated from the input WF sampling (number of pixels). The proper oversampling of the PSF is obtained by the proper zero padding of the image before computing the FFT. The image can then be convolved by the image of an extended source.

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 29 of 74
--	--	--

Because the zero padding is a discrete process, it is generally not possible to simulate the right **spot size** and **response to a tilt** for a given combination of pixel scale and wavelength. Indeed when generating a PSF by FFT, the sampling of this PSF (number of pixels per PSF diameter) is equal to ratio between the FFT image size (after zero padding) and the input image: an image of $N \times N$ pixels has to be zero-padded to $2N \times 2N$ pixels so that its FFT gives a Nyquist sampled image (2 pixels per PSF diameter).

To obtain any PSF sampling, one should in theory produce an infinitely well sampled PSF, and bin it to the desired sampling. It cannot be done in practice so one has to find the closest practical sampling. In the case of the AOF (6 WF pixels per sub-aperture, 589 nm sensing wavelength, 20 cm diameter aperture, 6 CCD pixels of 0.83" per sub-aperture), the minimum number of WF pixels required per sub-aperture is 17×17 , and the diffraction spot is 0.6074" wide. The closest match achievable (with a zero-padding to 25×25 pixels) is a diffraction spot of 0.6104" (0.5% larger) equivalent to the one at a wavelength of 592 nm.

As the spot size effect is very small in the WF sensing process, such deviation is acceptable. However the WF has to be scaled by the ratio of the wavelengths so that the response to a tilt is correct (0.83" tilt moves the spot of one pixel).

Finally, the PSF can be computed on a larger number of pixels (minimum factor to be inputted) before begin re-sampled by the pixels (as it happens in reality: an area of the image is averaged over a detector pixel, and is not only sampled on its central point). This is actually recommended if one wants to simulate properly the detector PSF (§6.7.1.2).

Note that when producing an image by FFT, it is centred on one pixel (point of zero frequency), so if one wants to have it centred at the crossing of 4 pixels (like it usually is the case in a SHWFS), a calibrated Tip-Tilt has to be added to the input WF. In the case of the PSIM code, this is done in the "SHWFS_PSIM" routine.

6.6.2 Other application: long-exposure PSF

The routine "makepsf_PSIM" can be used not only to generate PSFs at the focal plane of a SHWFS, but any PSF. For example we have used it to simulate a "long-exposure" image (at 0.6 μm) at the focal plane of the VLT through a layer of turbulence with known properties (actually 1000 uncorrelated realizations of turbulence): seeing = 1 arcsec, outer scale $L_0 = 22$ m.

From this image, we can measure the FWHM (see Figure 20), and validate the formula used to compute the FWHM of a long-exposure image from the seeing, L_0 and the wavelength (RD2 and RD3).

In this example the formula gives 0.779" and the measurement 0.781", in excellent agreement.

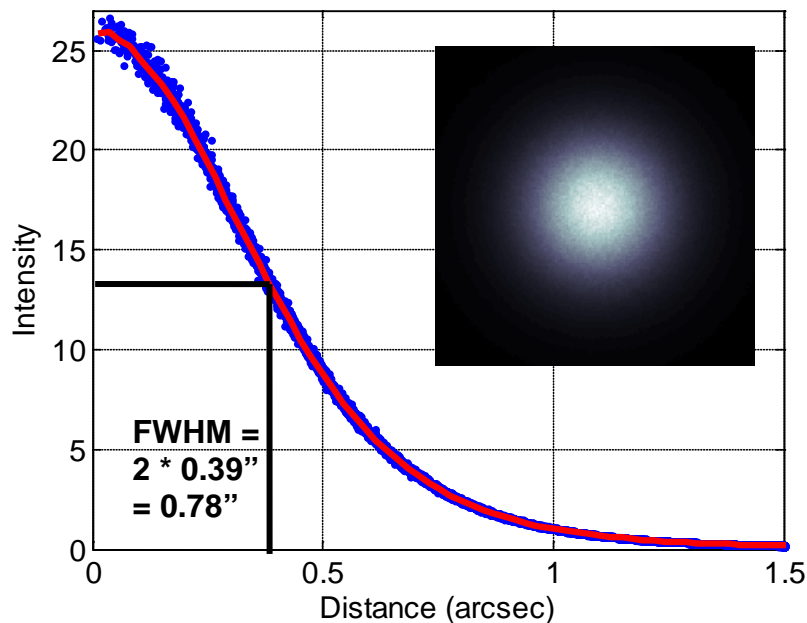


Figure 20: PSF imaging and determination of the FWHM.

6.7 Image sensor simulation (diffractive model)

6.7.1 Description of the code

6.7.1.1 General

The parts of the code described previously brought us from an input WF to the optical image produced by the lenslet array, sampled at the location of the detector pixels (or over-sampled, so that a later binning will be equivalent to signal averaging over the pixel area). Then the routine “ImSensor_PSIM” is in charge of adding photon noise to this optical image and converting it into the signal that will exit the detector read-out system, before or after digitization (i.e. signal in volts or ADUs respectively).

The detector parameters and noise sources taken into account are listed in the §6.1. They contribute to the modification of the image in the following order and manner:

- The input number of photons is multiplied by the QE
- Photon noise is added following Poisson statistics
- The detector PSF (charge diffusion) is added by convolution of the over-sampled image with a Gaussian of FWHM equal to the PSF value (see also §6.7.1.2)
- The over-sampled image is binned to match the correct pixel size
- Dark current is added following Poisson statistics
- A possible saturation is implemented (ceiling)
- Excess noise is added (see §6.7.1.3)
- The number of electrons are converted to Voltages (conversion gain in micro-Volts per electron)

- Pixel Response Non-Uniformity (PRNU) is added by multiplication by a map of responsivity values around 1
- Non-Linearity could be added as a polynomial function (not implemented yet)
- Bias is added (constant offset value over the whole detector)
- Column bias is added (offset value per column, added from an input map)
- Read-Out Noise (RON) is added following Gaussian statistics
- The image can be multiplied by a map of defective pixels (their value is set to zero)
- If the output is required in ADUs, the image is digitized (Conversion Factor CF in ADUs/Volts)

Applied to the case of the AOF (80 photo-electrons detected per sub-aperture, 1ms exposure time, PSF = 0.6 pixel, excess noise factor = 1.4, RON = 0.5 e⁻/pixel/frame), the degradation is illustrated on Figure 21.

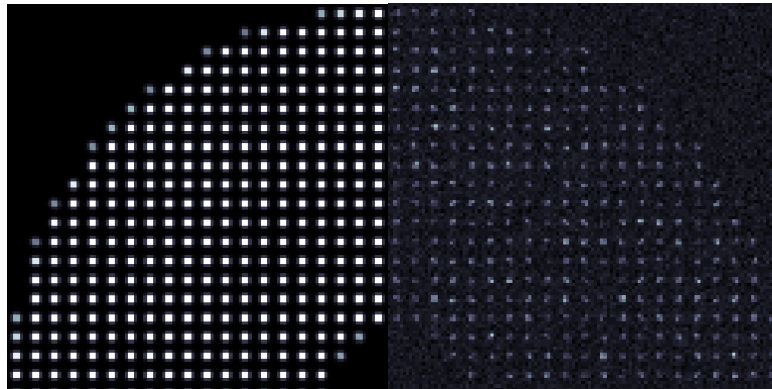


Figure 21: AOF case: perfect image (left) vs. image after detector read-out (right).

6.7.1.2 Detector PSF

The diffusion of charges in the detector material before they are allocated to one pixel is observable as an artificial enlargement of the image PSF; it is thus called Detector PSF. In practice it is measured by scanning a small light spot into a small sub-window of pixels (Figure 22), the edge of which acts like a virtual knife edge. A Gaussian fit (Figure 23) is performed on the derivative (w.r.t. position) of the measured charge as the spot is scanned into the small sub-window and the FWHM calculated. Figure 24 shows an example of PSF measurement on one of the AOF CCDs.

In the PSIM code the detector PSF is added by convolution of the image with a Gaussian of FWHM equal to the PSF value.

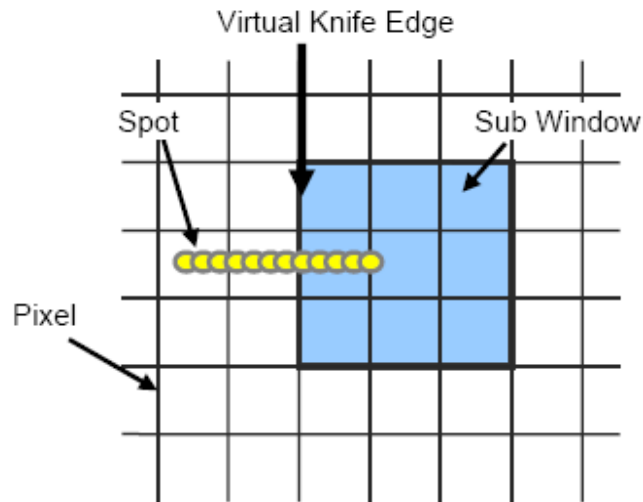


Figure 22: Diagram showing how the spot is stepped into a sub-window to create a sort of virtual knife edge when measuring PSF. The spot has been enlarged a little for illustration purposes only. The spot in reality is much smaller than the pixels (1um versus 24um).

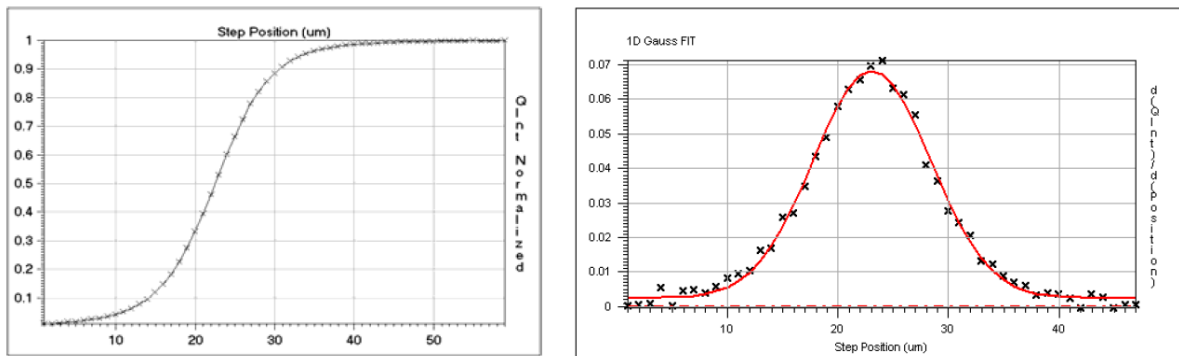


Figure 23: Left: Variation of charge (normalized) in sub-window versus scan position. The total charge in the sub-window is normalized to the total charge read out on the detector. Right: Derivative of charge in sub-window versus scan position. The smooth (red) curve is a Gaussian fit from which the FWHM of the PSF is calculated.

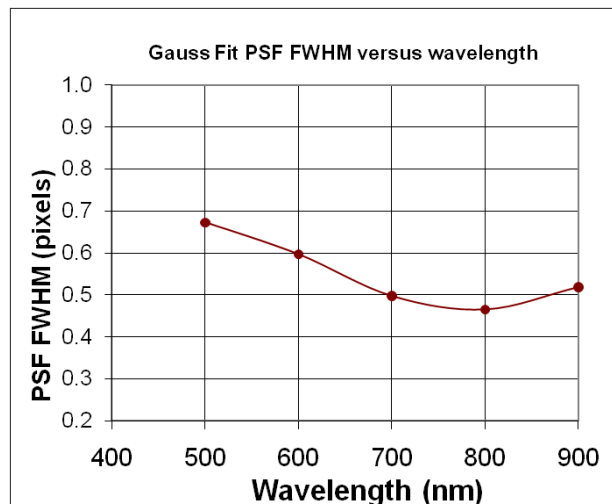


Figure 24: Measured PSF FWHM on one of the CCD220 chips from e2v that will equip the AOF.

6.7.1.3 Excess noise

An Electron-Multiplying CCD (EMCCD) is a charge-coupled device in which a gain register is placed between the shift register and the output amplifier (Figure 25). The gain register is split up into a large number of stages. In each stage the electrons are multiplied by impact ionization in a similar way to an avalanche diode. The gain probability at every stage of the register is small ($P < 2\%$) but as the number of elements is large ($N > 500$), the overall gain can be very high ($g = (1 + P)^N$), with single input electrons giving many thousands of output electrons. Reading a signal from a CCD gives a noise background, typically a few electrons. In an EMCCD this noise is superimposed on many thousands of electrons rather than a single electron; the devices thus have negligible readout noise.

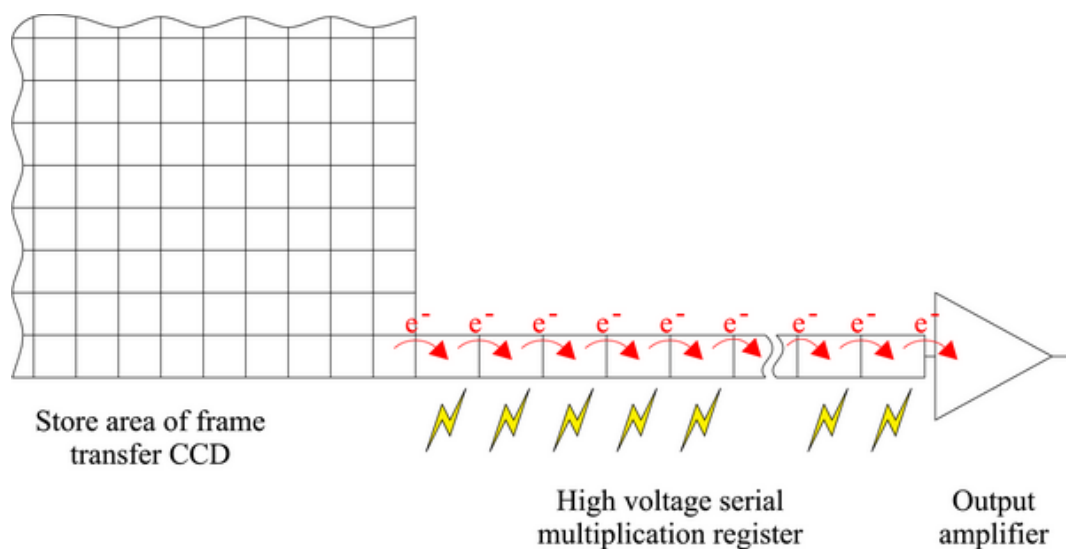


Figure 25: Electrons are transferred serially through the gain stages making up the multiplication register of an EMCCD. The high voltages used in these serial transfers induce the creation of additional charge carriers through impact ionisation.

In an EMCCD, the gain that is applied in the gain register is stochastic and the *exact* gain that has been applied to a pixel's charge is impossible to know. The excess noise factor is defined as the factor by which the multiplication process increases the shot noise on the image when multiplication gain is used.

In the PSIM code any value of excess noise can be added either:

- just before the read-out step, by amplifying the noise with the proper factor (as it happens physically). In that case photon noise has already been added to the image before detection, and we further add Poisson noise to the image divided by (XS^2-1) and multiply back the noisy image by (XS^2-1) , XS being the excess noise (≥ 1),
- or at the same time as the photon noise (saves one Poisson noise computation time). In that case we group photon noise and excess noise by adding Poisson noise once to the image divided by (XS^2) and multiply back the image by (XS^2) .

Indeed an excess noise of $\sqrt{2}=1.414$ has the same effect on the signal-to-noise ratio (SNR) as either halving the quantum efficiency with respect to operation with a gain of unity, or applying twice photon noise to the image.

6.7.1.4 Photon Transfer Curve

The response of any simulated detector can then be analysed using the so-called “Mean-variance method”, or “Photon Transfer Curve”. In this method the detector is lit with a uniform light and frames are recorded for different intensity levels (> 100 levels). Dark frames are also recorded and subtracted to the bright frames, the mean and standard deviation of the resulting images are computed, and reported on the Photon Transfer Curve (see example on Figure 26).

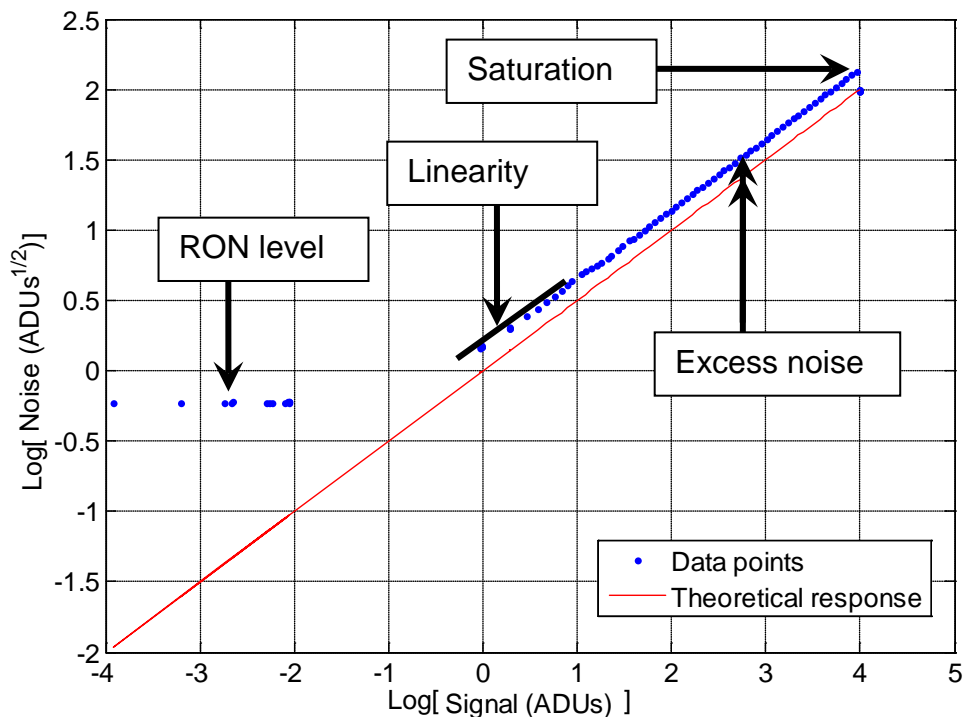


Figure 26: Photon Transfer Curve for the CCD220 simulation.

The slope of the curve should be $\frac{1}{2}$ (Photon noise goes with the square root of the signal). On such a curve one can also see and measure the RON level, saturation level, excess noise, non linearities... The method is described in length in RD4.

6.7.2 Other application: the EELT WFS detector

The routine “ImSensor_PSIM” has been written in such details for being applied to the simulation of the response of the CMOS detectors being developed for the WF sensing on the EELT project.

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 35 of 74
--	--	--

6.8 Slopes Computation (diffractive model)

6.8.1 Description of the code

From the WFS spot images generated by “ImSensor_PSIM”, the routine “Slopes_PSIM” slices the images in sub-apertures, computes for each of them the centroid to be stored and outputted as x and y slopes (sub-routine “Centroid_PSIM”), and provides the displays shown on Figure 3.

The centroiding can be performed using different algorithms (associated with parameters):

- Center of Gravity (CoG): it simply computes the barycentre of the image, which is not very accurate if the image is noisy and the spot off-centre. It can be associated with a thresholding parameter to reduce the sensitivity to noise: the threshold value is removed from the image and then all negative values are set to zero (threshold without pedestal, see Figure 41). Thresholding is studied in more details on the example of MAD in §6.9.1.8.
- Weighted CoG (wCoG): the image is multiplied by a weighting function before the CoG is computed. The associated parameter can be either a custom weighting function (than can be the average of many images), the FWHM of the Gaussian to use as weighting function or the power at which to take the input image itself to be used as weighting function. Note that the weighting affects the measurement of the spot position, which is actually not anymore the CoG of the star image. The wCoG can be used for relative spot position measurement in AO closed loop, but not for absolute WF measurement.
- Correlation (Corr): The correlation coefficient is computed between the image and a reference image which position over the FoV is scanned. The position of the reference image that gives the maximum of correlation is equal to the centroid. The associated parameter is then the FWHM of the Gaussian spot to use as correlation function. Faster than scanning a grid of positions with the moving image, the PSIM code uses a Matlab function of minimization of criterion (“fminsearch”), this criterion (opposite of the correlation of the 2 images) being computed by the sub-routine “Corr_PSIM”.

As illustration, Figure 27 shows the slopes computed by two of those methods (in the case of the AOF SHWFS measuring a turbulent Phase Screen): CoG with threshold at zero, and Corr with as correlation function a Gaussian 3 pixels in FWHM. The slopes measured are very similar and their difference is not correlated to the input phase, but is rather caused by 2 different realizations of noise (photon noise and WFS detector Read-Out Noise).

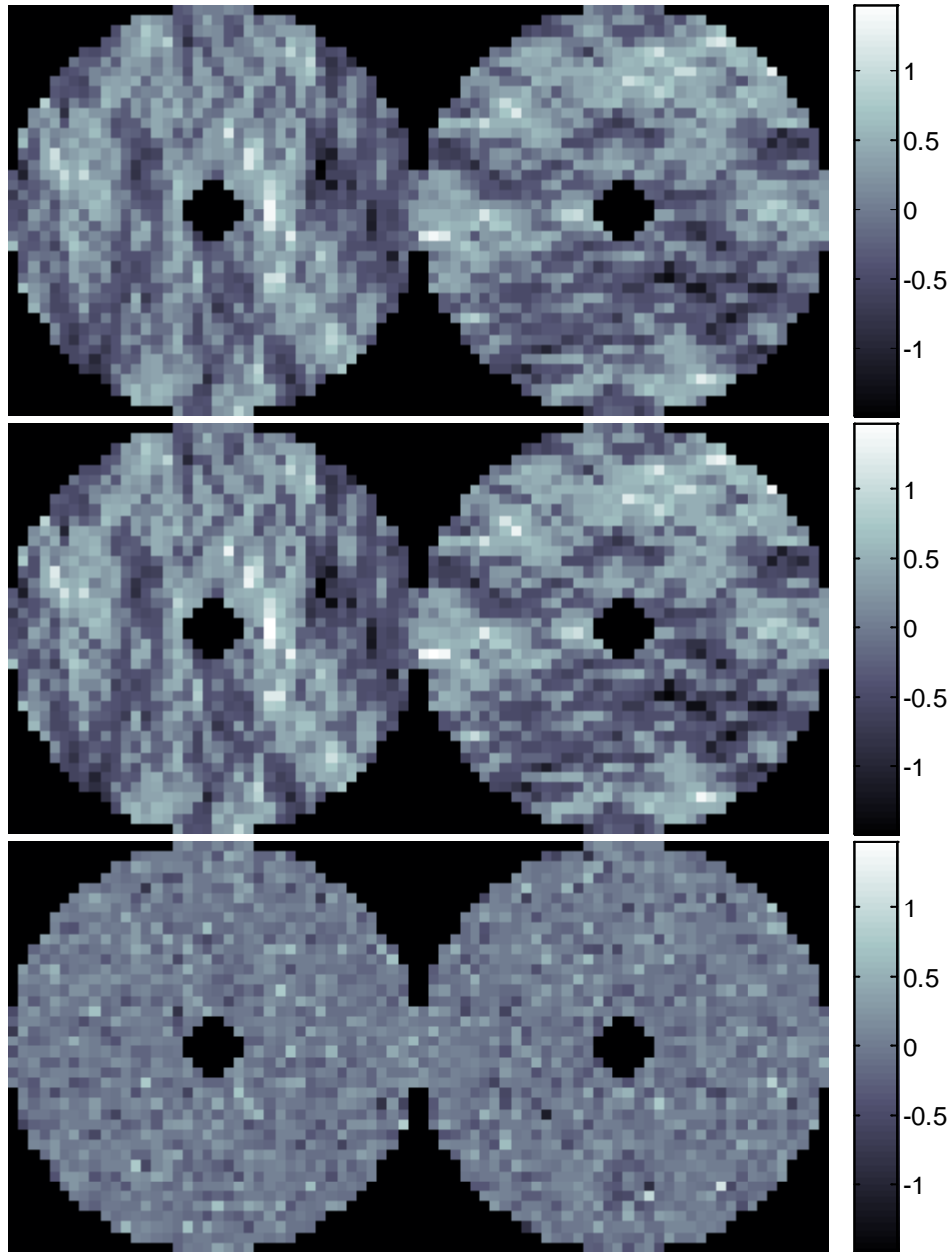


Figure 27: With a turbulent Phase Screen as input, X and Y slopes produced by the SHWFS model with CoG centroiding (top), Correlation (middle) and their difference (bottom).

6.8.2 Response verification

Similarly to what was done in §6.3.2 for the geometric model (with Z-tilt method), we wanted to verify the response of the diffractive model to a flat WF and to a tilt. As in Figure 6, the response to a flat WF is shown on Figure 28 in the case of no noise (infinite flux), and here also the largest slopes are of the order of 10^{-15} pixel (numerical precision).

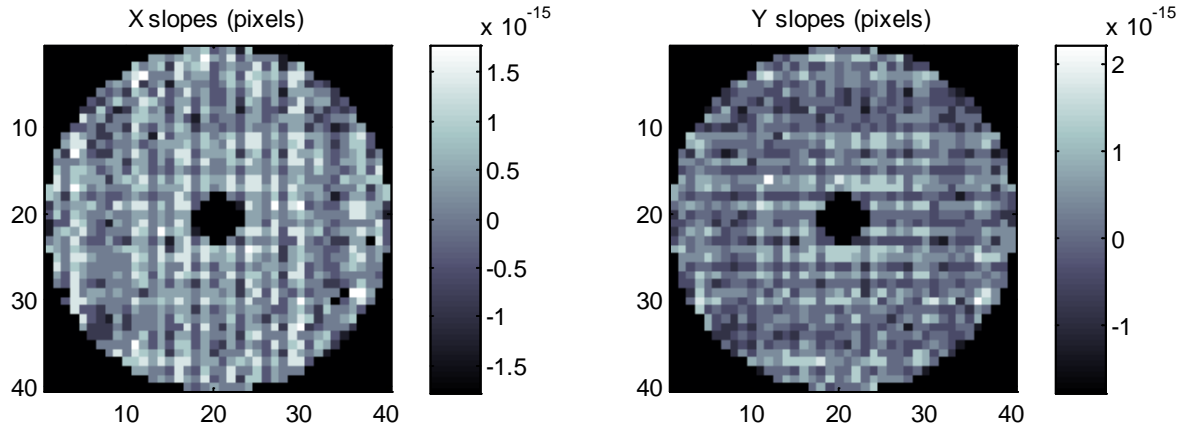


Figure 28: Response of the SHWFS diffractive model (CoG algorithm, no noise) to a flat WF.

As in Figure 7, the response to a pure tilt in X (of a 10^{th} of the pixel scale in amplitude) is shown on Figure 29, top. We can see that the response in X in most sub-apertures (actually the fully-illuminated ones) is close to 0.1, the difference being due to the CoG algorithm used over an image of finite FoV (see §6.9.1 for more details on this effect). As it is not easy to see on the figure, we verified that the response in Y on the same sub-apertures is of the order of 10^{-15} pixel (numerical precision). The striking difference with the geometric model of §6.3.2 is the response of the partially-illuminated sub-apertures: underestimated in the direction of the input tilt (X), and non zero in the orthogonal direction (Y). Indeed in those sub-apertures there seem to be a cross-talk of up to 1.2 % from the X to the Y tilt.

It is known that the PSF shape in the partially-illuminated sub-apertures is distorted, but its CoG doesn't change (proved by the same zero CoG in all sub-apertures of Figure 28). The reason for the cross-talk is different. At first we thought that it could be an effect of the sampling of the detector pixels, so we have done two different tests:

- Increasing the resolution of the focal plane image before it is sampled by the detector
- Increasing the number of pixels per sub-aperture, while keeping constant the total FoV.

All results showed the same cross-talk effect as on Figure 29, top. The next test was to increase the FoV of the sub-apertures by increasing the number of pixels while keeping the pixel scale constant. Figure 29 shows the results for sub-apertures FoV of 5, 10 and 20 arcsec. We see that the cross-talk effect decreases from 1.2 to 0.7 to 0.35 %. The images of the spots (in log scale) on the right side of Figure 29 provide the explanation on the reason for this cross-talk: the PSF in the truncated sub-apertures is indeed deformed and not centro-symmetric, so that a shift in X will truncate its wings in a non-symmetric way, resulting in a non zero slope measurement in the other direction (Y).

The linearity and response of the partially-illuminated sub-apertures has been partially measured on the MAD SHWFSs (RD14), but additional measurements would be required to compare the response of a real WFS with the one of our diffractive model.

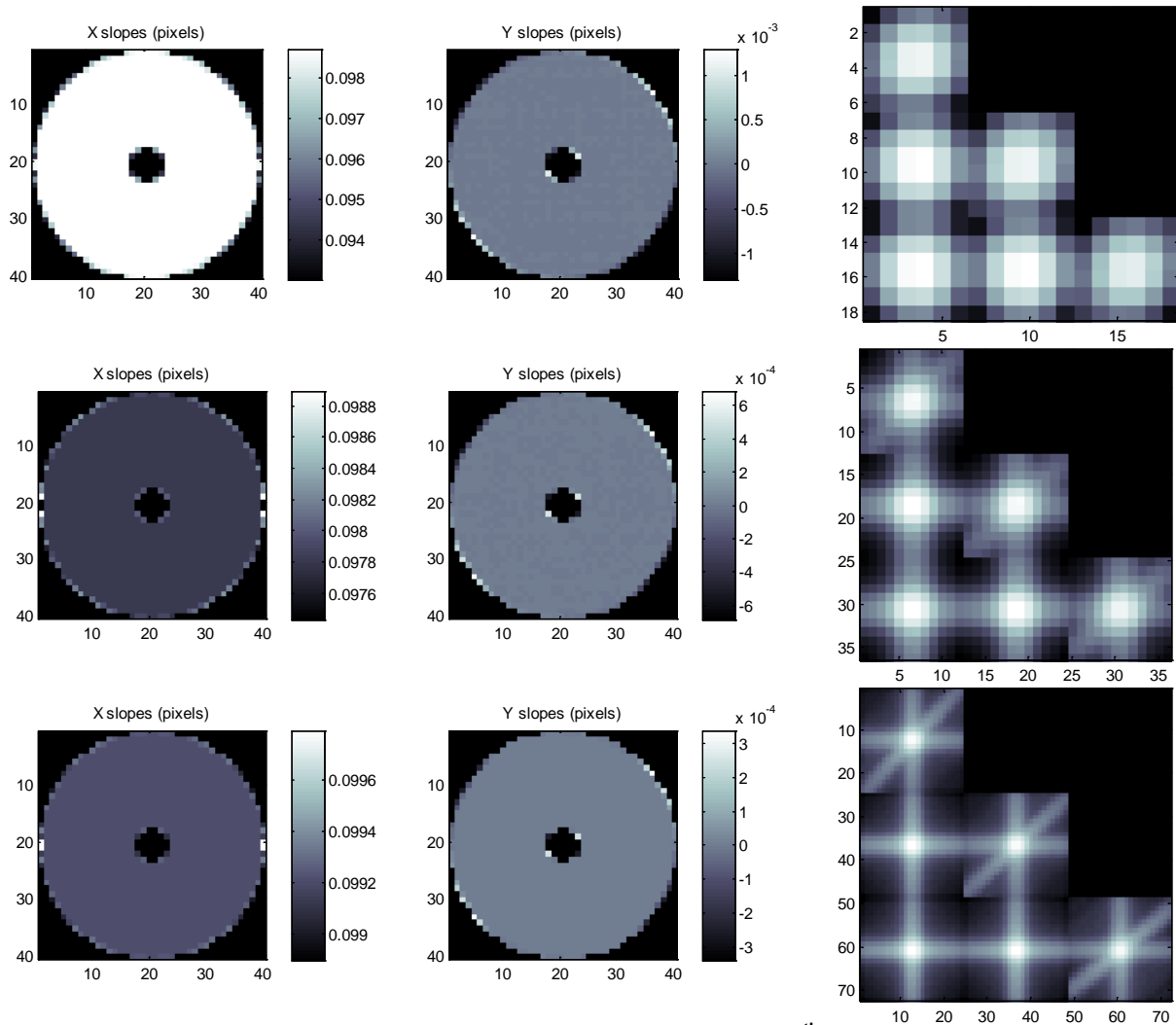


Figure 29: Diffractive WFS response to a perfect tilt in X of a 10^{-4} of the pixel scale (AOF case, no noise). Left to right: X slopes, Y slopes, close-up view of the PSF in some edge sub-apertures (log scale). Top to bottom: 5 arcsec sub-aperture FoV, 10 arcsec, 20 arcsec.

6.8.3 Other application: slopes computation from existing images.

Instead of having the centroiding routine (“Centroid_PSIM”) called by the SHWFS main code (“SHWFS_PSIM_Mainloop”), there is an intermediate routine (“Slopes_PSIM”) that allows computing slopes from WFS spot images not generated by the PSIM SHWFS simulation tool.

This function has been put to use for measuring the x and y slopes on short-exposure images taken with the Active Optics SHWFS at the Cassegrain focus of the VLT UT1 (24x24 sub-apertures of 22x22 pixels) in August 2010. This set of slopes have then been provided to another team at the University of Durham that will process them using the SLODAR reconstruction technique (spatio-temporal correlation of the slopes) in order to retrieve an estimation of the seeing value.

The raw images (Figure 30, top) have to be cropped first (Figure 30, bottom-left), and the sub-apertures selected and re-centred (Figure 30, bottom-right) for a better slopes measurement. This does not disturb the slopes evaluation as we are not

interested in absolute but relative WF measurement in this study. As side result, we have measured a constant rotation of **0.57 degrees** between lenslet array and detector.

In that case, the correlation algorithm was used to measure the centroid in the sub-apertures, in order to limit the effect of the large number of pixels per sub-apertures and their noise.

The slopes measured on the example image of Figure 30 (bottom-right) are displayed on Figure 31.

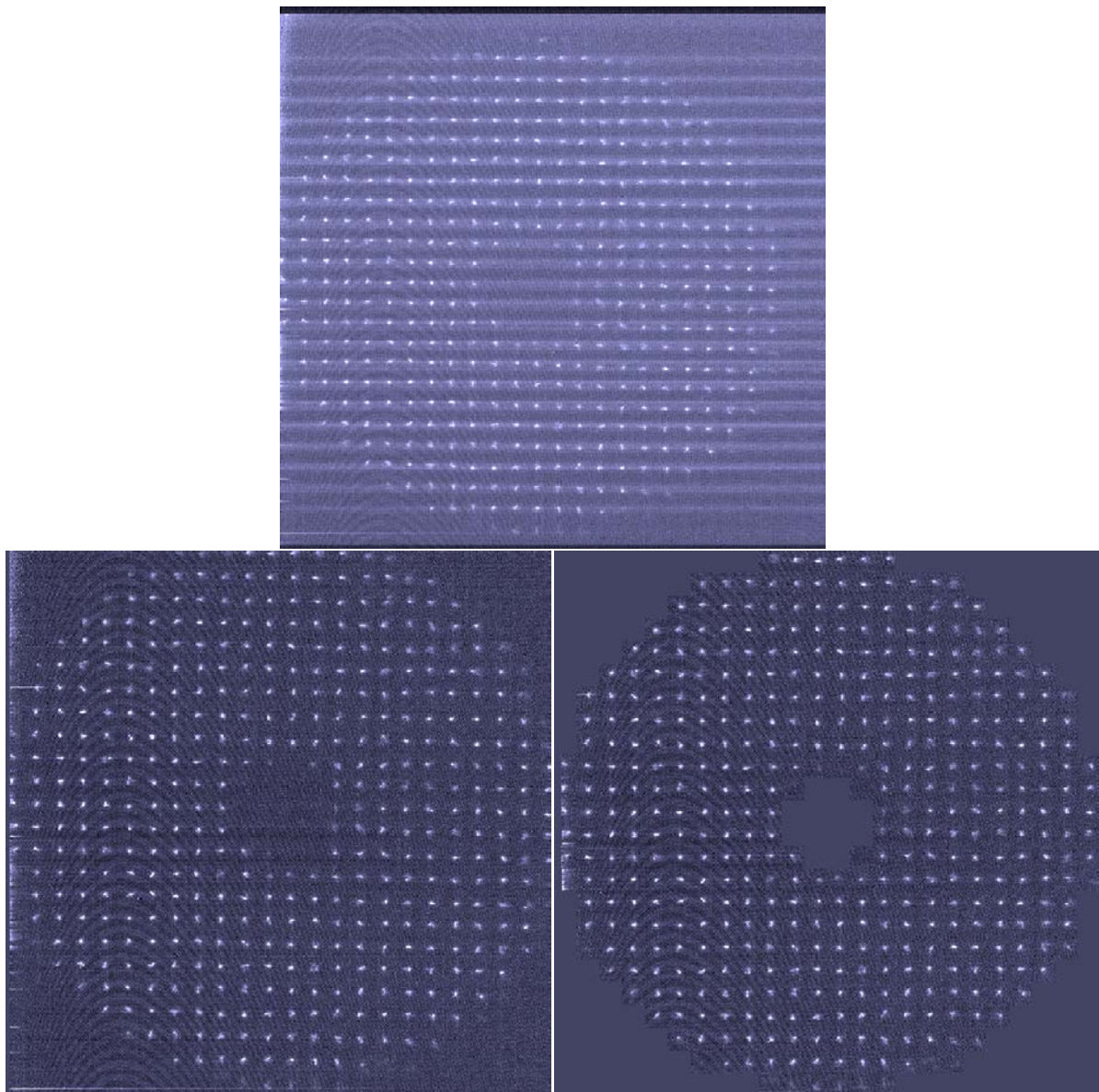


Figure 30: VLT Active Optics SHWFS short-exposure image. Top: raw. Bottom-left: after background subtraction and selection of the image area. Bottom-right: after selection and re-centring of the sub-apertures (includes correction of the rotation between lenslet array and detector).

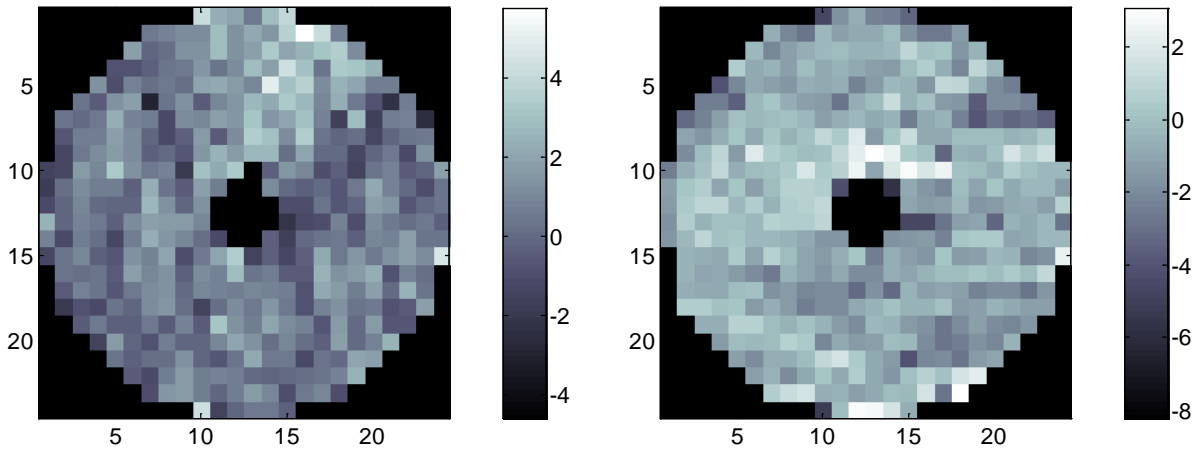


Figure 31: X (left) and Y (right) slopes (in pixels of 0.305 arcsec) measured on the image of Figure 30 (bottom-right).

6.9 Diffractive model

The full diffractive model allows taking into account many characteristics of the WFS that the geometric model cannot, the main interest of this latter model being that it is faster. Also it is already a very good approximation and can emulate photon noise.

In the geometric model, the size in pixels of the input WF doesn't need to be large and is only dictated by the number of sub-apertures; 2 x 2 pixels per sub-apertures are virtually enough to measure the WF. For example in the case of the AOF and its 40x40 WFS, a WF of 80x80 pixels is sufficient, although we prefer to use 240x240 for a better representation of the high orders in the WF.

In the diffractive model, the number of pixels is dictated by the FoV of the sub-apertures and the wish to avoid aliasing, while the pixel scale influences only the zero padding size required for computing the FFT. Again in the case of the AOF, 9 pixels per sub-aperture would be enough to image the 5 arcsec FoV of the sub-apertures, but to avoid aliasing (a spot moving to the left reappears on the right side of the same sub-aperture, when computed by FFT), we choose to use 19 pixels, thus a total of 720x720 pixels (so that aliasing starts to appear only when the spot is at the edge of the sub-aperture, a critical case anyway).

6.9.1 Other application – WFS Linearity

The model can be used to study the linearity in a SHWFS, i.e. the response of a sub-aperture to a tilt. It has been noticed in the MCAO Demonstrator MAD that the quality of the measurement of the linearity and of the pixel scale depends on many factors:

- Source size / pixel scale / diffraction limit of the sub-aperture
- Sub-aperture FoV
- Noise
- Cross-talk between the sub-apertures

Indeed a first method to measure the pixel scale was to apply an increasing tilt of calibrated value (by moving the source), measure the response of the SHWFS (measured slope in one axis) and compute the slope of this response. We will see (§6.9.1.7) that this is not the right way to proceed.

6.9.1.1 MAD WFS

As I have first noticed the kind of effect mentioned above on MAD, I wanted to simulated one sub-aperture of its SHWFS; in practice I simulated 3x3 and looked at the behaviour of the central one (Figure 32). The main characteristics are: 1 meter wide sub-apertures, 8x8 pixels per sub-aperture, 0.35 arcsec per pixel (while the diffraction limit is at about 0.15 arcsec), no noise and CoG algorithm used with threshold at 0.



Figure 32: Simulation of 3x3 sub-apertures of MAD under half a pixel tilt, for a diffraction limited source (left) and a source of 3 pixels in FWHM (right).

6.9.1.2 Source size

At first we want to isolate the effect of the source size, thus we plot linearity curves for source sizes between diffraction limited and 2 pixels of FWHM (Figure 33). We consider a perfect field stop (no cross-talk from one sub-aperture to its neighbours) and no noise (not even photon noise). As expected, we see strong non-linearities for source sizes smaller than 1 pixel in FWHM. At 2 pixels in FWHM, they are completely gone, but the linearity range starts to suffer. We verify that there is a trade-off between linearity and linearity range, and 2 pixels of source size is probably not required to get a good linearity; 1 pixel provides already an excellent linearity and a much better linearity range.

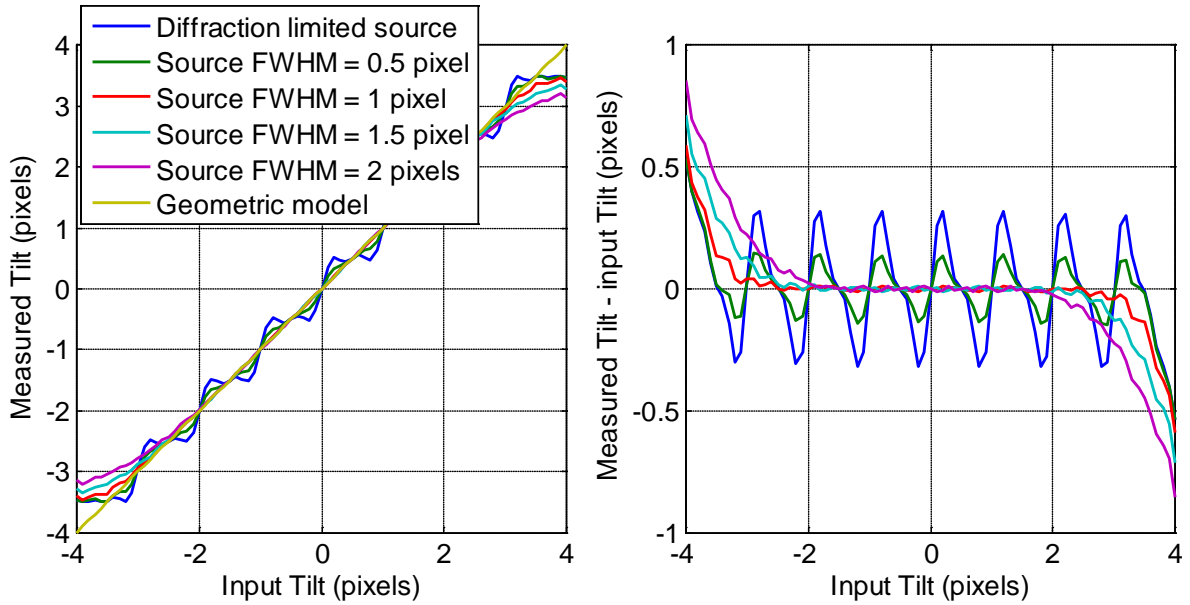


Figure 33: MAD SHWFS linearity curves for different source sizes.

6.9.1.3 Detector PSF

We have then set back the source size to diffraction limited. We still consider a perfect field stop but we add detector PSF (electronic cross-talk between one pixel and its neighbours, see also §6.7.1.2), increasing from 0.5 to 2 pixels.

We see on Figure 34 that the detector PSF has exactly the same effect as an extended source size, except that for large tilts, the cross-talk between sub-apertures enters in the game and produces saturation effects much stronger than in the case of an extended source.

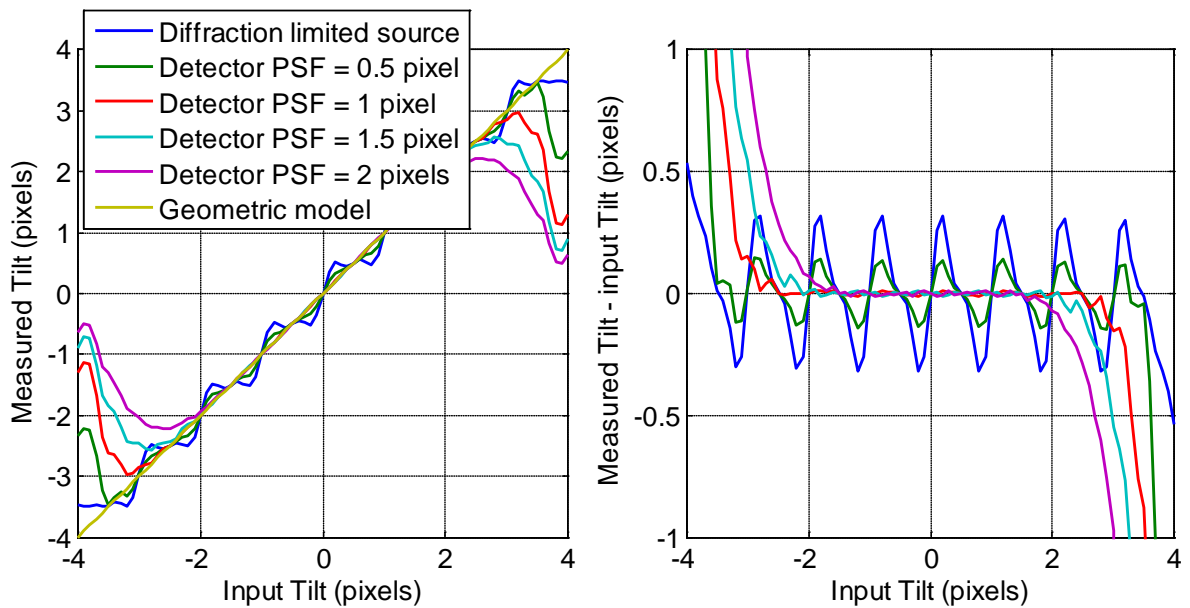


Figure 34: MAD SHWFS linearity curves for different detector PSFs.

6.9.1.4 Photon and Read-Out noises

For the next step we have considered a source of 2 pixels in FWHM and no detector PSF, but we have added photon and Read-Out noises representative of the measurements made on MAD: 8 electrons RON, a signal varying from 500 to 200000 photons per sub-aperture, a QE of 85% and saturation at 10000 electrons per pixel. The MAD WFS properties are the same (CoG algorithm used with threshold at 0). For each case 10 slopes measurements are performed and averaged; this will reduce the noise on the curve but not on the individual slope measurement.

The linearity curves are shown on Figure 35. We see that whatever the flux, there is a bias between the response measured and the expected one. The slopes are underestimated by 35% at 500 photons, 10% at 2000 ph, 5% at 5000 ph. There is little gain in going to 200000 photons, and the detector saturations affect the response (oscillations on Figure 35 right).

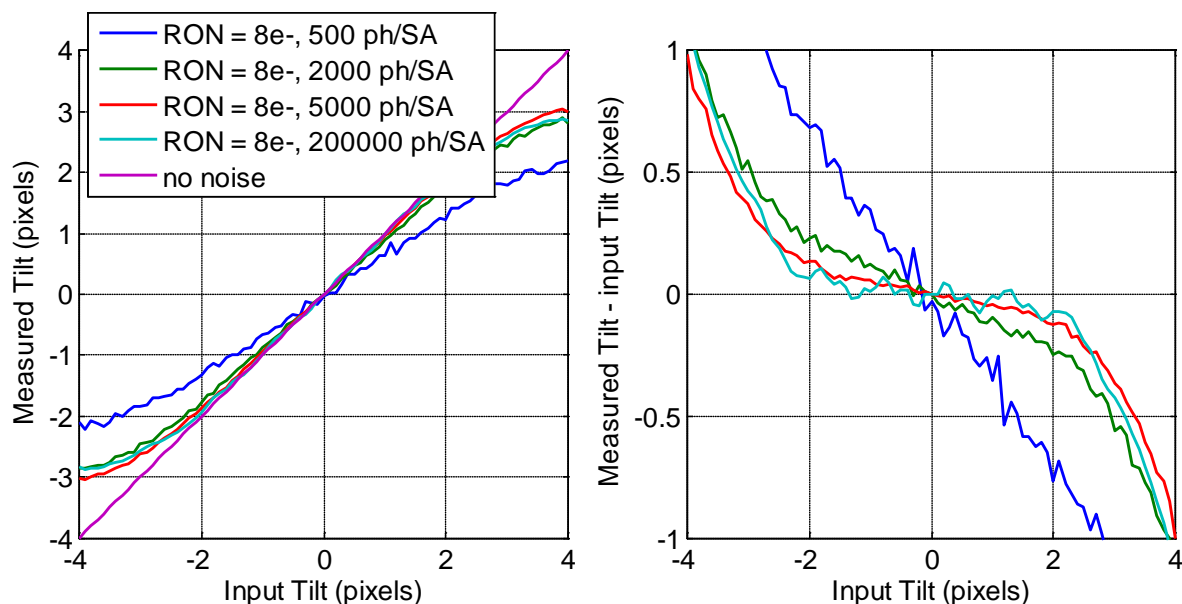


Figure 35: MAD SHWFS linearity curves for different flux level.

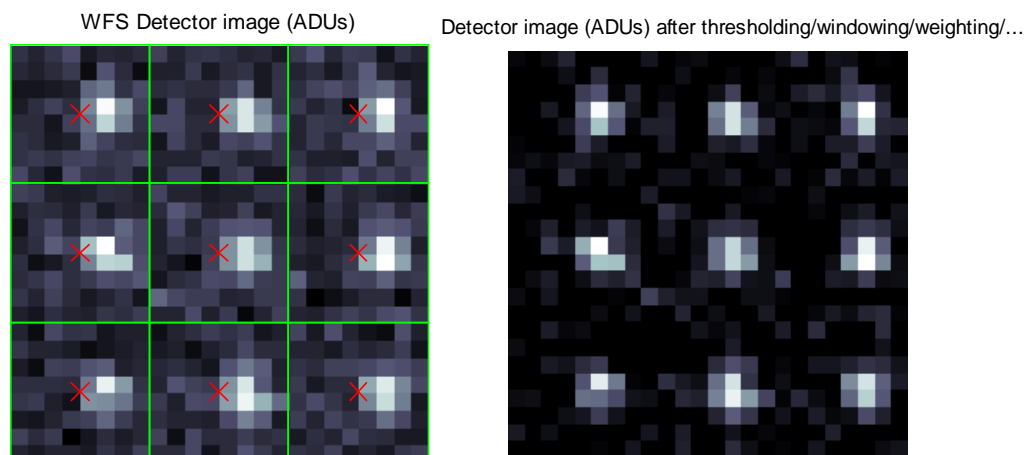


Figure 36: CCD image of MAD SHWFS (3x3 sub-apertures section) before (left) and after (right) thresholding at 0 for CoG computation.

The slopes underestimation is due to the fact that we use a CoG algorithm thresholded at 0 (Figure 36): without threshold the average of RON is equal to 0; after thresholding the average is positive and creates a bias in the CoG towards 0.

6.9.1.5 Measurement improvement on MAD

In order to counter the effects of detector PSF and of the noisy pixels, we have implemented in MAD a square sub-aperture mask (digital) of 6x6 pixels centred on the sub-aperture (windowing). This removes from the CoG computation 28 pixels out of 64, that usually don't contain signal (sub-apertures' FoV over-specified) (Figure 37). This modification led to an average improvement of 5% Strehl Ratio in K band. In addition to this, the best would have been to tune the threshold better, i.e. to about 3 times the RON rms, which could have in theory brought another increase of performance. But rapid tests in the laboratory haven't shown a significant improvement in the case of MAD.

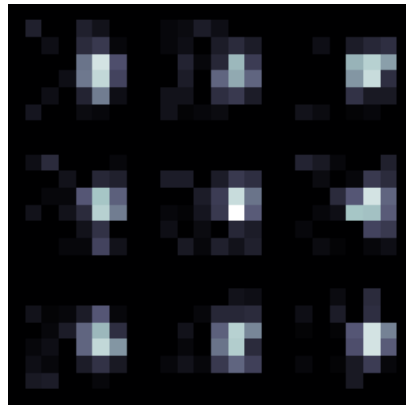


Figure 37: CCD image of MAD SHWFS (3x3 sub-apertures section) after 6x6 pixels windowing and thresholding at 0 for CoG computation.

6.9.1.6 Detector PSF on MAD

With all the MAD SHWFS parameters as described in the previous paragraphs, we are now able to reproduce the measured curve of linearity of the MAD SHWFS (Figure 38) by scanning the only 2 unknown parameters: the detector PSF and the flux at which the measurement was done (if we assume that the plot from Figure 38 is the only data we have, and the raw images that could have given us the flux are not available).

Figure 38 shows a linearity measurement that was done on the MAD bench. The slope of the linearity curve for the fully-illuminated sub-apertures and in the linear range is clearly smaller than the one given by the theoretical pixel scale by about 20%, which means (according to what we found out in 6.9.1.4) that the recording was done with 1000 photons per sub-aperture and per frame.

Moreover, we see on Figure 28 that even though the measurement was done using the diffraction limited fibre, the response does not show steps as pronounced as

could be expected (Figure 33). The reason is of course detector PSF (if we neglect lenslet aberrations and mis-alignment).

Figure 39 shows the simulated response with 1000 photons per sub-aperture and per frame and a detector PSF of 0.5 pixel. It is very similar to the measured one of Figure 38 (see close-up). The main difference is in the linearity range, larger in simulation because it considers a perfect field stop.

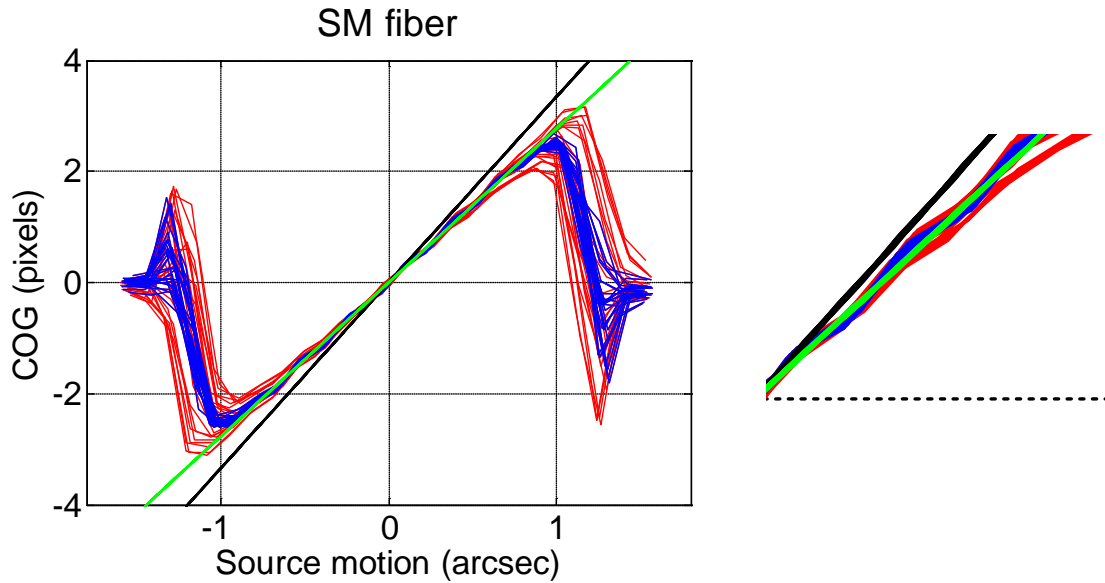


Figure 38: Linearity measurements on MAD with a diffraction limited fibre source. In blue the fully-illuminated sub-apertures, in red the others. In black the slope given by the theoretical pixel scale, in green the slope of the blue curve in the linear range. Right: close-up view.

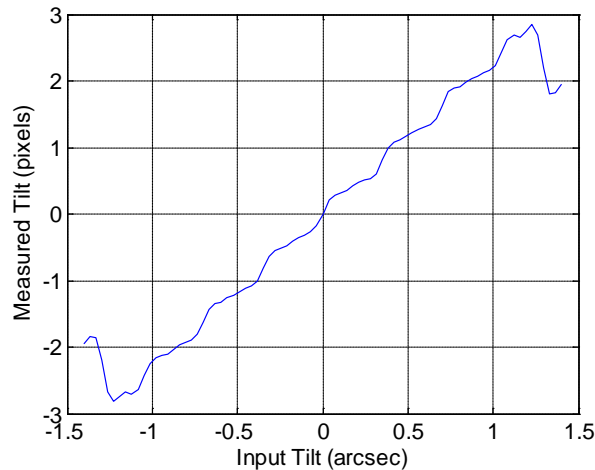


Figure 39: Simulated linearity measurements using the MAD parameters, plus the best matches of detector PSF of 0.5 pixel, and flux level of 1000 photons.

6.9.1.7 Pixel scale on MAD

As introduced in the §6.9.1 and demonstrated through the previous sections, measuring slopes is not the proper way to determine the pixel scale of the SHWFS,

as this measurement is subject to too much noise and too dependent on the WFS geometry and on the centroiding algorithm used.

Instead one should use a method that requires scanning a row of pixels with a spot and recording the flux in those pixels. This method was also used on the HOT bench and is described in RD10.

Figure 40 shows such measurement and its simulation. The waves in the response are caused by the simulation of a monochromatic light source, and disappear when adding the light response at 4 wavelengths (solid line). The curves can easily be fitted with a 2nd order polynomial or a Gaussian, and the distance between the 2 maxima is equal to the pixel scale. The same can be done with the positions where the flux equals in 2 adjacent pixels (same result, more accurate because less sensitive to the flat part of the response at the pixel centre). More couples of pixels can also be used to reduce uncertainty.

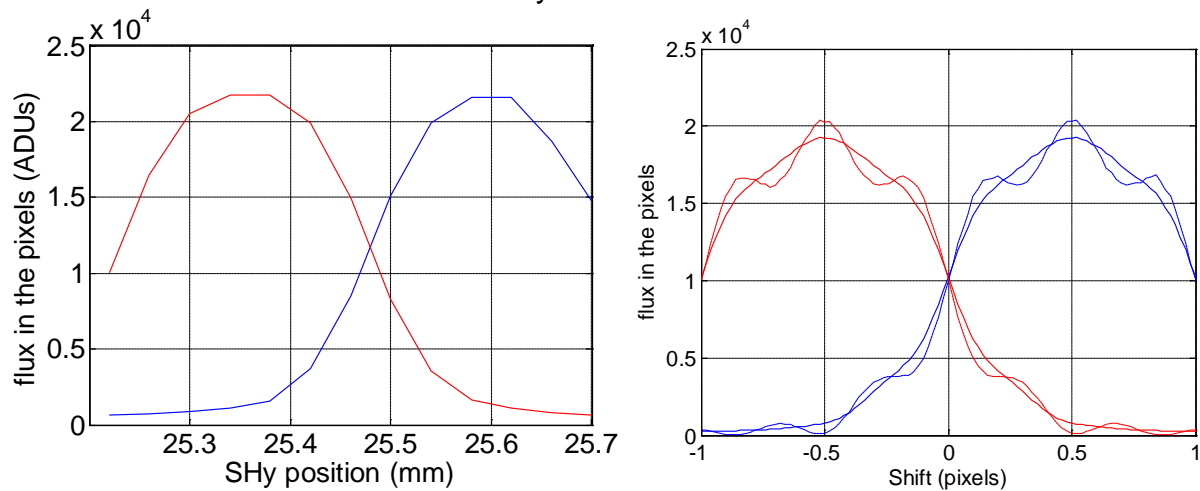


Figure 40: Pixel scale determination on MAD. Left: measurement. Right: simulation; dashed for monochromatic case, solid for equal addition of 400, 500, 600 and 700 nm light.

6.9.1.8 Centroiding method

We have seen in the §6.9.1.4 that CoG with threshold at 0 introduces a bias in the measurements when photon noise and RON are introduced. What is then the best way to measure slopes?

On the example of a MAD sub-aperture, we have compared various centroiding methods:

- Z-tilt with photon noise (on geometric model) for reference
- CoG in the case of no noise
- CoG with noise (MAD parameters: 8 e- RON, 100 photons)
- CoG with noise, threshold at 0
- CoG with noise, threshold at 3σ of the RON (i.e. 24 e-) with pedestal
- CoG with noise, threshold at 3σ of the RON (i.e. 24 e-) without pedestal

The difference between threshold with and without pedestal is illustrated on Figure 41.

The results of the algorithms comparison is shown on Figure 42 after averaging of 40 measurements. The analysis we can make of it is that:

- The geometric model gives a perfect measurement of the input pure tilt, even when photon noise is included
- With no noise, the CoG provides a good measurement of the input tilt, limited only by the FoV of the sub-aperture (truncation)
- In the presence of noise, the CoG measurement is still linear, but over a more limited range, and becomes more noisy
- Adding a threshold at 0 reduces the effect of the noise on the measurement, keeps the linearity range, but changes drastically the slope of the linearity curve (as already noticed in §6.9.1.4)
- Adding a threshold at 3σ of the RON gives the same linearity slope and range as simple CoG while reducing the effect of the noise. It seems to be the best way to measure slopes.

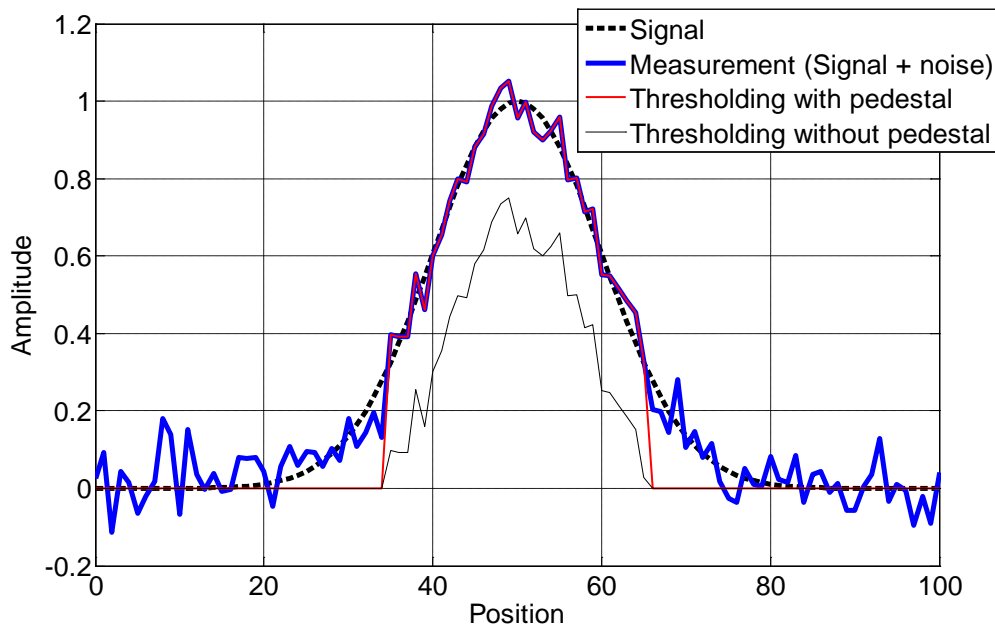


Figure 41: Illustration of the thresholding for CoG estimation. The measurement is the signal perturbed by noise. Pixels below a threshold can be set to zero for better CoG estimation (Thresholding with pedestal, on this example set to 3σ of the noise = 0.3 unit). The threshold value can be subtracted to the measurement before setting the pixels to zero (Thresholding without pedestal).

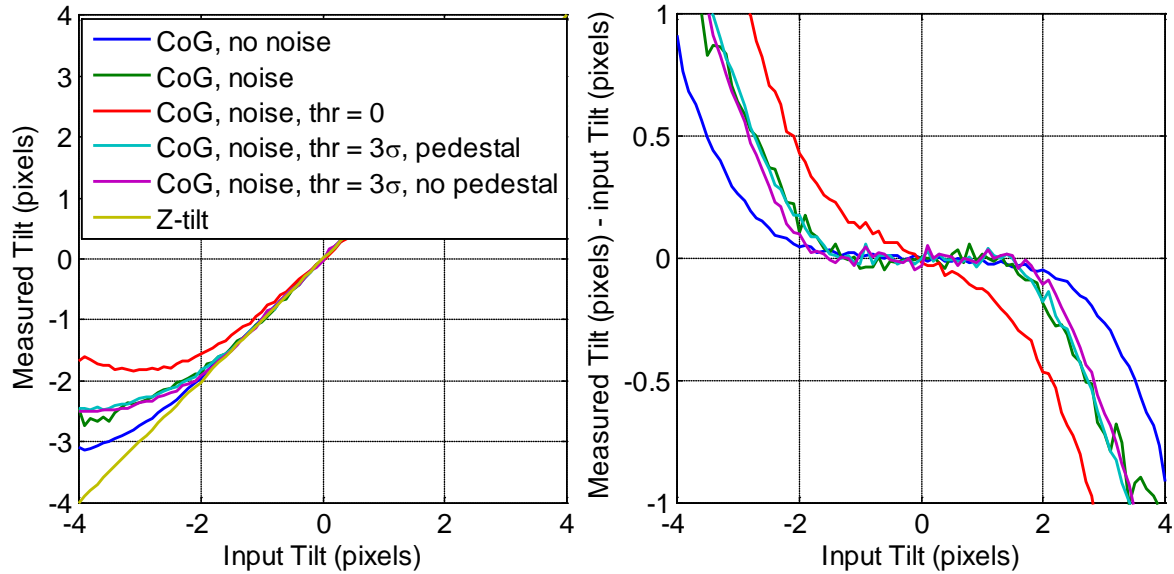


Figure 42: Comparison of different centroiding algorithms on the example of a MAD sub-aperture: CoG with different thresholding values.

The difference between keeping or not the pedestal is not obvious at this point (maybe the latter is better), thus we have redone another series of simulations, at weaker flux (500 photons) and less thresholding (1σ of the RON). The results are shown on Figure 43 and the interpretation is:

- Removing the pedestal seems to improve the linearity and its range.
- Using Correlation with a Gaussian of $\text{FWHM} = 2$ pixels provides a very accurate measurement of the tilt in spite of the noise, and this over a large range. But the required computing power is larger.
- The weighted CoG (with 2 or 4 pixels FWHM) is rather linear, over the same range as CoG, but the slope of the response is very different. This was to be expected as the weighting affects the measurement of the spot position, which is actually not anymore the CoG of the star image

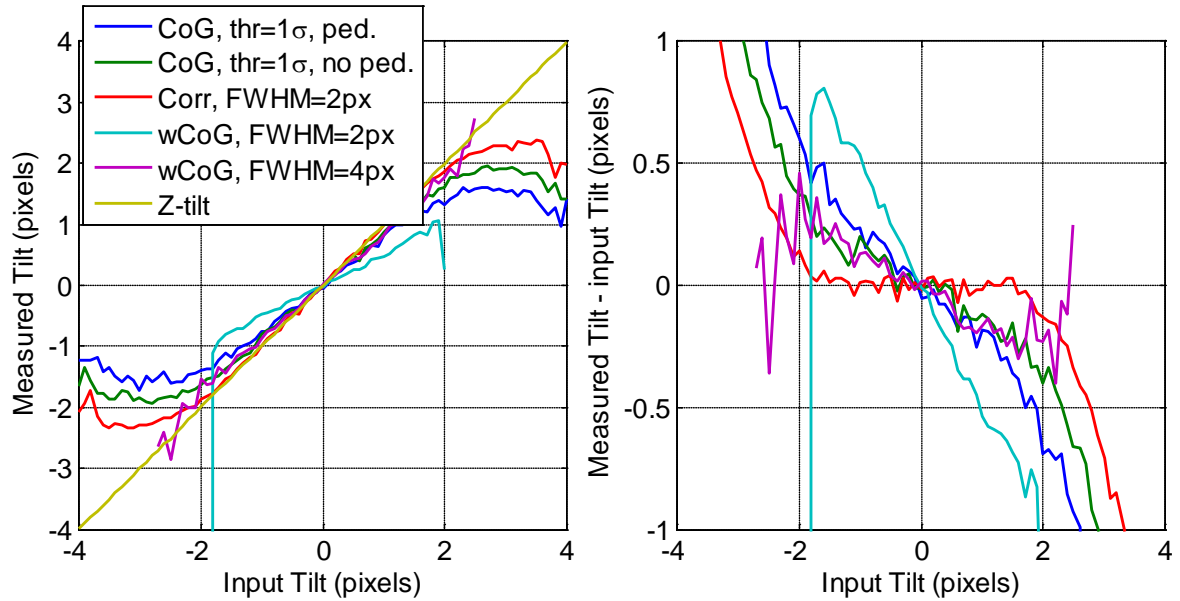


Figure 43: Comparison of different centroiding algorithms on the example of a MAD sub-aperture: CoG, wCoG and Correlation.

6.9.2 Other application - Long-exposure VLT Active Optics images

In the §6.8.3 we have seen how to use one part of the code to compute slopes from the VLT Shack-Hartmann short exposure images. For another application, we wanted to estimate the seeing from the FWHM of the spots of long-exposure images, still on the VLT Shack-Hartmann. Such images can be saved at the telescope, but to calibrate the FWHM measurement routines, it was important to generate images from known seeing (the Phase Screens generation and validation is described in §8.2). Figure 44 shows an example of such images. 1000 noise and turbulence realization are used to simulate the “long-exposure” effect. This study allows calibrating the FWHM measurement performed over the last 10 years by the VLT SW against images with a known seeing (Figure 45). The results of this study are presented in RD5 and RD6.

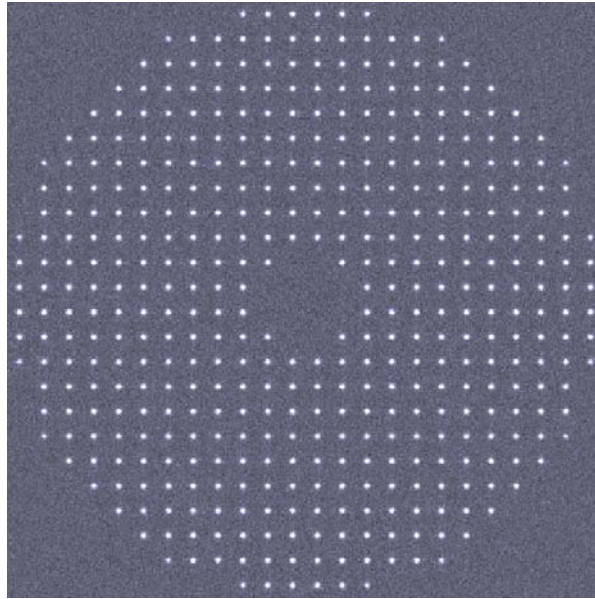


Figure 44: Simulated long-exposure image by the VLT Shack-Hartmann.

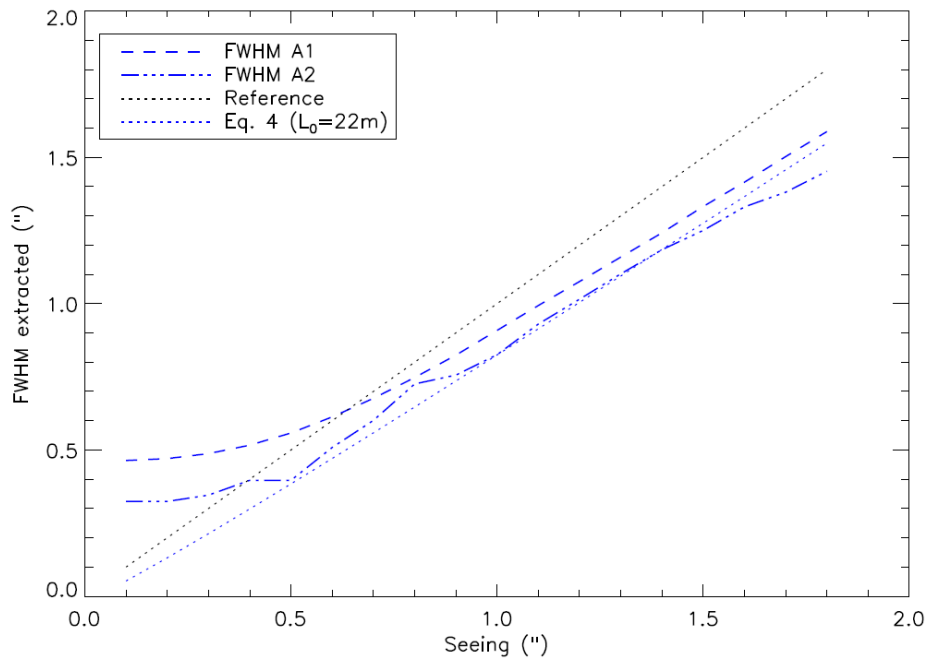


Figure 45: From RD6: Calibration of the FWHM values measured by the VLT active optics SH vs. the input seeing (Reference). FWHM A1 is a proposed algorithm to replace the current VLT one (FWHM A2). Eq. 4 ($L_0 = 22\text{m}$) is the prediction of the FWHM by a formula taking into account the effect of the outer scale (A. Tokovinin 2002).

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 51 of 74
--	--	--

7 Simulation of Interaction Matrix recording

7.1 Introduction

The function “makeIM_PSIM” goes through a list of pre-recorded DM surface shapes (each one defining an actuator’s displacement or a given mode), analyses them thanks to the SHWFS model and stores the slopes in the Interaction Matrix. Required inputs are:

- The SHWFS parameter file name
- A tag allowing identifying the folder where the DM surface shapes are stored (.mat files for faster loading). The maps must contain enough pixels so that the SHWFS measurement can be performed as specified.
- The maximum number of orders to use (number of modes when recording modal IM)
- A list of possible amplitudes to apply to the modes individually.
- A list of coefficients describing the mis-registration between the WFS and the DM:
 - x shift in % of a sub-aperture (>0 or <0)
 - y shift in % of a sub-aperture (>0 or <0)
 - rotation in degrees (>0 or <0)
 - x stretch in % (>0 or <0, >0 for shrinking of the DM on the WFS)
 - y stretch in % (>0 or <0, >0 for shrinking of the DM on the WFS).

For any set of coefficients, the DM shapes are interpolated to the required DM/WFS configuration (Cubic spline interpolation by Matlab).

7.2 Pupil shape

A special care was taken to reproduce accurately the behaviour at the edges of the pupil. Indeed in the case of the AOF, the DM is limited in dimensions (equivalent to a pupil mask on an infinite DM) and a pupil mask is deposited on the WFS lenslet array, so in case of mis-alignment (shift between the DM and the WFS), the illumination of the edge sub-apertures will change. Figure 37 illustrates this effect (truncation of the pupil). This mask is applied in front the WFS model so the it is taken into account when computing the centroids, both in the geometric and diffractive models.

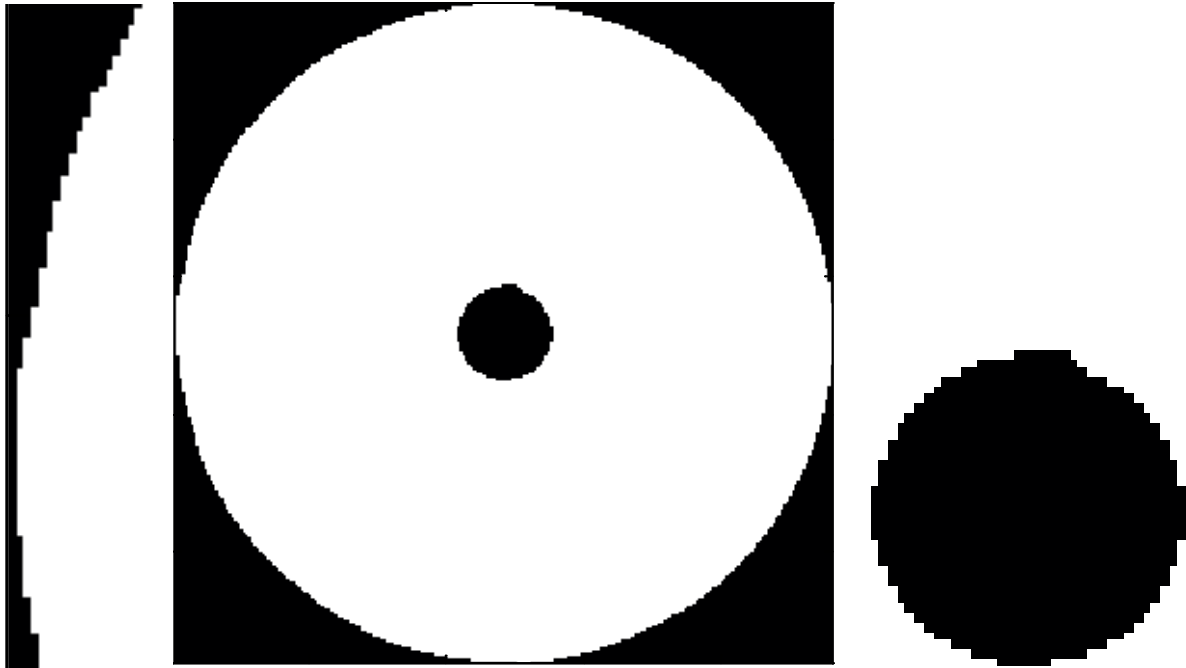


Figure 46: Example of pupil shape resulting from a mis-registration. Centre: full pupil. Left and right: close-up views on parts of the pupil edge.

7.3 AOF Influence Functions

7.3.1 Handling of FEA IFs

In the end, the PSIM for the AOF will be generated using the Influence Functions of the DSM measured on the ASSIST bench with the help of a fast interferometer. Meanwhile, the PSIM can be first simulated using Zernike polynomials as Influence Functions (as it was done for NAOS, see §6.3.6). A better approximation is to use the DSM shell shapes generated by Finite Elements Analysis by Microgate/ADS. Those were delivered to ESO in 2009. They have been analysed as described in RD7.

They have been generated on an irregular grid of points (more concentrated at the location of the actuators and at the shell's edges), thus I have interpolated them to a regular grid of 960 x 960 pixels to be used in the PSIM model (see Figure 47).

The Influence Functions are defined as the shapes that take the shell when one actuator is forced to a certain displacement (say 1 μm) while the others are forced to zero. An example is shown on Figure 48.

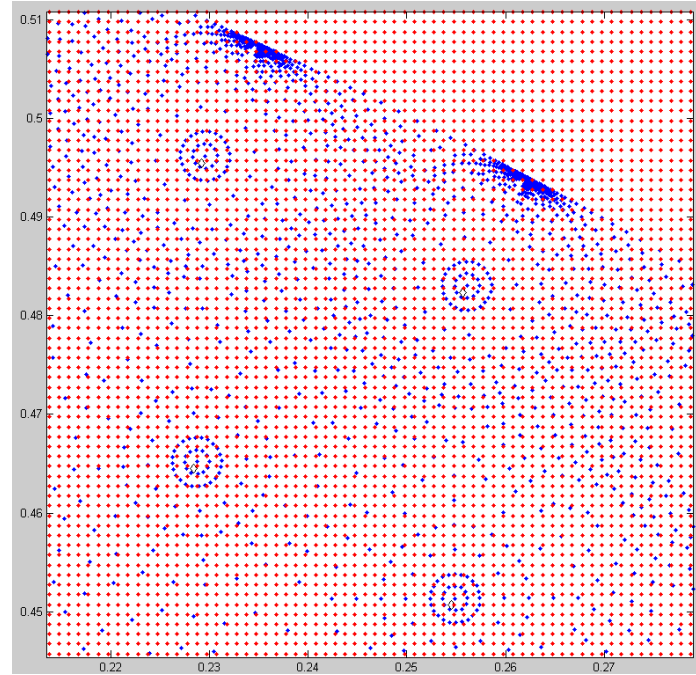


Figure 47: Close-up view of the grids of points sampling the DSM FEA Influence Functions. Blue: Microgate/ADS. Red: extrapolated. Diamond: actuators location.

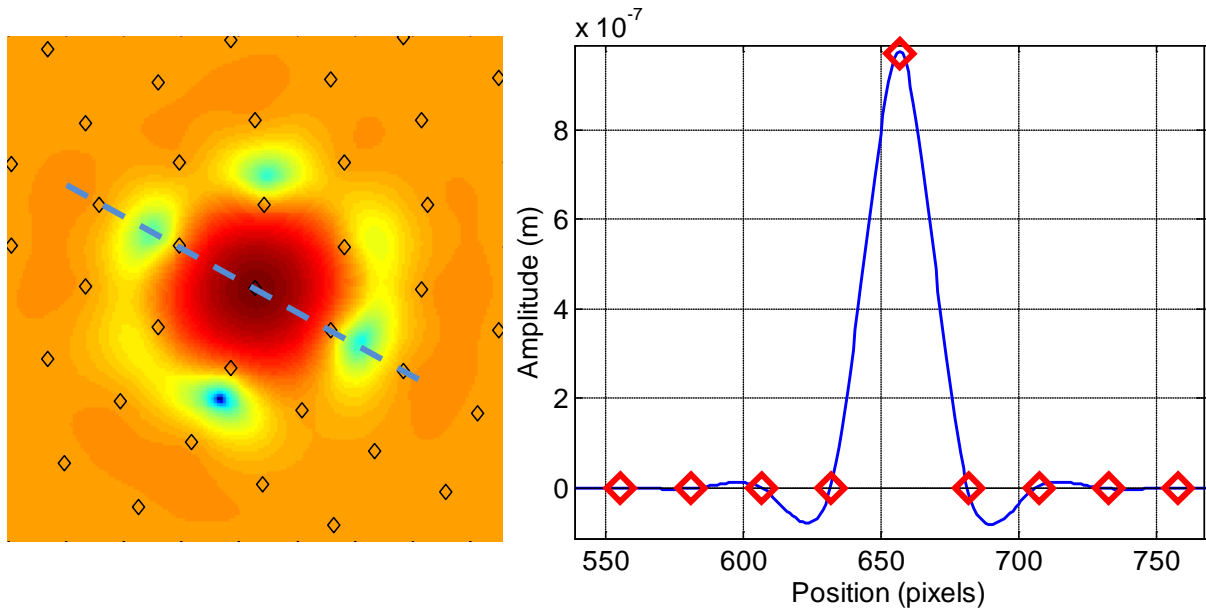


Figure 48: Left: Example of a DSM Influence Function, the actuators positions are marked by diamonds. Right: cross-cut of the left image (along the blue dotted line). The actuators positions are marked by red diamonds.

From those Influence Functions $\mathbf{S}^{(j)}$ one can build the base of stiffness modes of the mirror (projection matrix \mathbf{V}). There are thus 3 bases in which to express commands to the DSM (see RD7 for details):

- The actuators basis. The commands $\mathbf{f}^{(j)}$ are directly forces applied to the actuators.

- The Influence Functions basis. The commands $\mathbf{p}^{(j)}$ are shell displacement at the location of the actuators. The resulting total displacement of the shell is:

$$\mathbf{d} = \sum_{j=1}^{1170} \mathbf{S}^{(j)} \mathbf{p}^{(j)} \quad (7)$$

- The stiffness modes basis. The commands $\mathbf{m}^{(j)}$ are amplitudes to apply as linear combination of the stiffness modes. The resulting total displacement of the shell is:

$$\mathbf{d} = \sum_{j=1}^{1170} \mathbf{S}^{(j)} \mathbf{V} \mathbf{m}^{(j)}, \quad \mathbf{V} \text{ the projection matrix from IFs to stiffness modes} \quad (8)$$

In practice, the AOF RTC will command the DSM in amplitudes in the IFs basis. In that case the amplitudes \mathbf{p} . The AOF RTC could also command the DSM in the stiffness modes basis (amplitudes \mathbf{m}), or KL basis. The DSM electronics will take care of converting those amplitudes into forces to apply to the actuators thanks to the so-called feed-forward matrix.

The DSM IFs were used to study the trade-off between DSM shell thickness, corrected seeing, number of modes corrected, maximum force allowed on the actuators and ratio of saturated commands (in RD7).

7.3.2 Other application: shell surface error

Knowing the IFs of the DSM and the forces associated, it was possible to project on them the static shape of the shell (after polishing, Figure 49, left) to evaluate which surface error can be achieved after flattening (Figure 49, right), how many modes have to be corrected to reach the specifications (Figure 50, left), and how much force will that use (Figure 50, right).

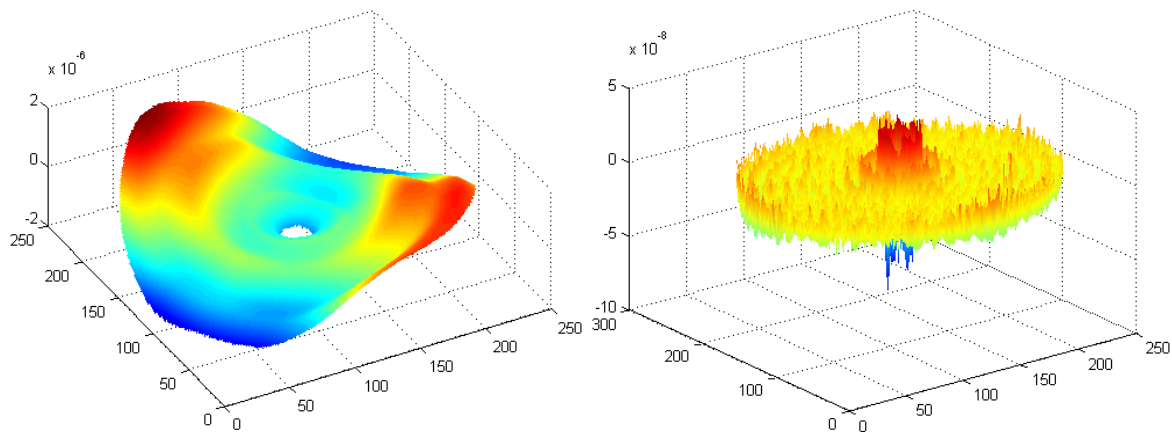


Figure 49: Left: measured static shape of the DSM shell. Right: simulated shape after flattening (correction of 1170 modes, the surface residual is 6.2 nm rms).

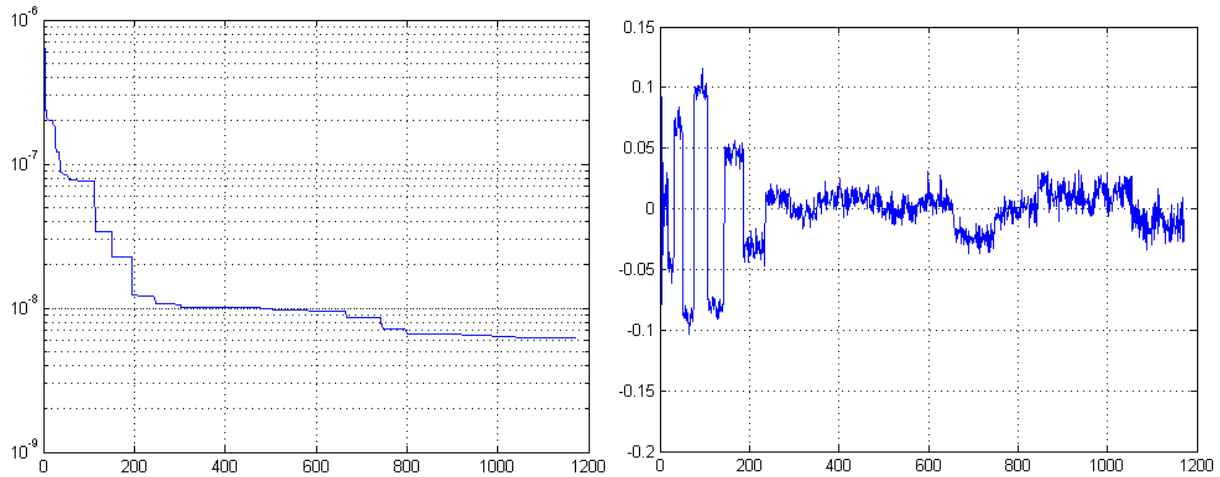


Figure 50: Left: residual rms in meters vs. number of modes corrected. Right: Force in N on the 1170 actuators when correcting 855 modes (minimum required to reach the goal spec of 6.6 nm rms).

7.3.3 Other application: offload to the VLT M1 elastic modes

When in operation, the AOF will take control of the VLT active optics. In particular the M1 shape will no longer be controlled by the Active Optics SHWFS, but by offloads from the DSM. All possible shapes of the DSM are known thanks to the FEA IFs (§7.3.1), and the possible shapes of the VLT M1 (“elastic modes”) are known theoretically (RD8, see Figure 51) so that ones can be projected on the others, and the residual fitting error can be computed.

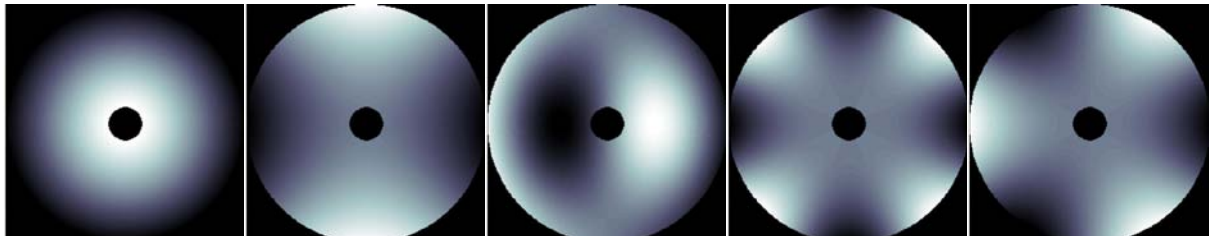


Figure 51: First elastic modes of the VLT M1.

7.4 Interaction Matrices

With all the pieces of code developed, it was now possible to generate Pseudo-Synthetic Interaction Matrices for the AOF. This was first done without noise in a zonal way (using the FEA IFs, Figure 52) or in a modal way (using the stiffness modes, Figure 53). Note that it is not easy to display such large matrices in a meaningful way.

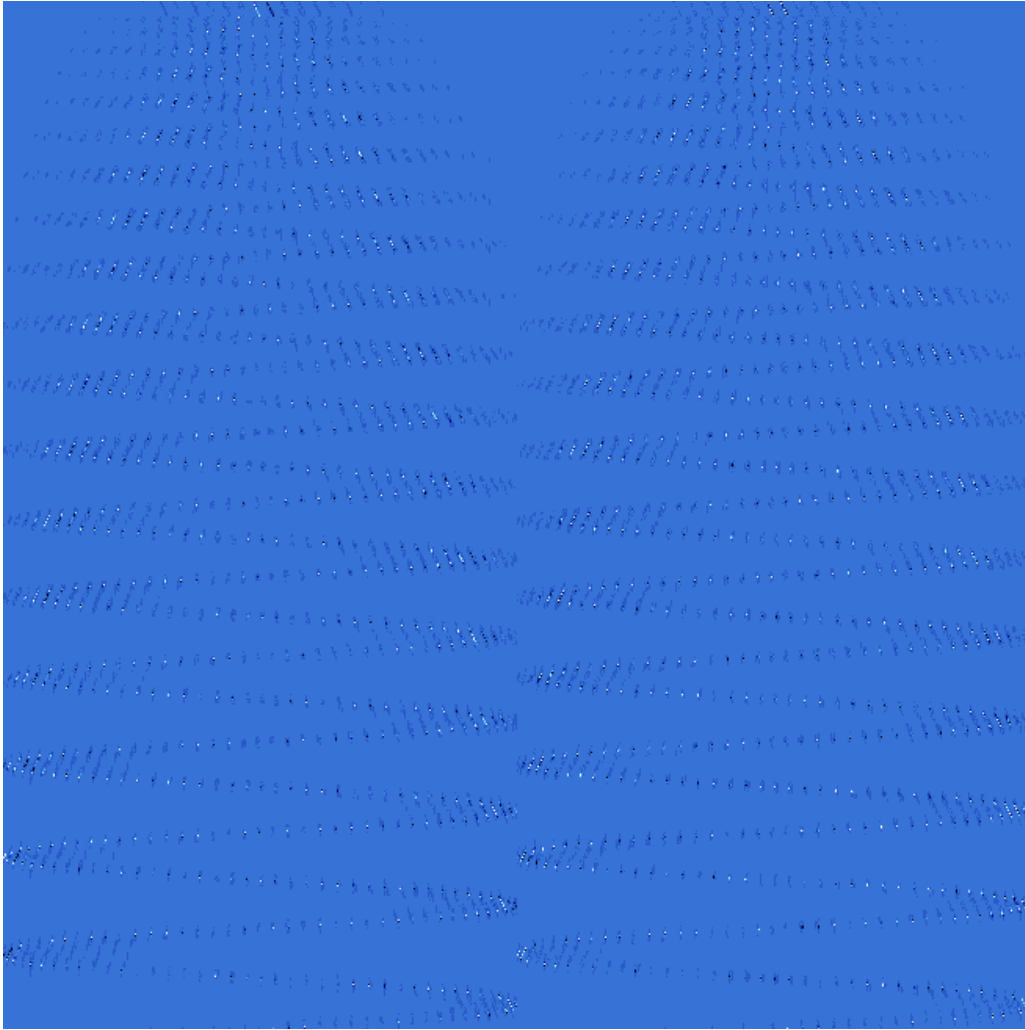


Figure 52: Zonal PSIM for the AOF: 1170 actuators vs. 1240 X slopes + 1240 Y slopes.

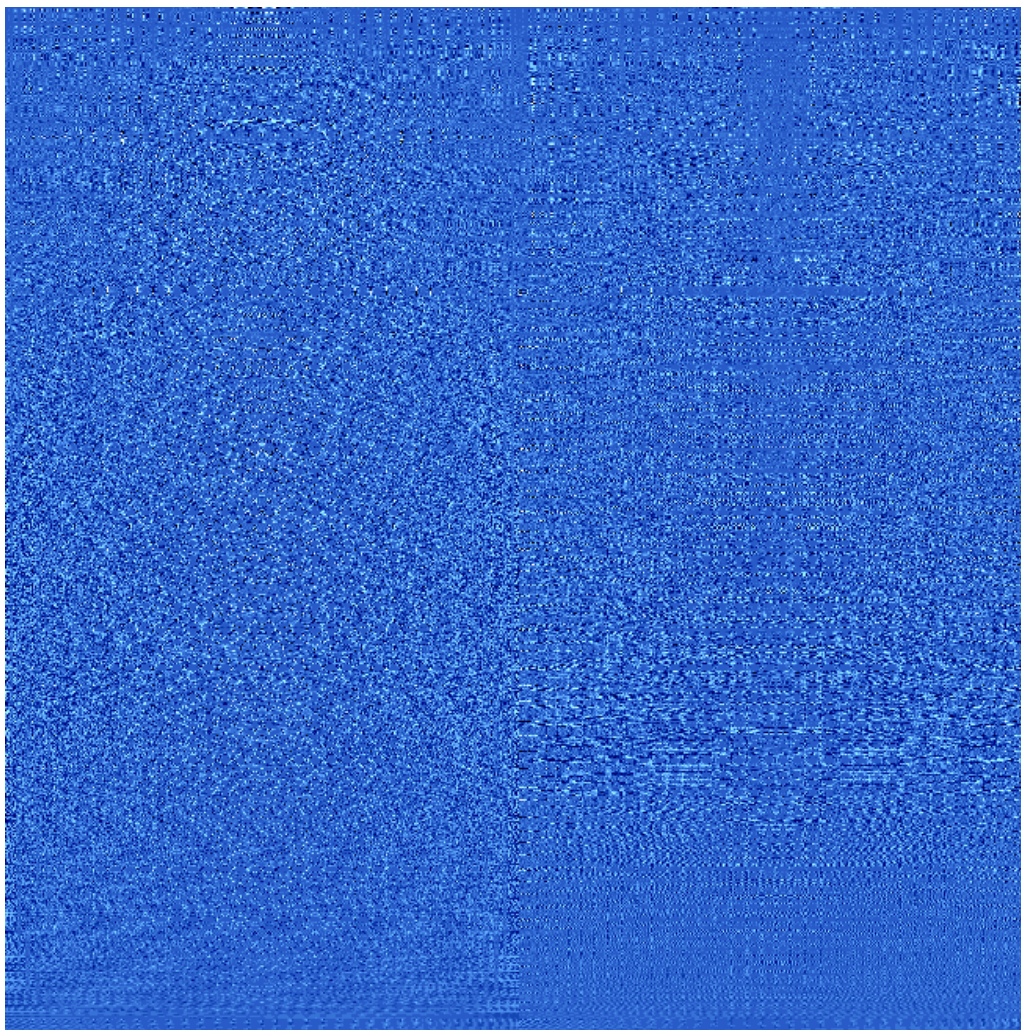


Figure 53: Modal PSIM for the AOF (stiffness modes): 1170 modes vs. 1240 X + 1240 Y slopes.

The clear difference is that the zonal IM has a very localized signal (sub-apertures near the displaced actuator) while the modal one has signal spread over all its elements.

These two bases have a different sensitivity to measurement noise, as would be shown if the IM was simulated with a noisy SHWFS. In practice (measurement of an IM on an instrument), the best base will have to be chosen to improve SNR, and measurements methods such as the Hadamard one for fast recording of the IM can be used to further increase the SNR.

7.5 Verification on NAOS

The AOF PSIM “quality” cannot be estimated until the system is manufactured and IMs are actually measured with it (on ASSIST). At that point we will be able to compare the matrices, and the ultimate test will be to use each of them to close the loop and to compare the achieved AO performance reached in both cases (measured vs. PSIM).

Meanwhile, I could use my code to generate PSIMs for other AO systems, and in particular NAOS installed at the VLT. Its SHWFS geometry and properties are well-known, as well as the position of the post-focal DM actuators. An IM is recorded occasionally on a fiber source (Figure 54, top-left) to assess the instrument ageing. On this IM is clearly visible an actuator with a weak response (# 182, fourth from the bottom). The DM influence functions were recorded long ago, so I thought that using a simulated Gaussian shape for the Influence Functions couldn't be that bad. I have then generated a PSIM from the SHWFS parameters, the simulated IFs and by minimizing the difference with the measured IM (parameters to optimize with the approach described in §7.6: X and Y shifts, rotation, X and Y stretch, IFs amplitude, IFs cross-coupling). The PSIM is shown on Figure 54, top-right. It looks very similar to the measured one except for the defective actuator, and the absence of background noise. The difference between the two (Figure 54, bottom) shows a regular pattern at the actuators location (probably due to the error committed in approximating the IFs to Gaussians) and an amplitude variation across the pupil (maybe due to pupil aberrations of higher order than shift, rotation and stretch).

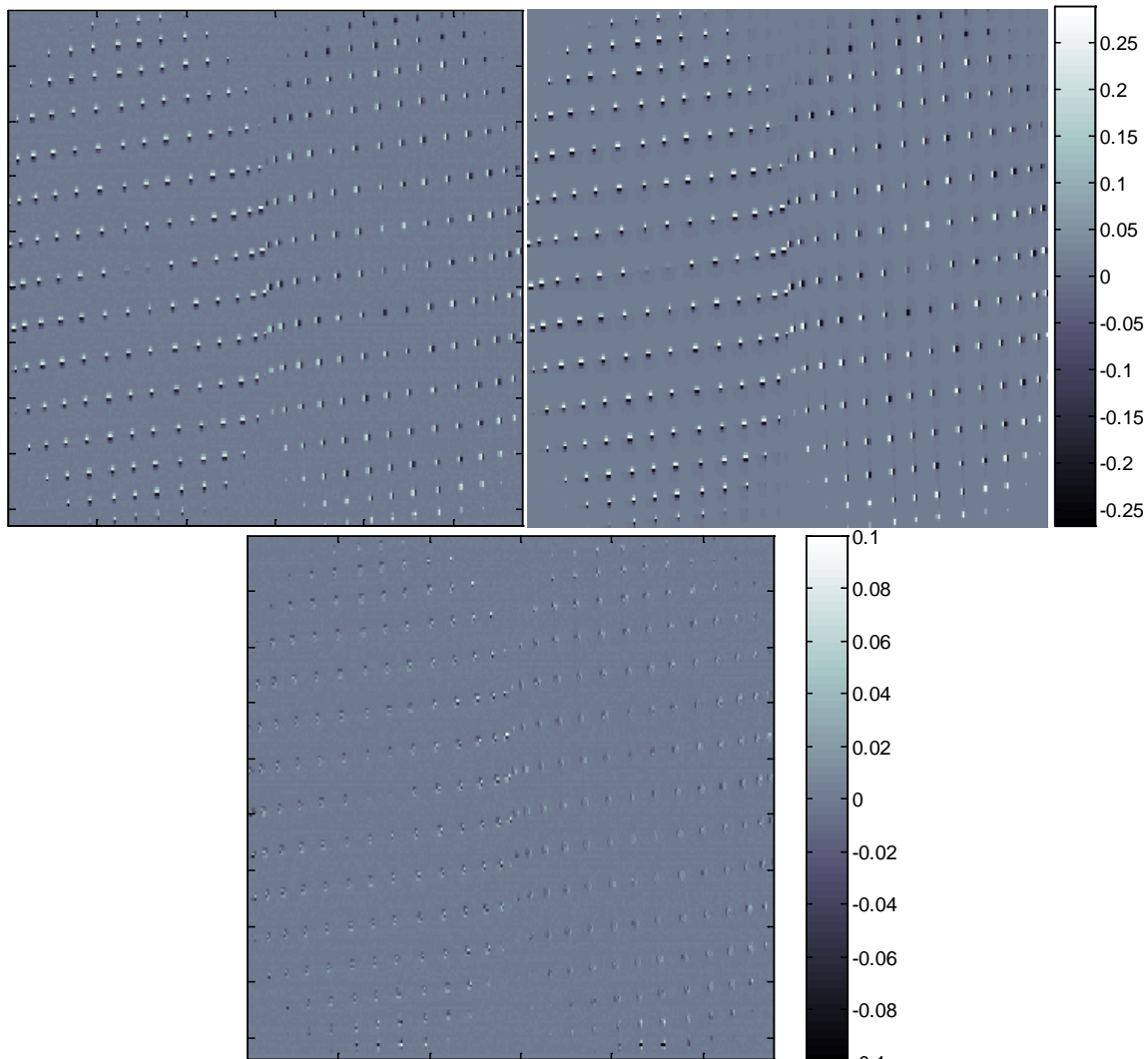


Figure 54: IMs on NAOS: Measured (top-left), simulated (top-right) and their difference (bottom, different scale).

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue: 1 Date: 03.12.2012 Page: 59 of 74</p>
---	--	--

The result is however very satisfactory. The PSIM was loaded in the NAOS RTC and the loop was successfully closed with it. A technical run at the VLT in May 2011 allowed further investigation; in particular we couldn't measure any performance loss from the use of the PSIM compared to the measured one. The measured IFs of the NAOS DM have also been used to generate a PSIM, and its performance seems to be slightly worse than the one generated from Gaussian IFs.

7.6 Mis-registration parameters estimation from comparison of IMs

For the AOF the approach adopted concerning the identification of mis-registration between the DSM and the WFSs is to combine sets of real-time data (slopes and DSM commands, measured in closed loop without adding any disturbance to the system besides the turbulence itself) with the knowledge of the actually used IM in order to retrieve the shifts, rotation... and update the CM. This approach requires the PSIM model but no **measured** IM. The results of this study are described in RD9.

Another approach (simpler, faster, but requiring a measured IM) requires only the PSIM code and will be presented here. It consists in generating one perfectly aligned PSIM and several "delta" IMs, i.e. PSIMs with small known mis-alignments. If we consider:

- 1) the real (measured) IM to be a linear combination of the perfect IM (PSIM) and of the delta IMs:

$$IM = PSIM + \alpha_{rot} \times \delta IM_{rot} + \alpha_{Xshift} \times \delta IM_{Xshift} + \alpha_{Yshift} \times \delta IM_{Yshift} + \dots \quad (9)$$

- 2) the delta IMs to have been recorded (by measurement or simulation) in the linear range of the mis-registration, and
- 3) the mis-registration delta IMs to be uncorrelated,

then the coefficients $\alpha_{rot} \dots$ can be computed by simple projection of the IM error (difference between the measured and the perfect IM) on the delta IMs:

$$\alpha_{rot} = \frac{\|(IM - PSIM) \times \delta IM_{rot}\|}{\|\delta IM_{rot}\|^2} \quad (10)$$

In practice the non-exactness of the assumptions above can be overcome by the use of several iterations. The starting point for each new iteration is a PSIM re-computed at the mis-registration parameters estimated at the previous iteration.

I have tested this method in simulation, i.e. the only difference with the real test is that I have simulated the "measured" IM with some mis-registration errors, and tried to retrieve those values blindly using the approach described above. The delta IMs were generated with 0.1 degree in rotation or 1% of a sub-aperture in shift. The Table 1 shows the input errors used, and the estimations after 2 iterations, which are very close to the inputs.

	α_{rot} (degrees)	α_{Xshift} (% of a sub-aperture)	α_{Yshift} (% of a sub-aperture)
Input errors	-0.24	-4.90	8.30
Estimated values after first iteration	-0.22	-4.68	7.80
Estimated values after second iteration	-0.24	-4.94	8.30

Table 1: Mis-registration parameters estimation.

The drawback of the method is that it requires a measured IM, which would make it not applicable to the AOF. However I have been working on a method allowing (partially) retrieving the IM directly from the measured slopes and voltages. This method is still under development so the equations below are preliminary. It is based on the fact that the slopes measured at an iteration (i) are linked to the voltages applied at the same iteration by the Interaction Matrix:

$$S(i) = V(i) \times \text{IM} + \text{Turbulence} + \text{Noise} + \dots \quad (11)$$

The same is true between the increment of slopes and voltages, with the advantages or cancelling out the turbulence and noise contributions (somehow):

$$\begin{aligned} \delta S(i) &= S(i) - S(i-1) = \\ [V(i) - V(i-1)] \times \text{IM} + \delta_{\text{Turb}} + \delta_{\text{Noise}} + \dots &= \delta V(i) \times \text{IM} + \delta_{\text{Turb}} + \delta_{\text{Noise}} + \dots \end{aligned} \quad (12)$$

Both sides of the equation can be multiplied by the pseudo-inverse $\delta V(i)^+$ of $\delta V(i)$, obtained by Truncated SVD:

$$\text{IM} = \delta V(i)^+ \times \delta S(i) + \dots \quad (13)$$

Figure 55 shows an IM obtained with this method (3 x 4096 loop cycles at 444 Hz), to be compared with the ones of Figure 54. It is noisy, but preliminary tests show that it should be good enough for projection on the delta IMs and estimation of the mis-registration parameters.

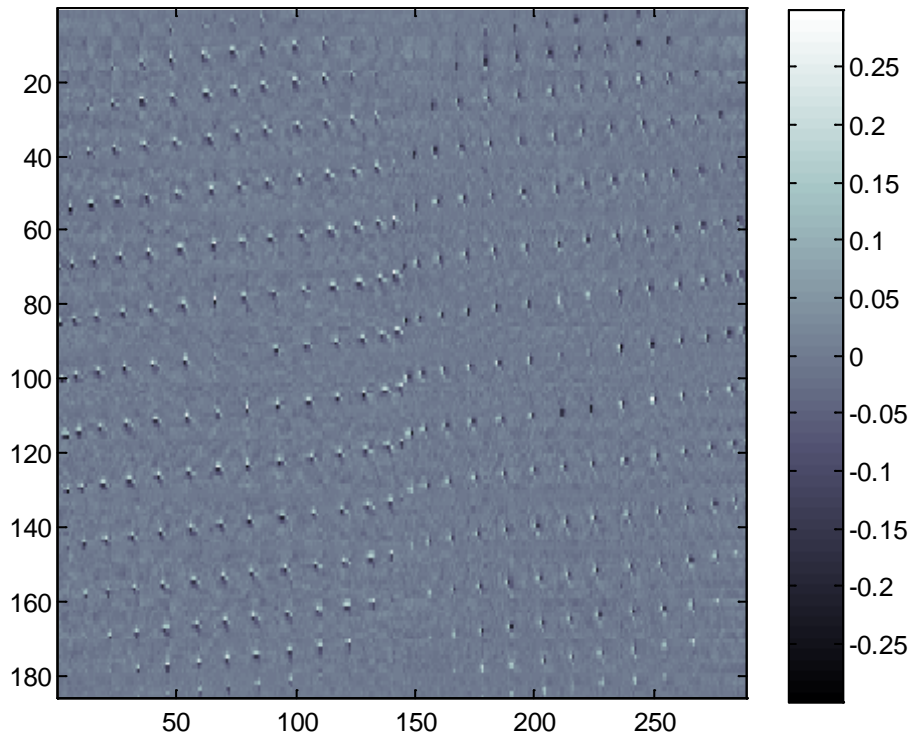


Figure 55: IM on NAOS estimated from the measured closed loop slopes and voltages.

7.7 Code efficiency

I have used the Matlab “Profiler” tool to optimize the efficiency of all routines. This tool provides the time spent on each routine, sub-routine, and eventually line of code. It also provides hints on how to reduce the computing time. Figure 56 shows the example of this tool applied to the “makeIM_PSIM” routine, i.e. computation of the Interaction Matrix for the AOF with 1170 modes. The total computation time was 110 seconds (i.e. 8.5 ms per mode in average). “makeIM_PSIM” calls “SHWFS_PSIM” and both take only little self time. “SHWFS_PSIM_Mainloop’ takes most of the time. One can see how many times each function is called. Obviously the ones called the most should be optimized. Figure 57 shows the details of the time spent in “SHWFS_PSIM_Mainloop”. We see that 5 functions or lines are called 1 450 800 times (1170 IFs x 1240 sub-apertures) and share almost equally the computing time.

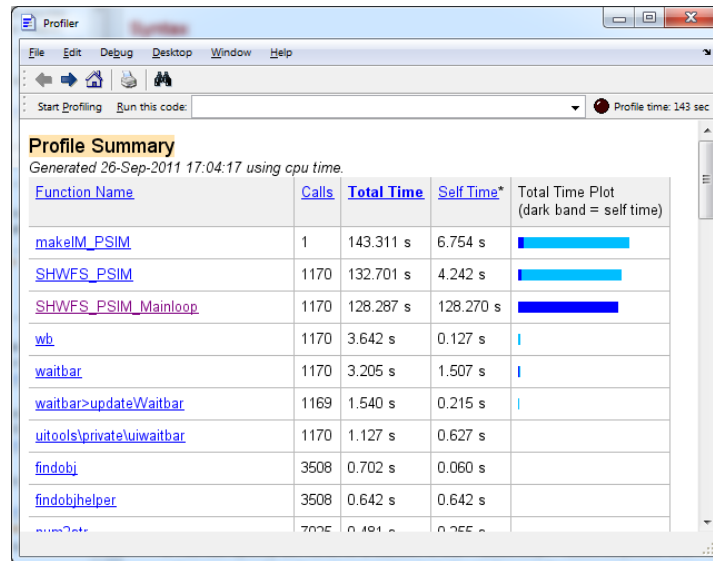


Figure 56: 'Matlab' Profiler tool applied to the routine 'makeIM_PSIM'.

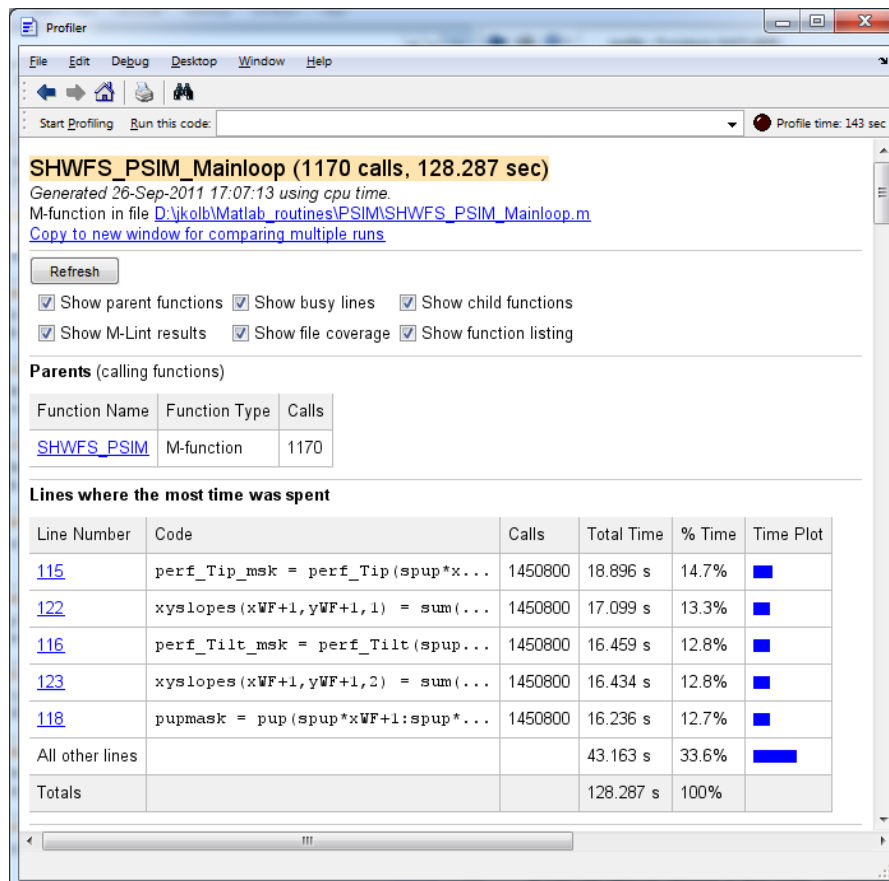


Figure 57: Details of the time spent on the routine 'SHWFS_PSIM_Mainloop'.

This tool can also be used to compare the time required to perform a certain operation (like computing an IM) in different configurations.

In the Table 2 are listed total PSIM computing times (on my laptop, an ESO standard DELL 6400 with 4Gb of RAM) as a function of the model used (geometric or

diffractive), the centroiding algorithm, the number of pixels in the input WF, the mis-alignments and the presence of noise. We see that if a simple geometric model with few pixels can run quite fast, a full diffractive model with noise is much slower.

Configuration: Model, Slopes computation method, noise, mis-registrations, number of pixels in the pupil, source size (when relevant)	Total computing time (s)
Geometric, Z-tilt, no noise, perfect alignment, 240 px	125
Geometric, G-tilt , no noise, perfect alignment, 240 px	140
Geometric, G-tilt, no noise, mis-alignment , 240 px	450
Geometric, G-tilt, no noise, perfect alignment, 960 px	455
Diffractive, CoG , no noise, perf algn, 960 px, diffraction limited source	1080
Diffractive, CoG, no noise, perf alignment, 960 px, extended source	1305
Diffractive, CoG, noise , perf alignment, 960 px, extended source	1475
Diffractive, Corr , noise, perf alignment, 960 px, extended source	5120

Table 2: Computing times as function of model complexity (increased complexity highlighted in red).

8 Closed loop

Once the IM measurement code written, it is actually fairly easy to simulate an AO closed loop on a single star (routine “Closed_Loop_PSIM”). One just needs to invert the IM, move Phase screens in the front of the WFS, compute the correction to apply and put in place the control loop.

8.1 IM inversion

The routine “invIM” checks the dimension of the input IM, computes its SVD and filters the required number of modes before outputting the Control Matrix:

$$IM = U \times S \times^t V \quad (14)$$

$$CM = V \times T \times^t U \quad (15)$$

where S is the diagonal matrix of eigen values and V the matrix of eigen modes. The values in T are the inverse of the ones in S, except for the eigen values of the filtered modes which are set to zero.

In a low-order AO system (for example MAD with 8x8 sub-apertures), when the IM is recorded zonally and inverted, the eigen modes (at least the low order ones) are very similar to the Atmospheric modes (Zernikes or Karhunen–Loève) and ordered in the same way. This is not anymore true for higher order systems (40x40 for sure, but even 14x14 like NAOS), where the low order modes are not as well seen as the higher order ones. We noticed that using modes with normalized amplitude doesn’t solve the problem. We have tried this with stiffness modes, Zernike modes and Zernike modes produced by the DM, i.e. projected on the stiffness modes, and we expect that it would be the same for normalized Karhunen–Loève modes.

To quickly overcome this problem we have used the Zernike modes with amplitudes following the Von Kármán statistics of the atmospheric turbulence to build the IM. The eigen modes resulting from the SVD are then ordered properly and can be

truncated by order of eigen values. This trick was used to build the CM before simulating the closed loop. The proper way to do the inversion is still to be implemented (§9.12).

8.2 Phase Screens

The routine “createPS_PSIM” generates Phase Screens using the inverse Fourier transform of a white noise spectrum coloured by the Von Kármán spectrum (example on Figure 58, left). Those Phase Screens were analysed to make sure that they produce the required turbulence: hundreds of them are generated, projected on the Zernike polynomials, and the variance of the coefficients are fitted with a Von Kármán spectrum (variables r_0 and L_0 , see Figure 45, right).

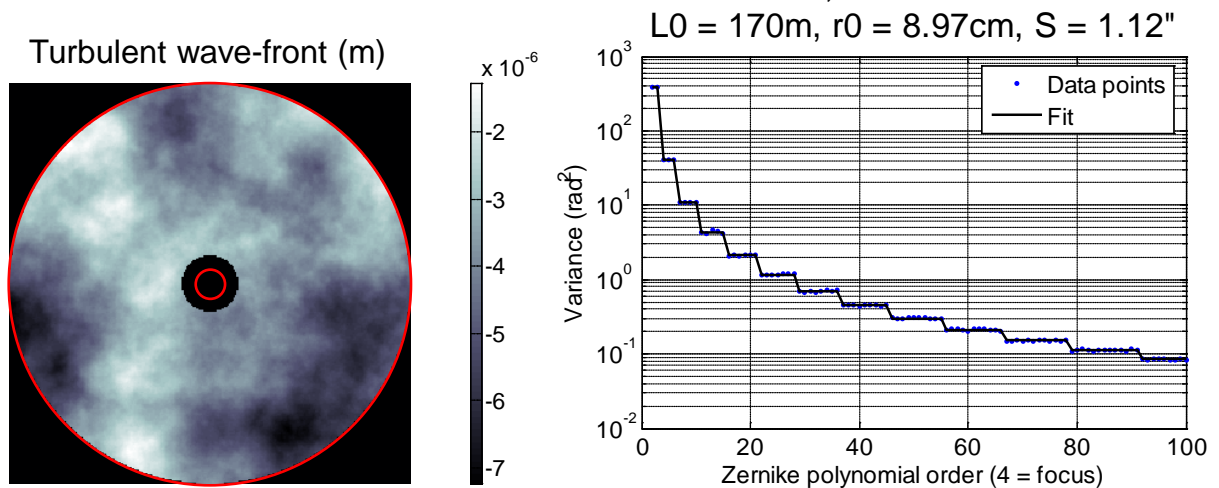


Figure 58: Example of Phase Screen (left) and verification of its properties (right). The red circles represent the dimension of the AOF DSM.

8.3 Control

At each loop cycle, the correction increment is computed by multiplying the vector of measured slopes, the Control Matrix and the loop gain (equation 16). It is then applied to the DM after the loop delay (equation 17). This means that at each loop cycle, a full DM shape should be computed and subtracted to the input turbulence before being analysed by the WFS. For a system with 1170 actuators, this is a lengthy process.

$$\text{DMcomm}(i) = \text{DMcomm}(i-1) + g \times \text{Meas} \times \text{CM} \quad (16)$$

$$\text{WF}(i) = \text{Turb}(i) - \text{DMcomm}(i - \text{delay}) \times \text{Z}, \quad (17)$$

Z contains the Influence functions of the DM

A simplification is possible when using a geometric model of the WFS and **no noise**: instead of being done in the DM space, the subtraction between input WF and DM shape is done in the WFS space. This simplification is rigorously true (but only in the case with **no noise**) because the geometric model of WFS is perfectly linear, so only

	<p style="text-align: center;">Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue: 1 Date: 03.12.2012 Page: 65 of 74</p>
---	---	--

the turbulent WF needs to be measured, and the WF associated to the DM shape is a linear combination of the Influence Functions responses, i.e. the IM (equation 18).

$$\begin{aligned} \text{Meas}(i) &= \text{SH}(\text{WF}(i)) = \text{SH}(\text{Turb}(i) - \text{DMcomm}(i - \text{delay}) \times Z) \\ \text{Meas}(i) &= \text{SH}(\text{Turb}(i)) - \text{IM} \times \text{DMcomm}(i - \text{delay}) \end{aligned} \quad (18)$$

8.4 Closed loop

After a phase of initializations, the following steps are repeated at each loop cycle (i):

- Shift turbulence layers according to wind speeds and compute input WF
- (Only in full model,) compute the DM shape from the voltages at the cycle (i), either from:
 - Zernike polynomials
 - measured IFs or modes
 - Gaussian IFs
- (Only in full model,) interpolate the DM shape to the actual pupil mis-registration characteristics
- Subtract input turbulence and DM shape (only in full model, see §8.3, equation 17)
- Generate AO-corrected PSF at scientific wavelength (from real or reconstructed DM shape depending on the loop model)
- Analyse input WF with SHWFS, including possibly noise
- (Subtract turbulence WF measurement and commands*IM, only in simplified model and **without noise**, see §8.3, equation 18)
- Reshape the WF measurements to fill the output matrix '*meas*' at iteration (i)
- In geometric WFS model, generate a "fake" WFS sensor image for display (see §6.3)
- Compute the DM correction (see §8.3, equation 16)
- Fill the output matrix of commands '*cmd*' at iteration ($i + \text{delay}$). The '*cmd*' matrix is thus '*delay*' lines longer than the '*meas*' matrix
- Estimate AO performance with different methods:
 - As variance of the residual slopes on the WFS
 - As quadratic sum of the Zernike coefficients (reconstructed DM shape)
 - As Strehl Ratio on the PSF at scientific wavelength (at image center or at image maximum)
- Update display of loop information (optional)
- Save outputs

An example of display is shown on Figure 59, in the case of NAOS (14 x 14 sub-apertures) simulated with the full Closed Loop model.

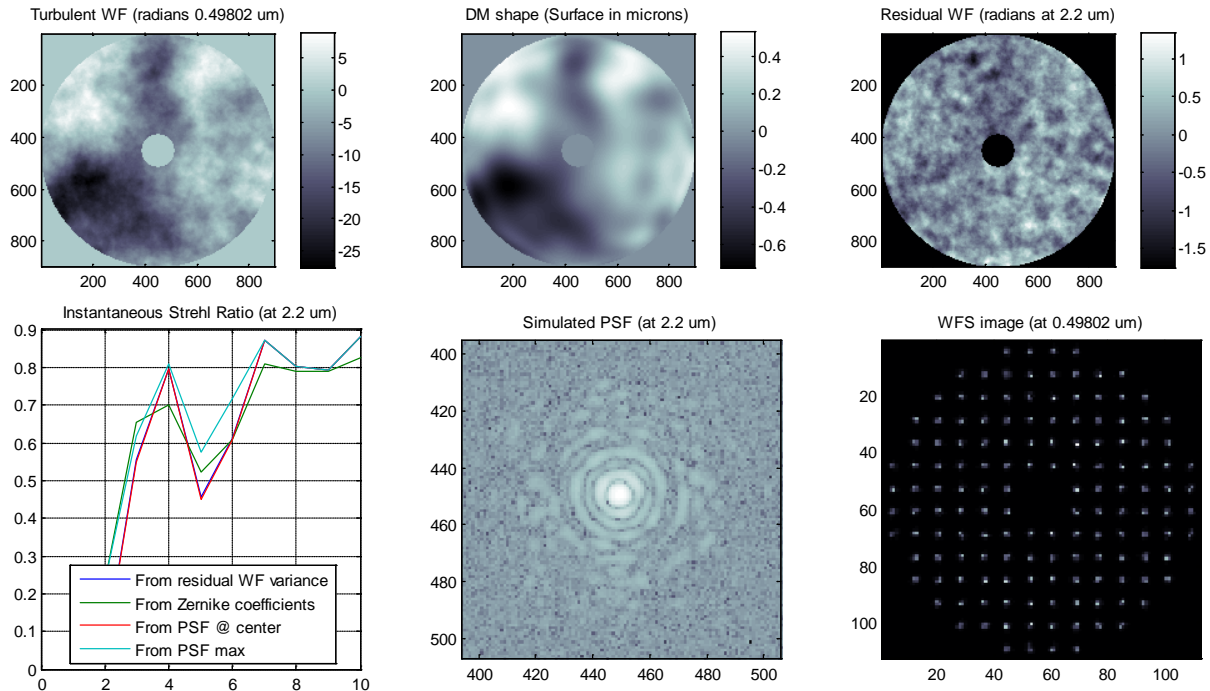


Figure 59: Example of Closed Loop simulation display. Top-left: Turbulent WF in radians at the WF sensing wavelength. Top-center: DM shape in microns. Top-right: residual WF in radians at the scientific imaging wavelength. Bottom-left: Instantaneous Strehl Ratio at the scientific wavelength (from different estimators, see text in §8.4). Bottom-center: PSF at scientific wavelength. Bottom-right: WFS detector image at sensing wavelength.

9 Future implementations

The PSIM code is fully functional to produce the first IMs for the AOF, but in this paragraph are listed some improvements that could be brought to the code in future releases (in no particular order), and first ideas of how to implement them. Their relative importance in producing a realistic PSIM should be evaluated.

9.1 **Elongated spot**

The AOF will use Laser Guide Stars which spots will be elongated and this shape could have an influence on the IM. Adding this feature to the code requires a convolution with an extended source like it is done already, but that varies from sub-aperture to sub-aperture. It should be simple and will not slow down the computation once I get the coordinates right.

9.2 **Rotation lenslet / detector**

A mis-alignment between the lenslet array and the detector will create an angle between the array of spots and the CCD pixels grid. This should not impact significantly the quality of the IM, but could change its sensitivity and add cross-talk between the X and Y signals thus could better be added to the PSIM simulations. I see two possible ways of implementing this:

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 67 of 74
--	--	--

- Applying a rotation matrix to the slopes after their measurement. This is less rigorous as it doesn't take into account the sub-aperture displacement w.r.t. the pixels... this implementation is the easiest.
- Rotating the WF image before it is sampled by the detector ('ImSensor_PSIM' routine). The interpolation will not be exact, but it is closer to reality, and shouldn't take too much time to compute.

9.3 Field stop

Until now we simulate a perfect square field stop of the exact size of the sub-aperture (as in the design of the AOF, see §6.4). A different shape of field stop, if smaller, would only require an additional mask at the WFS detector level. A larger field stop or a mis-alignment of it, generating cross-talk between the sub-apertures, requires a new definition of the sub-apertures' FoV and more pixels in the input WF.

9.4 Multi-lambda

As shown in the §6.5, the real polychromatic spot shape is different from the simulated monochromatic one. Adding this feature to the code could be done by convolving the lenslet PSF by an image of the polychromatic PSF. But in the AOF this is an issue only for the MCM of GRAAL (all other instrument modes are LGS i.e. monochromatic).

9.5 Lenslet aberrations

Lenslet aberrations (including defocus between the lenslet array and the detector) can change the spot size and thus the response to an incoming tilt (in a diffractive WFS model), the same way as an extended source (§6.9.1.2) or detector PSF (§6.9.1.3) do. They can easily be added as a static WF map in front of the WFS (changing the spot shape but keeping the same response to a flat WF or to a tilt).

9.6 Pixel scale per sub-aperture

Pixel scale can vary from one sub-aperture to the next, due to a tilt between the detector and the lenslet array. This could be measured on the real system and added to the PSIM model as a map of pixel scale per sub-aperture.

9.7 Higher order pupil aberrations

Until now we simulate only X/Y shifts, rotation and X/Y stretches between the DM and the WFS (§7.6). In the real system (in particular on ASSIST), higher order pupil aberrations (distortions) will be present and change locally the registration between the actuators and the sub-apertures. The implementation of those aberrations in the PSIM code requires a normalization of their definition, a coordinate convention, and then simply the interpolation of the IFs, as it is already done.

9.8 Different amplitude per actuator

Each actuator of the DM has a different behaviour, which may result in difference of amplitude of displacement between actuators for a same force applied. Those differences are taken into account in the measured IFs, but when using synthetic DM shapes they could be included as a map of actuators' response.

In the case of the DSM for the AOF which actuators are controlled in position, the inter-actuators differences should be taken into account in the internal loop feed-forward matrix and not in the IM.

9.9 DSM shape display / Forces display / saturations

The current PSIM model doesn't take into account the forces limitations on the DSM. It could be useful to display the DSM shape and map of forces applied, as well as warnings when saturation is reached. Figure 60 shows a tentative display of the forces applied to the DSM, but it could be improved by taking as example what was done for the LBT DSM, and extended to the DSM surface as well.

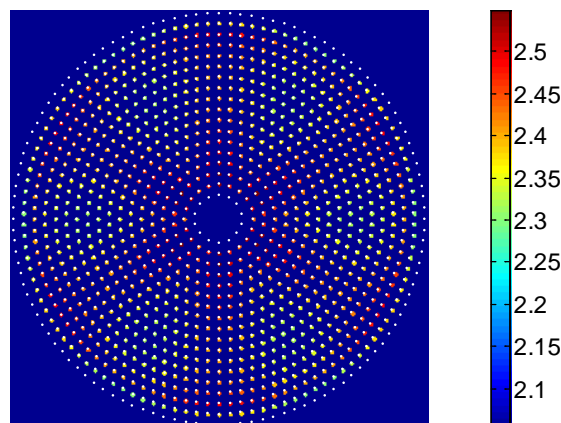


Figure 60: Tentative display of the forces applied to the DSM (in N)

9.10 Choice finite/infinite DM

As described in the §7.2, the current code uses a model of the DM finite in space (and defining the pupil), as it will be the case for the AOF. I could put in place an option to use either this model or the infinite DM one, in which the pupil is not truncated in case of mis-registration. This would be useful to understand if some behaviours of IM are related to the pupil truncation or not (as comparison to ESO's OCTOPUS AOF simulations).

9.11 Reduce extension of Influence Functions

A rather important computing time could be saved by setting to zeros areas of the IFs far from the actuator, or by using only the IFs of the actuators surrounding a sub-apertures when computing the WF to be measured by it.

	<p>Pseudo-synthetic Interaction Matrix simulations</p>	<p>Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 69 of 74</p>
---	---	---

9.12 Regularisation of IM inversion

As described in §8.1, a simple TSVD does not work for high order AO systems. The IM inversion is currently executed thanks to the routine “invIM”. This one could be improved to include regularization, i.e. the projection Karhunen–Loève modes before filtering, and then projection back to system modes (or other). This operation is described in RD10 and RD13.

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 70 of 74
--	--	--

10 Conclusion

The Pseudo-Synthetic Interaction Matrix simulation code I have written and described in this report has successfully provided the first AOF IMs including some measured parameters of the real system (namely the DSM Influence Functions, even if only generated by FEA for the moment).

The comparison of the quality between the PSIM and a measured IM will be assessed only once the DSM and one AO system (GRAAL or GALACSI) is mounted on the ASSIST test bench and ready to close AO loops. Meanwhile the code will be tested on the NAOS and MAD AO systems, in NGS and LGS SCAO for the first one, and NGS SCAO and GLAO for the second one. We expect that the PSIM will perform as well as a measured IM, and even better given that it can be generated free of noise.

The code was written in a modular way so that its components can be used independently for other applications. In this document are listed about 10 applications that already made use of parts of the code, be it slopes computations, linearity characterization, image generation, WF sensing, DSM performance prediction... Its versatility has proven to make it an excellent tool beyond its primal use of generating PSIMs.

Apart from the mandatory debugging phase, the next steps consist of further improvements of the code (as listed in this document), validation on existing AO instruments and incorporation of more and more measured parameters to bring the PSIM as close to reality as possible.

11 Appendices: How to run the code?

This paragraph gives quick examples on how to use some key pieces of the PSIM code. More detailed explanations on the routines are given in their header.

11.1 Input file for the function 'SHWFS_PSIM'

The input file for the function "SHWFS_PSIM" (generic name "SHWFS_input.m") is copied here after, with some example values (AOF case). In green are comments.

```
% SH model used: 1=geometric, 2=diffraction
SH_model = 2;

% Centroiding method (if diffraction SH model).
% - 'Bar': simple Barycenter, in pixels, from the *corner* of the image
% - 'CoG': Center of Gravity, in pixels from the *center* of the image. In
% that case 'param' is the threshold value to be applied to the
% image before computing the CoG (without pedestal, i.e. the
% threshold value is removed from the image and then all negative
% values are set to zero). No threshold is applied if 'param' is
% not inputted.
% - 'wCoG': weighted Center of Gravity. The input image is weighted before
% computing the CoG. 'param' can then be 1) the weighting map and
% should be the same size as the input image, or 2) the opposite
% of the power to which to take the input image, or 3) the FWHM
% (in pixels) of the Gaussian spot to use as weighting function.
% If 'param' is not inputted, then the weighting map is the image
% itself, which is equivalent to squaring the image before
% computing the simple CoG.
% - 'Corr': Correlation. 'param' is then the FWHM (in pixels) of the
% Gaussian function to be used as correlation map.
% - 'MF': Matched filter. 'param' is then the filter map and should
% be the same size as the input image. NOT IMPLEMENTED YET
% Centroiding method (if geometric SH model).
% - 'Z': Z-tilt method: the WF in each sub-aperture is projected on
% Zernike Tip and Tilt and the X and Y slopes are the values of
% the projection coefficients.
% - 'G': G-tilt method: the angle of arrival of the input WF is computed
Cen_meth = 'CoG';
param = 0;

D = 8; % Telescope diameter in meters
SHsize = 40; % Number of sub-apertures across the diameter
nbpixCCD = 6; % Number of pixels per sub-aperture
pix_scale = 0.83; % Detector pixel scale in arcsec/pixel
noise = 1; % 0 if perfect detector, 1 if noise added
wl = 0.589; % Wavelength (um)
source_size = 2*pix_scale; % Source size (FWHM) in " (= 0 if point source)
pixsize = 24; % Detector pixel size in um
outmax = 1; % External diameter of the pupil mask (>0,
between outmin and 1)
outmin = 1.12/8; % Internal diameter of the pupil mask (between
0 and outmin, <1)
```

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 72 of 74
--	--	--

```

subap_min_illu = 0.50;      % Min illumination required for validity of the
                             sub-apertures (between 0 and 1)
Cmin = 2;                  % Min binning of the WFS image to produce the
                             detector image

% Detector parameters, used only when the noise is set to 1:
Nphot = 100;               % Number of incident photons per sub-aperture and
                             per frame
Exptime = 10^-3;           % Exposure time (in s)
QE = 0.8;                  % Quantum efficiency from 0 to 1 [no unit]
PSF = 1.1;                 % Point Spread Function of the sensor (=0 if all
                             charges collected in one pixel) [pixels]
Dark = 0;                  % Dark current signal [electrons per px per second]
IMsat = 10000;             % Saturation level [electrons]
CF = 120;                  % Conversion Factor [microV/electron]
Bias = 0;                  % The overall Bias Voltage, converted into [e-]
                             % Bias [e] = 10^6 * Bias[V] / CF [microV/e]
XS = 1.4;                  % Excess Noise [no unit] (1.4 for EMCCD)
RON = 0.5;                 % Read-Out Noise [electrons rms per px per frame]
Digit = 1;                 % Gain of the ADC [electron/ADU]
Sensor_maps = 0;           % Filename where are stored the PRNU, column bias
                             and defective pixels maps. 0 is not used

% end of file

```

11.2 PSF computation

The routine responsible for PSF computation is “Makepsf_PSIM”.

For example, the PSF resulting from the following parameters:

- a wavefront **WFmask** defined in microns on a support of 240x240 pixels (physical dimension 8 meters)
- the pupil mask **pupmask** defined on the same support (ones inside the pupil and zeros outside)
- an imaging wavelength of 2.2 microns
- a pixel scale of 10 mas/pixel
- a detector of 100x100 pixels
- a Gaussian source of 2 pixels in FWHM
- at least 4x4 image points per detector pixel

is computed by the following command:

```
[psf, wl, C] = makepsf_PSIM(WFmask, pupmask, 2.2, 8, 0.01, 100, 2*0.01, 4);
```

Where **psf** is the output detector image, **wl** the effective imaging wavelength and **C** the actual number of image points per detector pixel.

11.3 Detector read-out simulation

The simulation of a detector output from a “perfect” optical image is done by the routine “ImSensor_PSIM”. The command line to call it is:

	<p>Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 73 of 74
--	---	--

```
IMout = ImSensor_PSIM(IMin, QE, PSF, Dark, Bias, CF, IMsat, XS, RON,
Exptime, Digit, Sensor_maps, C, IMdisp, IMout_type, noise_quick);
```

Where **IMin** is the input image, **IMout** the output one, and the other parameters are described in the §11.1. **IMdisp** should be set to 1 for a display of the output image, 0 otherwise. **IMout_type** can be either Voltages (**V**) or ADUs (**ADU**). **Noise_quick** has to be set to 1 for a faster but less rigorous computation of the excess noise.

11.4 WF sensing

The WF sensing is done thanks to the routine “SHWFS_PSIM” by the simple command line:

```
[CCDall, xyslopes, wl, C, xyvalid, xyfullill, xyf_of_v, Tag_params, CCDall2]
= SHWFS_PSIM(WF, 'SHWFS_input', Coeffs, WFdisp, Tag_params);
```

But the WFS parameters have to be properly set in a file save under the name “SHWFS_input.m” (see §11.1). **WF** is the wavefront to analyze, in microns. **Coeffs** are the mis-registration parameters (see example in §11.6). **WFdisp** should be set to 1 for a display of the WFS outputs, 0 otherwise. **Tag_params** can be set to 0 if the routine is not used a large amount of times like for an IM computation.

CCDall will be the output image of the WFS, **xyslopes** the measured slopes, **wl** the effective wavelength used, **C** the actual number of image points per detector pixel, **xyvalid** the list of valid sub-apertures, **xyfullill** the list of fully illuminated sub-apertures, **xyf_of_v** the fully illuminated among the valid, and **CCDall2** the output image of the WFS including image pre-processing like thresholding.

11.5 Slopes measurement

The routine responsible for slopes measurement is “Slopes_PSIM”. On the example of the VLT active optics image to analyse (§6.8.3), the command line would be:

```
[xyslopes, CCDall2, CCDint] = Slopes_PSIM(CCD_image, xyvalid, 24, 22,
'wCoG', weighting_map, 1);
```

where **CCD_image** is the existing WFS image, **xyvalid** the list of valid sub-apertures (can be obtained from a dry run of the WFS algorithm “SHWFS_PSIM” with the existing WFS parameters), 24 the number of sub-apertures across a diameter, 22 the number of pixels per sub-aperture, **wCoG** the centroiding method, **weighting_map** was obtained in that case by the averaging of many WFS images. The last ‘1’ is for a display of the slopes computation output.

xyslopes and **CCDall2** have been described in the §11.4, and **CCDint** is the map of intensity per sub-apertures.

	<p align="center">Pseudo-synthetic Interaction Matrix simulations</p>	Doc: VLT-TRE-ESO-22000-5114 Issue 1 Date 03.12.2012 Page 74 of 74
--	--	--

11.6 IM recording

The IM recording is performed by the routine “makeIM_PSIM”. For computing an IM using a WFS which parameters are described in the file “SHWFS_input.m”, and 1170 modes saved as individual .mat files (with names **Mode_[n].mat**, n from 1 to 1170, and variable name **DM**) in the folder **DM_IF_960/**, with an amplitude of 1/200, and mis-registration coefficients of:

- -5% of a sub-aperture in X shift,
- +4% of a sub-aperture in Y shift,
- 0.3 degrees in rotation,
- +2% of X stretch,
- -1% of Y stretch,

the command line is:

```
[IM, pupmask, wl, C, xyvalid, xyfulll1l1, xyf_of_v] =
makeIM_PSIM('SHWFS_input', 960, IF, 1170, 1/200, [-5 4 0.3 +2 -1]);
```

IM is the output Interaction Matrix, and the other outputs have been described previously.