

**CURVATURE SENSING FOR ADAPTIVE OPTICS:
A COMPUTER SIMULATION**

by
Nicolas Roddier

A Thesis Submitted to the Faculty of the
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
In Partial Fulfillment of the Requirements
For the Degree of
MASTER OF SCIENCE
WITH A MAJOR IN ELECTRICAL ENGINEERING
In the Graduate College
THE UNIVERSITY OF ARIZONA

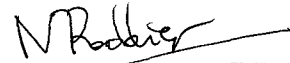
1 9 8 9

STATEMENT BY AUTHOR

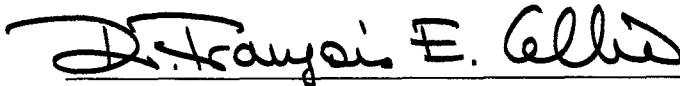
This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgement the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: _____

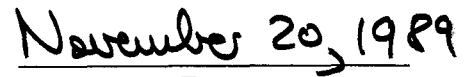
**APPROVAL BY THESIS DIRECTOR**

This thesis has been approved on the date shown below:



F. E. Cellier

Associate Professor of
Electrical and Computer Engineering



Date

ACKNOWLEDGMENTS

I would like to express my gratitude to my father Dr. François Roddier not only as my father, but for his permanent guidance throughout this project in spite of the far distance between Arizona and Hawaii.

I would like to also express my constant feeling of thankfulness to Dr. François Cellier, my major advisor, for his assistance and support throughout this research, but in particular for his help in turning this thesis report into a clear and easily understandable document.

I wish to thank Dr. John Rather from Kaman Aerospace Corporation for having given the necessary financial support to enable me to work on this exciting research project, and to promote curvature sensing as a technique for adaptive optics. In addition, I more than appreciated the way he was truly confident in this work and allowed me to proceed with my research in any direction that I considered to look promising.

Acknowledged should also be the interest of the thesis committee members Dr. John Reagan and Dr. Hal Tharp from this Electrical Engineering department in my research.

Finally, I want to thank my mother Dr. Claude Roddier for very helpful discussions relating to every aspect of the project, and in particular to those areas where we could compare the results from this thesis with her own ways to dealing with curvature sensing.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS	7
LIST OF TABLES	10
ABSTRACT	13
1. INTRODUCTION	14
1.1. Adaptive Optics	14
1.2. Effects of the atmosphere and definitions	14
1.2.a. Effects of the atmosphere	14
1.2.b. Principle of correction	15
1.2.c. Other involved parameters	16
1.3. Current development in Adaptive Optics	17
1.3.a. Estimating the incoming wavefront: estimation of slopes	19
1.3.b. Algorithms of reconstruction	20
1.3.c. Compensation using a piston position active mirror	22
1.4. Suggested new technique	23
2. CURVATURE SENSING AND COMPENSATION	26
2.1. Curvature sensing	26
2.1.a. Principle of curvature sensing	26
2.1.b. Mathematical expression of the curvature signal	27
2.1.c. Practical realization	27
2.1.d. Comparison of sensitivity and photon noise for classical and curvature sensing	30
2.2. Active mirrors	31

2.2.a. Membrane mirror	32
2.2.b. Bimorph mirror	34
3. OPEN LOOP SYSTEM SIMULATION	37
3.1. Simulation of wavefronts	37
3.1.a. Definition of Zernike polynomials	37
3.1.b. Simulation of simple wavefronts	42
3.1.c. Atmospheric wavefronts	42
3.1.d. Simulation of atmospheric wavefronts that obey Kolmogoroff's law	44
3.2. Simulation of the curvature sensor	51
3.2.a. Simulation of the ideal sensor characteristics	52
3.2.b. Some examples of such curvature signals for simple wavefronts	56
3.2.c. Simulation of the sensor by ray tracing method	58
3.2.d. Simulation of the sensor including diffraction effects	62
3.3. Reconstruction algorithms	64
3.3.a. General algorithms for elliptic equations	65
3.3.b. Noise amplification and scalability for the reconstruction algorithm	69
3.3.c. How to handle correctly the boundary conditions	77
3.3.d. Extension of NPOIS to include central obscuration	88
3.4. Open loop simulation	94
3.4.a. Study of non-linearities	94
3.4.b. Study of photon noise	100
3.4.c. Study of diffraction noise	102
3.5. Chapter summary	111

4. CLOSED LOOP SYSTEM SIMULATION	113
4.1. Closed loop system	113
4.2. Simulation of the seven element system	118
4.2.a. Our first system	119
4.2.b. Simulation for single aberrations	121
4.2.c. Simulation for linear combinations of aberrations	130
4.3. Comments and explanations for the seven element system	137
4.4. Simulation of the 13 element system	140
4.4.a. Our second system	141
4.4.b. Simulation for single aberrations	143
4.4.c. Simulation for linear combinations of aberrations	152
4.4.d. Simulation for random atmospheric wavefronts	156
4.5. Simulation of correction for images	157
4.6. Chapter summary	160
CONCLUSION	163
REFERENCES	164

LIST OF ILLUSTRATIONS

(1.3.1) A typical adaptive optics system	19
(1.3.2) A typical piston position active mirror	23
(1.4.1) The curvature-sensing membrane-mirror system	25
(2.1.1) Theoretical setup of the curvature wavefront sensor	26
(2.1.2) Practical setup with bi-refrigent lens and separation in space	28
(2.1.3) Practical setup with bi-refrigent lens and separation in time	29
(2.1.4) Practical setup with focal variable mirror	30
(2.2.1) A membrane mirror assembly	33
(2.2.1) A bimorph mirror assembly	35
(3.1.1a) Zernike polynomials	40
(3.1.1b) Zernike polynomials	41
(3.1.2) Covariance matrix C	48
(3.1.3) Diagonal matrix S	49
(3.1.4) Unitary matrix U	50
(3.2.1) A typical sensor configuration	53
(3.2.2) A typical edge detector	54
(3.2.3) A pure tilt curvature signal	56
(3.2.4) A pure defocus curvature signal	57
(3.2.5) A pure astigmatism curvature signal	58
(3.2.6) Ray tracing showing the effect of a perturbation in the \vec{x} direction	60
(3.3.1) Error propagation for several cases	74
(3.3.2) Example of reconstruction from random Laplacians	75

(3.3.3) Example of reconstruction from random Laplacians after removal of global tilt	76
(3.3.4) Typical curvature signal from an experiment	79
(3.3.5) Discretization along a radius	81
(3.3.6) Example of S-, B-, C-, and O-points near the pupil edge	82
(3.3.7) Simple case of the edge of a square pupil	83
(3.3.8) Example of two “support points” B_1 and B_2	84
(3.3.9) Reconstruction of the errors on the wavefront at the output of an achromat	88
(3.3.10) Example of S-, B-, C-, and O-points for the case with two boundary signals	90
(3.3.11) A typical out-of-focus image from the CFH Telescope	91
(3.3.12) Curvature signal from the CFH Telescope	92
(3.4.1a) Linearity for tilt	98
(3.4.1b) Linearity for defocus	98
(3.4.1c) Linearity for astigmatism	99
(3.4.2) Residual error when photon noise is present	102
(3.4.3) Typical image due to a single wavefront tilt	105
(3.4.4a) Cross-section through the left image	106
(3.4.4b) Cross-section through the right image	106
(3.4.5) Curvature signal for the tilt	107
(3.4.6) Cross-section of the curvature signal for the tilt	108
(3.4.7) reconstruction of the wavefront	109
(3.4.8) Astigmatism image	110
(3.4.9) Astigmatism curvature signal	110
(3.4.10) Astigmatism reconstruction	111

(4.1.1) Membrane response to a single edge electrode	116
(4.1.2) The new membrane configuration	117
(4.1.3) Discretization for solving the Poisson equation	118
(4.2.1) The seven element sensor	120
(4.2.2) The seven element membrane mirror	120
(4.2.3) Evolution of residual error for tilt initial wavefront without smoothing	123
(4.2.4) Evolution of residual error for tilt initial wavefront with 50% smoothing	124
(4.2.5) Evolution of residual error for defocus initial wavefront	126
(4.2.6) Evolution of residual error for astigmatism initial wavefront	127
(4.2.7) Evolution of residual error for coma initial wavefront	130
(4.2.8) Evolution of residual error for tilt and defocus aberrations together ...	132
(4.2.9) Evolution for tilt, defocus, astigmatism, and coma	134
(4.2.10) Error as a function of γ	136
(4.4.1) The 13 element sensor	142
(4.4.2) The 13 element membrane mirror	142
(4.4.3) Sensor and mirror without rotation	146
(4.4.4) Sensor and mirror with 30° rotation	146
(4.5.1) Uncorrected image for 21 samples	159
(4.5.2) Corrected image for 21 samples	159
(4.5.3) Uncorrected image for 21 samples	160
(4.5.4) Corrected image for 21 samples	160

LIST OF TABLES

(3.1.1) Zernike polynomials with particular order and normalization	39
(3.1.2) Mean Square coefficients for each of the Zernike polynomials for the atmosphere, where D is the pupil diameter, and r_o Fried's parameter ...	43
(3.1.3) Amplitude coefficients for each of the Zernike polynomials for the atmosphere, where D is the pupil diameter, and r_o Fried's parameter ...	44
(3.1.4) Degree and frequency for the first 14 Zernike polynomials	45
(3.3.1) Relation between the number of points across the pupil N , and the total number of points over the pupil n	71
(3.3.2) Error propagation coefficient from the convolution method	72
(3.3.3) Error propagation coefficient from the Monte Carlo method	73
(3.4.1a) Linearity for tilt	96
(3.4.1b) Linearity for defocus	97
(3.4.1c) Linearity for astigmatism	97
(3.4.2) Residual error when photon noise is present	101
(4.2.1) Residual error expressed in percents of the disturbed input tilt wavefront	122
(4.2.2) Residual error expressed in percents of the disturbed input defocus wavefront	125
(4.2.3) Residual error expressed in percents of the disturbed input astigmatism wavefront	126
(4.2.4) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.	129

(4.2.5) Residual r.m.s. error expressed in percents of the r.m.s. of the disturbed input wavefront. Tilts and defocus aberrations	131
(4.2.6) Residual error for tilt, defocus, and astigmatism combined	133
(4.2.7) Residual error for tilt, defocus, astigmatism, and coma in atmospheric proportion	133
(4.2.8) Residual error as a function of γ , ratio of coma amplitude over tilt amplitude	135
(4.2.9) Error for the mixture of the first five aberrations	137
(4.4.1) Some more Zernike polynomials	144
(4.4.2) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.	147
(4.4.3) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.	149
(4.4.4) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.	150
(4.4.5) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.	151
(4.4.6) Residual r.m.s. error expressed in percents of the r.m.s. of the disturbed input wavefront. Tilts and defocus aberrations	153
(4.4.7) Residual error for tilt, defocus, and astigmatism aberrations combined	153
(4.4.8) Residual error for tilt, defocus, astigmatism, and non-zero curvature coma aberrations	154
(4.4.9) Residual error for tilt, defocus, astigmatism, and all coma term aberrations	154
(4.4.10) Residual error for tilt, defocus, astigmatism, coma, and	

spherical aberrations	155
(4.4.11) Residual error for tilt, defocus, astigmatism, coma, spherical, and 5 th order astigmatism	155
(4.4.12) Residual error for the first five rows in the Zernike table. Hence, up to Z_{15}	155
(4.4.13) Correction for random atmospheric wavefronts	157

ABSTRACT

This thesis describes computer simulations of a new wavefront sensing technique for Adaptive Optics based on local wavefront curvature measurements, along with edge slope measurements. The output signal from the curvature sensor can be directly applied to the electrodes of a bimorph or membrane mirror. The mirror is used as an analog device to solve the Poisson Equation, providing a fast real time compensation for atmospheric disturbances.

The open loop characteristics of the system are presented. The ideal response is analyzed, and side effects such as non-linearity, photon and diffraction noises are discussed.

Closed loop simulations are presented thereafter. A seven actuator system showed a few unstable modes. A 13 actuator system with proper filtering corrects the atmospheric perturbations.

To simulate atmospheric distorted wavefronts, an algorithm based on spectral decomposition of the Zernike covariance matrix was derived.

This sensor can also be used to test large telescope mirrors using a modified program that solves the Poisson Equation with Neumann boundary conditions.

CHAPTER 1

INTRODUCTION

1.1. Adaptive Optics.

A principal cause of bad transmission of general electromagnetic waves or particular optical ones is the distortion due to atmospheric disturbances. The received wavefront phase has been affected. Therefore, a way to improve transmissions, or imaging in the optical case, is to restore the original phase of the wavefront to remove the distortions. The adaptive optics concept consists of making a real time estimate of the wavefront phase, have a deformable active mirror match this estimate, whereby the wavefront will automatically be corrected when light reflects on that mirror.

1.2. Effects of the atmosphere and definitions.

Atmospheric turbulence produces perturbations on both the amplitude and the phase of incoming wavefronts. Only phase perturbations can be corrected by putting a deformable mirror in a plane conjugate to the telescope entrance aperture. Fortunately the image degradation due to uncorrected amplitude fluctuations remains small and, image quality can be considerably improved by correcting phase errors only.

1.2.a. Effects of the atmosphere.

As we just said, the atmosphere mainly produces variation of phases on the wavefront. Because of those perturbations, deformations and corrugations on the

wavefront surface are produced, which involve distortions and blur on the image. The deformation $z(x, y)$ or $z(\vec{r})$ of the wavefront surface will be compensated by the adaptive optics scheme. The relationship between the phase and the surface is linear:

$$z(\vec{r}) = \frac{\lambda}{2\pi} \phi(\vec{r}) \quad (1.2.1)$$

where λ is the wavelength of the incoming wave.

With respect to the observed image, the expression is not so simple. The irradiance distribution $I(x, y)$ or $I(\vec{r})$ of the image is given by (Gaskill, 1978):

$$I(x, y) = \left| \mathcal{F}\mathcal{F} \left\{ P(x, y) e^{j\phi'(x, y)} \right\} \right|^2 \quad (1.2.2)$$

where $\mathcal{F}\mathcal{F}$ represent the two dimensional Fourier transform operator, and $P(x, y)$ is a gate function the value of which is 1 for any point (x, y) inside the pupil, and 0 for any point outside the pupil. $\phi'(x, y)$ is the error phase term that includes not only the $\phi(x, y)$ atmospheric phase error, but also the errors produced by the optical aberrations of the telescope.

1.2.b. Principle of correction.

The goal is to compensate for the surface deformation on $z(\vec{r})$. This can be achieved by producing a similar opposite deformation on an adaptive mirror.

Therefore, we really must know more about those $z(\vec{r})$ deformations. Mathematically, z can be described as a random Gaussian process. In order to build the system, we essentially need to know the amplitude and the scale of the deformations, which means the variance and the covariance of the process. The image is not changed by adding constant terms on $z(\vec{r})$, and thus the relative deformations we are interested in are best described by the structure function

$$D_z(\vec{\rho}) = \langle \left| z(\vec{r} + \vec{\rho}) - z(\vec{r}) \right|^2 \rangle \quad (1.2.3)$$

where the notation “ $\langle \dots \rangle$ ” stands for the average value.

For a good approximation (Rodier, 1981), $D_z(\vec{\rho})$ follows Kolmogoroff’s law (Kolmogoroff, 1941)

$$D_z(\vec{\rho}) = 6.88(\lambda/2\pi)^2(\rho/r_o)^{5/3} \quad (1.2.4)$$

where $\rho = |\vec{\rho}|$ and r_o is the Fried parameter (Fried, 1965), or turbulence strength parameter, which characterizes both the amplitude and the scale of the perturbations. r_o depends upon the wavelength we are looking at as $\lambda^{6/5}$. The larger r_o is, the better the weather is, thus, the better the seeing is. Typical values for r_o for good astronomical sites are:

$$r_o = 80\text{cm for } \lambda = 2.2\mu\text{m (infrared light)}$$

$$r_o = 12\text{cm for } \lambda = 0.5\mu\text{m (visible light)}$$

According to (1.2.4), the standard wavefront error, the square root of the deformation, over a distance $\rho = r_o$ is 0.42λ . Deformations over smaller distances can be considered “small” compared to λ and need not be corrected. Therefore, the Fried parameter r_o determines the smallest scale of the perturbations to be applied on the adaptive mirror.

1.2.c. Other involved parameters.

An important fact is that the wavefront perturbation is nearly wavelength independent. The surface change z can be approximated as being the total optical path fluctuation due to the atmosphere. Its wavelength dependence therefore is directly related to the refractive index of the air which varies very little with wavelength. However, the dependence of r_o upon λ cannot be ignored. The structure function D_z , as described earlier, does not depend upon the wavelength. It is then clear that r_o varies as $\lambda^{6/5}$.

Another important fact is the time evolution of the wavefront. The atmosphere is composed of inhomogeneities that produce deformations on the wavefront. These inhomogeneities not only vary with time, but also move because of wind across the optical beam. Nonetheless the so called Taylor approximation states that their life time is long compared to their transit time, and can be considered infinite. Because atmospheric turbulence is distributed over layers at different altitudes, the wavefront perturbation both propagates in an average direction, and changes more or less rapidly according to the dispersion of the wind velocities.

The last important parameter is the isoplanatic patch angle Ω . This is the angle over which the wavefront perturbation remains approximately the same. If a perturbation was just at the end of the telescope, any light coming from any source would just go through the same perturbation. But if the perturbation is built over several layers at various altitudes, two sources separated by an angle θ encounter different perturbations. This parameter is important when the sensing star, or star on which the perturbation is estimated, is different from the observed star. If a good correction for the observation is desired, the angle θ between the two stars must therefore be less than Ω .

1.3. Current development in Adaptive Optics.

Control theory and active systems have been used for a long time in many engineering fields, such as mechanics or electronics. However, it is not the case in optical sciences. People have tried to build elements, such as lenses and mirrors, as precise as possible, which are supposed to change as little as possible with external perturbations. The advantage of such optical systems is that, once tuned up and adjusted, they really worked fine. But, on the other hand, they are totally unable to take care of any external change or disturbance.

In astronomy, or any other field where the light has to go through air, a limitation of the systems was due to air turbulence. At that point, it was realized that it would be very convenient to use adaptive control there too.

The first adaptive optics systems were built a little more than a decade ago, and since then, there have been improvements in the technology and applied theory, but no real change in fundamental operations.

Any adaptive optics system works exactly like any adaptive control system. There is a first stage that consists of estimating the disturbance or perturbation that has to be corrected. A second stage consists of processing this estimate, and the last stage consists of applying a signal in the control loop that will compensate for the disturbance. The compensation is done with a deformable active mirror. Figure (1.3.1) represents a typical adaptive optics system (Goad, 1989).

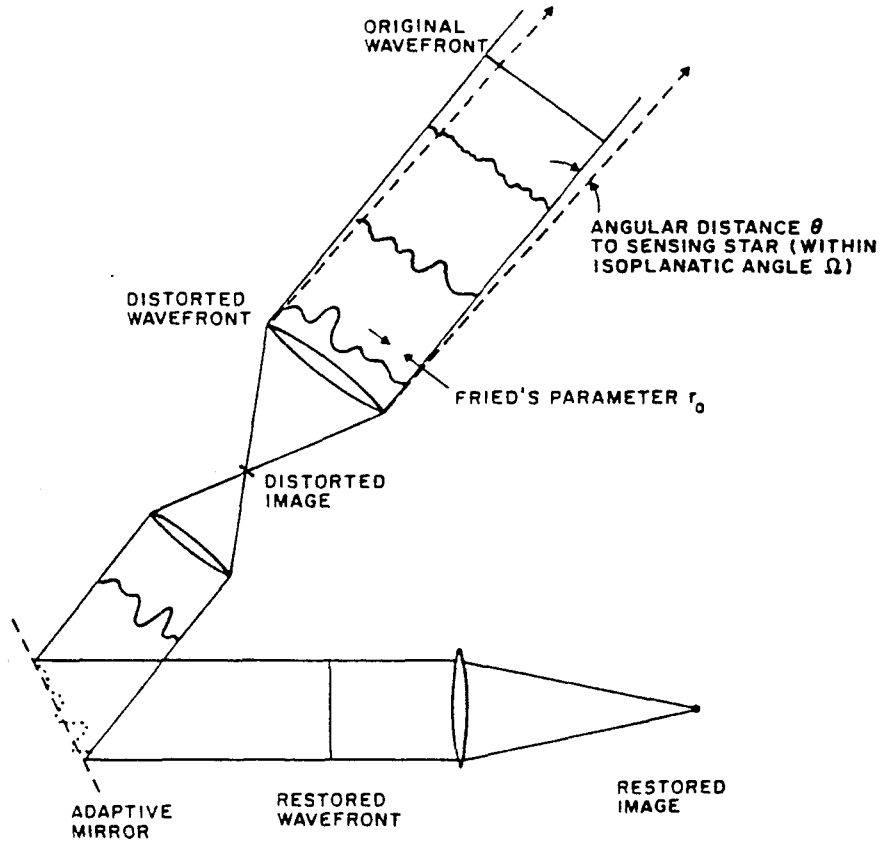


Figure (1.3.1) A typical adaptive optics system.

1.3.a. Estimating the incoming wavefront: estimation of slopes.

What is important in a distorted optical wavefront is the phase difference between the different positions on that front. Any absolute value of the phase is irrelevant for imaging. Therefore, people have looked toward detecting phase differences for possible correction.

A wavefront is a moving surface, say in a two-dimensional space. Therefore, estimating the phase difference requires estimation of the slopes along two distinct directions. To have an optimal decoupling and as much information as possible

from the slope measurements, the two directions are chosen to be orthogonal to each other. There currently exist several kinds of slope sensors. They can be split in two categories.

The first solution consists of putting an array of lenslets in the optical beam. Each lenslet corresponds to a zone where two values of the slopes will be known. If there was no local distortion of the wavefront, the light would focus on the focus point of the lenslet, but on the other hand, if there is a small wavefront distortion, the light will focus on a point distinct from the lenslet focus point. The distance and position of this point gives the information of the local slopes of the wavefront. This kind of sensor is called a Shack-Hartmann sensor (Hardy, 1978). The practical realization of the lenslet array and the detector varies with the materials and technology used, but they all use this same Hartmann theory.

The second solution consists of using interferometry theory (Koliopoulos, 1980). The wavefront is sheared along the two directions by a small amount. Interference patterns are produced due to local changes of phase. These changes of phase can then be directly related to the local slopes of the incoming wavefront. Such a sensor has been developed at Itek (Hardy, 1976).

1.3.b. Algorithms of reconstruction.

After the estimation stage, estimation of local slopes in two orthogonal directions, the wavefront must be reconstructed. Much work has been done for that purpose, and several algorithms have been derived. Another type of research was devoted to theoretically studying how well the slope estimates, followed by reconstruction, matched the true wavefront, and to analyzing the properties and imperfections of the procedure as a whole.

With respect to the reconstruction algorithms, research has proceeded in two different directions. The first solution consists of directly solving for the phase values from those known phase differences. The second solution consists of first computing the Laplacian array from the slope data, and then solving the resulting Poisson equation (Hudgin, 1977).

Knowing the slopes or phase differences over a discrete array actually means that there are finite difference equations to deal with. Such problems are well known in applied mathematics, and can be rewritten in terms of n equations with n unknowns, where n is the number of to-be-found phases for which two local slopes are known. This linear system of equations can be solved, usually with matrix operations.

The other approach consists of computing the Laplacian at every solution point from the four adjacent slopes, according to the discrete Laplacian definition. Then, the phase values are obtained by solving the Poisson equation. This is usually performed by either of three methods:

- (1) numerically through an iterative algorithm, e.g. of the Jacobi type, speeded up with some successive over-relaxation technique,
- (2) digitally by writing a set of difference equations and solving the corresponding linear matrix equation, or finally
- (3) analogically by using some resistive plates onto which currents proportional to Laplacians are injected at each point. The resulting voltage distribution indeed follows Poisson's equation (a so-called Beuken model).

All of those algorithms must, of course, be optimized for each particular type of situation.

Along with the research related to the algorithms themselves, theories have been developed around them. The main subjects of interest have been noise sensitivity, scalability to large array size, optimization, and least squares fit.

Herrman (1981) worked on the sensitivity of the system to noise created by sampling, aliasing, and cross-coupling between several frequency modes in the atmosphere. Fried (1977), Hudgin (1977), and Noll (1978) looked at the scalability of such systems with different hypotheses concerning the geometry of the pupil, and the type of sampling. In all these situations, they found that the propagation error ratio σ_{out}/σ_{in} increases as $\text{Log}(n)$, where n is the number of solution points. Fried (1977) and Herrman (1980) worked on the error produced in the least squares fit of the reconstruction. Wallner (1983) compared several configurations for the sensors in x and y directions to try to find an optimal setup.

1.3.c. Compensation using a piston position active mirror.

The last element that is missing in the complete adaptive optics loop as described so far is the active element. Once the reconstruction algorithm has produced an estimate of the distorted wavefront, this estimate has to be applied to a flexible mirror that will match it as accurately as possible.

This estimate is an estimate of the phase of the wavefront which means it corresponds to a displacement on the mirror surface. This displacement is produced by a push-pull piston used as an actuator. This is why the active mirrors most commonly used are called “piston-position mirrors”.

As technology has evolved, so has the construction of those mirrors. They used to be flexible mirrors with several piezo-electric ceramic cylinders as command actuators. Currently, a more monolithic construction is preferred. Such mirrors were e.g. developed at Itek (Hudgin, 1975). Basically, the mirror consists of an

aluminized piece of flexible glass, under which a single piece of piezo-electric ceramic is glued. Command electrodes produce non-uniform electric fields. By applying voltages to those electrodes, the mirror surface is being displaced. By controlling this displacement properly, we can reconstruct the wavefront.

The current development and construction have reached a point where commercialized versions of such mirrors contain up to 500 actuators, their maximum displacement is about $\pm 7\mu m$ from equilibrium which corresponds to a correction of 25 wave perturbations, their resonant frequencies are in the $10kHz$ range, and their command voltages are about $\pm 3000V$ for maximum displacement. Figure (1.3.2) represents one of those mirrors, commercialized by Itek (Goad, 1989).

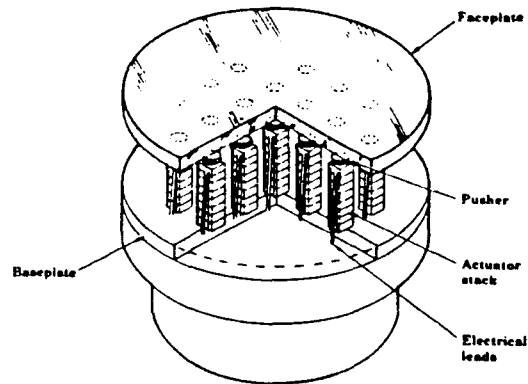


Figure (1.3.2) A typical piston position active mirror.

1.4. Suggested new technique.

The currently used technology in adaptive optics as it has been outlined in the previous section has several drawbacks which we shall describe in the sequel. A suggested new technique provides improvements to the classical approach without sacrificing many of its advantages.

Currently used wavefront sensors measure local wavefront tilts or wavefront first derivatives along two orthogonal directions. The newly suggested sensor measures the total local wavefront curvature or the Laplacian of the wavefront surface $z(x, y)$ together with the wavefront tilts at the edges. Those tilts provide boundary conditions to solve the Poisson Equation obtained with the Laplacians, to obtain an estimate of the wavefront surface for reconstruction. This method presents several advantages.

The setup of this sensor is much simpler. It is more difficult to measure a vector field, as with a Hartmann sensor, than a scalar field for Laplacians. We now only need half the number of detectors.

One of the most successful algorithms to reconstruct wavefronts detected with a Hartmann sensor is the so-called Hudgin algorithm which consists of computing the Laplacians from the tilts, and thereafter solving the Poisson equation (Hudgin, 1977). The proposed sensor provides the Laplacians directly.

For atmospheric distorted wavefronts, local curvatures are nearly statistically independent. Curvature sensing measurements are therefore expected to be more efficient than tilt measurements.

In adaptive optics applications, membrane or bimorph mirrors can be used as analog devices to solve the Poisson equation. In this case, the signal from a curvature sensor can be fed directly to the mirror without computation, providing an extremely fast control loop. Figure (1.4.1) represents a block diagram of the complete new suggested system.

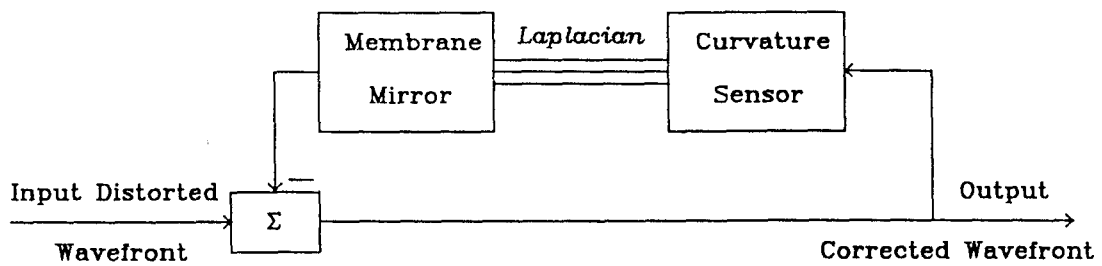


Figure (1.4.1) The curvature-sensing membrane-mirror system.

CHAPTER 2

CURVATURE SENSING AND COMPENSATION

2.1. Curvature sensing.

2.1.a. Principle of curvature sensing.

To compensate for a distorted wavefront using adaptive optics, we surely must first estimate it. Nowadays, most of the optical atmospheric sensors are Hartmann type sensors. They look at a point source, such as a star, and estimate the slopes of the incoming wavefront. Those slopes are estimated along both the x and the y axes over an array of points. Another way to get an estimate is to use a curvature sensor which does not estimate the slopes, but instead estimates the Laplacians at every point of the wavefront. This sensor also gives the normal derivatives at the edges of the pupil. They are required to fully determine the wavefront.

Theoretically, this sensor works physically in the following way:

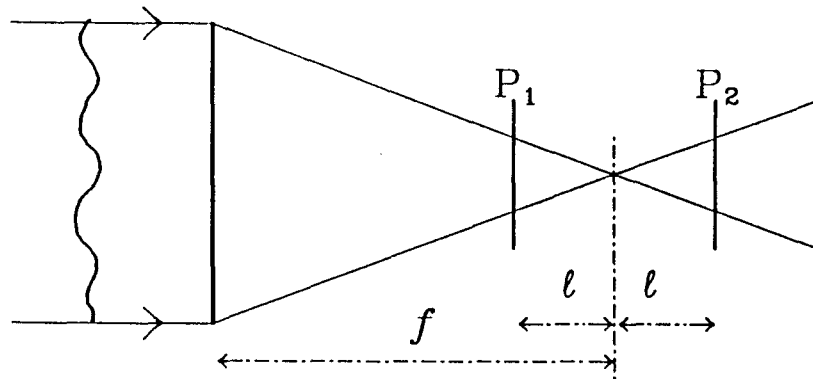


Figure (2.1.1) Theoretical setup of the curvature wavefront sensor.

2.1.b. Mathematical expression of the curvature signal.

The sensor measures the two irradiance distributions I_1 and I_2 on each side of the focal plane. From Fresnel's diffraction theory, it can be shown that (Roddiier, 1987):

$$\frac{I_1(\vec{r}) - I_2(-\vec{r})}{I_1(\vec{r}) + I_2(-\vec{r})} = C_0 \cdot \left[\nabla^2 z(c_1 \vec{r}) - \frac{\partial}{\partial \vec{n}} z(c_1 \vec{r}) \delta_c \right] \quad (2.1.1)$$

The $\nabla^2 z$ term represents the Laplacians of the wavefront surface we are looking for, and the circular Dirac distribution δ_c represents the outward-pointing normal derivatives on the edge of the signal. The constants C_0 and c_1 depend upon the optical setup: focal length f , and distance ℓ of the detectors from the focal plane.

$$C_0 = \frac{f(f - \ell)}{\ell} \quad c_1 = \frac{f}{\ell} \quad (2.1.2)$$

The output signal of this sensor indeed provides enough information to reconstruct an estimate of the wavefront.

The procedure to use such a sensor is to first know both the irradiances I_1 and I_2 , then subtract one from the other, and divide by the sum to normalize. At that point, only the reconstruction of the wavefront by solving the Poisson equation is needed.

2.1.c. Practical realization.

There are several ways to practically realize such a sensor. It is of course not possible to directly use the setup suggested in figure (2.1.1) since any photodetector, such as a *CCD* for example is not transparent, and therefore *CCD*₁ as positioned would prevent all light from going on to lens L and *CCD*₂.

The two possible solution suggested so far (Roddiier, 1988) differ from each other a lot. The first one involves polarized light and a special lens used for L , the other method involves a small focal-variable mirror used for L .

The goal of the sensor is to record two extra-focal images simultaneously. This can be achieved by using a special lens for L , a bi-refracting achromate similar to those used in Dyson interferometers (Dyson, 1957). In this setup, we no longer need CCD_1 . The camera or CCD used will be placed at the position CCD_2 . For one light polarization, the focal length is $f/2$, half that of the telescope, and the camera records the illumination originally on CCD_2 . For the other polarization, the focal length is equal to $\ell/2$, and the image originally on CCD_1 is reimaged on CCD_2 with the required 180° rotation. The two new images can be separated through space, by decentering the bi-refracting lens laterally. Thereby, the two images will be projected beside each other onto the same plane.

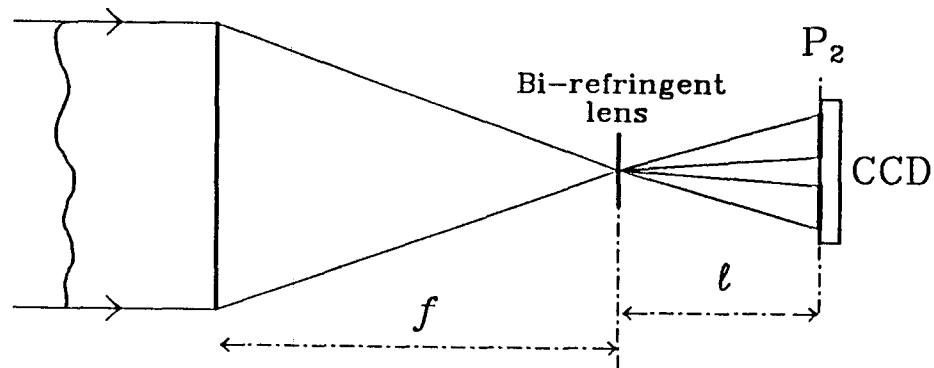


Figure (2.1.2) Practical setup with bi-refracting lens and separation in space.

Alternatively, the two images can be separated in time. In that case, a rotating polarizer has to be used in the main beam, and the two images will appear sequentially on the detector, producing a modulated output. The required curvature signal can be reconstructed through synchronous demodulation of this signal.

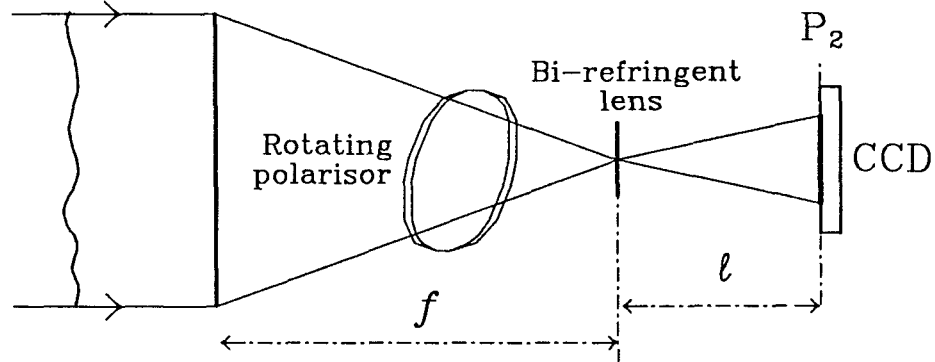


Figure (2.1.3) Practical setup with bi-refractive lens and separation in time.

The other approach consists of using a small focal variable mirror with focal length f_v instead of lens L . Again, only one detector or camera will be needed. When f_v is chosen equal to $f/2$, the image originally on CCD_2 will be in focus at the location the camera. When f_v is chosen equal to $l/2$, the image originally on CCD_1 will be focused at the location of the camera. Therefore, by toggling the variable focal length periodically between $f/2$ and $l/2$, the two images will alternatively appear on the detector. Again, a demodulation scheme can be used.

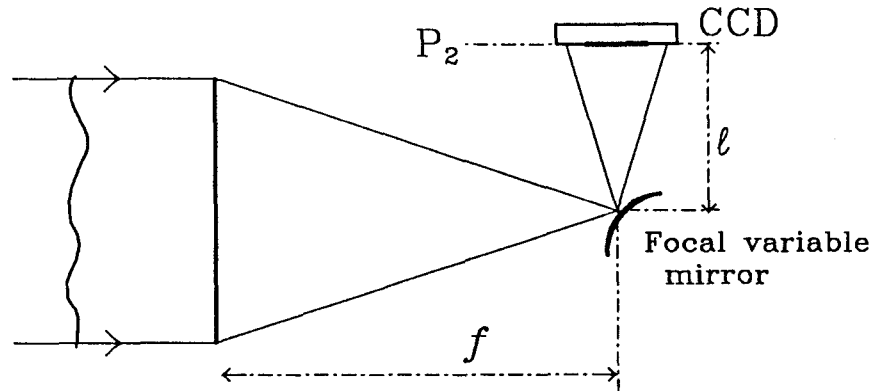


Figure (2.1.4) Practical setup with focal variable mirror.

Both methods have their advantages and disadvantages. Using the bi-refrigrant approach, it is possible to observe both images simultaneously and continuously. Unfortunately, this approach has some drawbacks. First, only half the light goes to each image which can be a problem for very weak sources. Also the bi-refrigrant lenses are rarely used. They must be custom made and are therefore very expensive. The advantage of the mirror approach is that it is much easier to use, and works with classical components. Unfortunately, no continuous imaging is possible.

2.1.d. Comparison of sensitivity and photon noise for classical and curvature sensing.

Let us compare the sensitivity of our method to that of a Shack-Hartmann sensor. To resolve wavefront fluctuations of scale r_o , a Shack-Hartmann sensor will use a lenslet array with lenslet size r_o . The angular width of the image produced by a single lenslet is λ/r_o . Hence, the standard deviation of a slope measurement obtained from a single photon position is of the order of $\lambda/2r_o$. For n photons the standard deviation is \sqrt{n} times smaller. Most of the reconstruction processes use

the so-called Hudgin's algorithm that actually computes the Laplacians from the slope measurements. The Laplacian is computed by taking the difference between adjacent slopes in the x direction plus the difference between adjacent slopes in the y direction. The result is divided by the sampling interval r_o . The standard deviation σ_s of the Laplacian is therefore

$$\sigma_s = \frac{\lambda}{r_o^2 \sqrt{n}} \quad (2.1.3)$$

We now can perform a similar analysis for our curvature sensing method. We measure the illumination over an area of width r_o in the pupil plane. The standard deviation on our measurement is $1/\sqrt{n}$. Assuming $\ell \ll f$ in the optical setup, the standard deviation σ_c on the Laplacian will be

$$\sigma_c = \frac{\ell}{f^2 \sqrt{n}} \quad (2.1.4)$$

The maximum sensitivity is achieved when $\ell = \lambda f^2 / r_o^2$ in which case it is clear that our σ_c is exactly the same as σ_s , the one we found for a Shack-Hartmann sensor.

The sensitivities of the two methods are therefore identical. Nonetheless, in our method, sensitivity can be easily balanced against spatial resolution on the pupil by just modifying ℓ , that is just moving the photodetectors. However, with a Shack-Hartmann sensor, the same operation requires changing the lenslet array.

2.2. Active Mirrors

The curvature sensor, as described in the previous section, delivers a signal that can be processed to solve the Poisson equation. A solution for our adaptive optics system would be to numerically solve this equation on a digital computer, thereby obtaining an estimate of the phase values of the wavefront, which can then

be applied to a classical piston-position type mirror. Such a system looks exactly like the currently used adaptive optics systems, except that the blackbox “Hartmann sensor” has been replaced by the blackbox “curvature sensor”. However, both have exactly the same input/output characteristics.

The drawback of such a solution is that, as we stated, we would have to numerically solve the Poisson equation. This is a number crunching problem which requires much computer time, especially for large systems. Therefore, this solution may prevent us from following the atmosphere in real time due to insufficient speed.

Another approach is to consider other types of mirrors that would not require to solve the Poisson equation explicitly. The suggested curvature sensor delivers a signal which can be amplified and applied directly as a potential distribution on a bimorph or a membrane mirror without requiring any computation. The signal produces, on such mirrors, local curvature changes proportional to the voltage applied, which provides for a direct compensation for the measured curvature. Here, bimorph or membrane mirrors are used as an analog device which automatically solves the Poisson equation, and produces the desired wavefront shape. The method is potentially able to correct wavefronts with high spatial resolution at very high speed.

2.2.a. Membrane mirror

A membrane mirror typically is an assembly including an aluminized polymer foil, a transparent electrode, conducting pads, and control electrodes. The membrane itself is made of reflecting material and is used as the flexible mirror. It is positioned between the transparent electrode containing a bias voltage V_o and, on the other side, a region of actuators consisting of conducting pads having a voltage

$V_o + V_s$. The membrane is always at ground. Figure (2.2.1) shows a membrane assembly.

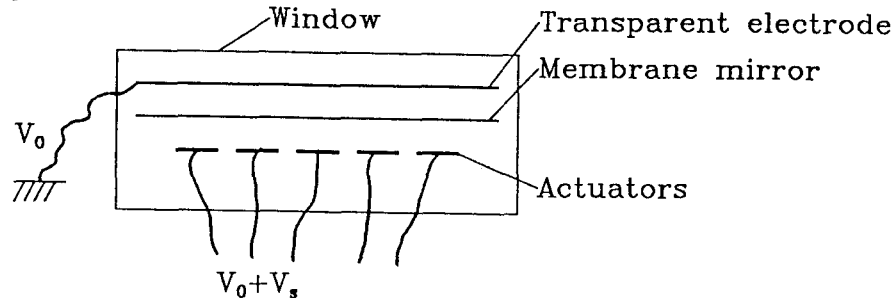


Figure (2.2.1) A membrane mirror assembly.

When V_s is equal to zero, no net force is applied to the membrane, and therefore, there is no deflection. When a voltage V_s is applied to any electrode, there will be a deflection over the corresponding pad. This deflection is such that the Laplacian at every point over the pad is the same, and proportional to V_s . It is then very clear that such a membrane mirror is perfectly suited for our application.

Since we have derived an analytic transfer function for the sensor, it would be very convenient to have one for the membrane active mirror as well. It can be shown that, for an ideal membrane, the equation of motion is (Morse, 1948):

$$\sigma \frac{\partial^2}{\partial t^2} z(x, y, t) = T \cdot \nabla^2 z(x, y, t) + P(x, y, t) \quad (2.2.1)$$

where z is the vertical displacement at point (x, y) and time t . σ is the membrane mass per unit area, P is the local pressure function applied to the membrane, T is the tension per length or stress multiplied by the thickness, and $\nabla^2 z$ is the Laplacian operator acting on the mirror surface $z(x, y)$. Atmospheric wavefront disturbances propagate at wind speed which is typically a hundred times smaller than propagation of sound in a membrane. In other words, the membrane will always be activated at frequencies well below its first resonance frequency. Therefore, we can

assume that the membrane is always in a state very close to its steady state given by equation (2.2.2).

$$\nabla^2 z + \frac{P(x, y)}{T} = 0 \quad (2.2.2)$$

A real membrane would certainly differ from this ideal model. The model has been developed without consideration of damping effects, i.e., assumes perfect elasticity. The effects of the finite boundary have been neglected as well, and the equation has been linearized which means that the results apply to small voltages and small resulting deflections only. Also, a real membrane would have a resonance frequency which limits the bandwidth of operations.

Therefore, a real membrane may approximately satisfy this equation, but only for sufficiently small excitations (deflections). Grosso and Yellin (1977) showed that this is indeed possible. The damping effects can be controlled by introducing a pressurization of the membrane cavity, the effects of finite boundary and boundary conditions can be handled, and with respect to the non-linearity, they found that the linear region is quite large compared to the required range for our application. The resonance frequencies of a membrane mirror are several kHz, and therefore, as we already said, are sufficiently large not to interfere with the atmospheric turbulence correction. For these reasons, a membrane mirror indeed presents us with a very good approach for this adaptive optics system.

2.2.b. Bimorph mirror

Bimorph mirrors are very similar to the previously discussed membrane mirrors. Also bimorph mirrors are active devices with electrodes, and they undergo a deformation as voltages are applied.

As for the case of the membrane mirror, the bimorph is an assembly that includes several components. First, the optical element is a piece of flexible glass

that is polished and coated. This is the deformable mirror. The control of the deformation is done with piezo-electric elements. The glass is firmly glued to an inner electrode on which a constant biasing voltage is applied. Beneath this inner electrode, a plate of piezo-electric ceramic with normal polarization is also glued, and still below this ceramic, there are placed differently shaped control electrodes. Figure (2.2.2) represents such a bimorph assembly.

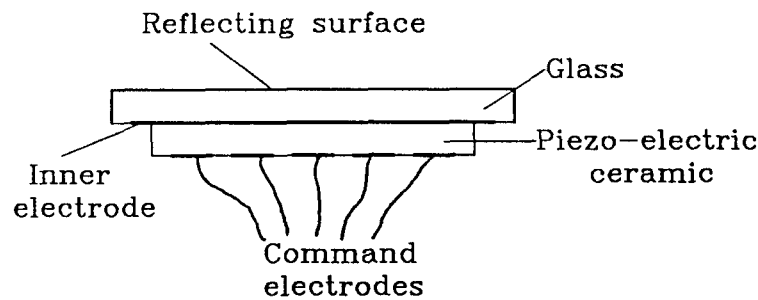


Figure (2.2.2) A bimorph mirror assembly.

When a voltage is applied to those, the lateral dimensions of the piezo-electric element changes locally, and the bimorph bends like a bi-metallic strip around the electrode.

Equations for this device can be derived from the physics of the materials. After minimization of the expression, and again assuming steady state like for the membrane, we obtain (Steinhaus, 1979):

$$\nabla^4 z + A\nabla^2 V = 0 \quad (2.2.3)$$

where A is a constant that should be determined experimentally. $V(x, y)$ is the voltage distribution on the inner electrodes, and $z(x, y)$ is the bimorph deflection.

This equation looks very similar to the Poisson equation. Nonetheless its solution is more complex. To numerically obtain $z(V)$, a regular Poisson equation would have to be solved with given boundary conditions, and this would yield to

yet another Poisson equation to be solved, but now with new boundary conditions and a right hand side, or forcing term. The whole procedure would hence be much harder.

Note that for a free forcing term, the membrane solution $V = \frac{1}{A} \nabla^2 z$ is a solution of the bimorph equation. Therefore, the main difference between a bimorph mirror and a membrane mirror is related to the boundary conditions, i.e., how the mirror is mounted at the edges.

The advantage of the bimorph over the membrane is that it is less sensitive to acoustic perturbations, and is less fragile. But its boundary conditions are much harder to handle. Similarly to the membrane mirror, also the bimorph should not be excited too much in order to reduce the effects of the non-linearities. As with the membrane mirror, the resonance frequencies of the bimorph are several kHz , and therefore, do not interfere.

CHAPTER 3

OPEN LOOP SYSTEM SIMULATION

3.1. Simulation of wavefronts

The model of our system must include a model of the disturbance the effects of which we wish to correct. We shall begin our modeling process by describing the disturbance.

3.1.a. Definition of Zernike polynomials

Both input and output signals are optical wavefronts. The input wavefront is estimated and corrected to produce the output wavefront. Both of them can be considered two-dimensional functions, following certain models and statistics. Since they may be very complex, people have looked toward finding a base to decompose them. Usually functions are decomposed into cosine and sine components using the Fourier transform. For wavefront aberrations, since they are mostly observed over a circular domain such as the mirror of a telescope, it is much more interesting to decompose them onto polar coordinate basic functions.

Such a decomposition exists, and the basic functions are the Zernike polynomials (Zernike, 1934). They have been used for a long time in optics to describe aberrations. They have the particularity of being separable in angle and radius. Their analytical definition can be found to be (Born and Wolf, 1965):

$$\begin{aligned}
 Z(\rho, \theta) &= R_n^m(\rho) \text{Cos } m\theta && \text{for } m \text{ even} \\
 Z(\rho, \theta) &= R_n^m(\rho) \text{Sin } m\theta && \text{for } m \text{ odd} \\
 Z(\rho, \theta) &= R_n^0(\rho) && \text{for } m = 0
 \end{aligned}
 \tag{3.1.1}$$

where $R_n^m(\rho)$ is a recursive polynomial of degree n . The parameters m and n represent the frequency variables. m is called the azimuthal frequency, and n the radial degree. Exactly like for the Fourier decomposition, higher values of n or m denote higher frequency components. It is convenient to order and number them according to a desired sequencing scheme. Several such schemes exist, but we shall use the one suggested by Noll (1976). The first terms are given in table (3.1.1). Some of those polynomials are plotted on figures (3.1.1a) and (3.1.1b).

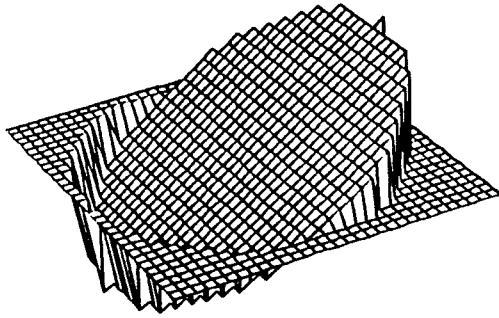
Each row includes all the polynomials with the same radial degree n , and each column includes the terms that have the same azimuthal frequency m . The numbering sequence of the index j of every polynomial $Z_j(\rho, \theta)$ proceeds row by row, and for a given value of n , modes with a lower value of m are ordered first. In the case where two polynomials have the same n and m , even j terms correspond to the symmetric modes in $\text{Cos}(m\theta)$, while odd j terms correspond to the antisymmetric modes in $\text{Sin}(m\theta)$.

With respect to the normalization, there exist several alternatives. Sometimes all the polynomials are normalized such that $\text{Max}(Z(\rho, \theta)) = 1.0$ on the unit circle. Nonetheless, we prefer to use here the normalization that Noll suggested: the root mean square (rms) value of the polynomial over the unit disk is 1. This will simplify the later statistical analysis when simulating random atmospheric wavefronts. The fact that the rms value is indeed 1 can be checked by computing the integral:

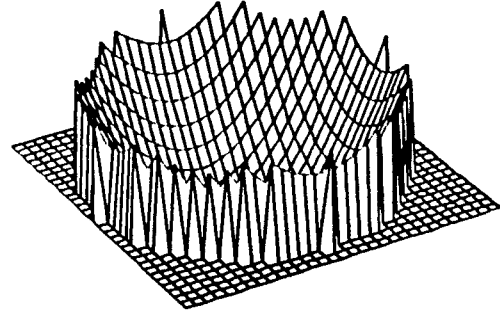
$$Z_{rms} = \frac{1}{\pi} \sqrt{\int_0^{2\pi} \int_0^1 Z^2(\rho, \theta) \rho \, d\rho \, d\theta} = 1 \quad (3.1.2)$$

Table (3.1.1) Zernike polynomials with particular order and normalization.

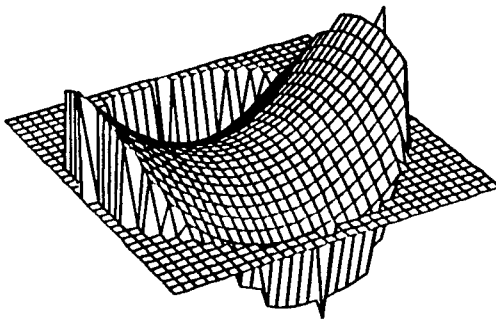
Radial degree (n)	0	1	Azimuthal frequency (m)	3
0	$Z_1 = 1$ Constant		2	
1		$Z_2 = 2\rho\cos\theta$ $Z_3 = 2\rho\sin\theta$ Tilts		
2	$Z_4 = \sqrt{3}(2\rho^2 - 1)$ Defocus		$Z_5 = \sqrt{6}\rho^2\sin 2\theta$ $Z_6 = \sqrt{6}\rho^2\cos 2\theta$ Astigmatism	
3		$Z_7 = \sqrt{8}(3\rho^3 - 2\rho)\sin\theta$ $Z_8 = \sqrt{8}(3\rho^3 - 2\rho)\cos\theta$ Pure Coma		$Z_9 = \sqrt{8}\rho^3\sin 3\theta$ $Z_{10} = \sqrt{8}\rho^3\cos 3\theta$ Zero curvature Coma
4	$Z_{11} = \sqrt{5}(6\rho^4 - 6\rho^2 + 1)$ Spherical		$Z_{12} = \sqrt{10}(4\rho^4 - 3\rho^2)\cos 2\theta$ $Z_{13} = \sqrt{10}(4\rho^4 - 3\rho^2)\sin 2\theta$ 5 th order Astigmatism	



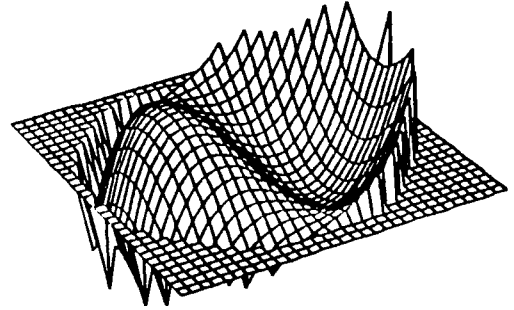
Z_2 : Tilt aberration



Z_4 : Defocus aberration

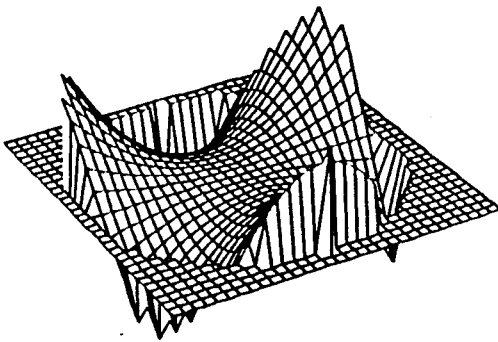


Z_6 : Astigmatism aberration

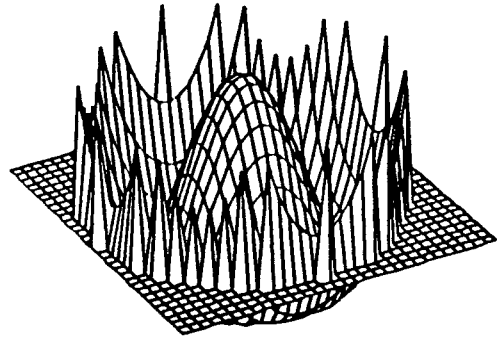


Z_8 : Pure coma aberration

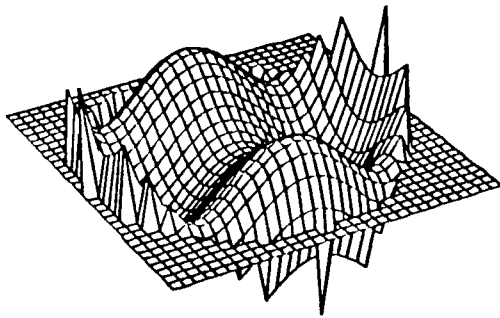
Figure (3.1.1a) Zernike polynomials.



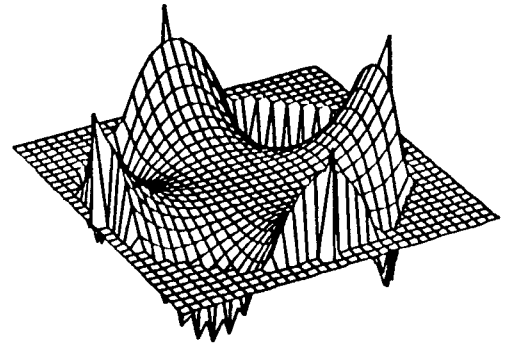
Z_9 : Zero curvature coma



Z_{11} : Spherical aberration



Z_{12} : 5th order astigmatism



Z_{14}

Figure (3.1.1b) Zernike polynomials.

Since $Z(\rho, \theta)$ can be separated in ρ and θ , $Z^2(\rho, \theta)$ can be separated too, and thus, the work for the integral can be much simplified. Also, the evaluation of some integrals can be used for more than only one Zernike term.

Note that for our application, the Z_1 constant term is mostly unused since we are working with zero mean terms only.

Any physical function $F(\rho, \theta)$ can be expanded into a linear combination of Zernike terms. Similarly to the Fourier decomposition, the coefficients will go to zero as the order of the polynomial increases.

3.1.b. Simulation of simple wavefronts

The simplest possible wavefronts are single Zernike polynomials. To simulate any of these is a simple task. The program just has to evaluate a given polynomial over an array of points. The Zernike term may be any of those defined earlier, or even a term of higher order.

A less simple wavefront is a linear combination of those, with different weighting coefficients. Since we want to correct physical functions, the low order terms will have higher coefficients than high order ones.

One advantage of simulating all the terms, and simple combinations thereof, separately is that we can verify that they are all decoupled and, thereby determine up to what level the overall system can be considered to be linear.

3.1.c. Atmospheric wavefronts

If we look at an atmospheric wavefront, our eventual goal for correction, we surely can decompose it into Zernike terms, too. The difference between simple wavefronts and atmospheric wavefronts is that the atmosphere has random disturbances. We only know a statistical distribution of its behavior. If we want to indeed compensate for such disturbances, we first must know something about them.

As we said earlier, the atmosphere can be described as moving waves with a given shape and speed which depends upon the weather conditions. We already presented an expression for its structure function (1.2.3), and we said that it obeyed Kolmogoroff's law (1.2.4).

Now, the problem is to relate all this to the Zernike polynomials that are so convenient to use. Noll wrote a very interesting paper on this subject (1976). The relationship that we are looking for can be obtained through energy considerations. The Wiener power spectrum of the phase fluctuations in the atmosphere due to the Kolmogoroff turbulence is given by:

$$W_{\phi}(\vec{f}) = \frac{0.023}{r_o^{5/3} f^{11/3}} \quad (3.1.3)$$

where f is the wave spatial frequency, and r_o is Fried's parameter characterizing the seeing (Fried, 1966).

The Zernike representation of this spectrum can be obtained by looking at the covariance of the terms in a Zernike expansion. The coefficient for every term can be said to be a Gaussian random variable with zero mean. The mean square of each of the Zernike coefficients as computed by Noll (1976) is given in table (3.1.2).

Table (3.1.2) Mean Square coefficients for each of the Zernike polynomials for the atmosphere, where D is the pupil diameter, and r_o Fried's parameter.

Zernike terms	Expected Mean Square value
Z_2, Z_3	$0.4557 (D/r_o)^{5/3}$
Z_4, Z_5, Z_6	$0.0236 (D/r_o)^{5/3}$
Z_7, Z_8, Z_9, Z_{10}	$0.0063 (D/r_o)^{5/3}$
$Z_{11}, Z_{12}, Z_{13}, Z_{14}, Z_{15}$	$0.0025 (D/r_o)^{5/3}$

These data are fundamental to the understanding of the Zernike decomposition of the atmosphere. Nonetheless, they are not directly exploitable for our

simulation. We need random amplitude coefficients. The expected values of the amplitudes are simply the square roots of the expected mean square values given in table (3.1.2), since the polynomials are normalized such that the rms value is equal to the amplitude coefficient. The expected values of the amplitude coefficients are given in table (3.1.3).

Table (3.1.3) Amplitude coefficients for each of the Zernike polynomials in the atmosphere, where D is the pupil diameter, and r_o Fried's parameter.

Zernike terms	Amplitude value
Z_2, Z_3	$0.6750 (D/r_o)^{5/6}$
Z_4, Z_5, Z_6	$0.1535 (D/r_o)^{5/6}$
Z_7, Z_8, Z_9, Z_{10}	$0.0793 (D/r_o)^{5/6}$
$Z_{11}, Z_{12}, Z_{13}, Z_{14}, Z_{15}$	$0.0499 (D/r_o)^{5/6}$

However, even the new table cannot be used directly because there exists a second problem. Some terms are statistically dependent on each other and must be decoupled first.

3.1.d. Simulation of atmospheric wavefronts that obey Kolmogoroff's law

In order to simulate random atmospheric wavefronts, we must assign random variables to each Zernike term. Unfortunately, we cannot use the amplitude coefficients in table (3.1.3) directly. This approach would work only if we dealt with independent random variables, which means that all the terms should be decoupled. This is however not the case. If we look at the complete covariance matrix, we notice that it is not diagonal. Therefore, in order to simulate random wavefronts that are in accordance with the atmospheric characteristics, we must apply a linear variable transformation such that, in the new variables, the covariance matrix is diagonal.

This is necessary since we don't have an algorithm to generate properly correlated random variables directly.

The covariance matrix in the Zernike basis can be evaluated. Let z_j and $z_{j'}$ be two Zernike polynomials as defined in Section 3.1.a with amplitude a_j and $a_{j'}$. Their covariance $\langle a_j, a_{j'} \rangle$ can be evaluated from equation (3.1.4). This expression is similar to equation (25) of Noll's (1976) paper, except that it has been corrected for printing errors and some other inaccuracies.

$$\langle a_j, a_{j'} \rangle = \frac{K_{zz'} \delta_z \Gamma[(n + n' - \frac{5}{3})/2] (D/r_o)^{5/3}}{\Gamma[(n - n' + \frac{17}{3})/2] \Gamma[(n' - n + \frac{17}{3})/2] \Gamma[(n + n' + \frac{23}{3})/2]} \quad (3.1.4)$$

where n , m , n' , and m' are the radial degree and azimuthal frequency of z_j and $z_{j'}$. Table (3.1.4) shows the relation between j , m , and n for a few terms. This information can be read out from table (3.1.1). δ_z is a Kronecker symbol the value of which is 1 if $m = m'$ and if j and j' do not have the same parity in the case $m \neq 0$; and the value of which is 0 otherwise. In other words, this means that the covariance is 0 for all terms that are not in the same column. Furthermore, for the covariance to be non zero, the terms must have the same parity on the index, i.e., the covariance between a sine term and a cosine term is zero (a fact that had been overlooked in Noll's expression).

Table (3.1.4) Degree and frequency for the first 14 Zernike polynomials

j	2, 3	4	5, 6	7, 8	9, 10	11	12, 13	14, 15
n	1	2	2	3	3	4	4	4
m	1	0	2	1	3	0	2	4

Back to equation (3.1.4), $\Gamma(x)$ is the usual Gamma function that involves Bessel integrals. Also:

$$K_{zz'} = \frac{0.046}{\pi^2} (2\pi)^{11/3} \frac{\Gamma(\frac{14}{3})}{2^{14/3}} (-1)^{(n+n'-m-m')/2} \sqrt{(n+1)(n'+1)} \quad (3.1.5)$$

or, after computation of the numerical coefficient,

$$K_{zz'} = 2.2802 (-1)^{(n+n'-m-m')/2} \sqrt{(n+1)(n'+1)} \quad (3.1.6)$$

Using these expressions, we can evaluate the covariance matrix of the first n Zernike terms. n is arbitrary, but the larger n is, the better the approximation to the real atmosphere will be. From now on, we shall work with $n = 15$, which means that we shall work with the first 14 zero mean Zernike polynomials, or the first four rows in table (3.1.1). This will include more than 97% of the energy in the spectrum (Noll, 1976), which is good enough for our purpose.

Let us now describe the procedure to simulate random atmospheric wavefronts. We want to generate random coefficients for the first 14 Zernike terms in the Zernike basis with a given covariance matrix.

Let \vec{A} be the desired vector the components of which are the random coefficients for the Zernike polynomials $Z_2 \dots Z_{15}$. \vec{A} can be represented as a column vector:

$$A = \begin{pmatrix} a_2 \\ a_3 \\ \vdots \\ a_{15} \end{pmatrix} \quad (3.1.7)$$

The covariance matrix can then be written as:

$$C = E[A \cdot A^T] = \begin{pmatrix} E(a_2 a_2) & E(a_2 a_3) & \dots & E(a_2 a_{15}) \\ E(a_3 a_2) & E(a_3 a_3) & \dots & E(a_3 a_{15}) \\ \vdots & \vdots & \ddots & \vdots \\ E(a_{15} a_2) & E(a_{15} a_3) & \dots & E(a_{15} a_{15}) \end{pmatrix} \quad (3.1.8)$$

where the notation $E(a_j a_{j'})$ is used instead of $\langle a_j, a_{j'} \rangle$ as defined in (3.1.4) for the expected values. This covariance C matrix is given in figure (3.1.2). We can notice that it is almost diagonal. This fact makes it easy to determine a change of basis which will completely diagonalize the matrix.

From its analytic expression, we know that C is Hermitian. Therefore there exists a unitary matrix U , ($U^{-1} = U^T$), such that $U \cdot C \cdot U^T$ is diagonal.

This U matrix can be obtained by computing a singular value decomposition (SVD) over the covariance matrix C . From the SVD routine available in CTRL-C, we obtain two matrices, a unitary matrix X , and a diagonal matrix S such that $C = X \cdot S \cdot X^T$. From there it is clear that

$$X^T \cdot C \cdot X = X^T \cdot X \cdot S \cdot X^T \cdot X = S \quad (3.1.9)$$

and therefore the matrix U we want is simply equal to X^T . Matrices S and U are given in figures (3.1.3) and (3.1.4) respectively.

Now, let us define another vector \vec{B} the components of which (b_2, b_3, \dots, b_{15}) are given by

$$B = U \cdot A \quad (3.1.10)$$

We then have for the new covariance matrix

$$E[B \cdot B^T] = E[U \cdot A \cdot A^T \cdot U^T] = U \cdot E[A \cdot A^T] \cdot U^T = S \quad (3.1.11)$$

which clearly is diagonal. The covariance matrix of B being diagonal means that the 14 components of B are uncorrelated. Since we deal with Gaussian random variables, they are statistically independent. Thus, we have found the basis we were looking for.

To conclude the complete procedure, we generate 14 Gaussian random variables that will be the components of the vector B . To obtain the coefficients we want, we simply compute the A matrix using the relation

$$A = U^T \cdot B = U^{-1} \cdot B \quad (3.1.12)$$

To simulate a Gaussian random variable, there exist many different procedures. One of the easiest and most popular ones was introduced by Frieden (1984). Starting from two independent uniform random variables z_1 and z_2 , immediately obtained with the RAN function on a computer, we can obtain two independent Gaussian random variables g_1 and g_2 using formulas (3.1.13a) and (3.1.13b).

$$g_1 = \mu + \sigma \text{Cos}(2\pi z_1) \sqrt{-2 \log(z_2)} \quad (3.1.13a)$$

$$g_2 = \mu + \sigma \text{Sin}(2\pi z_1) \sqrt{-2 \log(z_2)} \quad (3.1.13b)$$

where μ represents the desired mean of the Gaussian variable, and σ denotes its desired standard deviation.

For our application, μ of course will be 0, while σ is the square root of one of the coefficients of the diagonal matrix S corresponding to each term. Once these 14 independent numbers have been obtained, we simply multiply them by U^T as explained earlier to obtain the coefficients of the Zernike polynomials with correct variance and covariance from the atmospheric spectrum.

3.2. Simulation of the curvature sensor.

The task of this sensor is to be able to determine, for any given wavefront, the Laplacians at every inside point, and the normal derivatives at every edge point. In real operation, this is done optically, and the desired signal is obtained by

subtracting the two irradiance signals as explained earlier, or by using a modulation scheme such as the bi-refrigent lens.

For the simulation, in some cases, we do not necessarily have to really know what happens within the sensor itself. We may be able to consider it as a black box which, for any given input signal, produces the desired output signal. In other cases, we may wish to completely study the sensor itself, and model in detail its non-ideal characteristics. We must then simulate what really happens inside the sensor. However, these are separate cases, for which we created separate simulation routines. Accordingly, we can use the routine that is appropriate for any given problem.

3.2.a. Simulation of the ideal sensor characteristics.

Since we know the sampled two-dimensional wavefront, we can mathematically compute the data we want, that is, average the Laplacians over given areas, and average the normal derivatives over given parts of the edges.

Our configuration may involve a sensor with a potential shape as shown in figure (3.2.1), although the number of distinct elements used in figure (3.2.1) is exemplary.

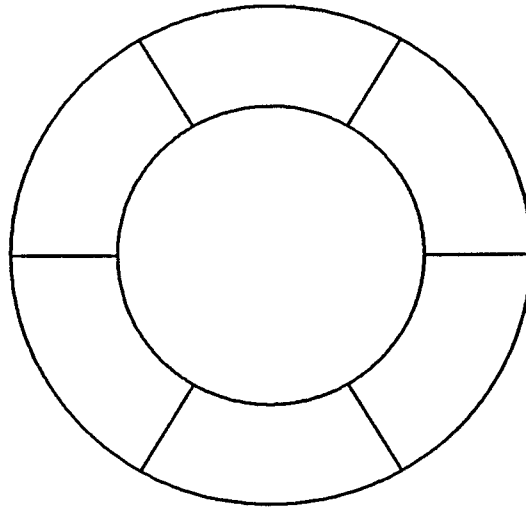


Figure (3.2.1) A typical sensor configuration.

Any of the central detectors measures the average Laplacian over its area. Any of the edge detectors measures the average normal derivatives at its position.

To evaluate the average Laplacian over a detector, we compute the Laplacian at every point that lies inside that particular detector, and compute an average value. Mathematically, to obtain an expression for the Laplacian at a given point (x, y) , we start from its definition (Kreysig, 1979):

$$\nabla^2 \phi(x, y) = \frac{\partial^2}{\partial x^2} \phi(x, y) + \frac{\partial^2}{\partial y^2} \phi(x, y) \quad (3.2.1)$$

Assuming the sampled second derivative is the difference between the first derivatives on each side, divided by the spacing, we get an expression for the discrete Laplacian:

$$\nabla^2 \phi(x, y) \approx \frac{\left(\frac{\partial \phi}{\partial x}(x+1, y) - \frac{\partial \phi}{\partial x}(x, y) \right)}{h} + \frac{\left(\frac{\partial \phi}{\partial y}(x, y+1) - \frac{\partial \phi}{\partial y}(x, y) \right)}{h} \quad (3.2.2)$$

Now, assuming the derivative to be the difference between the function itself on each side, and again dividing by the spacing, we get:

$$\begin{aligned} \nabla^2 \phi(x, y) \approx & \left(\frac{\phi(x+1, y) - \phi(x, y)}{h} - \frac{\phi(x, y) - \phi(x-1, y)}{h} \right) / h \\ & + \left(\frac{\phi(x, y+1) - \phi(x, y)}{h} - \frac{\phi(x, y) - \phi(x, y-1)}{h} \right) / h \end{aligned} \quad (3.2.3)$$

$$\nabla^2 \phi(x, y) \approx \frac{\phi(x+1, y) + \phi(x-1, y) + \phi(x, y+1) + \phi(x, y-1) - 4\phi(x, y)}{h^2} \quad (3.2.4)$$

where h is the normalized distance between two adjacent points.

To get the normalized derivative at each of the edge detectors, we can, using a least square algorithm, fit a plane through all the points that lie inside that particular element, and the normal derivative will be proportional to the plane coefficient along the normal direction.

Any one of the edge detectors looks similar to the one shown in figure (3.2.2), where \vec{n} is the unit vector along the normal axis, and \vec{t} is the unit vector along the tangential axis.

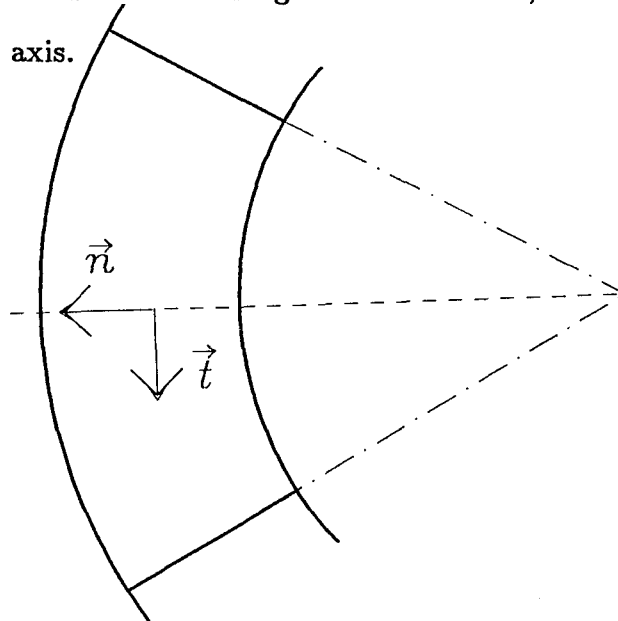


Figure (3.2.2) A typical edge detector.

We therefore want to fit the plane $ax_n + by_t + c = 0$.

A least square algorithm states that $\sum |\phi - (ax_n + by_t + c)|^2 \stackrel{!}{=} \min$ for all $\phi(x_n, y_t)$ inside the element. Therefore,

$$\frac{\partial \sum |\phi - (ax_n + by_t + c)|^2}{\partial a} = 0 \quad (3.2.5a)$$

$$\frac{\partial \sum |\phi - (ax_n + by_t + c)|^2}{\partial b} = 0 \quad (3.2.5b)$$

$$\frac{\partial \sum |\phi - (ax_n + by_t + c)|^2}{\partial c} = 0 \quad (3.2.5c)$$

This is a system of three equations with three unknowns: a , b , and c . With some algebraic computations and manipulations, we obtain the following expressions for the coefficient a in which we are interested:

$$a = \frac{A \sum x\phi - B \sum y\phi + C \sum \phi}{A \sum x^2 - B \sum xy + C \sum x} \quad (3.2.6)$$

where

$$A = (N \sum y^2 - (\sum y)^2) \quad (3.2.7a)$$

$$B = (N \sum xy - \sum x \sum y) \quad (3.2.7b)$$

$$C = (\sum xy \sum y - \sum x \sum y^2) \quad (3.2.7c)$$

and where N is the total number of $\phi(x_n, y_t)$ points on the segment.

With such a scheme for all the central and all the edge elements, we indeed get the output required to solve the Poisson equation with Neumann boundary conditions. We have simulated the overall curvature sensor without simulating the optical sensor itself. This describes its ideal characteristics.

3.2.b. Some examples of such curvature signals for simple wavefronts.

We shall look at three different curvature signals with Laplacians inside, and with edge slopes on their boundaries. Let us start with a tilt aberration.

A tilted wavefront produces a shift on the image. The shift will be equal in size but with opposite direction for each of the two recorded images. The image does not change in size, which implies that the irradiances are the same if they are centered correctly. This shift is given by equation (3.2.8) in terms of an ϵ aberration amplitude, and in terms of f and ℓ , the setup parameters of the sensor as explained before. The measurement units of displacement are expressed in multiples of the pupil radius.

$$Shift = \frac{\epsilon f(f - \ell)}{\ell} \quad (3.2.8)$$

Therefore, we can expect the subtraction to yield a 0 Laplacian zone inside the region, and some high positive and negative signals along the edges where there is no overlap of the two images. Figure (3.2.3) represents such an expected signal. The simulation routine confirms this.

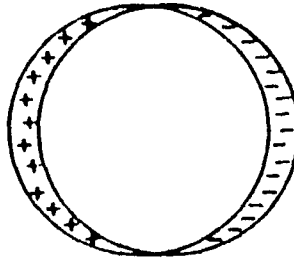


Figure (3.2.3) A pure tilt curvature signal.

Now, let us take a defocus wavefront. A defocus aberration optically corresponds to a shift of the focus point along the optical axis. The actual focus is closer or further from where it is supposed to be. Therefore, the two images will be centered, but will not have the same diameter, hence, the local irradiance will also change. The radius of our two images are given by equation (3.2.9) with the same notation as for equation (3.2.8).

$$\text{Radius} = 1 \pm \frac{4\epsilon f(f - \ell)}{\ell} \quad (3.2.9)$$

Therefore, we can expect the subtraction to yield a constant Laplacian zone inside the region corresponding to the difference between the two intensities, and some high signal zone along the edges corresponding to the non-overlap due to the fact that one image is larger than the other. Such a signal is shown on figure (3.2.4). As for the tilt, the simulation confirms the analytical results.

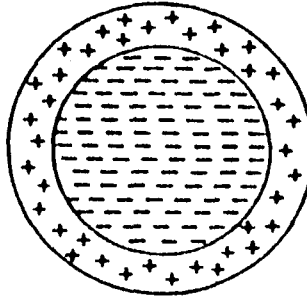


Figure (3.2.4) A pure defocus curvature signal.

The last example is a pure astigmatism wavefront. Such an aberration optically deforms a circle into an ellipse. Only the orientation of the ellipse will differ on the two images, but not their size, and not their intensity. The major and minor axes of the two ellipses are given in equation (3.2.10), with the same notation as before.

$$Axes = 1 \pm \frac{2\epsilon f(f - \ell)}{\ell} \quad (3.2.10)$$

Therefore, we can expect the subtraction to yield a 0 Laplacian zone inside the region, and four high signal zones along the edges corresponding to the non-overlap of the two ellipses. Such a signal is shown in figure (3.2.5). Again, the simulation routine confirms the results.

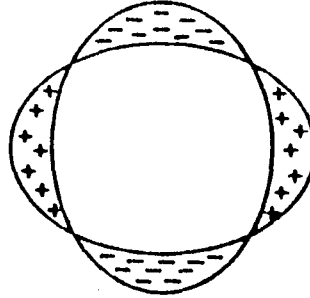


Figure (3.2.5) A pure astigmatism curvature signal.

3.2.c. Simulation of the sensor by ray tracing method.

This second method will further involve the physics of the sensor itself, although we shall still neglect side effects such as diffraction. We consider now the sensor as an optical setup including the main element, usually the telescope, the small lens to correct for complete symmetry, and the two irradiance detectors in planes P_1 and P_2 . We use geometrical optics laws.

Similarly as for the previous simulation routine, we use the distorted wavefront as input, and we wish to obtain an output signal that corresponds to its curvature with boundary conditions. The principle of the ray tracing method is very simple. The light rays are simulated one by one. Each starts from a position on the wavefront surface and propagates in a direction orthogonal to the surface at

that position. The propagation is of course a perfect straight line. Therefore, that ray will intercept plane P_1 , and the spot of impact on the corresponding detector will be recorded. Then the ray will pursue its route and will pass through lens L_2 . Here a deviation occurs. The lens is being used to symmetrize the setup. This means that the ray will proceed into a direction which will let it hit the symmetric point of the original place on the wavefront with respect to the focus point of the telescope. On its way, it will intercept plane P_2 , and again, the spot of impact on the photodetector will be recorded.

This procedure can be executed a large number of times, once for each ray simulated. Of course, the position on the wavefront has to be different for all rays so that the whole surface is scanned. This scanning can be done along a determined grid, or using random positions (Monte Carlo technique).

Once the desired number of rays has been simulated, the two irradiance distributions I_1 and I_2 are known, since the local irradiance on any surface is proportional to the number of rays that had an impact on that particular surface. This is called a spot diagram.

From there, it is trivial to obtain the desired curvature output signal since only the ratio $\frac{I_1 - I_2}{I_1 + I_2}$ has to be evaluated with a complete spatial symmetry on I_2 .

Let us now describe how to practically implement a routine for this. The input data is a wavefront $z(x, y)$ that is known over a 2D array. From this array, and the knowledge of the optical parameters f and ℓ , it is possible to compute the irradiance arrays $I_1(x, y)$ and $I_2(x, y)$.

First, it is clear that the process which generates a vector position deviation out of a vector slope is completely linear. Therefore, the problem can be separated

into two problems, one along each of the two orthogonal directions \vec{x} and \vec{y} . Figure (3.2.6) shows what is happening with a local wavefront perturbation in the \vec{x} direction.

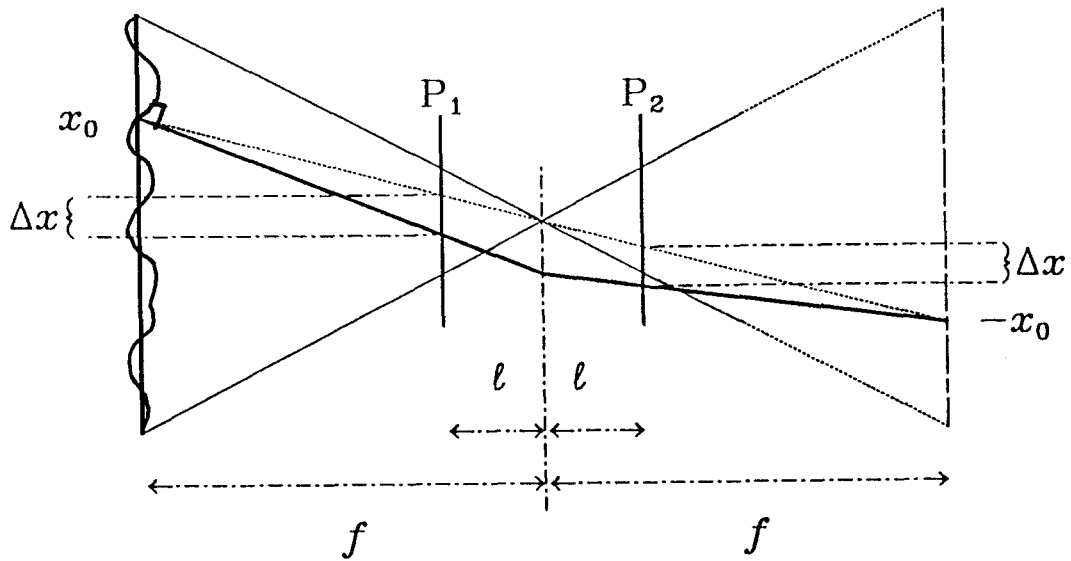


Figure (3.2.6) Ray tracing showing the effect of a perturbation in the \vec{x} direction.

The whole picture is in a plane $y = y_0$. We are interested in finding the spot of impact of the ray emanating from position $x = x_0$ where the wavefront has a local disturbance. If there were no disturbance, the spots of impacts would be $x_1 = x_0$ and $x_2 = -x_0$, if we consider a normalized scale removing the reduction l/f due to the convergence toward the focus point. But due to the disturbance an additional correction Δx is required. Due to symmetry, the correction is the same

for both impact spots. Since the wavefront is known at every point, it is clear that the true deviation Δx_t can be computed from

$$\frac{\Delta x_t}{(f - \ell)} \approx \frac{z(x_0 + h, y_0) - z(x_0 - h, y_0)}{2h} \quad (3.2.11)$$

where h represents the spacing between two adjacent points on the wavefront array. Denormalizing, we obtain the following value of Δx :

$$\Delta x \approx \frac{f(f - \ell) [z(x_0 + h, y_0) - z(x_0 - h, y_0)]}{2h \ell} \quad (3.2.12)$$

Therefore, we obtain for the points of impact x_1 and x_2 :

$$x_1 \approx \frac{f(f - \ell) [z(x_0 + h, y_0) - z(x_0 - h, y_0)]}{2h \ell} + x_0 \quad (3.2.13a)$$

$$x_2 \approx \frac{f(f - \ell) [z(x_0 + h, y_0) - z(x_0 - h, y_0)]}{2h \ell} - x_0 \quad (3.2.13b)$$

Similarly, for the deviations due to perturbations in the other direction, we find:

$$y_1 \approx \frac{f(f - \ell) [z(x_0, y_0 + h) - z(x_0, y_0 - h)]}{2h \ell} + y_0 \quad (3.2.14a)$$

$$y_2 \approx \frac{f(f - \ell) [z(x_0, y_0 + h) - z(x_0, y_0 - h)]}{2h \ell} - y_0 \quad (3.2.14b)$$

From any starting point (x_0, y_0) on the wavefront, and the knowledge of the local wavefront surface, we can therefore compute the positions of impact (x_1, y_1) and (x_2, y_2) of the ray in the planes P_1 and P_2 .

In Section 3.4, we shall present simulation results using this algorithm.

3.2.d. Simulation of the sensor including diffraction effects.

This simulation model is the most accurate of the three models. The algorithm produces a simulation of the true picture that will be generated on the detectors. This model enhances the previous model by taking diffraction phenomena into account. By comparing the results of this very accurate simulation with the results obtained from the previous two models, we can decide if, and if so, to what extent these effects can be neglected in the overall simulation.

As we said in previous parts, the sensor signal is $\frac{I_1 - I_2}{I_1 + I_2}$ where I_1 and I_2 are the irradiance distributions in the two planes on each side of the focus point of the telescope. So far, we have had only approximations for these distributions. We have not computed them accurately yet. This is the goal of this section.

To obtain a mathematical expression for such an irradiance, we have to make use of the diffraction theory. For our case, since the out-of-focus planes where I_1 and I_2 are detected are close enough to the focus, we shall apply the Fresnel diffraction theory (Gaskill, 1978).

The input data are again a wavefront $z(x, y)$, and the knowledge of the optical setup and the parameters ℓ (the distance of the out-of-focus planes from the focus point), λ (the wavelength), and f (the focal length of the telescope).

According to the Fresnel diffraction theory, we have to work with the complex amplitude of the wavefront, and realize that the transmission through a lens, or the propagation between lenses is characterized by a two-dimensional convolution. It can be shown that the complex amplitudes A_1 and A_2 of the wavefronts reaching the two detectors are (Rodier, 1987):

$$A_1(\vec{r}) = \frac{1}{j\lambda(f-\ell)} e^{+j\pi \frac{\vec{r}^2}{\lambda(f-\ell)}} \int P(\vec{\rho}) \Psi(\vec{\rho}) e^{+j\pi \frac{\ell \vec{\rho}^2}{\lambda f(f-\ell)}} e^{-2j\pi \frac{\vec{r}\vec{\rho}}{\lambda(f-\ell)}} d\vec{\rho} \quad (3.2.15a)$$

$$A_2(\vec{r}) = \frac{1}{j\lambda(f-\ell)} e^{-j\pi \frac{\vec{r}^2}{\lambda(f-\ell)}} \int P(\vec{\rho}) \Psi(\vec{\rho}) e^{-j\pi \frac{\ell \vec{\rho}^2}{\lambda f(f-\ell)}} e^{-2j\pi \frac{\vec{r}\vec{\rho}}{\lambda(f-\ell)}} d\vec{\rho} \quad (3.2.15b)$$

where $P(\vec{r})$ is the gate function of the pupil entrance, which is 1 inside the pupil region, and 0 outside the pupil region. $\Psi(\vec{r})$ is the complex amplitude of the incoming wavefront:

$$\Psi(\vec{r}) = e^{j\phi(\vec{r})} = e^{\frac{2j\pi}{\lambda} z(\vec{r})} \quad (3.2.16)$$

To relate the complex amplitudes A_i to the irradiances I_i which are the functions that we really need, we use the relationship $I_i(\vec{r}) = |A_i(\vec{r})|^2$. We obtain:

$$I_1(\vec{r}) = \frac{1}{\lambda^2(f-\ell)^2} \left| \int P(\vec{\rho}) \Psi(\vec{\rho}) e^{+j\pi \frac{\ell \vec{\rho}^2}{\lambda f(f-\ell)}} e^{-2j\pi \frac{\vec{r}\vec{\rho}}{\lambda(f-\ell)}} d\vec{\rho} \right|^2 \quad (3.2.17a)$$

$$I_2(\vec{r}) = \frac{1}{\lambda^2(f-\ell)^2} \left| \int P(\vec{\rho}) \Psi(\vec{\rho}) e^{-j\pi \frac{\ell \vec{\rho}^2}{\lambda f(f-\ell)}} e^{-2j\pi \frac{\vec{r}\vec{\rho}}{\lambda(f-\ell)}} d\vec{\rho} \right|^2 \quad (3.2.17b)$$

For the simulation, it is of course very inconvenient to evaluate such two dimensional integrals. However, we can notice that we have an $e^{-2j\pi}$ term in the integrals, which is by definition a kernel for a Fourier transform. Thus, these two I_1 and I_2 irradiance distributions can be evaluated using 2D FFT routines. A general public domain easy to use 2D FFT routine is FOUR2, written by Brenner (1968).

The complete procedure to simulate our curvature sensor is the following. From the wavefront $z(x, y)$ or $z(\vec{r})$, the function $\Psi(\vec{r})$ is evaluated, and then gated with $P(\vec{r})$. The result is multiplied by the Fresnel terms $e^{\pm j\pi r^2}$, then the 2D FFT is taken, its modulus is squared, and finally, the result is multiplied by the constant coefficient.

Using this algorithm, we can derive the true values for the irradiance distributions $I_1(x, y)$ and $I_2(x, y)$.

3.3. Reconstruction algorithms.

We are now at a point where the output signal from the curvature sensor is known. From this information, we must derive the corresponding wavefront, so that we can correct it. Therefore we need some kind of wavefront reconstruction algorithm.

As we said in the introductory chapters, that reconstruction will eventually be done for the closed loop system analogically by feeding the curvature signal directly to the electrodes on a membrane or bimorph mirror. On the other hand, for the purpose of open loop operations, or for this closed loop simulation, we still need to reconstruct the wavefront on a computer.

From now on let us talk about that reconstruction process as a subsystem, or element, of the overall system. An input signal will be applied to this subsystem which can be either the output of the previously discussed subsystem or a signal taken from real measurements. The input is the curvature signal with edge conditions, i.e., a set of values of the Laplacian over the pupil, and a set of normal slopes at the edges of that domain. Therefore, in order to reconstruct the wavefront, we need to solve the Poisson equation which is a particular type of an elliptic partial differential equation.

For such kinds of differential equations, a number of algorithms had been previously developed, and an extensive set of software routines has been made available (Rice, 1985). Nonetheless, because our application is very special (non-rectangular domain with Neumann conditions given as Dirac δ_c distributions), we had to write our own algorithm to speed up the numerical solution, and at the same

time, to increase its accuracy. This was decided after our initial attempts to use ELLPACK (Rice, 1985) were not very successful. However, we made use of existing techniques and code fragments as much as possible in order to minimize the amount of algorithmic development and coding necessary.

We now are going to talk about the kernel of the selected algorithm, an iterative solution with successive overrelaxation which turned out to be the most promising algorithm for our application. Thereafter, we shall discuss how noise is amplified through this kernel, which is an essential characteristic to be considered in a fair comparison between the curvature sensing method and the classical slope measurement technique for adaptive optics. Finally, we shall explain our complete algorithm with its particular way of handling our special boundary conditions. The algorithm will be demonstrated at hand of several open loop situations.

3.3.a. General algorithms for elliptic equations.

Two main approaches are widely used in the solution of elliptic partial differential equations. In both cases, the equation is discretized over an array, and therefore, there are a finite number of points to work with, although this number may be quite large.

The first approach consists of writing one equation per solution point, which produces a set of n equations with n unknowns. Solving the system gives the n solution point values. Each of these equations is obtained by writing a limited 2^{nd} order Taylor series expansion around the point. For the point (x, y) we obtain an equation of the form:

$$\phi(x, y) \approx \frac{\phi(x+1, y) + \phi(x-1, y) + \phi(x, y-1) + \phi(x, y+1) - \text{lapl}(x, y)}{4} \quad (3.3.1)$$

where $lapl(x, y) = h^2 \cdot \nabla^2 \phi(x, y)$ is the Laplacian at the point (x, y) for normalized sampling.

The drawbacks of this method are that a very big matrix may have to be inverted, e.g. by Gaussian elimination, which gives rise to other numerical problems, and that even if we “have an idea” of the solution that we are looking for, there is no way we can “help” the computer. This last option is very useful for our closed loop system.

Indeed, in closed loop operation, the curvature sensor sees only the residual error that still has to be corrected. Since this error is hopefully very small compared to what has already been corrected, it seems to be a “good idea” to use the previous reconstruction as a starting point for the next inversion. Unfortunately, with a direct matrix algorithm, it is not possible to do so, and therefore, each inversion consumes as much time as the previous one.

Clearly, it would be possible to use this linear algebra technique to reconstruct the wavefront, but this would not be very efficient.

However, there exists an alternative approach which offers many advantages. The main difference is that, instead of using direct inversion routine, we use an iterative algorithm, which starts with an initial guess of the solution which is subsequently improved in an iterative fashion. As the algorithm goes on, the solution comes closer and closer to the true answer. The whole procedure is very easy to program, and we can indeed “help” the algorithm by supplying an intelligent first guess to it. If our initial guess is good, it will take the computer much less time to converge toward the final solution. In a closed loop operation, we just have to give as initial guess the solution of the previous loop.

This algorithm is one of the standard algorithms that are widely used in numerical linear system solutions, and, of course, this algorithm was also available

within ELLPACK (Rice, 1985). The problems that we faced when using ELLPACK were related to how ELLPACK treats the boundary conditions, and were not related to the kernel algorithm itself.

Now that we have seen that it would indeed be better to use an iterative algorithm, let us precisely described what the requirements are for it to work, i.e., what conditions must be satisfied to guarantee convergence. Any linear iterative (recursive) scheme describing the transition from step $(m - 1)$ to step (m) may be represented in the following form

$$u^{(m)} = Tu^{(m-1)} + k \quad (3.3.2)$$

where T is called the iteration matrix, k is a constant vector, and (m) indicates the iteration count. Let

$$e^{(m)} = u^{(m)} - u_{sol} \quad (3.3.3)$$

be the remaining error vector after (m) iterations, where u_{sol} is the true solution, which is therefore unchanged by equation (3.3.2). Hence, u_{sol} satisfies the equation

$$u_{sol} = Tu_{sol} + k \quad (3.3.4)$$

Subtracting (3.3.4) from (3.3.2) gives

$$e^{(m)} = Te^{(m-1)} = T^m e^{(0)} \quad (3.3.5)$$

It can be shown by spectral decomposition of T that the norm of $e^{(m)}$ as $m \rightarrow \infty$ satisfies (Young, 1972):

$$\| e^{(m)} \| \leq \mu^m(T) \cdot \| e^{(0)} \| \quad (3.3.6)$$

where $\mu(T)$ is the spectral radius (the length of the largest eigenvalue) of T . It is therefore clear that, for the iteration equation (3.3.2) to converge, the largest

eigenvalue of the iteration matrix T must be less than unity. Furthermore, we can notice that the smaller the eigenvalues are, the faster the convergence will occur.

Now that we know the requirements for the method to converge, let us describe how the algorithm is implemented. The basic algorithm is the so-called Jacobi algorithm, which however converges very slowly. This algorithm requires on the order of n^2 iterations to converge where n is the number of solution points. On the other hand, we can speed up the convergence dramatically by using a *successive overrelaxation method* (SOR). At each step, the algorithm tries to anticipate what to do according to what it did on previous steps. It tries to “learn” how to work more efficiently. This method involves extrapolation of the solution, and therefore, is a two-step scheme. At a solution point (x, y) the algorithm first computes the predictor $\tilde{\phi}(x, y)$ from the knowledge of both the value of the normalized Laplacian $lapl(x, y)$ at that point, and of the nearest-neighbor average $\bar{\phi}(x, y)$ (Southwell, 1980):

$$\tilde{\phi}(x, y) = \bar{\phi}(x, y) - lapl(x, y)/4 \quad (3.3.7)$$

where,

$$\bar{\phi}(x, y) = \frac{\phi(x+1, y) + \phi(x-1, y) + \phi(x, y-1) + \phi(x, y+1)}{4} \quad (3.3.8)$$

Then, this predictor $\tilde{\phi}^{(m)}(x, y)$ is used to define the corrector,

$$\phi^{(m+1)} = \phi^{(m)} + \omega \left\{ \tilde{\phi}^{(m)} - \phi^{(m)} \right\} \quad (3.3.9)$$

where ω is called the relaxation parameter. Hence, the m -th complete iteration equation for each point is (Southwell, 1980):

$$\phi^{(m+1)}(x, y) = \phi^{(m)}(x, y) + \omega \left[\bar{\phi}^{(m)}(x, y) - lapl(x, y)/4 - \phi^{(m)}(x, y) \right] \quad (3.3.10)$$

Comparing equation (3.3.9) with equation (3.3.2), we find that the T matrix for successive overrelaxation, which we shall call T_{sor} satisfies the equation:

$$T_{sor} = (1 - \omega)I + \omega M \quad (3.3.11)$$

where I is the identity matrix, and M computes the equation (3.3.8).

T_{sor} is a function of the relaxation parameter ω which still must be chosen. To achieve the best convergence, the optimal SOR parameter ω_{opt} must be selected such that the eigenvalues of the iteration matrix T_{sor} are as close as possible to the origin. The solution to this problem leads to the well known optimal SOR parameter (Young, 1972):

$$\omega_{opt} = \frac{2}{1 + \text{Sin} \frac{\pi}{n+1}} \quad (3.3.12)$$

where n is the total number of points for the solution.

Under those conditions, using the optimal SOR parameter, the convergence rate is reduced to a factor of n instead of the previous n^2 as for the regular Jacobi method.

With such an algorithm, we eventually will be able to reconstruct the wavefront, or simulate the active mirror, in a closed loop mode. However, we still need to handle the boundary conditions correctly. However, before proceeding with this discussion, let us look at the noise amplification for this subsystem on the Laplacian alone.

3.3.b. Noise amplification and scalability for the reconstruction algorithm.

The main purpose of a complete noise and noise amplification study is to be able to compare the capability of a curvature sensing based system to the one of a slope sensing system. In Chapter 2, Section 2.1.d, we showed for the sensor

alone that photon noise affected both sensor types in exactly the same manner. Nevertheless, since the ultimate goal is to compare the reconstructed wavefronts, we need to analyze how the reconstruction algorithm is effected by noise. Because the two sensor types are equivalent from a noise point of view, the only difference could be caused by the reconstruction algorithms themselves.

An estimate of the so-called noise amplification is a ratio that relates the error on the reconstructed wavefront to the error on the measurements. Two complete studies have been performed for classical systems. Fried (1977) and Hudgin (1977) have analyzed the propagation of errors in the case of slope measurements. They both define the error coefficient as the ratio of the mean square wavefront error over the mean square slope error for a unit sampling interval. Fried's analysis applies to a Shack-Hartmann sensor with a square lenslet array yielding phase estimates at the corners of the squares, whereas Hudgin's algorithm applies to the same sensor, but yielding phase estimates at the center point of each square.

For our comparison, we redefine the error coefficient as the ratio of the mean square wavefront error to the mean square Laplacian error for a unit sampling interval. In the case of slope measurements, the mean square Laplacian error is four times larger than the mean square slope error. According to our new definition, the noise propagation factors become for Hudgin's and Fried's analysis:

$$(\sigma_{out}/\sigma_{in})_{Hudgin} = 0.1403 + 0.0258 \ln(N) \quad (3.3.13a)$$

$$(\sigma_{out}/\sigma_{in})_{Fried} = 0.1640 + 0.0801 \ln(N) \quad (3.3.13b)$$

where N represents the number of solution points across the pupil. For a square pupil, the total number of points n would therefore be $n = N^2$, and for a circular aperture, we would have $n \approx \pi(N/2)^2$. Table (3.3.1) relates typical values for N to

the number n of solution points, or number of actuators for an active systems, in the case of a circular aperture.

Table (3.3.1) Relation between the number of points across the pupil N , and the total number of points over the pupil n

N	2	4	6	8	10
n	3	13	28	50	79

In order to compare our curvature sensing technique to the previously used slope detection technique, we performed a similar analysis to those made by Fried (1977) and Hudgin (1977). We were interested to find the propagation error coefficient in the case of curvature measurements. One way to achieve this was to theoretically derive an expression for this quantity. This was partly done at Kaman Aerospace, Corp. (Shellan, 1987). Shellan tried to express the phase function through a Fourier series expansion, and used the Fourier coefficients in his computations. Unfortunately, the computations and expressions grew too large to obtain a general result. To get rid of some terms, he made some approximations such as assuming a “very large” number of actuators, in the 10000^+ range. That assumption of course was not admissible for our purpose since, for astronomical applications, we can hardly use more than 50 independent active elements. Nevertheless, Shellan’s work anticipated that noise propagation would be much larger for curvature than for slope measurements, especially as the number of actuators increases.

Not being able to have an exact analytic derivation of the error propagation coefficient, we went along a different route, and tried to obtain the desired noise propagation from simulation results. There are two ways to do this.

The first solution consists of determining all the coefficients of the convolution matrix which relates the reconstructed wavefront values to the Laplacian

coefficients. The required error coefficient is simply the sum of the squared matrix coefficients. We ran such a procedure for many different array sizes, and pupil shapes. The results for σ_{out}/σ_{in} are summarized in table (3.3.2). The drawback of this first method is that, for large arrays, it takes a long time to compute this convolution matrix due to the large number of terms, so we did this only for relatively small array sizes. On the other hand, a big advantage of this approach is that the coefficients computed in this way are highly accurate.

Table (3.3.2) Error propagation coefficient from the convolution method

Array size	“square”	“circular”
3	0.24	0.07
5	0.78	0.17
7	2.17	0.35
9	2.75	0.58
11	4.11	1.03

The second solution consists of running a so-called Monte Carlo analysis. Random Laplacian arrays are simulated with a given variance, and the algorithm reconstructs the wavefront the variance of which is measured. σ_{out}/σ_{in} is simply the ratio of those two numbers. Of course, for the Monte Carlo analysis, a large number of runs has to be executed for each data point since it is a statistical approach and since we do not want to be sampling limited. The values of the propagation ratio are summarized in table (3.3.3). The reason why we made some runs with independent removal of the overall tilt will be explained later.

Table (3.3.3) Error propagation coefficient from the Monte Carlo method.

Array size	“square”	“circular”	“circular” with no tilt
7	2.19	0.32	0.17
9	2.78	0.61	0.23
11	4.06	1.01	0.44
15	7.65	1.69	0.78
23	17.99	3.97	1.80

Fortunately, the results in tables (3.3.2) and (3.3.3) are in good agreement with each other. This gives us a good confidence in the accuracy of these results. The small difference may stem from the fact that the Monte Carlo approach is a statistical one, and we would require an infinite number of runs to obtain the true answer. Now, about the data themselves, the error coefficient was found to increase as the square of the linear array size (as expected by Shellan for very large arrays). Using the slope measurement technique, it increases only with the logarithm of the size. An explanation for this unfortunate difference is that the values for the Laplacian derived from the slope are not independent of each other. Indeed, adjacent values share a common slope value (with opposite sign) which limits the error propagation, whereas curvature sensing yields independent Laplacian values. A complete comparison of the error propagation in the two techniques is presented in figure (3.3.1).

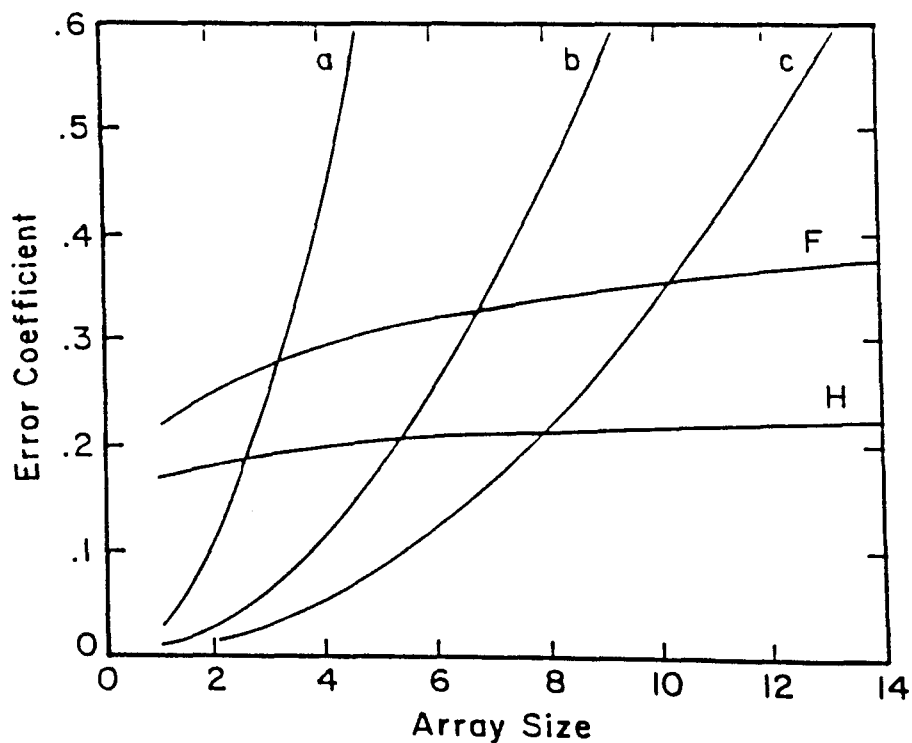


Figure (3.3.1) Error propagation for several cases.

The “H” and “F” curves are for a slope sensor. They are plots of equations (3.3.13a) and (3.3.13b), respectively. The other three curves represent the propagation error using our curvature sensor. The “a” curve is a plot of the “square” assumption in the tables and is for the case of a square pupil discretized along a square grid in the array. The “b” curve is for the “circular” assumption, and deals with a circular pupil discretized along an hexagonal grid. We note that the error is less severe for the latter case. With the square pupil, the corners produced high errors. Inspection of the distribution of errors in the reconstructed wavefronts shows that the main contribution is an overall tilt error. Figure (3.3.2) shows a reconstruction of a completely random Gaussian distributed Laplacian. We would

expect to obtain a completely flat image. Figure (3.3.2) shows clearly the tilt error introduced in the reconstruction.

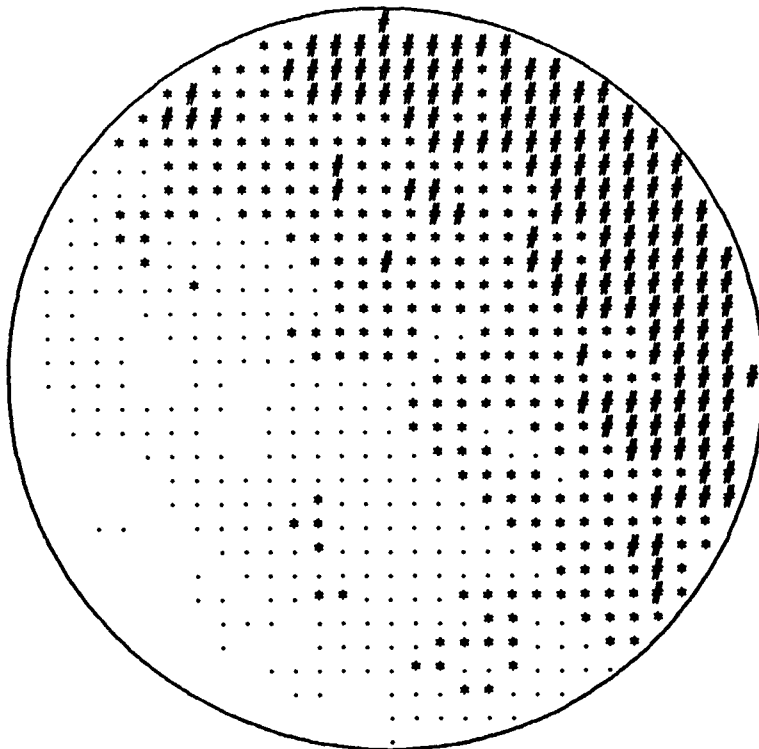


Figure (3.3.2) Example of reconstruction from random Laplacians.

The error can be considerably reduced if this tilt is removed by subtracting a plane wave, least-square fitted to the reconstructed wavefront. The resulting errors are the 3rd case, or curve “c” in figure (3.3.1). If we now look at the dominant error term after elimination of the tilt error, as shown in figure (3.3.3), we can clearly identify it as an astigmatism term.

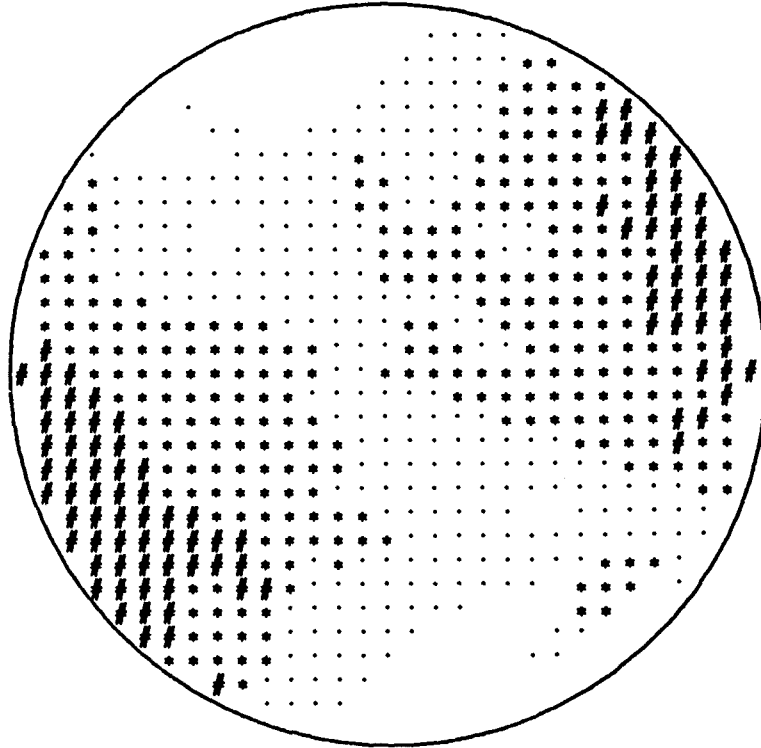


Figure (3.3.3) Example of reconstruction from random Laplacians after removal of global tilt.

Hence, the error propagation is essentially due to low order terms with zero Laplacians, the terms that are on the first diagonal in the Zernike table. The results could therefore be considerably improved if some modal estimations of these terms were performed.

From this analysis and comparison, we can conclude that error propagation is probably the main drawback of curvature sensing. However, for small sensor arrays such as required to measure and compensate low order aberration terms of optical telescopes during astronomical observations, it is clear that the curvature performance may be at least comparable to, but possibly even better than those of a Hartmann sensor with the advantage of a lesser complexity.

3.3.c. How to handle correctly the boundary conditions.

So far we have studied how the kernel of the reconstruction algorithm functions, and how the propagation error increases with the array size. From this, we were able to assess the maximum array size with which we can still obtain an acceptably low noise amplification.

However, this discussion considered the Laplacian part of the Poisson equation only, that is, the effects of the boundary conditions were ignored. Unfortunately, to correctly reconstruct the wavefront from our curvature sensor signal, these boundary conditions are not only required, but they are in fact as important as the Laplacian curvature signal itself; and they must therefore be handled properly.

For our application, all boundary conditions are of the Neumann type, which means that we force the slope at the edges of the domain rather than the positions themselves as for Dirichlet type boundary conditions. Such a problem does not have a unique solution since any constant position added to the overall solution will also satisfy the Poisson equation and the Neumann type boundary conditions. We forced a unique solution by making the additional assumption of a zero mean value of the overall image. This was actually the major problem that we faced when we initially tried to use ELLPACK (Rice, 1985). ELLPACK was not able to handle elliptic partial differential equations with a non-unique solution over a non-rectangular domain properly.

With Neumann boundary conditions, boundary points are to be updated according to a slope, which means a relative update rather than an absolute update. This is numerically difficult to handle. Furthermore, the pupil is circular, while the discretization grid is square. Hence, the solution will call for accurate interpolations to obtain accurate results.

The initial signal out of the curvature sensor is the combination of two irradiance distributions as explained in Chapter 2, Section 2.1. These irradiances are recorded with any kind of detector, but the point is that we can expect them to be recorded with a very large spatial resolution, such as with a CCD array. However, for the reconstruction, we do not want the resolution to be so high, since first we want above all to correct low frequency terms, and second, we are limited by the noise propagation factor for large arrays as shown in the previous section. Therefore, we need to compress the information from a large array to a small one.

Another important consideration becomes clear when we look at a typical curvature signal. One such signal is shown in figure (3.3.4). Two CCD frames were taken on each side of the focus point, and were then combined. The resolution of the CCD is 512 pixels in each direction. Two disjoint zones are clearly visible.

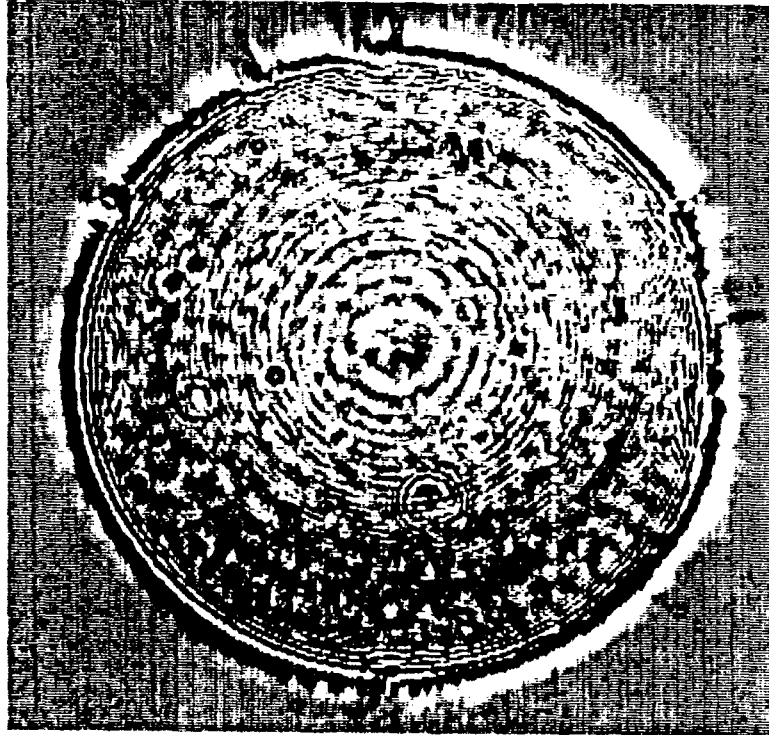


Figure (3.3.4) Typical curvature signal from an experiment.

A first zone is the center zone which covers almost the entire pupil. The signal here corresponds to the Laplacian of the wavefront, and represents the normalized difference between the two irradiances. Therefore, the amplitude of that signal is small compared to the irradiance values.

A second zone clearly visible is a ring around the edge of the pupil. That ring represents the signal stemming from the boundary conditions. It is produced by the non-overlapping of the edges of the two irradiances. Therefore, its amplitude is large over a small area. Theoretically, it is supposed to be a Dirac distribution.

It is obvious that, in order to obtain an accurate reconstruction, the algorithm must be able to treat each of these two zones differently.

We have explored several approaches before we were able to derive an algorithm which indeed gave satisfactory results for our special boundary value problem: a Neumann problem with circular domain for which the boundary slope is not *a priori* known, but must be computed as well. As we shall demonstrate, we indeed found an algorithm which reconstructs wavefronts from curvature signals, such as the one shown in figure (3.3.4), such that they match the true solution well.

This algorithm is currently coded in a Fortran program called NPOIS. Before starting with the iteration, NPOIS preprocesses the image data from a topological point of view, i.e., determines which points belong to the Laplacian domain (so-called solution points, or S-points), which points belong to the edge region (so-called boundary points, or B-points), which points are just outside the edge but still influence the solution (so-called close neighbor points, or C-points), and finally, which points are so far outside the domain that they can be ignored in the reconstruction (so-called outside points, or O-points).

NPOIS then computes the boundary slopes using an algorithm that will be explained in due course, and only then starts iterating on the Poisson equation using the normal derivatives at the boundary positions that were previously computed.

As stated earlier, the number of reconstruction points is much smaller than the initial number of points. This is the first decision to be made during the preprocessing phase. Let us call N_{in} the number of data along a pupil radius on the CCD curvature signal, and N_{out} the same quantity but for the reconstruction array. It is clear that N_{out} has to be much smaller than N_{in} , and that $(\frac{N_{in}}{N_{out}})^2$ CCD pixels will correspond to a single reconstruction point. Therefore, the Laplacian signal at every solution point is an average over a small surface of the CCD signal.

Now, let us look at the boundary conditions. To solve the Poisson equation, a discretization is needed. We decided to use N_{out} points along a radius. From this

data, if we assume a scaling such that the pupil obtains a unit radius, the sampling distance h must be equal to $1/N_{out}$. To correctly reconstruct the wavefront, the size of the normal derivative zone must be that particular h as shown on figure (3.3.5), where P stands for an interior point of the solution, and X denotes an exterior point with respect to the pupil. Hence, the routine must consider CCD data points as an edge slope derivative signal, if they lie on a ring with a normalized outer radius of $1 + \frac{1}{2N_{out}}$ and with an inner radius of $1 - \frac{1}{2N_{out}}$. If we consider CCD pixel coordinates, the radii will be $N_{in} \pm \frac{N_{in}}{2N_{out}}$.

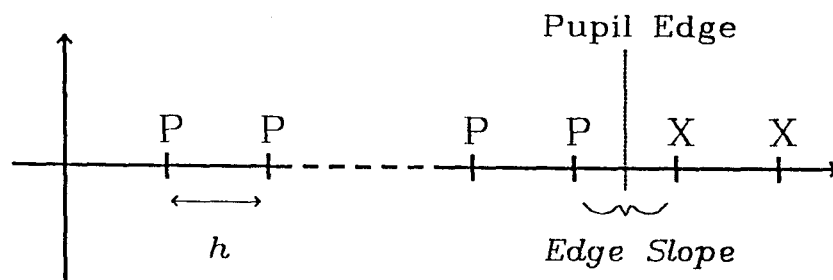


Figure (3.3.5) Discretization along a radius.

Therefore, the first task of the algorithm is to scan all the data points that come from the CCD array, and check whether they are inside the pupil, on the edge ring, or outside. A flag with three different values is set for each point.

Now, NPOIS knows which points have to be considered directly as Laplacian, which must be further processed to obtain slope values, and which can simply be ignored. The next move is to explore the reconstruction array, and perform some topological analysis on it.

Indeed, for all the solution points, the iteration process has to use the kernel equation (3.3.10) as explained in Section 3.3.b. Nonetheless, for solution points close

enough to the edge, it may happen that one or two neighboring points do not lie on the pupil. Those actually are the points that will provide the boundary conditions. To be able to use the kernel equation (3.3.10), all points must have four neighbors that are periodically updated. Therefore, NPOIS checks all inside points and will provide “close neighbor points” to those that are missing some neighbors. Figure (3.3.6) illustrates this concept of close neighbor points. On that figure, S stands for solution points, B denotes boundary solution points with missing neighbors, C marks close neighbor points, and O depicts outside points that can be totally ignored.

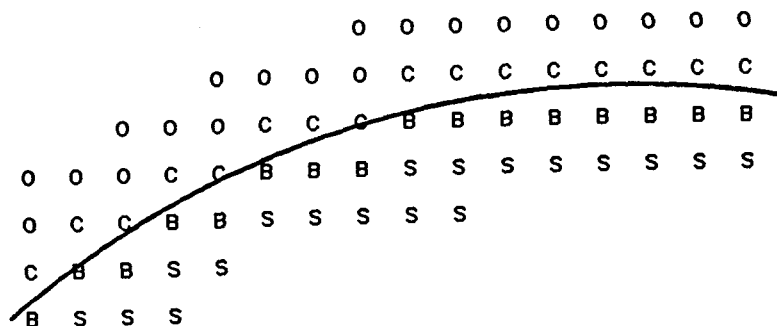


Figure (3.3.6) Example of S-, B-, C-, and O-points near the pupil edge.

At that point, the iterative process works smoothly for all the S- and B-points, but still, the problem is that the C-points also have to be updated.

From what we said so far about the Neumann conditions, it is clear that those C-points have to be updated according to the forcing slope information. This would be trivial if we worked with a square pupil. Figure (3.3.7) shows such an example. The update equation for any C-point would simply be

$$\phi(C) = \phi(B) + h \cdot Slope \quad (3.3.14)$$

where h is the sampling interval, B marks the next point on the left of that particular C-point, and $Slope$ denotes the local slope between the B- and the C-point.

S	S	S	S	B	C	0	0	0
S	S	S	S	B	C	0	0	0
S	S	S	S	B	C	0	0	0
S	S	S	S	B	C	0	0	0
S	S	S	S	B	C	0	0	0
S	S	S	S	B	C	0	0	0

(Figure 3.3.7) Simple case of the edge of a square pupil.

Let us now consider a circular domain. We want to find the update equation for any given C-point. If we draw a line \vec{n} , such that it is normal to the boundary and passes through C, it is clear that \vec{n} will also pass through the center of the pupil, but will probably not pass through any B-point.

It is possible to assign an angle variable to \vec{n} . This means that the location of the C-point from the center is described in polar coordinates. The same can be done for all B-points. If the size of the reconstruction array is given by N_{out} , then the number of B-points will approximately be $n_B = 2\pi N_{out}$. Hence we obtain n_B angle values that can be sorted from 0 to 2π .

Polar coordinates are required for the “support” point concept. To be updated, any C-point must have a neighbor in the direction toward the pupil center. As we said, there is no reason why a B-point should happen to be present in that direction. However, it is certain that two B-points, one on each side of \vec{n} , always exist. Therefore, by interpolating between those two B-points, it is possible to obtain an approximation for the value of the solution at any point P on \vec{n} which can then be used in the update equation for the C-point. The easiest way to find the best two candidates for supporting B-points is to look at the angle α of the C-point, and choose those B-points the angles of which are closest to α such that

one B-point has an angle that is just a little larger than α , say $\alpha + \epsilon_1$, while the other has an angle that is just a little smaller than α , say $\alpha - \epsilon_2$. This is why it is very convenient to have all the B-angles already sorted in an array. Figure (3.3.8) illustrates the support point concept.

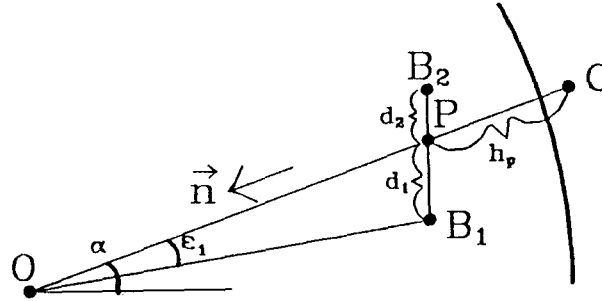


Figure (3.3.8) Example of two “support points” B_1 and B_2

The update equation for the C-point can now be written in a straightforward manner. Generalizing equation (3.3.14), we find:

$$\phi(C) = \phi(P) + h_p \cdot Slope = \frac{d_2\phi(B_1) + d_1\phi(B_2)}{d_1 + d_2} + h_p \cdot Slope \quad (3.3.15)$$

where B_1 and B_2 are the two support points, d_1 and d_2 their distances to P , and h_p the distance from P to C .

In order to compute d_1 , d_2 , and h_p we need the coordinates of P . We can find them by noting that P is at the intersection of \vec{n} with $\overrightarrow{B_1B_2}$. The equations of those lines are:

$$y = \frac{y_C}{x_C} x \quad (3.3.16a)$$

$$y = \frac{y_{B_2} - y_{B_1}}{x_{B_2} - x_{B_1}} (x - x_{B_1}) + y_{B_1} \quad (3.3.16b)$$

where (x_{B_1}, y_{B_1}) , (x_{B_2}, y_{B_2}) , and (x_C, y_C) are the coordinates of B_1 , B_2 , and C respectively.

Solving for the intersection coordinates x_P and y_P yields:

$$x_P = \frac{x_C(y_{B_1}x_{B_2} - x_{B_1}y_{B_2})}{(x_{B_2} - x_{B_1})y_C - (y_{B_2} - y_{B_1})x_C} \quad (3.3.17a)$$

$$y_P = \frac{y_C(y_{B_1}x_{B_2} - x_{B_1}y_{B_2})}{(x_{B_2} - x_{B_1})y_C - (y_{B_2} - y_{B_1})x_C} \quad (3.3.17b)$$

To clarify, let us summarize the complete procedure to obtain the update equations for all C-points.

- Obtain all the B-points for a given array and sort them according to their angle.
- For any C-point, compute its angle, and find the two closest B-angles embracing the C-angle. Those two B-points are called the two support points.
- Obtain the coordinates of the P point by computing the intersection of \vec{n} with $\overrightarrow{B_1B_2}$.
- Compute the values d_1 , d_2 , and h_p , and then the coefficients $K_1 = \frac{d_2}{d_1+d_2}$ and $K_2 = \frac{d_1}{d_1+d_2}$.
- The update equation is:

$$\phi(C) = K_1\phi(B_1) + K_2\phi(B_2) + h_p \cdot Slope \quad (3.3.18)$$

Note that it is not necessary to compute an explicit estimate of the function value at P . NPOIS will simply store the three coefficients K_1 , K_2 , and h_p during the preprocessing phase for use during the iterations. Those coefficients are constant, and therefore, it would be a waste of time to recompute them ever again.

The last point that is still missing for the algorithm to be complete is how to obtain the *Slope* factor. If we knew the boundary conditions as true Neumann conditions, nothing more would be needed, but we actually do not know these slopes precisely. All we know is the quantity $\frac{\partial}{\partial n}\phi(z)\delta_c$. It is clear that we wish to remove

the Dirac term. This can be accomplished if we remember the sifting property of the $\delta(x)$ distribution. For every function $f(x)$, it is true that:

$$\int_{-\infty}^{+\infty} f(x)\delta(x - x_o)dx = f(x_o) \quad (3.3.19)$$

Similarly, we can integrate the boundary signal from the CCD along its normal direction. As we said previously, that signal is within a ring the inner radius of which is $1 - \frac{1}{2N_{out}}$ while its outer radius is $1 + \frac{1}{2N_{out}}$. NPOIS already set a flag for those values. Integrating along this route, we can find the integral, and the result we were looking for:

$$\int_{\rho_{min}}^{\rho_{max}} \frac{\partial}{\partial n} \phi(z) \delta_c(\rho - 1) d\rho = \left. \frac{\partial}{\partial n} \phi(z) \right|_{\rho=1} = Slope \quad (3.3.20)$$

where $\rho_{min} = N_{in} - \frac{N_{in}}{2N_{out}}$ and $\rho_{max} = N_{in} + \frac{N_{in}}{2N_{out}}$.

Now, we still need to evaluate the normal derivative. The data on the ring are used by approximately n_B C-points, since there are about the same number of C-points as B-points. Therefore, each C-point is influenced by all points that have similar angles within $\pm \frac{\pi}{n_B}$. Hence, our 1D integration can be translated into a 2D sum with an averaging coefficient:

$$Slope = \frac{\sum_{\theta_{min}}^{\theta_{max}} \sum_{\rho_{min}}^{\rho_{max}} curv(\rho, \theta)}{\sum_{\theta_{min}}^{\theta_{max}} \sum_{\rho_{min}}^{\rho_{max}} 1} \quad (3.3.21)$$

where $curv(\rho, \theta)$ is the direct signal from the CCD array, ρ_{min} and ρ_{max} the same as for the integral (3.3.20), and θ_{max} and θ_{min} are equal to $\alpha_C \pm \frac{\pi}{n_B}$.

This is the procedure to obtain the *Slope* factor. Of course, this algorithm too can be executed once and for all during the preprocessing phase and can be

stored away for each of the C-points. This concludes the description of the preprocessing phase.

For the iterative phase, NPOIS knows which points are to be directly updated with the kernel equation (3.3.10), and which points are close neighbor points that have to be updated with the special equations the coefficients of which are in memory. During the iteration process, NPOIS updates all the inside points first, and then computes all the close neighbor points.

Once per given number of iterations, usually about 100, NPOIS computes the *rms* value of the reconstruction. It assumes that the solution has converged if the difference between this *rms* value, and the previously computed one is smaller than a given accuracy factor. When convergence has been achieved, the final solution is used as the reconstructed wavefront the curvature signal of which was the curvature sensor output.

Let us now look at an example for reconstruction. The signal we looked at in figure (3.3.4), as we already said, was taken with a CCD. The optical setup consisted of a light point source, an achromat (a pair of lenses) considered as the main optical element, and a CCD that was moved from one side of the focus plane to the other in order to take the two pictures. The exposure time was in the seconds range, and since it was taken in a quiet lab, there was no distortion on the images due to atmospheric turbulence. Therefore, the only aberrations were due to the lens itself. Hence, our curvature sensor along with the complex reconstruction algorithm can precisely measure aberrations of optical elements. The achromat had a focal length $f = 120\text{cm}$, a radius $r = 3\text{cm}$, and the CCD planes P_1 and P_2 were at a distance $\ell = 2\text{cm}$ from the focus point.

Running NPOIS produced an estimate of the aberrations of the optical element itself. The errors on its surface are shown in figure (3.3.9). It shows

essentially a coma aberration probably produced by a small error in the centering of the lens components.

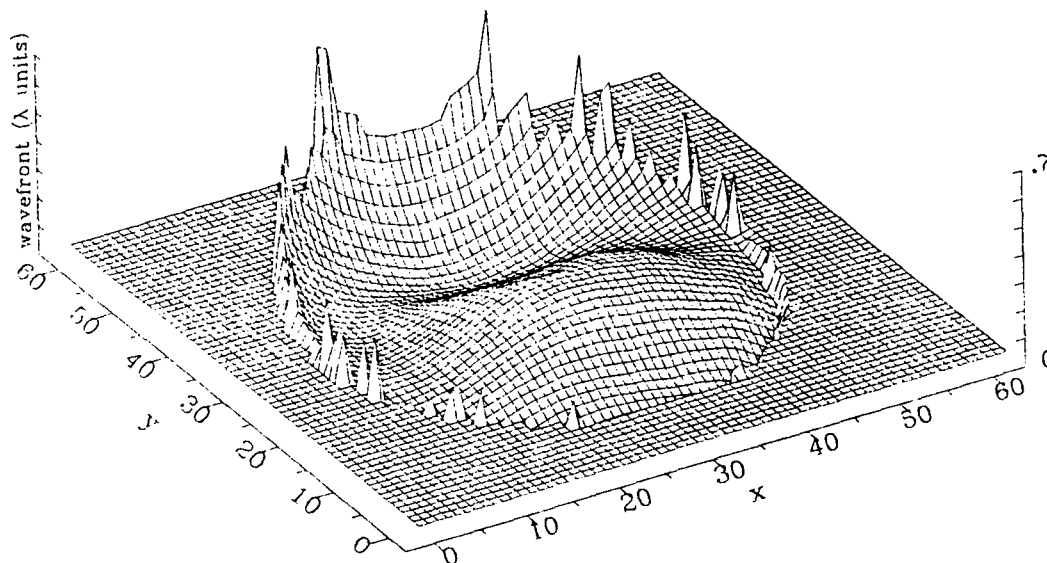


Figure (3.3.9) Reconstruction of the errors on the wavefront at the output of an achromat.

The NPOIS program provides us with a very quick and easy way to obtain both qualitative and quantitative information on optical elements.

3.3.d. Extension of NPOIS to include central obscuration.

The next step consists of pursuing this new method to test optical equipment a little further before going back to the open loop simulation for adaptive optics systems.

From the previous section, it was clear that NPOIS was perfect to reconstruct lens deformations from the information taken out of two out-of-focus images.

Unfortunately, lenses are not really of a huge interest for testing. Indeed, usually, they are of relatively small diameter, rarely larger than a foot, and people know quite well how to make them. On the other hand, what is still hard to make are telescope mirrors. They can be as much as 6 meters in diameter from a single piece of glass, and due to this large area and weight, it is impossible to produce them with as accurate a surface as desired. Therefore, it would be very convenient if it were possible to perform on the telescope mirror the same kind of testing as we did for the lens. Unfortunately, most telescopes are made such that their mirrors have a big hole right in the middle, which produces a central obstruction. This central obstruction imposes a new boundary condition on our Poisson equation. Therefore, it is clear that NPOIS cannot be used as is for testing telescope mirrors.

If we compute a curvature signal as before, taking two images on each side of the focus plane, and evaluating a normalized difference, we shall obtain the same signal as before, with an additional high intensity circular δ_c distribution at the position of the central obscuration due to the non-overlapping of the hole.

Therefore, with our special algorithm, we need to handle not only the outer ring, but also this new inner ring. Due to the modularity of the NPOIS program developed previously, the extended version, called CPOIS, was very easy to develop. As before, we must create S-, B-, C-, and O-points for the inside border as well. An example is presented in figure (3.3.10). The rest of the procedure to obtain the update equations for the close neighbor points is the same, except we now must consider the normal derivative to be directed toward the mirror center for the inner ring, and toward the outside for the outer ring.

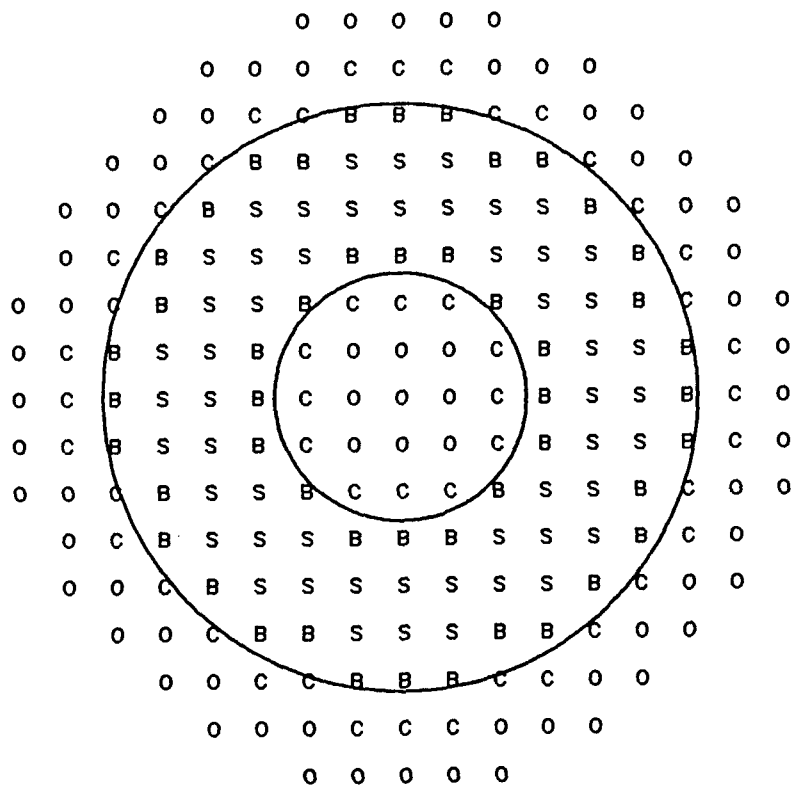


Figure (3.3.10) Example of S-, B-, C-, and O-points for the case with two boundary signals.

To test CPOIS, we used CCD frames of out-of-focus images from the Canada-France-Hawaii (CFH) Telescope located on Mauna Kea in Hawaii. Originally, those pictures were taken to be processed with another algorithm (C. Roddier, 1989), but it was worthwhile doing some work in parallel and checking for result discrepancies. The tested primary mirror is 3.6 meters in diameter, its focal length is 13.5 meters, and the pictures were taken at 10cm from its focus point.

Both CCD pictures look basically like figure (3.3.11).

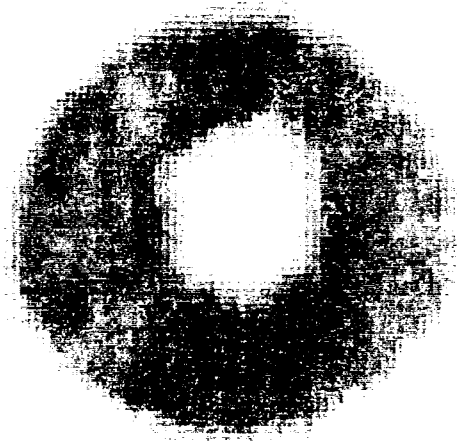


Figure (3.3.11) A typical out-of-focus image from the CFH Telescope.

The corresponding curvature signal is shown in figure (3.3.12). We can clearly see the zone of the hole, the curvature zone, and both the inner and outer edge slope signal zones.

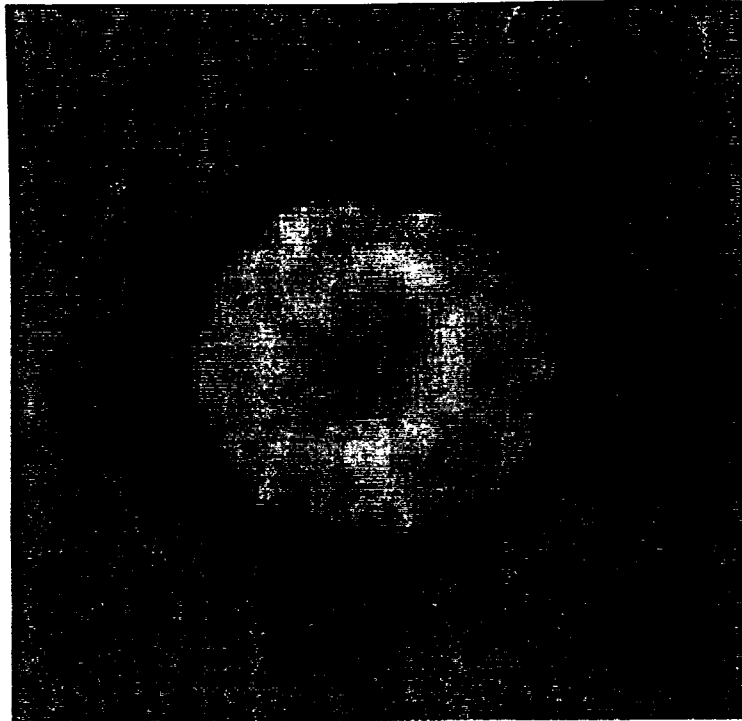


Figure (3.3.12) Curvature signal from the CFH Telescope.

The reconstruction with CPOIS, and the Zernike decomposition on the corresponding reconstructed wavefront could be compared with another approach of interpreting the same out-of-focus images, still using the same curvature sensing concept, but without a complete reconstruction (C. Roddier, 1989).

To summarize the other algorithm: it determines the Zernike coefficients directly from the images or from the curvature signal, and corrects the images or the signal directly by subtracting one after the other of the newly found Zernike polynomials, to make it possible to determine the next coefficient. There is no accumulation error because all the Zernike polynomials are orthogonal to each other.

From two typical out-of-focus images, taken at the same distance on each side of the focus point, it is possible to directly determine the amount of tilt, the defocus, and the spherical aberrations without computing the curvature signal first. Indeed, a shift corresponds to a tilt, a change of pupil size corresponds to a defocus, and a change of ratio of the outer diameter to the hole diameter corresponds to a spherical aberration. Hence, just from the two independent pictures, it is possible to extract qualitative as well as quantitative informations on the Zernike polynomials Z_2 , Z_3 , Z_4 , and Z_{11} . Once known, these polynomials can be subtracted from the pictures by means of shift (tilt), overall reduction or expansion (defocus), and special mapping (spherical).

At that point, the curvature signal is computed in the usual fashion, i.e., a least square fit of a plane through the Laplacian zone provides the amplitude and orientation of the pure coma term (Zernike polynomials Z_7 and Z_8). Again, these are immediately subtracted including their corresponding edge signals.

The last sequence of the algorithm consists of decomposing the edge signal into a Fourier series. The first harmonic is zero since it corresponds to tilt aberration that has already been removed, the second harmonic corresponds to astigmatism (Z_5 and Z_6), and the third harmonic corresponds to the zero-curvature coma (Z_9 and Z_{10}).

In this way, it is possible without solving the Poisson equation at all to determine the Zernike coefficients up to Z_{11} for the telescope mirror aberrations.

The results obtained in this manner were compared for both lenses and telescope mirrors with those determined by NPOIS and CPOIS and were found to be approximately 90% to 95% compatible.

While CPOIS was not really needed for our overall problem, it was a useful exercise to develop the code since it helped verify the correctness of the needed

NPOIS code. Moreover, both methods could be used in parallel, and it may also be possible to improve the reconstruction even further by combining the two methods, i.e., by subtracting some Zernike polynomials first, and then solve the Poisson equation for the remaining ones. However, this idea has not yet been pursued any further.

3.4. Open loop simulation.

We now have explained how to write simulation routines for each of the system elements. The open loop operation starts from a known simulated wavefront, and ends up with a reconstructed one, that must match as closely as possible the first one. In between, the wavefront has been decomposed into Laplacians and edge slopes.

Working with several different aberrations and amplitudes, we can study non-linear effects and noise due to photons and diffraction.

3.4.a. Study of non-linearities.

It is no surprise that the main non-linear problem of this system arises at the edges. Especially for the reconstruction, NPOIS expects the edge signal to be within a ring of a certain diameter for a given reconstruction resolution.

What if that edge signal is larger than the ring? Some of it will clearly escape on the exterior side and get lost in a zone that is never scanned by the algorithm. Or, on the interior side, some high intensity signal can interfere with a low intensity Laplacian signal. It is obvious that we cannot expect the result to be good for that kind of a situation.

The edge signal may become large if either the distance of the photodetectors to the focus point is too small, or if the aberration amplitude is too large. In

a real situation, we can expect ℓ to be fixed, and therefore, we would be faced with this too large edge signal problem only when the aberration amplitude becomes too large for a given optical setup.

Therefore, we have to run some simulations to study this problem more carefully. Since we do not want to be bothered by any additional noise, we have to use the mathematical routine described in Section 3.2.a to simulate the wavefront sensor.

About aberrations, we again choose tilt, defocus, and astigmatism since for them we know the corresponding edge signal width W for a given situation and amplitude coefficient ε . It is straightforward to show by using equations (3.2.8), (3.2.9), and (3.2.10) that this width W will be respectively for a tilt, a defocus, and an astigmatism:

$$W_{\text{tilt}} = \frac{2\varepsilon(f - \ell)f}{\ell} \quad (3.4.1a)$$

$$W_{\text{defocus}} = \frac{8\varepsilon(f - \ell)f}{\ell} \quad (3.4.1b)$$

$$W_{\text{astigmatism}} = \frac{4\varepsilon(f - \ell)f}{\ell} \quad (3.4.1c)$$

We now have to choose typical values for all the parameters, including the reconstruction resolution, and perform open loop simulations with different aberration amplitudes to observe if a breakpoint is indeed present when the edge signal becomes larger than that which NPOIS expects.

With the exception of the non-linear effect we want to study, the rest of the procedure is completely linear. Therefore, we do not have to be very careful about values for parameters as long as they are within a typical range. We chose $f = 10$ and $\ell = 1$ for the optical setup. For the reconstruction, we chose 10 points on a radius where, as shown in Section 3.3.b, the error propagation is still acceptable.

Therefore, under those conditions, we can expect to obtain a breakpoint when $W = 0.1$, which implies for the amplitudes:

$$\epsilon_{break}^{tilt} = 50 \cdot 10^{-5} \quad (3.4.2a)$$

$$\epsilon_{break}^{defocus} = 12.5 \cdot 10^{-5} \quad (3.4.2b)$$

$$\epsilon_{break}^{astigmatism} = 25 \cdot 10^{-5} \quad (3.4.2c)$$

The procedure for the simulation is very simple. We simulate the wavefront by evaluating the correct Zernike terms with the corresponding amplitudes, run the mathematical routine to get a perfect curvature signal, run the reconstruction program to compute an estimate of the initial wavefront, and compare them. The results are in tables (3.4.1a), (3.4.1b), and (3.4.1c) for the three different aberrations.

Table (3.4.1a) Linearity for tilt.

Initial amplitude	Reconstructed amplitude
$10 \cdot 10^{-5}$	$9.99 \cdot 10^{-5}$
$20 \cdot 10^{-5}$	$20.13 \cdot 10^{-5}$
$30 \cdot 10^{-5}$	$29.80 \cdot 10^{-5}$
$40 \cdot 10^{-5}$	$39.88 \cdot 10^{-5}$
$50 \cdot 10^{-5}$	$50.08 \cdot 10^{-5}$
$60 \cdot 10^{-5}$	$59.06 \cdot 10^{-5}$
$70 \cdot 10^{-5}$	$65.94 \cdot 10^{-5}$
$80 \cdot 10^{-5}$	$73.01 \cdot 10^{-5}$

Table (3.4.1b) Linearity for defocus.

Initial amplitude	Reconstructed amplitude
$1 \cdot 10^{-5}$	$0.96 \cdot 10^{-5}$
$2 \cdot 10^{-5}$	$1.91 \cdot 10^{-5}$
$4 \cdot 10^{-5}$	$3.73 \cdot 10^{-5}$
$6 \cdot 10^{-5}$	$5.65 \cdot 10^{-5}$
$8 \cdot 10^{-5}$	$7.36 \cdot 10^{-5}$
$10 \cdot 10^{-5}$	$9.31 \cdot 10^{-5}$
$15 \cdot 10^{-5}$	$12.61 \cdot 10^{-5}$

Table (3.4.1c) Linearity for astigmatism.

Initial amplitude	Reconstructed amplitude
$2 \cdot 10^{-5}$	$2.17 \cdot 10^{-5}$
$5 \cdot 10^{-5}$	$5.02 \cdot 10^{-5}$
$10 \cdot 10^{-5}$	$9.91 \cdot 10^{-5}$
$20 \cdot 10^{-5}$	$20.24 \cdot 10^{-5}$
$30 \cdot 10^{-5}$	$29.73 \cdot 10^{-5}$
$40 \cdot 10^{-5}$	$35.72 \cdot 10^{-5}$
$50 \cdot 10^{-5}$	$42.23 \cdot 10^{-5}$

The corresponding plots are in figures (3.4.1a), (3.4.1b), and (3.4.1c) respectively.

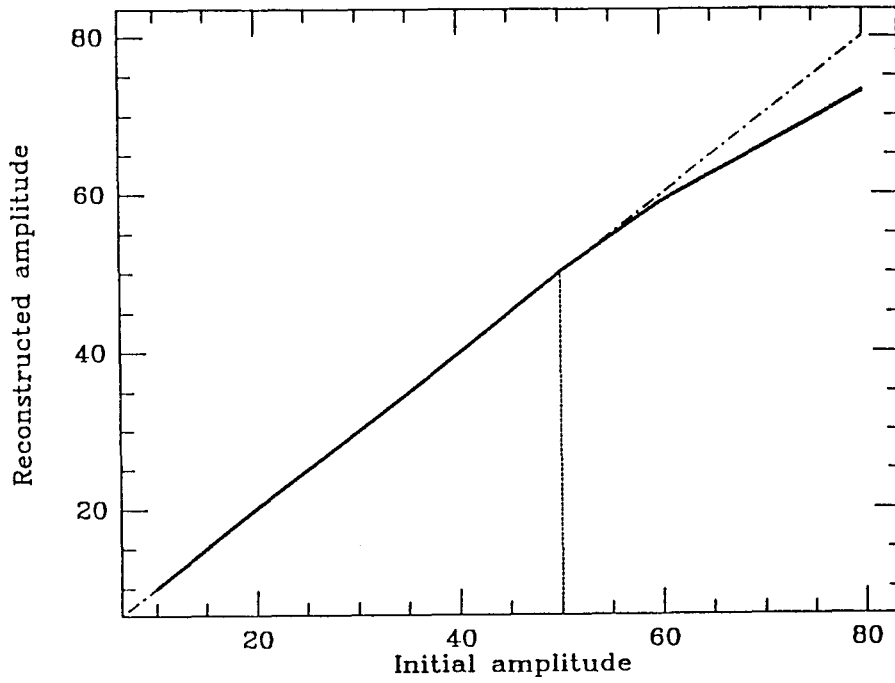


Figure (3.4.1a) Linearity for tilt. (10^{-5} radius units)

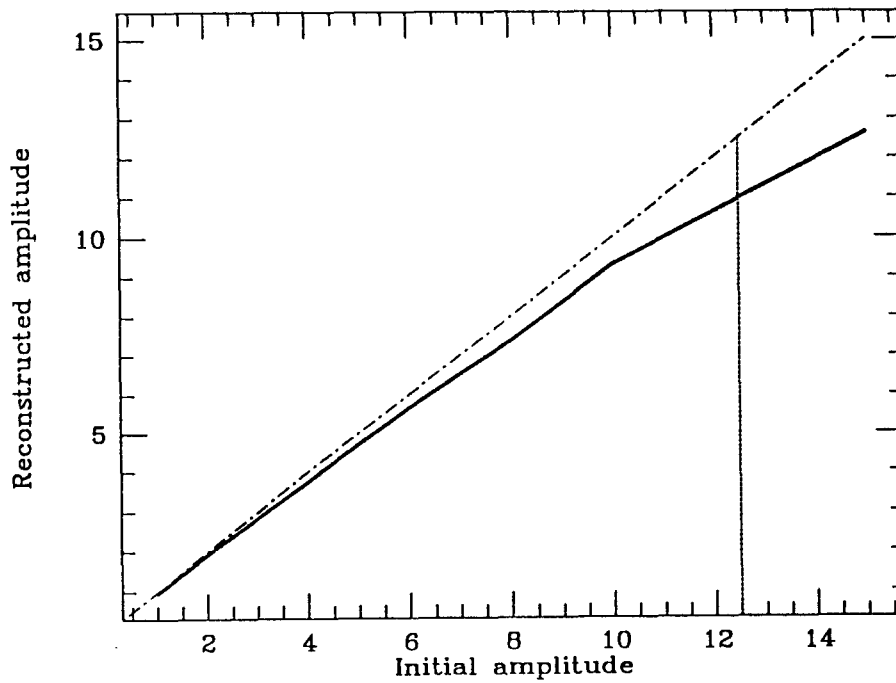


Figure (3.4.1b) Linearity for defocus. (10^{-5} radius units)

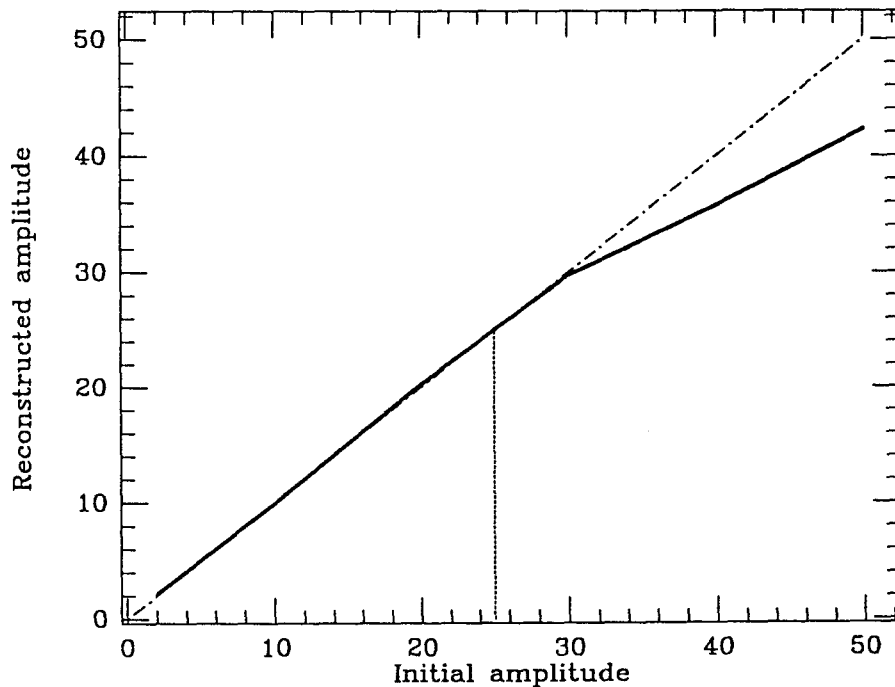


Figure (3.4.1c) Linearity for astigmatism. (10^{-5} radius units)

From the plots, we clearly see that the foreseen breakpoints are indeed present, and almost where expected. Probably, for the tilt and astigmatism, we could not hope for a picture that matched our predictions as well. However, for defocus, we see a systematic error, even before the breakpoint region, an error that we had not anticipated. It is not clear at this point where this error comes from. However, it is rather small and therefore, in practice, not very important.

From these runs, we conclude that the non-linear effect we studied is not dramatic. The system after the breakpoint seems to under-estimate the aberration, rather than over-estimate it. This point is important since we do not want to have oscillations in the later closed loop system.

3.4.b. Study of photon noise.

Now we have a better idea of the intrinsic non-linearities of the system, we can look at some external noise.

Photon noise exists due to the fact that the irradiance distribution is not continuous. The light intensity depends on the number of photons that fall on the detector, and this is a discrete number. Furthermore, for photons coming from a star, the distribution over the different detectors is surely totally random, and it may happen that some detectors receive much more photons than others for a given sampling period. This of course creates bad estimates, and therefore errors in the reconstruction.

As the average number of photons increases, the uncertainty of the estimate decreases. Hence, it is very important to know on the average how many photons are required to have an acceptable error.

For that purpose, we made simulations using the ray tracing routine. Each ray photon started from a uniformly distributed random position on the wavefront. The impact point was considered as the photon impact. The detector array has eight points on a pupil radius which corresponds roughly to 200 points on the reconstruction. We made runs for five different aberrations with three different photon noise coefficients for each. We sent 5000, 10000, or 20000 photons overall, which correspond to 25, 50, or 100 photons per detector respectively.

The amplitude coefficient for the aberration was chosen such that the algorithm would not produce any non-linearity in the reconstruction. The amplitude was set such that the edge signal width was about half the ring width, about the optimal case for NPOIS.

The setup was exactly as before: $f = 10$, $\ell = 1$, with 10 points for reconstruction. The aberrations we chose to work with were tilt, defocus, astigmatism,

coma, and spherical. The corresponding aberration amplitudes for an edge signal of half the ring size were:

$$\begin{aligned} \varepsilon_{tilt} &= 25 \cdot 10^{-5}, & \varepsilon_{defocus} &= 6.25 \cdot 10^{-5}, & \varepsilon_{astigmatism} &= 12.5 \cdot 10^{-5} \\ \varepsilon_{coma} &= 3.57 \cdot 10^{-5}, & \varepsilon_{spherical} &= 2.08 \cdot 10^{-5} \end{aligned} \quad (3.4.3)$$

Our indicator parameter is the ratio of the r.m.s. value of the residual error over the initial wavefront r.m.s. value. Runs were made for 25, 50, and 100 photons per detectors. The results are summarized in table (3.4.2).

Table (3.4.2) Residual errors when photon noise is present

Aberration	25 ph.	50 ph.	100 ph.
tilt	.24	.08	.02
defocus	.25	.10	.05
astigmatism	.28	.10	.03
coma	.31	.15	.09
spherical	.33	.19	.14

The corresponding plots are in figure (3.4.2).

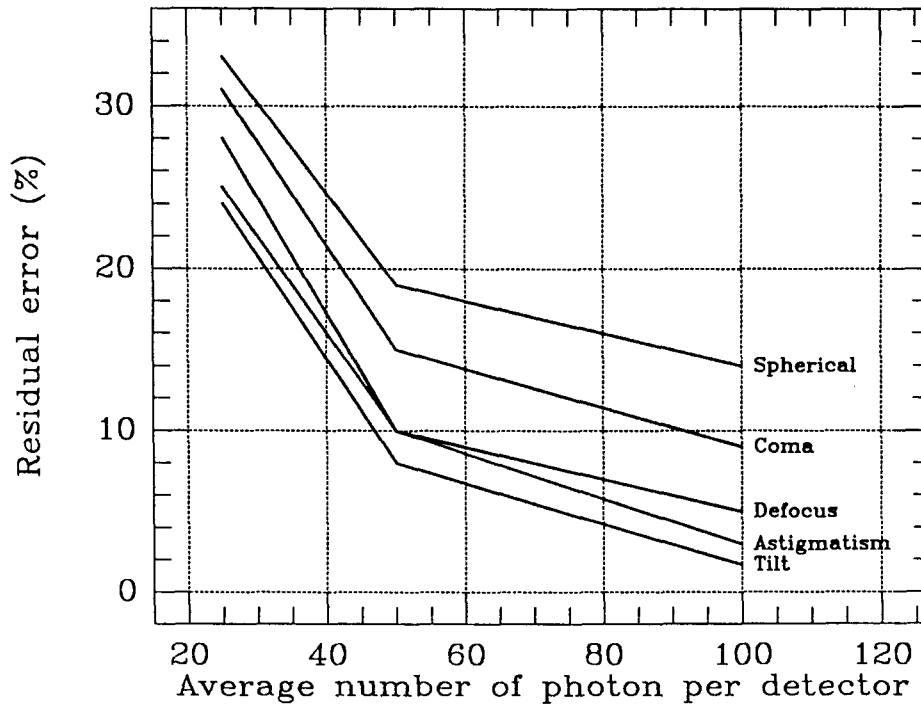


Figure (3.4.2) Residual error when photon noise is present

It is clear that the system response is not satisfactory for 25 photons. It becomes acceptable for about 50 photons per detector.

3.4.c. Study of diffraction noise.

This last non-ideal characteristic of the open loop system was more difficult to study exhaustively. Indeed, the way to go should have been similar to the one for the study of photon noise, except that the sensor simulation routine should now have been the one described in Section 3.2.c rather than the previous ray tracing method. However, even if theoretically the tough part of the routine was only to evaluate two 2D FFTs, to do the complete study would have required doing 2048×2048 FFTs!

Although our FOURT routine was able to handle this, the computer space and calculation time required to do it was out of our possibilities. Therefore, we

did not perform a complete study, but had to be content with a number of worst case simulations instead.

If we look at a typical setup for the operations, such as the two examples presented in Section 3.3.c and 3.3.d, we can obtain a rough idea for the values of the involved parameters. Assuming a 2 meter telescope, and working with a unit pupil radius (1 meter) for the main optical element, we find roughly the following values for the other parameters:

$$f \approx 20, \quad \ell \approx 0.05, \quad \lambda \approx 0.5 \cdot 10^{-6} \quad (3.4.4)$$

which imply in the FFT equations (3.2.17a) and (3.2.17b) an angle for the Fresnel terms of about:

$$\pi \frac{\ell \rho^2}{\lambda f(f - \ell)} = \pi \frac{0.05 \cdot 1.0^2}{0.5 \cdot 10^{-6} \cdot 20 \cdot 20} = 250 \cdot \pi = (125)(2\pi) \quad (3.4.5)$$

Therefore, if we want to sample this correctly, we need to sample at least at the Shannon limit which means a strict minimum of $2 \cdot 125$ points for a pupil radius, or at least 500 pixels across the pupil diameter. To get a little away from this theoretical limit, we should at least double that number, i.e., we need 1000 pixels.

Now, when taking an FFT, we must also include that amount of zeros on the side for zero-padding. Hence, we need a 2048×2048 FFT. It is therefore clear that this approach is not reasonable. Fortunately, we were able to find an alternative approach which still allowed us to study diffraction noise.

What we want to study is how diffraction affects and deforms the estimates and reconstructions. If we choose the setup parameters so that the diffraction is

much larger than that of a typical real situation, we can study the diffraction noise limitation, and we may be quite close to the true limit.

Changing the parameters so that diffraction would be worse than for a real situation means to either decrease ℓ or increase λ . Hopefully, changing any of those two parameters in this way will reduce the required sampling, and thus the FFT dimension.

We decided to work with complex arrays of size 256×256 for the FFT, which was the largest size we could handle on our VAX if we still wanted to have some space available out of the allocated disk quota, and if we did not want to wait too long for a single simulation to complete. Under those conditions, we had to reduce the quantity $\lambda \cdot \ell$ by at least a factor of 10. We therefore moved the photodetectors closer to the system focus point by a factor of 5 from their typical positions, and simulated infrared light with a wavelength three times larger than the usual visible light.

With such a configuration, the setup is in the worst situation that possibly could happen during regular operations.

Even in this setup, the requirements for the simulation could not be ignored, and we limited ourselves to study tilt and astigmatism aberrations only. One simulation run required roughly a quarter of an hour of CPU time on a VAX-11/750.

The images produced over the planes P_1 and P_2 for a mid-size tilt are disks with diffraction rings. A typical such image is shown in figure (3.4.3).

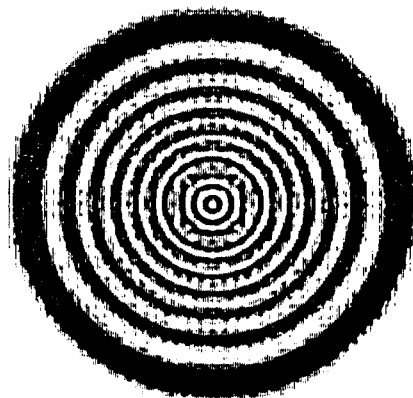


Figure (3.4.3) Typical image due to a single wavefront tilt.

The irradiance distributions I_1 and I_2 differ from each other by a simple shift, as could be anticipated knowing the nature of the tilt aberration. Cross-sections right through the middle of each of the two images are shown in figures (3.4.4a) and (3.4.4b). We clearly see that they are indeed quite the same except for the shift. We also clearly see the variation in the intensity as a function of the distance from the center due to interference.

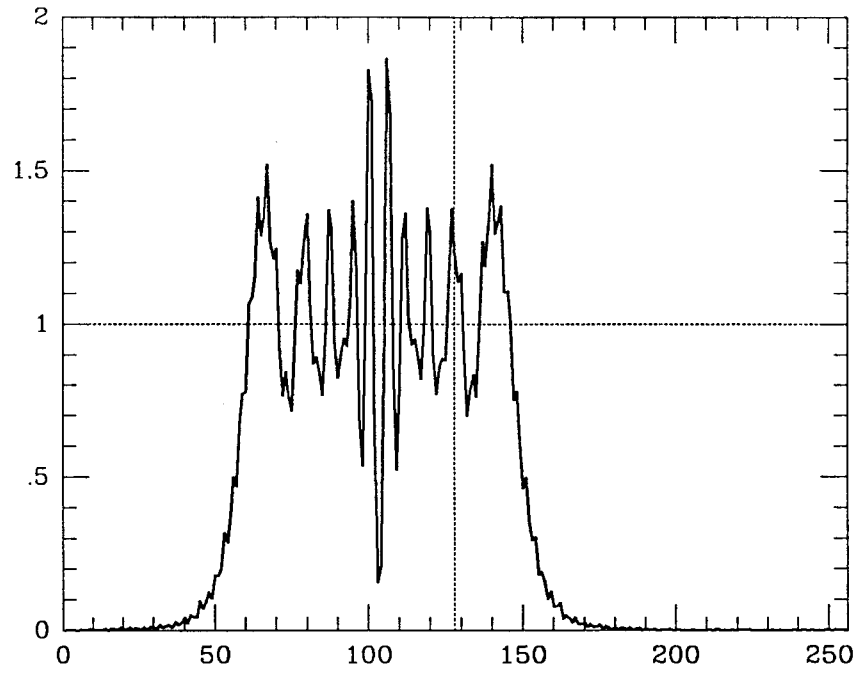


Figure (3.4.4a) Cross-section through the left image.

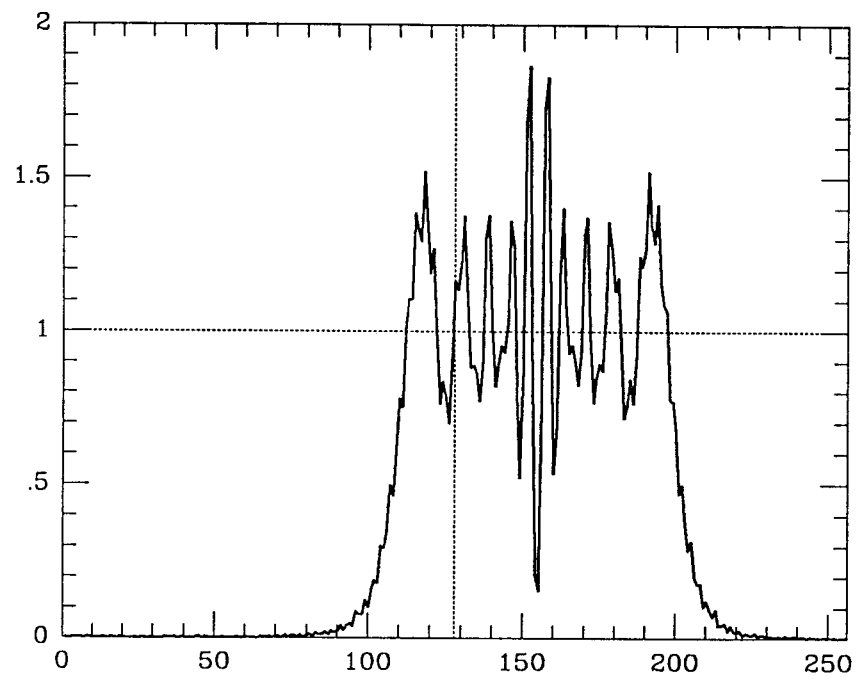


Figure (3.4.4b) Cross-section through the right image.

Now, combining these two irradiance distributions (difference divided by normalizing sum) gives rise to the curvature signal. Such a signal, as shown in figures (3.4.5) and (3.4.6), looks indeed very noisy. The Laplacian zone is no longer a flat "O".

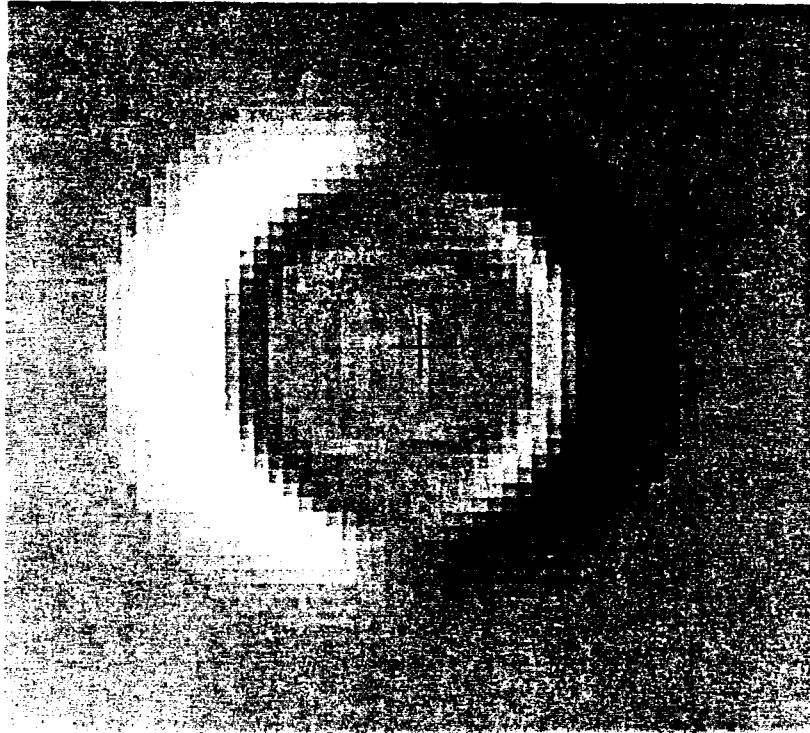


Figure (3.4.5) Curvature signal for the tilt.

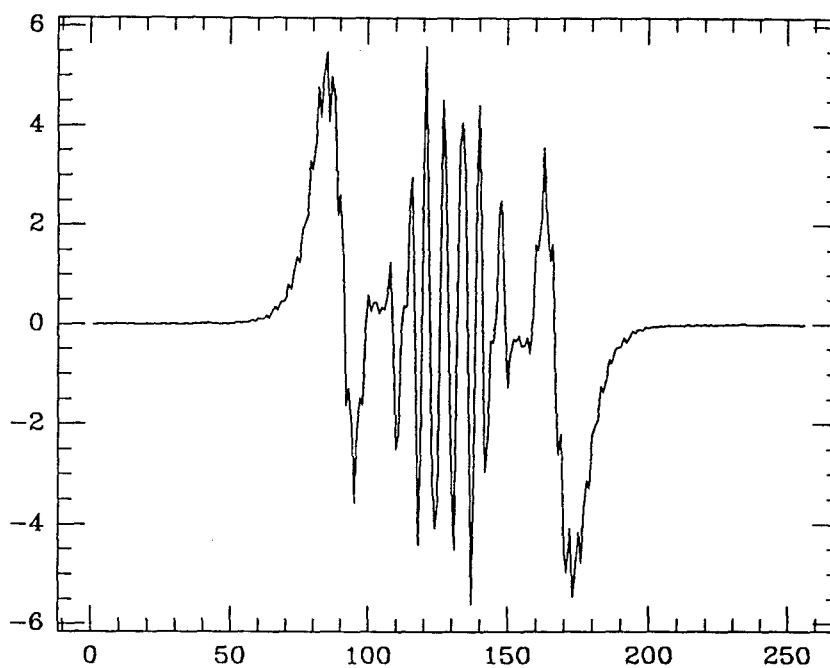


Figure (3.4.6) Cross-section of the curvature signal for the tilt.

Running the reconstruction algorithm on this “bad” signal produces an unexpected result: The diffraction noise is completely filtered out, and the reconstructed wavefront, as shown in figure (3.4.7), is a pure tilt with the correct amplitude within $\pm 1\%$.

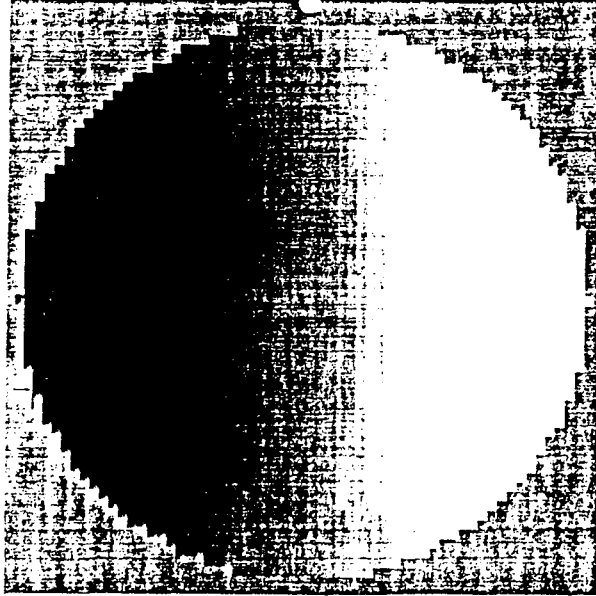


Figure (3.4.7) Reconstruction of the wavefront.

An explanation for this unanticipated good result is that fringes have mostly high spatial frequency components compared to the aberration frequency and sampling for the reconstruction. Therefore, when solving the Poisson equation which actually performs a second order integration, the diffraction noise is smoothed out. In the Fourier plane, solving a Poisson equation corresponds to an $1/\omega^2$ attenuation, therefore completely eliminating high frequency noise. This is a good point for curvature sensing versus classical sensing, since for a Hartmann sensor, the attenuation would be only $1/\omega$.

For the case of astigmatism, there is not much new in addition to the conclusions drawn from the tilt runs. A typical astigmatism irradiance distribution is shown in figure (3.4.8). It looks similar to what could be expected: an ellipse with diffraction rings. The other image over the second plane is very similar except for a 90° rotation of the ellipse axes.

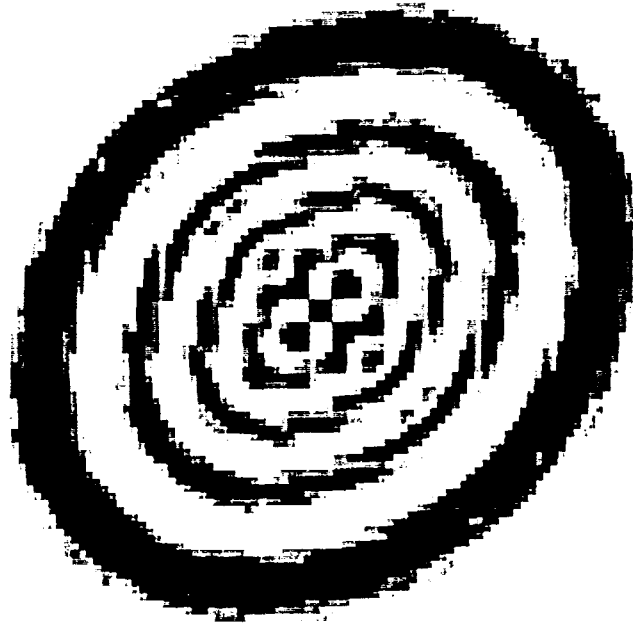


Figure (3.4.8) Astigmatism image.

The corresponding curvature signal is shown in figure (3.4.9). Similarly to the tilt case, the inside Laplacian signal no longer is zero, but contains some noise instead.

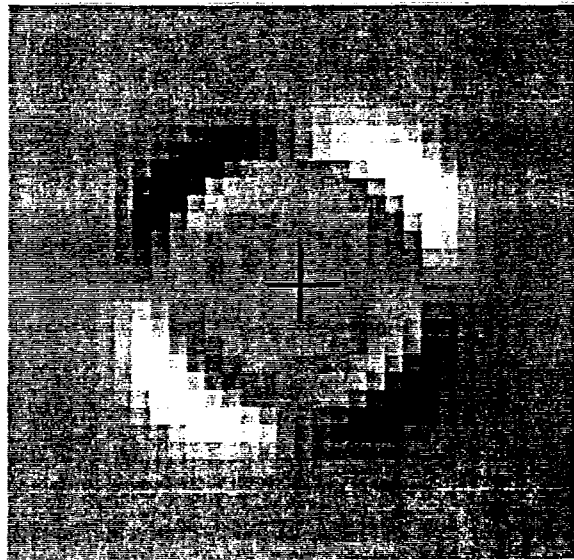


Figure (3.4.9) Astigmatism curvature signal.

The reconstruction wavefront, figure (3.4.10), as expected from the “good” results for the “tilt”, is a pure astigmatism with a correct amplitude also within $\pm 1\%$.

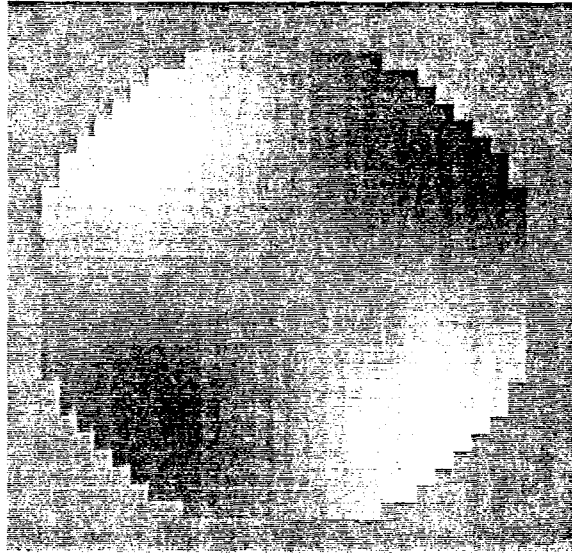


Figure (3.4.10) Astigmatism reconstruction.

3.5. Chapter summary.

In this chapter, we first analyzed the input, i.e., the atmosphere. Thereafter, we looked at the process of wavefront detection, i.e., the sensor. Finally, we described the wavefront reconstruction, in our case this consisted of solving the Poisson Equation. These three processes together describe the open loop simulation which we then applied to our physical configuration to analyze the behavior of our overall system in open loop.

The atmosphere had to be described using a mathematical model. For this purpose, the Zernike polynomials were introduced. We then derived an algorithm that allows us to simulate random atmospheric wavefronts with correct amounts of variance and covariance for all the Zernike terms.

The next step of our work consisted of simulating the wavefront sensor. Three different algorithms were derived, and corresponding procedures were implemented in computer code. Each of these algorithms was designed to deal with a different situation. A mathematical routine provided the perfect theoretical signal that we could expect out of the sensor. A ray tracing routine provided the response of the system under the influence of a finite number of photons. Finally, a routine using the Fresnel theory was coded for studying the effects of diffraction noise.

The third step was concerned with the process of wavefront reconstruction. An estimate of the noise propagation through this second order reconstruction was presented in two different ways. The error seems to increase as the square of the number of solution points. The Poisson equation had to be solved with special Neumann boundary conditions: the boundary conditions were normal slopes, the values of which were, however, not known in advance. A program called NPOIS was written to handle this problem. In addition, an extension of this program, called CPOIS, was also developed for the purpose of analyzing the central obstruction of telescope mirrors. This program has most of its utility for mirror testing.

The last step of our study was to run open loop simulations to carefully study the complete system. Side effects or secondary effects consisted first of nonlinearities due to our non-ability to properly handle the edge signal when it becomes too large. The second study was devoted to problems related to photon noise. Finally, we studied the effects of diffraction noise. However, neither of these effects are critical for typical configurations as they may be used in astronomical observations.

CHAPTER 4

CLOSED LOOP SYSTEM SIMULATION

The open loop response of our curvature sensing based adaptive optics system has been studied, including the problems of noise and scalability. It seems that for astronomical applications, none of these non-ideal behaviors are a major drawback. On the contrary, curvature sensing showed advantages in comparison with classical slope sensing.

The next step for our work is to simulate closed loop operations to see how the correction really works, and toward which equilibrium the system converges: how the initial wavefront is estimated, corrected, and how important the resulting error is in comparison with the initial error.

4.1. Closed loop system.

In previous chapters and sections, we have clearly defined each part of the overall system. We worked with subsystems for which we have simulation routines.

A distorted wavefront is estimated through its Laplacian and edge derivatives, and a membrane mirror or bimorph mirror by itself solves the Poisson equation to reconstruct the estimate of the initial wavefront on its surface.

The wavefront is reflected by the mirror. The phase after the reflection is the initial phase minus the reconstructed phase. The phase difference is the residual error. Closing the loop simply means estimating the residual phase, and correcting it. The feedback control loop was previously shown in figure (1.4.1). Assuming that the system is stable, the estimate of the incoming wavefront will

converge. Theoretically, this settled estimate approaches the disturbed wavefront. Nevertheless, because of noise in the system and high spatial frequency components in the atmosphere, a residual error will always be present. It is the aim of this chapter to discuss how large this residual error will be.

It is clear that the mirror reconstructs an estimate of the shape of the wavefront we want to correct, while the sensor measures only the difference between what has been corrected so far with what has to be corrected, hence, it measures what still needs to be corrected. Therefore, the signal we want to apply to the mirror must be the sum of all the successive differences: an integration through time.

For our closed loop simulation, at each system iteration, the curvature signal will be added to what is already in memory, and the complete sum will be applied to the simulated membrane electrodes. Hence, we need one more small routine beyond those that we used in the open loop situation. This routine does not present any difficulty though.

The last modification to adapt our open loop simulation to a closed loop simulation concerns the solution of the Poisson equation. We could use NPOIS, but because of the preprocessing phase, this approach is very slow, and if we want to look at the system convergence, we need at least a few dozens of complete system iterations. Therefore, it would be much better if we could find another way to solve the Poisson equation. We look for a new routine that executes faster than NPOIS, but which on the other hand does not need to be as accurate. The high accuracy of NPOIS was required since we wanted to study secondary and side effects during the open loop simulation, but now, we can content ourselves with an approximate solution, since the closed loop operations should provide for the necessary additional corrections.

Also, the NPOIS algorithm seems unrealistic for the real system, the one that will be built. We have to remember that one of the goals of this system is to come up with a solution that is realizable with analog technology. This would not be the case for NPOIS. We used this program initially to perform a complete study and to obtain a reference solution, but, for the real system, we must come up with an alternative solution.

Let us consider a real membrane. We know that there are electrodes attached to the membrane that bend themselves such that the Laplacian of the membrane surface is proportional to the applied voltage. However, we must consider the boundary conditions, i.e., the slopes, which are of the Neumann type. It seems difficult to find any physical actuator system that would force that slope as desired at the edge of a membrane. Here too we have to find an alternative.

The slope signal from the sensor output can also be considered a Laplacian since it actually is a distribution of such. If we apply this signal to an electrode, it will bend the membrane dramatically in the vicinity, and thereby produce a discontinuity in the slope. In addition, we can force the discontinuity to be along only one desired direction by shaping the electrode accordingly. Hence, we can produce a change of slope in the normal direction of the membrane in accordance with the applied voltage to an electrode.

As we already said, it is difficult to force a derivative at a given point on a membrane. It is easier and almost mandatory to have fixed positions as boundary conditions, i.e., conditions of the Dirichlet type. For our particular problem, it is possible to convert our former Neumann conditions to Dirichlet conditions. Indeed, if we consider a very large membrane, several times the size of the pupil, and we apply a constant (e.g. zero) position all along its boundary, and we do not apply any signal in between the outer edge of the pupil and that "far away" new edge,

then, controlling some edge electrodes as described earlier, we can force the correct boundary conditions at the pupil edges.

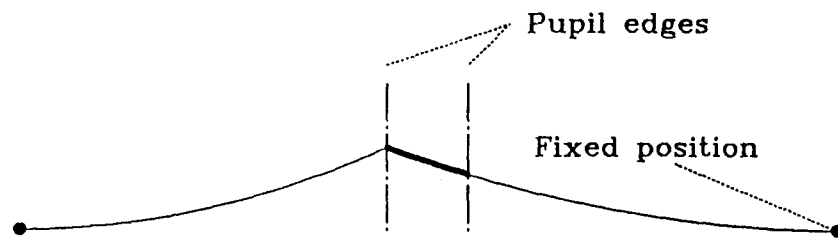


Figure (4.1.1) Membrane response to a single edge electrode.

Figure (4.1.1) shows what happens when a single edge electrode is excited. If the overall membrane is large enough compared to the pupil zone, we can assume symmetry, i.e. we can assume that the slope on both sides of the electrode is equal in magnitude. Hence, the slope at the pupil edge is half the overall change of slope. Therefore, to obtain the desired edge slope at the pupil boundary, the overall change of slope around the electrode must be twice that amount. Hence, in order to obtain the correct response, we must multiply the sensor output signal by a factor of 2 before converting it to the control voltage that then is being applied at the edge electrodes.

The suggested solution consists of a very large membrane that is fixed along its edge, with the pupil being at its very center with inner Laplacian electrodes, and boundary electrodes that will directly receive the Dirac distributions from the curvature sensor. The boundary electrodes have to be a small distance outside the pupil region so that the slope discontinuity will be outside the zone where

light reflects on. Such a new membrane configuration is shown in figure (4.1.2). The number of independent elements or actuators here stands exemplary for a real configuration.

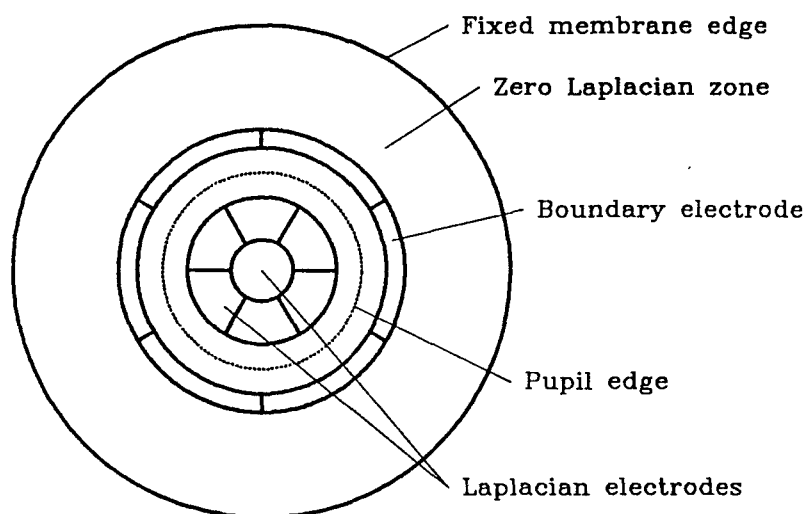


Figure (4.1.2) The new membrane configuration.

This is a physically realizable solution for the real system. In the sequel, we wish to simulate this configuration to assess its physical properties. We now have to solve a Dirichlet problem which is much simpler than the previously solved Neumann problem. There arises no longer any need to preprocess the data or perform any topological analysis. We simply have to use the kernel equation (3.3.10) in the iterations. The points that do not lie on the new membrane surface are simply fixed at a zero position and are never updated.

Since we want to simulate an astronomical configuration, i.e., operate on a small overall number of photons for the total image, and since each actuator should be hit by at least 50 photons (a larger number of photons being better), the

number of actuators in the system must be rather small. The procedure to solve the Poisson equation is to work with a much larger array and use several discretization points per electrode zone. For a given electrode, all the points will receive the same Laplacian information, and the electrode will bend itself with the same curvature all over its area. The zero Laplacian zone as described earlier will of course be discretized also, but it will receive a constant zero signal. A typical working array to solve this Poisson equation is outlined in figure (4.1.3).

We now have all the elements needed for running closed loop simulations.

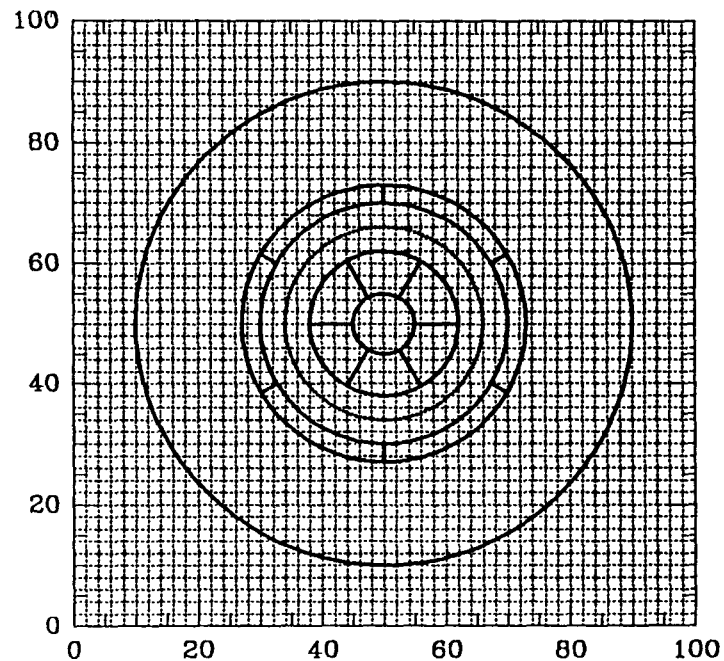


Figure (4.1.3) Discretization for solving the Poisson equation.

4.2. Simulation of the seven element system.

Now that we have clearly defined the system, have made choices for each of its elements, and have devised algorithms for each of them, we want to apply all these algorithms together to perform closed loop simulations.

4.2.a. Our first system.

We have to select a realistic configuration for the complete system, i.e., select the number of active elements, choose the number of elements for the estimation, decide on the proportions of Laplacian and edge elements, determine the shapes of all these elements, and specify all other parameters.

To be able to correct faint and weak stars, without being limited by photon noise, we want to use as few detectors as possible.

If we look at the error participation for each of the Zernike terms in the atmosphere as we explained in Chapter 3, Section 3.1.c, we see that the atmosphere wavefront r.m.s. error can be reduced by almost a factor 7 when correcting x and y tilts, defocus, and x and y astigmatisms. Therefore, if we limit our correction to those five terms, we need at least five detectors and actuators.

The first basic system configuration we decided to use consisted of seven elements, a central element, and six edge elements placed along a hexagonal pattern. The central element corresponds to a Laplacian, while the six edge elements are the boundary actuators. Figures (4.2.1) and (4.2.2) represent the sensor and membrane mirror, respectively. With such a seven element configuration, we can estimate seven independent parameters, which means that we can at best correct the first seven Zernike terms. In practice, due to the so-called Shannon limit, we probably will not correct them completely.

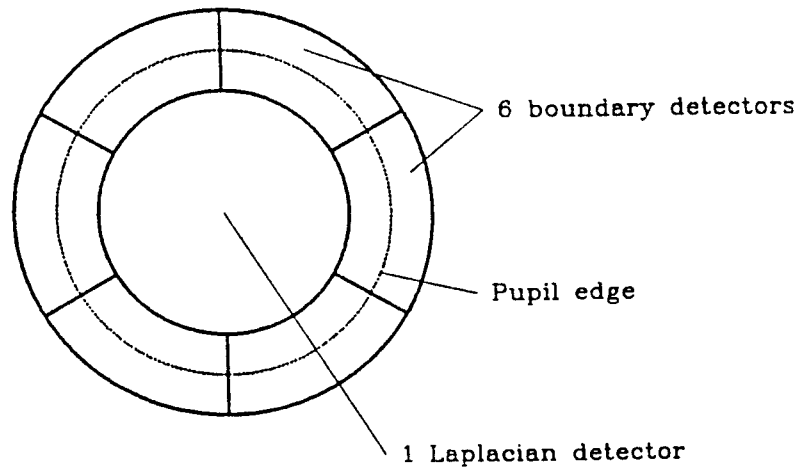


Figure (4.2.1) The seven element sensor.

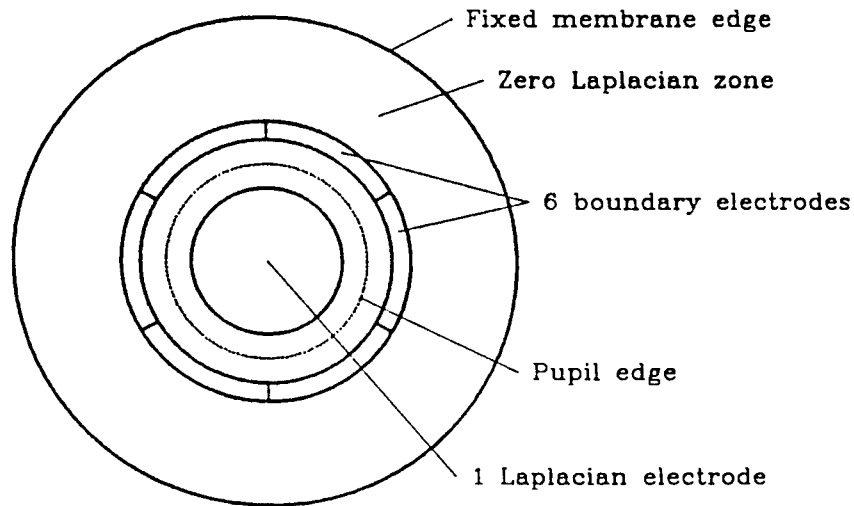


Figure (4.2.2) The seven element membrane mirror.

For the simulation, we must simulate the desired wavefront and pass data successively through all the parts of the system. During the estimation and reconstruction phases, the data will mainly be Laplacian and slope arrays, while for the feedback loop, it will mostly consist of wavefronts for partial correction.

The simulation makes it easy to investigate the influence of every parameter. This enables us to judge their effectiveness and sensitivity. We can also try

to to optimize the geometry of each element, and once we approach the optimal configuration, we can obtain an idea as to what is the best performance that we can expect of the system.

4.2.b. Simulations for single aberrations.

To have a better idea of the response of all the different atmospheric modes, we shall first run the simulations for only one Zernike polynomial at a time. Once this is done, we shall try linear combinations of those polynomials with the correct atmospheric proportions.

a. Simulation and correction of tilt.

We applied an initial wavefront with a tilt, and ran several simulations with different sets of parameter values. Each simulation was terminated when it became clear that the system had converged to its final or equilibrium state. At that moment, we were able to assess the estimate of the disturbance, and we could look at the correction and the residual error. In the following, we describe the results obtained for 20 simulations.

We used 10 different sets of electrode positions and sizes, and applied both smoothing and non-smoothing on the curvature sensor signal. The smoothing here consists of calculating an average signal for the edge electrodes from two neighbors. The two signals are equally weighted. This technique is termed “50% smoothing”. Introducing this smoothing technique provides us with an excellent means to reduce high frequency noise, but also affects the reconstruction of the terms we want to correct. Probably, a tradeoff will have to be found. The first five sets are for “large” edge electrodes, about one fifth of the pupil radius, placed at different distances from the center. The five next sets are for “thin” edge electrodes, about one tenth of the

pupil radius, also placed at different distances from the center. Those electrodes should better approach the required δ_c distribution. With respect to the central element, since for tilt the inside Laplacian is zero anyway, the central electrode shape and size is irrelevant to the problem under consideration.

Table (4.2.1) Residual error expressed in percents of the disturbed input tilt wavefront.

Interior radius <i>i.r.</i>	Exterior radius <i>o.r.</i>	No smoothing	50% smoothing
0.9	1.1	∞	∞
1.0	1.2	∞	37%
1.1	1.3	11%	13%
1.2	1.4	9%	11%
1.3	1.5	8%	9%
1.0	1.1	18%	72%
1.1	1.2	14%	12%
1.2	1.3	9%	14%
1.3	1.4	9%	9%
1.4	1.5	7%	8%

We notice that, exactly as we had expected, if the electrodes are inside the pupil or very close to the pupil's edge, the discontinuity in the radial slope makes the system unstable. As the electrodes are placed farther away, the residual error becomes smaller. Also, as predicted, it seems that the best convergence occurs when the electrodes are thinner, i.e., about one tenth of the pupil radius.

We now want to describe how this convergence was reached. For example, let us consider the last simulation, the one with $i.r. = 1.4$ and $o.r. = 1.5$ for the two cases (50% smoothing and no smoothing). Figures (4.2.3) and (4.2.4) show plots of the residual error graphed versus the number of iterations. In each case, we begin with no correction (error=100%), and after about 20 iterations, the system has almost converged. Figure (4.2.3) shows this error reduction without smoothing.

It just regularly decreases to reach its final value of 7%. Figure (4.2.4) shows the same graph but this time with 50% smoothing. We notice that the estimate is the best at iteration #13. Later, the error increases again until it settles at about 8%. Our interpretation of this result is such that the algorithm without smoothing underestimates the error whereas the algorithm with 50% smoothing overestimates the error. Since the residual error, as computed by the algorithm, represents the absolute value of the error, the seemingly excellent result at iteration #13 is a fluke, i.e., simply represents the transition of the error through zero. This indicates that we actually might be better off by using a linear combination of the two estimates, i.e., using a smoothing of less than 50%.

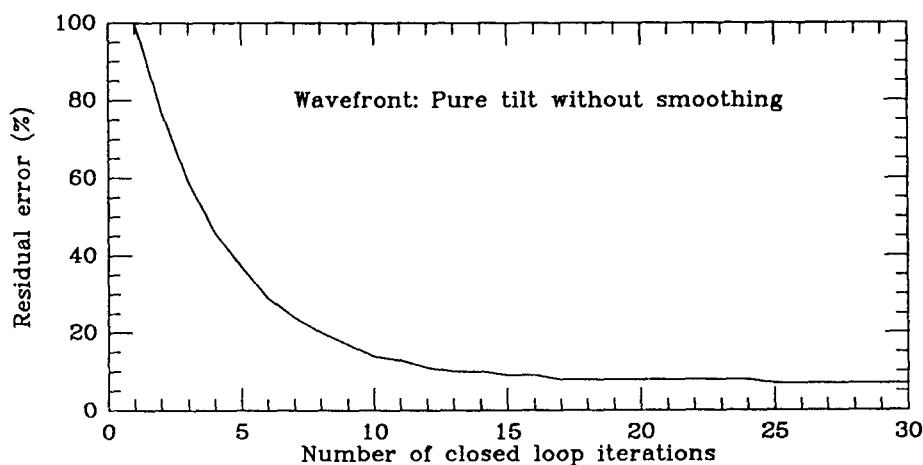


Figure (4.2.3) Evolution of the residual error for tilt initial wavefront without smoothing.

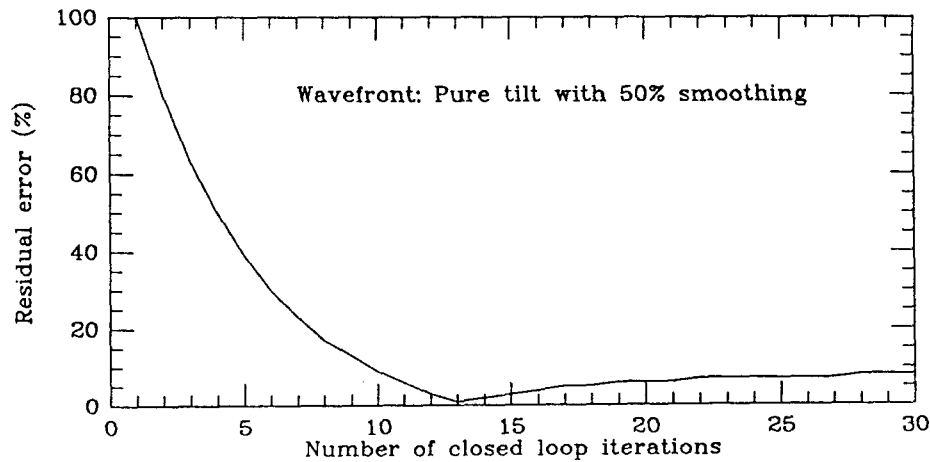


Figure (4.2.4) Evolution of the residual error for tilt initial wavefront with 50% smoothing.

b. Simulation and correction of defocus.

This experiment is similar to the previous example, but this time, we applied a defocus aberration rather than a tilt aberration. The selection of the geometric parameters was the same as in the previous case, however, we did not simulate the examples that were unstable for the tilts (since system stability is independent of the applied input). Also, due to the circular symmetry of the aberration, it is not necessary to repeat the experiment with and without smoothing. With respect to the central element, we are no longer in a position where we can ignore this parameter since, for defocus, the Laplacian is a non-zero constant. Hence, we chose two separate configurations, both using circular central electrodes, but one with a radius of 80% of the pupil radius, and the other with a radius of 90%. The results are:

Table (4.2.2) Residual error expressed in percents of the disturbed input defocus wavefront.

Interior radius <i>i.r.</i>	Exterior radius <i>o.r.</i>	Central radius 80%	Central radius 90%
1.1	1.3	2%	1%
1.2	1.4	4%	1%
1.3	1.5	4%	1%
1.1	1.2	6%	1%
1.2	1.3	8%	1%
1.3	1.4	9%	1%
1.4	1.5	10%	1%

For the 80% set, contrary to the tilt, the convergence turned out to be better if the electrodes are placed closer to the pupil. This can be explained by the fact that the defocus aberration is mainly determined by the Laplacian, the edge signals contain redundant information. For the 90% set, the convergence error is negligibly small. This result was found independent of the placement of the edge electrodes. This confirms that, for defocus, the boundary conditions are less important than the Laplacian. Also, the residual error is in general much smaller than for the tilt. Figure (4.2.5) shows a plot of the case where $i.r. = 1.2$, $o.r. = 1.3$, and the 80% central electrode. The error of the estimate decreases rapidly at the beginning, and then gradually converges to its final 8% value.

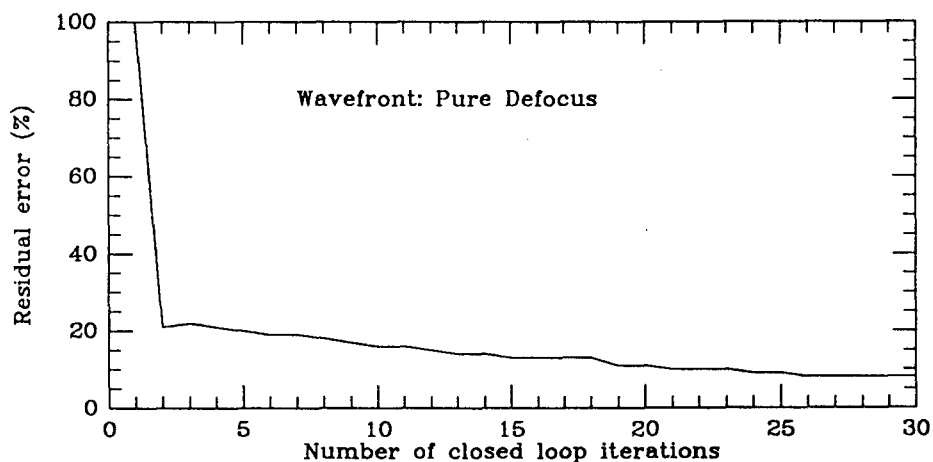


Figure (4.2.5) Evolution of residual error for defocus initial wavefront.

c. Simulation and correction of astigmatism.

We ran simulations with the same conditions as before, again with only one central electrode type since, as for tilt, the inside Laplacian is always zero. The results were about the same with or without smoothing. Since the tilt results looked better without smoothing, we chose to present only those data in the table below.

Table (4.2.3) Residual error expressed in percents of the disturbed input astigmatism wavefront.

Interior radius <i>i.r.</i>	Exterior radius <i>o.r.</i>	No smoothing
1.1	1.3	28%
1.2	1.4	26%
1.3	1.5	24%
1.1	1.2	28%
1.2	1.3	28%
1.3	1.4	25%
1.4	1.5	23%

For astigmatism, the correction is not as good as for the two previous types of wavefronts. As predicted, we reach the Shannon limit for seven elements, and we were therefore able to correct only about 77% of the disturbance. Since astigmatism is completely determined by boundary conditions, it could be expected that the correction looks better for thinner electrodes positioned away from the pupil. However, the improvement is not remarkable since most of the error stems from the Shannon limit which is reached independently of the electrode location and shape. Figure (4.2.6) shows a plot of the residual error as a function of the iteration count for $i.r. = 1.4$, and $o.r. = 1.5$. The graph looks similar to figure (4.2.3) for tilt, except that the final value is much larger. Also, the final value is reached faster, namely as soon as the Shannon limit becomes the dominant factor in the error estimate.

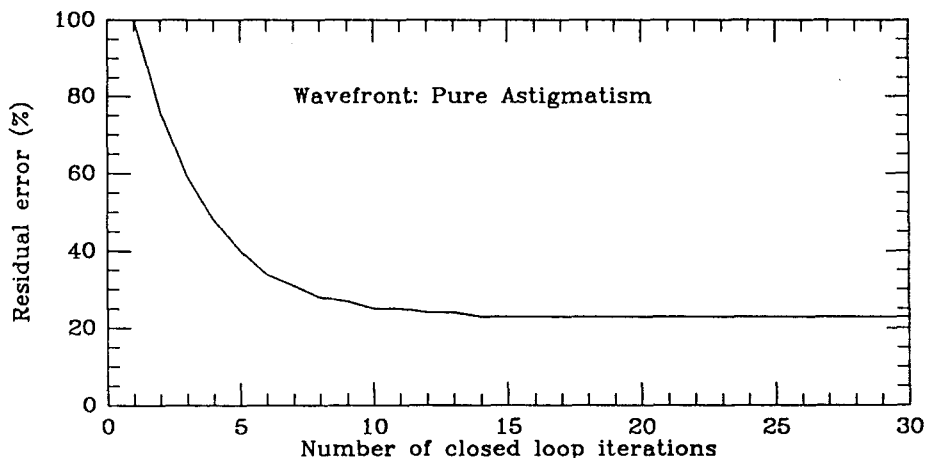


Figure (4.2.6) Evolution of residual error for astigmatism initial wavefront.

d. Other Zernike polynomials.

From those first few simulations, we acquired a fairly good idea of the system response to the three single aberrations, or five first Zernike terms that we wish to correct.

How about the higher terms? Although we cannot correct them with the system as configured (since we cannot correct more than the first seven Zernike terms with our seven electrode setup), we do not want them to interact with those terms that we were able to correct. Therefore, when running simulations with higher order terms as input only, we can obviously not correct these terms, but we wish to ensure that they do not interact with lower order terms, i.e., we wish to make certain that the residual error contains the uncorrected terms only. We therefore would like to guarantee that the various terms are, in their effects on the residual error, as much as possible decoupled from each other.

In order to verify this, we performed simulations for all Zernike terms up to Z_{15} independently, i.e., for all the first four rows in table (3.1.1). We tested several setups for the different parameters similar to those that we used in the previous examples. Since those runs are of lesser importance, we shall not print all the results here, but provide a summary only. Table (4.2.4) shows the residual error and convergence rate for the first 14 zero average Zernike polynomials with $i.r. = 1.3$, and $o.r. = 1.4$, hence thin electrodes, that were placed not too closely to the pupil; the central electrode radius being equal to 90% of the pupil radius.

Table (4.2.4) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.

Zernike term (aberration)	Convergence rate: # of iterations	Residual error: rms_{conv}/rms_{ini}
Z_2, Z_3 (Tilt)	17	8%
Z_4 (Defocus)	24	1%
Z_5, Z_6 (Astigmatism)	13	24%
Z_7, Z_8 (Coma)	9	660%
Z_9 (0-curv Coma)	16	13%
Z_{10} (0-curv Coma)	4	104%
Z_{11} (Spherical)	16	266%
Z_{12}, Z_{13} (5^{th} Astig)	10	307%
Z_{14}, Z_{15}	10	234%

The most interesting data in table (4.2.4) clearly is the residual error for the coma terms. The “corrected” wavefront is almost 7 times as bad as the “uncorrected” one! The evolution for that particular run is plotted in figure (4.2.7). We anticipated that there would be no correction, but at least we did not want the error to increase. Furthermore, the residual error is mainly produced by a huge tilt. Therefore, there exists coupling between the terms. Similarly, spherical aberration produce a defocus error. High order aberrations produce low order errors for terms in the same column in table (3.1.1), i.e., the same azimuthal frequency.

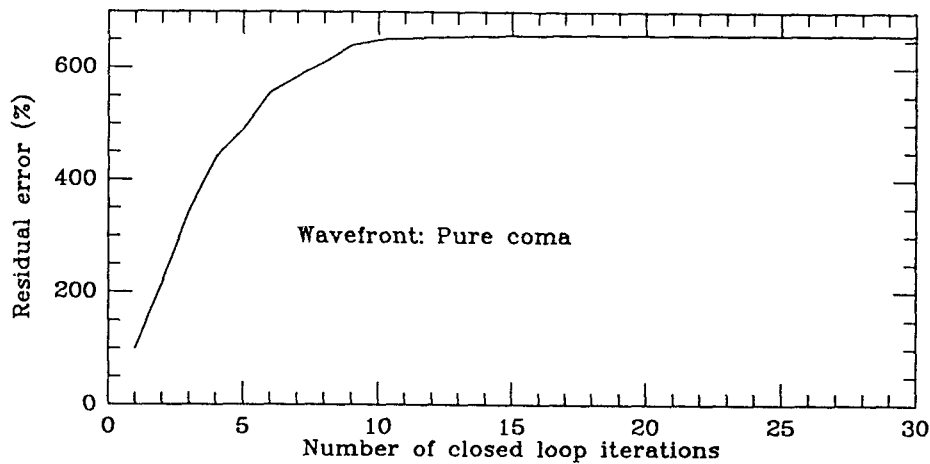


Figure (4.2.7) Evolution of residual error for coma initial wavefront.

We suspect that the same observation is true for other adaptive optics configurations as well, e.g. those making use of the Hartmann sensor, but we had no chance to investigate and prove this suspicion.

The overall system with all terms present will still correct disturbances but only if the proportion of high order terms to low order terms is sufficiently small. We know that, in the atmosphere, the low order terms contain most of the energy, but whether the energy content of the higher order terms is indeed sufficiently small, that we don't know for a fact. Only simulations with linear combinations of different terms can tell.

4.2.c. Simulations for linear combinations of aberrations.

Now that we have obtained results for single aberrations, we can try to investigate the combined effects of linear combinations of those aberrations, applying weights that are characteristic for a random atmospheric wavefront since our eventual goal is to correct aberrations of the atmosphere. Derived from table (3.1.3),

equation (4.2.1) provides a rough estimate of the r.m.s. proportion for the first few Zernike terms.

$$\begin{aligned} \text{Tilt}(2) \approx 28\%, \quad \text{Defocus} \approx 7\%, \quad \text{Astigmatism}(2) \approx 6\%, \\ \text{Coma}(4) \approx 3\%, \quad \text{Spherical} \approx 2\% \end{aligned} \quad (4.2.1)$$

In order to investigate the effect of individual aberrations in the overall concert, we decided to start with tilt alone (already done), then add defocus in the right proportion, then add astigmatism, and so forth. For each of these cases, simulations of the obtained corrections were performed. When we have evaluated the results for the five aberrations of the list above, we will know the response to 96% of the atmospheric components (Noll, 1976) which will give us a fairly good idea of what happens in the real situation.

a. Tilts and defocus combined in a 4 to 1 proportion.

Of course, we ran the simulations only for stable and good configurations as evaluated in the previous examples, i.e., we applied no smoothing and used a large central electrode.

Table (4.2.5) Residual r.m.s. error expressed in percents of the r.m.s. of the disturbed input wavefront.
Tilts and defocus aberrations.

Interior radius <i>i.r.</i>	Exterior radius <i>o.r.</i>	Convergence Error
1.1	1.3	11%
1.2	1.4	9%
1.3	1.5	8%
1.1	1.2	12%
1.2	1.3	11%
1.3	1.4	8%
1.4	1.5	7%

The convergence is pretty good, similar to the tilt alone. We have to remember that for defocus alone, the error was in the 1% range all the time. Their combination has not affected the correction capability of the system. The evolution for the particular case $i.r. = 1.4$, and $o.r. = 1.5$ is plotted in figure (4.2.8).

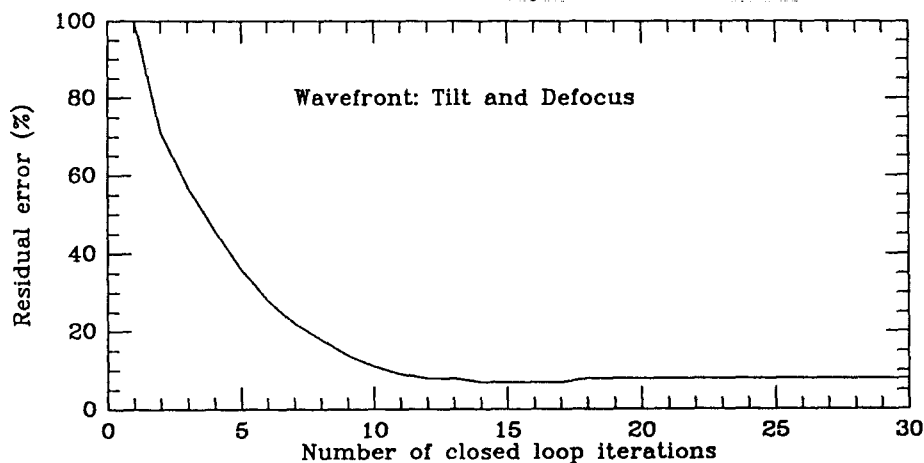


Figure (4.2.8) Evolution of the residual error for tilt and defocus aberrations combined.

b. Tilt, defocus, and astigmatism in the required proportion.

We ran almost the same simulations as for the previous case, except that now we add the 6% aberration corresponding to each astigmatism.

Table (4.2.6) Residual error for tilt, defocus, and astigmatism combined.

interior radius <i>i.r.</i>	exterior radius <i>o.r.</i>	Convergence error
1.1	1.3	∞
1.2	1.4	10%
1.3	1.5	9%
1.1	1.2	13%
1.2	1.3	10%
1.3	1.4	10%
1.4	1.5	8%

Again, the combination has not affected the convergence except for one case in which the system seemingly turned unstable. At this point, we have no explanation for this unexpected behavior. The evolution plots look similar to the previous one without astigmatism, figure (4.2.8), and are therefore not presented.

c. Addition of the coma terms to the previous wavefront.

Again, we ran the simulations exactly as before, except that we added 3% aberration corresponding to each of the two non-zero curvature coma terms, i.e., the Zernike polynomials Z_7 and Z_8 .

Table (4.2.7) Residual error for tilt, defocus, astigmatism, and coma in atmospheric proportion.

interior radius <i>i.r.</i>	exterior radius <i>o.r.</i>	Convergence error
1.1	1.3	∞
1.2	1.4	75%
1.3	1.5	75%
1.1	1.2	60%
1.2	1.3	75%
1.3	1.4	75%
1.4	1.5	70%

It is obvious that, this time, the combination has dramatically affected the convergence. Adding just a few percent coma made the error jump from the previous 10% range to a 70% range. The “corrected” wavefront is hardly better than the “uncorrected” one. We had partly foreseen this as a possible problem when we analyzed the runs for coma alone in Section 4.2.b. At that time we noticed that the residual error was mostly an uncorrected huge tilt. Again, the errors in table (4.2.7) are mainly produced by tilts.

An evolution plot, shown in figure (4.2.9), demonstrates that the system starts with a good behavior, the error decreases, but after a while, it no longer works as it should. This can be easily explained. Figure (4.2.9) is simply a superposition of figures (4.2.8) and (4.2.7), i.e., initially, the three correctable error terms make the total error decrease, but after a small number of iterations, the virtually unstable coma term takes over and makes the total error grow again.

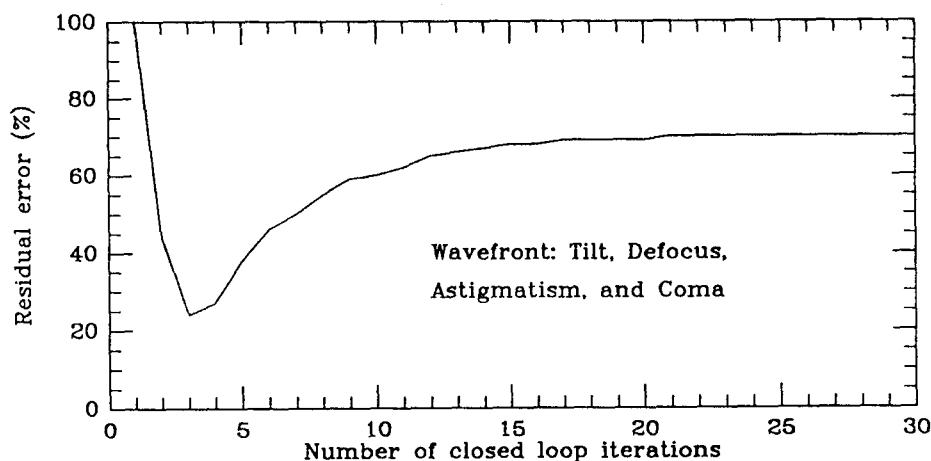


Figure (4.2.9) Evolution for tilt, defocus, astigmatism, and coma.

To further investigate this, let us perform some runs with tilt and coma alone, without any of the other aberrations. We know that, when there is only tilt, the error is about 10%, and that for coma alone, it is about 660%. Let us call γ

the proportion ratio of coma against tilt, for the same direction, that is either both *Cos* terms, or both *Sin* terms. This means that we shall have combination of Z_2 with Z_8 , or Z_3 with Z_7 . Clearly, for no coma we have γ equal to 0, and the error is 10%, while for coma only, we have γ equal to 1, and the corresponding error is 660%.

We want to evaluate the error as a function of γ when varying γ from 0 to 1. The results of those simulations are presented in table (4.2.8) and figure (4.2.10).

Table (4.2.8) Residual error as a function of γ , ratio of coma amplitude over tilt amplitude.

γ	Error
0.0	10%
0.1	69%
0.2	217%
0.3	364%
0.4	433%
0.5	564%
0.6	586%
0.7	603%
0.8	641%
0.9	657%
1.0	660%

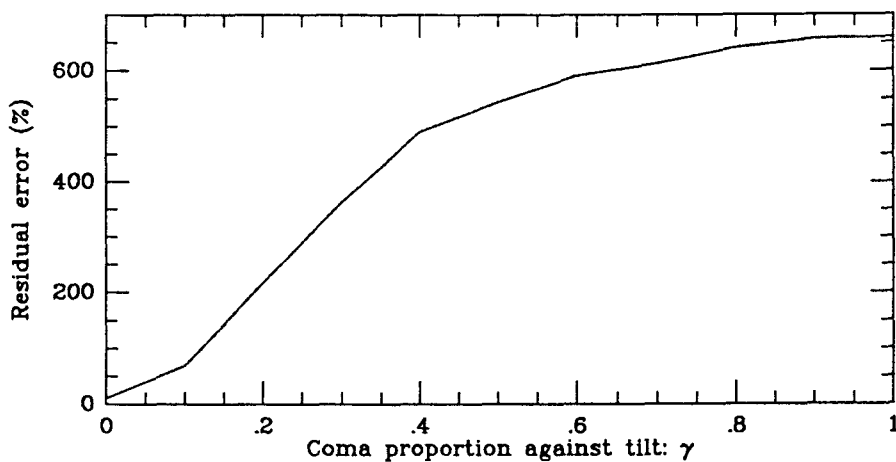


Figure (4.2.10) Error as a function of γ .

As can be seen, the error increases overproportionally with the introduction of even a modest percentage of coma. The system would work satisfactorily if the γ value were much smaller than 0.1, but unfortunately, this is not the case for Earth atmosphere. The true γ value for the atmosphere is $\frac{3}{31} \approx 0.1$.

d. Addition of spherical aberration.

Although we already know that our system will not work for Earth atmosphere, we decided to analyze the effect of adding spherical aberration to see if the system performance deteriorates even further. We ran simulations for the same case as for table (4.2.7), with the required amount of spherical aberration added. The results of these simulation runs are summarized in table (4.2.9). Evolution plots look the same as in figure (4.2.9), and were therefore omitted.

Table (4.2.9) Error for the mixture of the first five aberrations.

Interior radius <i>i.r.</i>	Exterior radius <i>o.r.</i>	Residual error
1.1	1.3	∞
1.2	1.4	72%
1.3	1.5	78%
1.1	1.2	72%
1.2	1.3	78%
1.3	1.4	78%
1.4	1.5	78%

Again, as we already knew, the combination has affected the convergence. Adding the spherical term has not removed the negative effect of the coma. The error now is in the 75% range. As we noticed when we ran the spherical aberrations alone, the error was mainly due to an uncorrected huge defocus. The combination spherical-defocus shows the same behavior as the combination coma-tilt. As for the former case, the proportion of spherical in the atmosphere compared to defocus is way too high for our system to be effective for correction. However, it seems that the system at least remains stable as long as the boundary electrodes are sufficiently far away from the pupil.

4.3. Comments and explanations for the seven element system.

We can draw some important conclusions from all the simulations we performed with this first system. It is very clear that this setup will not work properly as is to correct the atmosphere. However, we saw that the procedure worked quite well for a number of simple cases. Our goal was to be able to correct with seven electrodes the two tilts, the defocus, and the two astigmatisms. We were able to correct those aberrations and linear combinations of those with atmospheric weighting

almost perfectly, when they occurred alone, i.e., when no other types of aberrations (coma, spherical) were present. This is a positive step.

Nonetheless, we ran into problems. When we added higher order Zernike terms, present in the Kolmogoroff spectrum, we, of course, were unable to correct those, as we had anticipated. But, worse, under such conditions, the system even did not correct the simple terms any longer, terms that it had been able to correct, when alone. This was due to the fact that the higher order terms gave cause to new lower order contributions through non-linear coupling between the various terms.

An explanation of all of this may be the following. We can split the Zernike polynomials into three distinct categories. The first class contains the five low order terms Z_2 to Z_6 which are theoretically correctable by this particular seven electrode system configuration. The second class contains the Zernike polynomials Z_7 to Z_{11} which are right in the Shannon limit zone. Finally, the third category which includes polynomials Z_{12} and above, contains higher frequency terms.

The first class does not present any problems as we were able to demonstrate at hand of the tilt, defocus, and astigmatism simulations as well as any combinations thereof.

The third class does not interfere either since those terms are high frequency terms that are filtered out. Indeed, they do not affect the correction either due to the thickness of the detectors or due to the thickness of the electrodes. These are stable, undetectable, unobservable, and uncorrectable terms. Since they contain very little energy in the power spectrum, this is still fine.

On the other hand, the second category of terms is the one that causes problems. They cannot be corrected, but are not smoothed either, and their presence dominates the estimation to such a high extent that they render the overall

correction useless for atmospheric wavefronts. They represent unstable uncontrollable modes that contain little energy in the beginning, but become more and more important as the number of iterations increases. This is why some of the evolution plots (e.g. figure (4.2.9)) start decreasing first while those terms still contain little energy, but as soon as they pick up a sufficient amount of energy, they start dominating the controllable terms, and the residual error increases again. In order to have the system work properly, we must keep these terms fully controllable.

How can we find a solution? It is mandatory to be able to control this second class of polynomials, since we clearly cannot tolerate any unstable uncorrectable modes in the system.

The first idea might be to move those terms down into the first class. To this end, we would need to be able to correct more terms, and therefore, we would need more electrodes and detectors. Indeed, we could correct all the terms belonging to the first two classes if we were using 13 electrodes instead of the previously used seven. Unfortunately, this approach does not work since there would exist a new Shannon limit, further away but nevertheless real, which would again make some modes uncontrollable, polynomials that currently are in the third class, i.e., we simply would create a new second class at higher frequencies than before. Therefore, the proposed idea is not a good one.

The solution is therefore to move those terms up into the third class. If we can achieve this, we can make those terms stable uncontrollable and unobservable modes. Since their original energy content is quite low, this would be perfectly acceptable. For this purpose, we need additional smoothing. We can create artificial smoothing, and this will work. We shall end up with a small number of stable observable and controllable polynomials (class one), and an infinite set of stable unobservable and uncontrollable modes (class three). Fortunately, most of the energy

of the wavefront is contained in the few low order terms that make up class one. The price that we would have to pay for this modified design is that the correction of the class one terms will be somewhat affected. The system will still correct those terms to a certain degree, but it will not do so as radically as before because of the additional smoothing.

If we want the first five terms (which are very important in the atmosphere spectrum) to be corrected accurately and yet make all higher terms stable, we need to improve this last solution further. For example, we could use 13 electrodes with such a strong smoothing that the Shannon terms would not be present, and the terms Z_{12} and above would thereby be stabilized. The terms Z_7 to Z_{11} , which would theoretically be correctable with 13 elements, would no longer be perfectly corrected as a consequence of the artificial smoothing, however, this does not pose any problem since these terms do not contain much energy in the spectrum. More importantly, the first five polynomials would still be correctable with high accuracy, since they would be sufficiently far away from the smoothing to prevent serious system degradation. This 13 element configuration with extra smoothing may be a good solution.

4.4. Simulation of the 13 element system.

In the previous section, we drew conclusions about why our attempted seven element system failed to correct a satisfactory spectrum of Zernike terms. Our interpretation was that, owing to undersampling, mid-order terms were partially detected but could not be properly corrected, thus producing errors on lower order terms. In order for such a system to work, it must be blind to terms which are around the Shannon limit for the particular system. For instance, they can be

smoothed out by spatial filtering. If one still wants to effectively correct tilt, defocus, and astigmatism, a higher number of detectors becomes necessary.

4.4.a. Our second system.

In section 4.3, we suggested a 13 element system could possibly work according to our hypotheses. Therefore, we again have to decide about a realistic configuration for our new system. We chose 13 elements, but we still have to choose the proportion of Laplacian against edge elements, and the shapes of the electrodes.

From the previous modal response given in table (4.2.4) and the combination runs, we clearly see that the terms that we do not wish to be amplified, are primarily the non-zero curvature coma terms (Z_7 and Z_8), and the spherical term (Z_{11}). Those terms in the Zernike table (3.1.1) are not on the main diagonal, and all of the terms on that main diagonal (zero curvature polynomials) received very satisfactory corrections with the seven element system. Therefore, we do not want to change the edge actuators in any way. The difference between our two systems has to concern Laplacian elements only.

Now, if we consider defocus or spherical aberrations, we see that due to their circular symmetry, it is desirable to use a central actuator. We are thus left with six other Laplacian elements which we can easily distribute over another ring. Figures (4.4.1) and (4.4.2) represent the new sensor and the new membrane mirror, respectively. The overall electrode system now consists of a central Laplacian element, a ring of six additional Laplacian elements neighboring the central element each of which is contained in a 60° sector of the pupil, and finally, a boundary ring of six edge elements, each of which is also contained within a 60° sector, rotated 30° against the other rings in order to improve the equilibrium in the azimuthal direction.

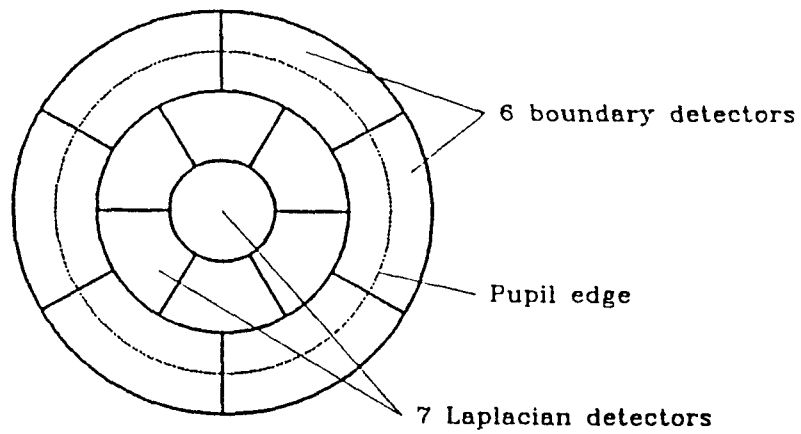


Figure (4.4.1) The 13 element sensor.

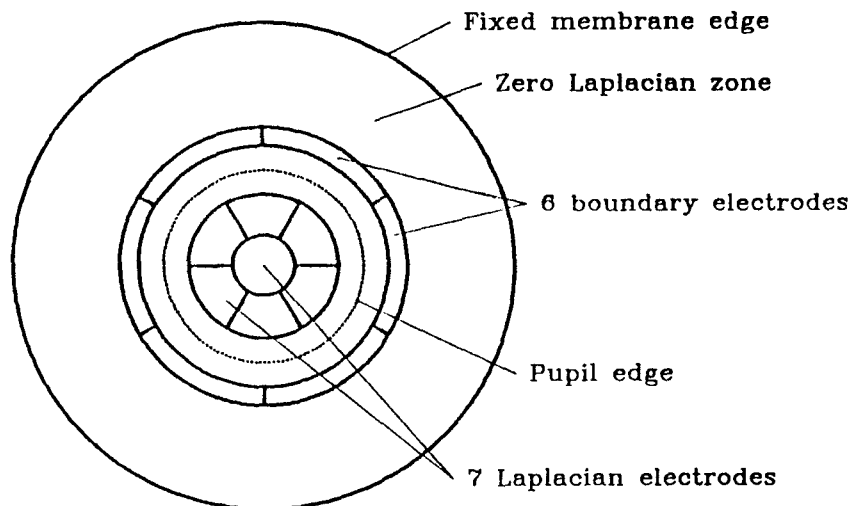


Figure (4.4.2) The 13 element membrane mirror.

We shall run simulations similar to those that we performed for our previous system. Again, we shall try to find an optimum setup with respect to the size and position of each electrode. Also, we must remember that we have to apply some smoothing to make the system blind to all terms that would otherwise be unstable. Therefore, we must also try to optimize the amount of smoothing.

4.4.b. Simulations for single aberrations.

This time, we are in the first place interested in the stability of all the terms. For the seven element system, we looked at the corrected terms and their rate of convergence, before noticing the stability problems caused by the higher order terms. This was not the right approach since no-one really cares about how fast the low order aberrations are corrected if the overall system is unstable!

Hence, we shall start by checking stability, or rather non-amplification of as many terms as we can for several different setups and smoothing. Since the Shannon limit is now close to the 13th term, we need more than the previous 14 Zernike polynomials to work with. Hence, we have to add some more to our list. Table (3.1.1) went up to Z_{15} , i.e., contained the first five rows, i.e., radial degrees zero to four. Let us now add a sixth row. This contains the terms Z_{16} to Z_{21} . They are listed in table (4.4.1). With the same normalization procedure as before, the r.m.s. value is still equal to 1.

Table (4.4.1) Some more Zernike polynomials.

Zernike number Z_i	Azimuthal frequency m	Definition $Z(\rho, \theta)$
16	1	$Z_{16} = \sqrt{12}(10\rho^5 - 12\rho^3 + 3\rho)\text{Cos}\theta$
17	1	$Z_{17} = \sqrt{12}(10\rho^5 - 12\rho^3 + 3\rho)\text{Sin}\theta$
18	3	$Z_{18} = \sqrt{12}(5\rho^5 - 4\rho^3)\text{Cos}3\theta$
19	3	$Z_{19} = \sqrt{12}(5\rho^5 - 4\rho^3)\text{Sin}3\theta$
20	5	$Z_{20} = \sqrt{12}(\rho^5)\text{Cos}5\theta$
21	5	$Z_{21} = \sqrt{12}(\rho^5)\text{Sin}5\theta$

The procedure to test a particular setup is therefore to run the simulation for each of those 20 Zernike terms, and look at the residual error and the convergence rate. For the former seven element system, we concluded that the response was usually better when we worked with thin edge electrodes, placed not too closely to the pupil. For our new 13 element system, we can expect a similar behavior. Hence, it seemed wasteful to spend more time trying to find out an optimal position for the edge actuators since we already have a fairly good idea of where we should put them. On the other hand, we have not worked yet on smoothing techniques. Therefore, we shall stress more on this new aspect of our problem.

Spatial filtering and smoothing means that some curvature signals are combined in a weighted sum before they are applied to the membrane electrodes. The farther away from each other the elements are that are combined in the weighted sum, the larger the amount of smoothing will be. Thereby, the bandwidth of the spatial filter is reduced which, in turn, determines which of the Zernike terms are filtered out.

Hence, the smallest possible smoothing occurs when a signal is mixed only with its immediate neighbor. Increasing the number of neighbors involved in the mixing increases the amount of smoothing. In the extreme situation, taking averages

from all the elements at the same time will result in an overall zero signal since the overall sum is zero. That would mean an infinite smoothing, and no terms would be corrected at all.

What we want is the correct amount of smoothing that ensures that the terms around the Shannon limit do not disturb the operation of the system with respect to the low order terms.

The procedure we want to follow is to start with only a small amount of smoothing, i.e., the low order terms will certainly not be affected, while the high order terms will probably not be sufficiently filtered, and therefore, cause overall system instability. Then, we increase the amount of smoothing, until overall system stability is preserved, i.e., until the high order terms are satisfactorily filtered out by the system, while the low order terms are still corrected accurately.

The 13 signals to which the smoothing technique will be applied contain the six edge signals, the six outer Laplacian signals, and the central Laplacian signal. In the smoothing process, the 13 signals are linearly mixed to produce 13 new signals which are then converted to be applied to the 13 membrane electrodes. As we showed in figures (4.4.1) and (4.4.2), the overall configuration of the detector and actuator systems are very similar. A one to one connection can be made. Nonetheless, this is not mandatory. We could e.g. assume that the actuator system is rotated by 30° against the detector system. In that case, any electrode will be affected by two neighboring detectors which will generate some smoothing. Figure (4.4.3) shows the system without any rotation, while figure (4.4.4) shows the system with a 30° rotation.

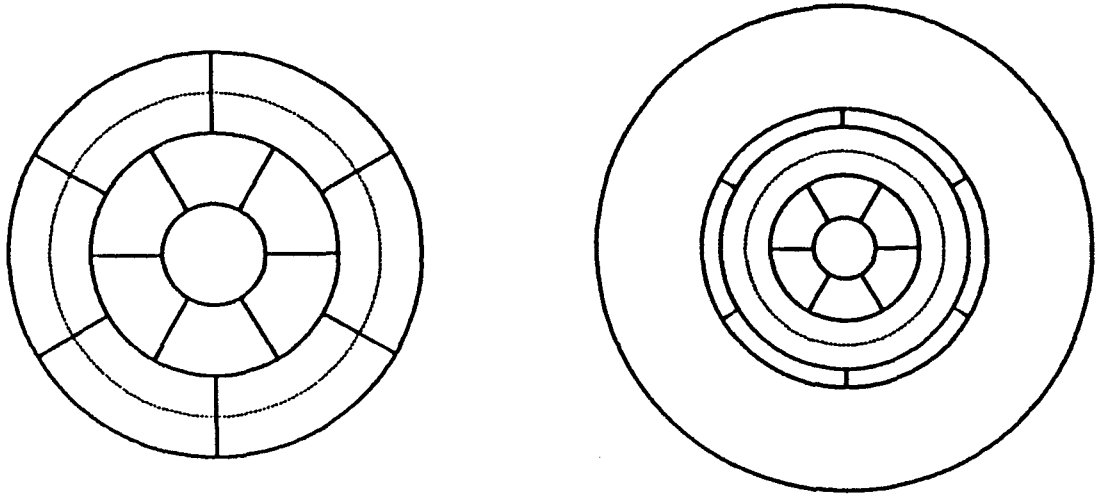


Figure (4.4.3) Sensor and mirror without rotation.

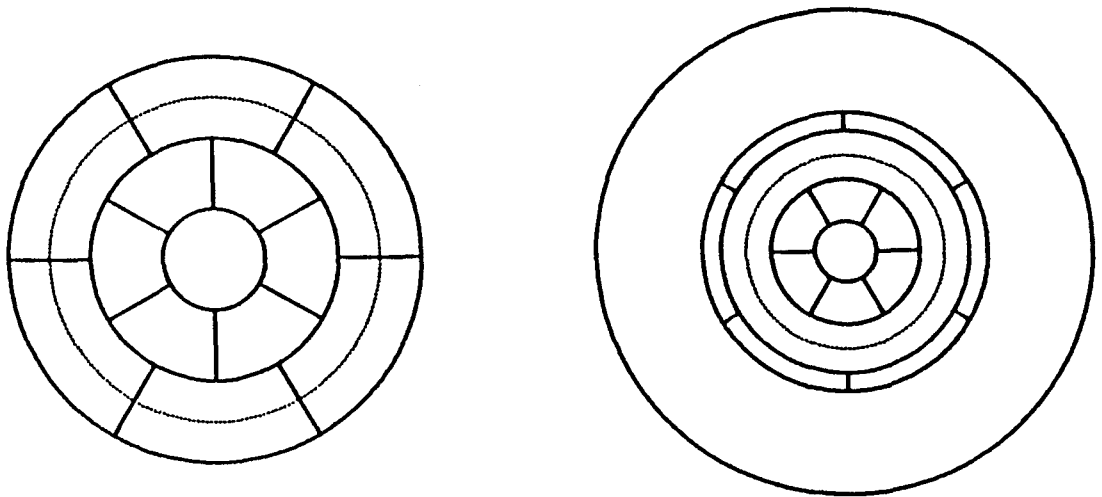


Figure (4.4.4) Sensor and mirror with 30° rotation.

Now that we have outlined all the possibilities for smoothing and assigned a geometrical interpretation, from a signal processing point of view, to at least one such smoothing configuration, let us discuss the mathematical properties of some of them.

a. First smoothing.

As we said, the simplest smoothing occurs when only two signals are mixed. The basic configuration for this is to use the 30° rotation alignment, and have the applied signal to any electrode be the average of the two detector signals that the electrode covers. Clearly, the center element will not be affected, and there is absolutely no interaction between elements of one ring and the other, in other words between Laplacians and edge slopes.

The simulation is run for the 20 Zernike polynomials mentioned earlier, and the results are summarized in table (4.4.2).

Table (4.4.2) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.

Zernike term (aberration)	Convergence rate: # of iterations	Residual error: rms_{conv}/rms_{ini}
Z_2, Z_3 (Tilt)	11	2%
Z_4 (Defocus)	9	1%
Z_5, Z_6 (Astigmatism)	14	22%
Z_7, Z_8 (Coma)	8	17%
Z_9, Z_{10} (0-curv Coma)	-	100%
Z_{11} (Spherical)	5	66%
Z_{12}, Z_{13} (5 th Astig)	10	71%
Z_{14}, Z_{15}	7	147%
Z_{16}, Z_{17}	5	122%
Z_{18}, Z_{19}	-	100%
Z_{20}, Z_{21}	5	128%

The correction for the terms we want to correct is really excellent, the smoothing does not seem to affect the accuracy with which the low order Zernike terms are corrected. The terms that we wish to suppress but not necessarily correct are unfortunately still rather prominent. However, their amplification is way smaller than it was in the case of the seven element system. While we previously were confronted with errors in the range of 660%, now the largest error is below 150%.

To improve the situation further, we should add some more smoothing.

b. Second smoothing.

The previous smoothing was in the azimuthal direction only. We now want to try some smoothing in the radial direction as well. For that purpose, we decided to take the configuration without rotation. In a first step, we smooth only the edge actuator signals, but don't smooth the Laplacian elements yet at all.

Each Laplacian electrode will receive directly the corresponding detector signal. Each edge electrode will receive 50% from its associated detector, and 50% from the average of its two edge neighbors and its two Laplacian neighbors.

Again, the simulation is run for our 20 Zernike polynomials, and the results are summarized in table (4.4.3).

Table (4.4.3) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.

Zernike term (aberration)	Convergence rate: # of iterations	Residual error: rms_{conv}/rms_{ini}
Z_2, Z_3 (Tilt)	4	2%
Z_4 (Defocus)	5	1%
Z_5, Z_6 (Astigmatism)	12	17%
Z_7, Z_8 (Coma)	20	37%
Z_9 (0-curv Coma)	22	4%
Z_{10} (0-curv Coma)	-	100%
Z_{11} (Spherical)	3	68%
Z_{12}, Z_{13} (5^{th} Astig)	9	67%
Z_{14}, Z_{15}	7	116%
Z_{16}, Z_{17}	4	115%
Z_{18}	15	119%
Z_{19}	4	86%
Z_{20}, Z_{21}	4	125%

As in the previous case, the correction of the terms we wish to correct is excellent, the correction for astigmatism has been even improved. The new smoothing procedure has further reduced the terms that we wish to suppress. The maximum amplification dropped meanwhile to 125%.

c. Third smoothing.

To further pursue the study of smoothing in the radial direction, we shall smooth not only the elements on the outer ring but also the ring of Laplacian elements. The center element too will be mixed with its neighbors.

Each ring Laplacian electrode will receive 50% from its associated detector, and 50% from the average of its two edge neighbors, its two Laplacian neighbors, and the center detector. The center electrode will receive 50% from the center detector, and 50% from the average of the six Laplacian detectors on the first ring. Each edge electrode will receive, like in the previous smoothing case, 50% from its

associated detector, and 50% from the average of its two edge neighbors and its two Laplacian neighbors.

Once again, the 20 Zernike polynomials were tested, and the simulation results are summarized in table (4.4.4).

Table (4.4.4) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.

Zernike term (aberration)	Convergence rate: # of iterations	Residual error: rms_{conv}/rms_{ini}
Z_2, Z_3 (Tilt)	10	2%
Z_4 (Defocus)	14	1%
Z_5, Z_6 (Astigmatism)	13	17%
Z_7, Z_8 (Coma)	13	37%
Z_9 (0-curv Coma)	21	4%
Z_{10} (0-curv Coma)	-	100%
Z_{11} (Spherical)	9	66%
Z_{12}, Z_{13} (5 th Astig)	9	68%
Z_{14}, Z_{15}	5	116%
Z_{16}, Z_{17}	4	115%
Z_{18}	9	119%
Z_{19}	7	86%
Z_{20}, Z_{21}	4	125%

Compared to the previous case, there is hardly any difference. The convergence rate now seems a little slower, but the residual error is the same. Therefore, it was probably not a useful strategy to allow too much smoothing along the radial direction.

d. Fourth smoothing.

We now test a smoothing with complete separation between Laplacian and edge signals. The smoothing will be only along the azimuthal direction for the edge actuators, but will be along both directions for all the Laplacian elements.

Each ring Laplacian electrode will receive 60% from its associated detector, and 40% from the average of its two Laplacian neighbors, and the center detector. The center electrode will receive 60% from the center detector, and 40% from the average of the six surrounding Laplacian detectors on the first ring. Each edge electrode will receive 60% from its associated detector, and 40% from the average of its two edge neighbors.

The results of the simulation for the 20 Zernike polynomials are summarized in table (4.4.5).

Table (4.4.5) Residual error expressed as the ratio of the r.m.s. error when convergence is assumed over initial wavefront r.m.s.

Zernike term (aberration)	Convergence rate: # of iterations	Residual error: rms_{conv}/rms_{ini}
Z_2, Z_3 (Tilt)	7	2%
Z_4 (Defocus)	11	1%
Z_5, Z_6 (Astigmatism)	17	17%
Z_7, Z_8 (Coma)	10	37%
Z_9, Z_{10} (0-curv Coma)	-	100%
Z_{11} (Spherical)	8	66%
Z_{12}, Z_{13} (5 th Astig)	5	66%
Z_{14}, Z_{15}	10	116%
Z_{16}, Z_{17}	6	115%
Z_{18}	-	100%
Z_{19}	13	86%
Z_{20}, Z_{21}	4	125%

Compared to all the previous cases, this one is the best considering residual error. With respect to the convergence rate, it is faster than the last case, and of the same order as the one before. Therefore, this smoothing technique was considered the best among all those that we tested so far.

4.4.c. Simulations for linear combinations of aberrations.

Similarly as for the former seven element system, we first obtained results for every single aberration up to the 20th Zernike term. We can now try to simulate linear combinations of those again under the condition of atmospheric weighting. The proportions that we shall use here are the same as before, and that were given in equation (4.2.1). This time, we can feel more confident since now all the terms are more or less controllable, and none of them is submitted to a huge amplification factor as was previously the case for some of the “bad” terms.

Exactly as before, to obtain a complete picture of the overall system performance, we have to move step by step, starting with a single tilt and adding, one at a time, the higher order aberrations to the input. With a complete Zernike spectrum for the first 14 terms, this represents more than 97% of the atmosphere perturbation energy. Since the last 3% is present only in very high order terms that are neither detected nor corrected by the system, we can feel confident that we know the complete atmospheric system response with those 14 first terms.

We ran the simulations for each of the four cases for which we studied the single term response. From now on, we shall call them configurations *a*, *b*, *c*, and *d* in accordance with the paragraph numbers in the previous section 4.4.b.

Here are the results for all the successive combination runs.

Table (4.4.6) Residual r.m.s. error expressed in percents of the r.m.s. of the disturbed input wavefront. Tilts and defocus aberrations.

Case number	Convergence Error
<i>a</i>	2%
<i>b</i>	2%
<i>c</i>	2%
<i>d</i>	2%

This could be easily anticipated: from the seven element system, we knew that tilts and defocus aberrations were completely decoupled, and the results for each of them separately were in the 2% range for tilts, and in the 1% range for defocus.

Table (4.4.7) Residual error for tilt, defocus, and astigmatism aberrations combined.

Case number	Convergence Error
<i>a</i>	5%
<i>b</i>	4%
<i>c</i>	4%
<i>d</i>	4%

Again, the linear combination of the term is exactly what could be expected from single runs. The error is worst for case *a*, but we have to remember that the astigmatism error was 22% for that case alone compared to 17% for the other ones. This is enough to explain the percentage difference on the combination. In comparison with the seven element setup, astigmatism is corrected much more effectively. This is due to the fact that, in the seven element setup, the astigmatism terms were already somewhat affected by the Shannon limit which is not the case here.

Table (4.4.8) Residual error for tilt, defocus, astigmatism, and non-zero curvature coma aberrations.

Case number	Convergence Error
<i>a</i>	7%
<i>b</i>	6%
<i>c</i>	7%
<i>d</i>	6%

The non-zero curvature coma was the term that blew up the seven element system. We made hypotheses as to how to design a new system that would eliminate those problems. With our 13 element system, the addition of these coma terms does not affect the convergence dramatically. As desired, these terms are almost filtered out, and the system is basically blind to them.

Table (4.4.9) Residual error for tilt, defocus, astigmatism, and all coma term aberrations.

Case number	Convergence Error
<i>a</i>	8%
<i>b</i>	7%
<i>c</i>	7%
<i>d</i>	8%

The zero curvature coma terms did not give any trouble with the old system. They do not do so with the new one either.

Table (4.4.10) Residual error for tilt, defocus, astigmatism, coma, and spherical aberrations.

Case number	Convergence Error
<i>a</i>	9%
<i>b</i>	9%
<i>c</i>	8%
<i>d</i>	9%

Spherical was the second worst aberration for the seven element system. It produced a huge uncorrectable error upon defocus. With the new system, this seems no longer to be the case.

Table (4.4.11) Residual error for tilt, defocus, astigmatism, coma, spherical, and 5th order astigmatism.

Case number	Convergence Error
<i>a</i>	10%
<i>b</i>	10%
<i>c</i>	9%
<i>d</i>	10%

Table (4.4.12) Residual error for the first five rows in the Zernike table. Hence, up to Z_{15} .

Case number	Convergence Error
<i>a</i>	13%
<i>b</i>	12%
<i>c</i>	12%
<i>d</i>	12%

We are now at a point where the first five rows of the Zernike table are present with correct atmospheric coefficient in our simulated disturbed input. The residual error is about 12% which is very satisfactory. This means that the r.m.s.

corrected wavefront is about a factor of 8 improved over the uncorrected wavefront. This is more than the required factor to go from typical atmospherically distorted images to diffraction limited images for astronomical observations.

4.4.d. Simulations for random atmospheric wavefronts.

Since we now know that the system works for fixed linear combinations, we have to introduce the statistical approach as described in Section 3.1.d to really simulate the atmosphere. We now include Zernike terms up to Z_{21} , which represents 98% of the total energy in the atmosphere (Noll, 1976).

The computer simulates random wavefronts, and exactly as before, we look at the residual error at convergence, and the corresponding convergence rate. Since, this is a statistical approach, several runs are needed for our estimate of the complete correction not to be sample limited. Table (4.4.13) shows for 15 different runs the r.m.s. value of the uncorrected random wavefront, the r.m.s. value of the residual error after convergence of the correction scheme, and the percentage residual error as defined earlier. We use the best configuration we determined from the previous runs, i.e. case *d*.

Table (4.4.13) Correction for random atmospheric wavefronts.

Uncorrected input wavefront (rms) (radius units)	Corrected output wavefront (rms) (radius units)	Residual error (%)
$49 \cdot 10^{-6}$	$70 \cdot 10^{-7}$	14.3%
$56 \cdot 10^{-6}$	$52 \cdot 10^{-7}$	9.3%
$67 \cdot 10^{-6}$	$71 \cdot 10^{-7}$	10.6%
$47 \cdot 10^{-6}$	$88 \cdot 10^{-7}$	18.7%
$67 \cdot 10^{-6}$	$10 \cdot 10^{-6}$	14.9%
$54 \cdot 10^{-6}$	$79 \cdot 10^{-7}$	14.6%
$51 \cdot 10^{-6}$	$10 \cdot 10^{-6}$	19.6%
$75 \cdot 10^{-6}$	$11 \cdot 10^{-6}$	14.7%
$27 \cdot 10^{-6}$	$55 \cdot 10^{-7}$	20.4%
$62 \cdot 10^{-6}$	$83 \cdot 10^{-7}$	13.4%
$75 \cdot 10^{-6}$	$11 \cdot 10^{-6}$	14.7%
$76 \cdot 10^{-6}$	$87 \cdot 10^{-7}$	11.4%
$49 \cdot 10^{-6}$	$84 \cdot 10^{-7}$	17.1%
$65 \cdot 10^{-6}$	$11 \cdot 10^{-6}$	16.9%
$71 \cdot 10^{-6}$	$82 \cdot 10^{-7}$	11.5%

Over those few values, the residual error has an average value of 14.81%. This represents about a factor of 7 of reduction for the r.m.s. value of the wavefront. This is a good result for an adaptive optics system.

4.5. Simulations of correction for images.

The 13 element system that we designed gave satisfactory results for wavefront correction. After convergence of the closed loop operations, only a small fraction of the initial error persists. However, it would be interesting to see what this will do for the astronomer, or for the observing camera. Indeed, a wavefront is more of a theoretical tool, but is not of much interest in an observation. What we really want are images, good images.

Back in Chapter 1, Section 1.2.a, we summarized the relations between a wavefront surface and its corresponding observed image, equations (1.2.1) and

(1.2.2) (Gaskill, 1978). We are now at a point where we would like to know not only the input and output wavefront of our active system, but also the corresponding input and output images, so that we can compare them, and be conscious of the improvement.

To go from wavefront surface to images, we first have to go from wavefront surface to phase. This is given by equation (1.2.1), and is an easy task. We simply need the wavelength which we are observing at.

However, to go from phase values to images, we need to use again a 2D FFT routine. Fortunately, this time, we do not have problems with the sampling since the FFT size has to be the wavefront array size.

The procedure from a wavefront surface $z(x, y)$ to obtain the corresponding image irradiance $I(x, y)$ is the following:

- Multiply $z(x, y)$ by $2\pi/\lambda$ to obtain $\phi(x, y)$.
- Take the complex exponential $e^{j\phi(x, y)}$ of the phase.
- Gate the result by the pupil function: 1 inside, 0 outside the pupil.
- Take the 2D FFT of the result.
- Evaluate the square of the modulus. This last computation gives the irradiance distribution, the observed image.

With such a procedure, we are able to visualize the performances of our system and its correction capabilities. We simulated a realistic situation in that we used real values for wavelength and pupil parameters. We simulated what would happen if our 13 element system was behind the 2.2m telescope of the University of Hawaii on Mauna Kea, with a typical seeing parameter of $r_o = 56cm$ at a typical infrared wavelength of $\lambda = 2.2\mu m$.

Figure (4.5.1) shows a 3D plot of the irradiance distribution for an average of 21 uncorrected images for 21 wavefront samples. Figure (4.5.2) shows the same average image with closed loop correction.

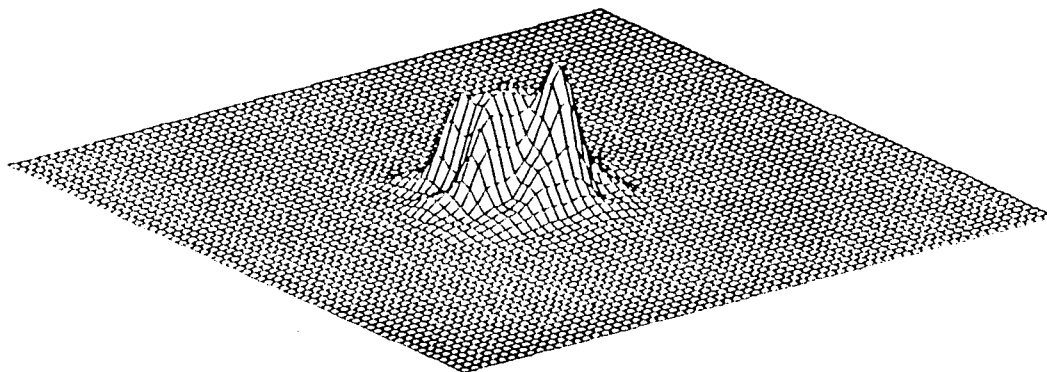


Figure (4.5.1) Uncorrected image for 21 samples.

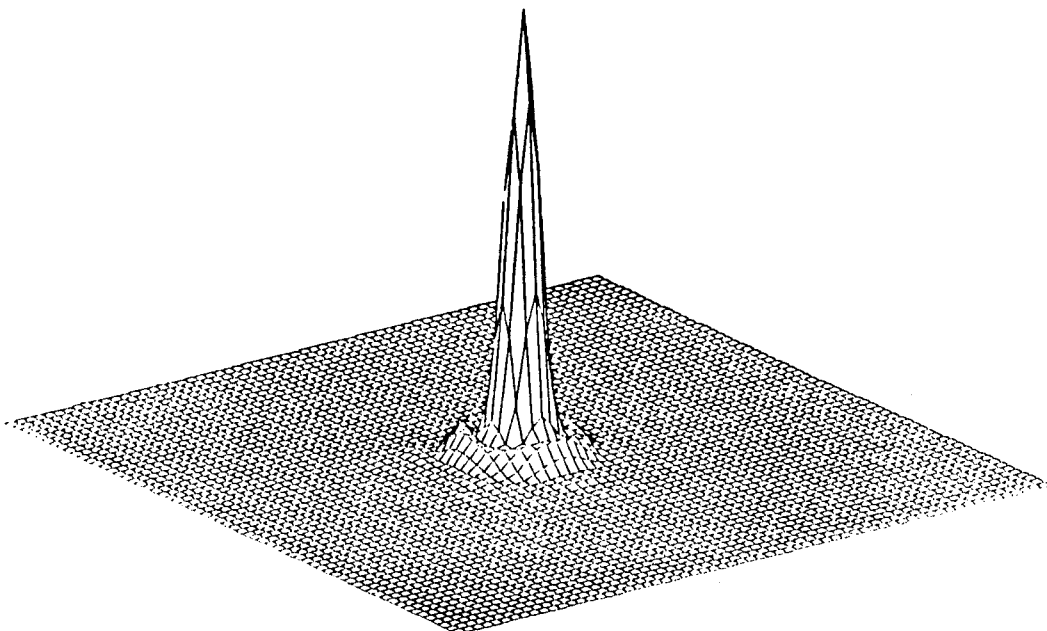


Figure (4.5.2) Corrected image for 21 samples.

Figures (4.5.3) and (4.5.4) show the corresponding contour plots of the previous 3D figures. Note the perfect circular rings on the corrected image.

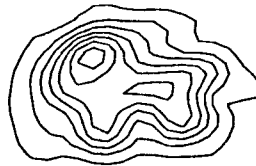


Figure (4.5.3) Uncorrected image for 21 samples.



Figure (4.5.4) Corrected image for 21 samples.

4.6. Chapter summary.

In this chapter, we first derived the necessary requirements for a real physical membrane mirror to be used in our adaptive optics system. We then looked at a configuration with seven actuators. Unfortunately, several unstable uncontrollable

modes were present in this configuration. We then came up with hypotheses how to modify the system to make it stable. A second configuration with 13 elements was designed. This configuration was stabilized with the aid of a specifically designed filtering technique. The stability of the configuration was tested to be very reliable. The adaptive optics system was then simulated with typical atmospheric random wavefronts used as disturbed input.

A physical membrane mirror in a realistic experiment has to be fixed at its edges to a position. It is therefore technically not feasible to use Neumann boundary conditions. We have to use Dirichlet boundary conditions. We showed that it is possible to convert the edge signals out of our curvature signals to high density signals which produce slope discontinuities at some places on a very large membrane. This new large membrane has to be fixed at its edges. The pupil zone now represents only a small disk at the center of the membrane.

A seven element configuration was tested in closed loop operations. The convergence was very efficient and fast enough when the disturbed input wavefront was composed of tilt, defocus, and astigmatism aberrations, or combinations of those only. However, when higher order Zernike terms were introduced, our system did not only fail to correct for those, but the higher order terms even introduced new low order terms that were no longer corrected. The instability of the coma and spherical aberrations rendered the whole system unstable.

Faced with the instability problems of our first system, alternatives had to be found. We separated the Zernike polynomials into three class, each of which was considered with a different perspective. The first class contains the polynomials that we want to observe and correct as best we can. The second class contains the unstable modes that we wish to eliminate. The third class finally contains all the high frequency terms that are of no concern to us since they are stable and

since their total energy content is negligible. The solution we came up with was to expand the number of actuators to 13, and at the same time apply strong filtering to suppress all those terms that are located around the Shannon limit, i.e., those terms that are by nature uncontrollable and unstable.

In order to guarantee the stability of all terms, the following filtering technique was applied: each membrane electrode receives a linear combination of signals from its associated detector and from the detectors that are associated with its immediate neighbours. Even with this smoothing technique, the system can still be considered entirely analog. The good convergence properties of this algorithm made it possible to correct random atmospheric wavefronts so that they would give rise to diffraction limited images.

CONCLUSION

In this thesis, we showed that the newly introduced “curvature sensing and compensation” concept can work for an adaptive optics system. The performance of this system is quite good compared to the classical approach using a Hartmann sensor from the point of view of noise sensitivity.

However, curvature sensing cannot be used as is for applications where a very large number of independent actuators is required. For astronomy, this is not a problem.

In closed loop operations, the system has to be designed carefully with proper filtering to avoid any instability. The filtering we used was mostly found empirically. The filter as well as the geometry of the actuators could be further optimized.

Our system uses a membrane mirror. It would be interesting to study the behavior of the curvature sensing algorithms also for a bimorph mirror where the boundary conditions must be handled differently.

REFERENCES

- Born, M. and Wolf, E., *Principles of Optics*, (Pergamon, New York), Sec 9.2, (1965)
- Brenner, N., "FOUR2 procedure", MIT, (1968)
- Culbachini, R., "Modal wavefront estimation from phase derivative measurements", *J. Opt. Soc. Am.*, **69**, pp972-977, (1979)
- Dyson, J., "Common-path interferometer for testing purposes", *J. Opt. Soc. Am.*, **47**, pp386-390, (1957)
- Fried, D.L., "Statistics of a geometric representation of wavefront distortion", *J. Opt. Soc. Am.*, **55**, pp1427-1435, (1965)
- Fried, D.L., "Optical resolution through a randomly inhomogeneous medium for very long and very short exposures", *J. Opt. Soc. Am.*, **56**, pp1372-1379, (1966)
- Fried, D.L., "Least-square fitting a wave-front distortion estimate to an array of phase-difference measurements", *J. Opt. Soc. Am.*, **67**, pp370-375, (1977)
- Frieden, B.R., *Probability, statistical optics, and data testing*, (Springer-Verlag), p165, (1984)
- Gaskill, J.D., *Linear systems, Fourier transforms, and Optics*, (John Wiley & Sons), (1978)
- Goad, L. and Beckers, J., "A near infrared astronomical adaptive system", *Proc. SPIE 1114*, Orlando, (1989)
- Grosso, R.P. and Yellin, M., "The membrane mirror as an adaptive optical element", *J. Opt. Soc. Am.*, **67**, pp399-406, (1977)
- Hardy, J.W., Lefebvre, J.E., and Koliopoulos, C.L., "Real-time atmospheric compensation", *J. Opt. Soc. Am.*, **67**, pp360-369, (1976)

- Hardy, J.W. "Active optics: a new technology for the control of light", Proc. IEEE **66**, pp651-697, (1978)
- Herrmann, J., "Least-squares wave front errors on minimum norm", J. Opt. Soc. Am., **70**, pp28-35, (1980)
- Herrmann, J., "Cross coupling and aliasing in modal wave-front estimation", J. Opt. Soc. Am., **71**, pp989-992, (1981)
- Hudgin, R.H., "Wavefront reconstruction for compensated imaging", J. Opt. Soc. Am., **67**, pp375-378, (1977)
- Hudgin, R.H. and Lipson, S.G., "Analysis of a monolithic piezoelectric mirror", J. Applied Physics, **46**, pp510-512, (1975)
- Kokorowski, S.A., "Analysis of adaptive optical elements made from piezoelectric bimorphs", J. Opt. Soc. Am., **69**, pp181-187, (1979)
- Koliopoulos, C.L., "Radial grating lateral shear heterodyne interferometer", Applied Optics, **19**, pp1523-1528, (1980)
- Kolmogoroff, A.N., "On degeneration of isotropic turbulences in an incompressible viscous liquid", Dan. SSSR, **30**, p229, (1941)
- Kreyszig, E., *Advanced Engineering mathematics*, (Willey, New York), Chap 19, (1979)
- Morse, P.M., *Vibration and Sound*, 2nd ed., (McGraw-Hill, New York), (1948)
- Noll, R., "Zernike polynomials and atmospheric turbulence", J. Opt. Soc. Am., **66**, p207-211, (1976)
- Noll, R., "Phase estimates from slope-type wave-front sensors", J. Opt. Soc. Am., **68**, p139-140, (1978)
- Rice, R., *Solving elliptic problems using ELLPACK*, (Springer-Verlag, New-York), (1985)
- Roddier, C., "Report on the figure of the primary mirror of the CFH Telescope", to be published, (1989)

- Roddier, F., *The effects of atmospheric turbulence in optical astronomy*, Progress in Optics, E. Wolf ed. (North Holland, pub.) **19**, p33, (1981)
- Roddier, F., "Curvature sensing: a diffraction theory", NOAO ADP R&D Note, 87-3, (1987)
- Roddier, F., Roddier, C., and Roddier, N., "Curvature sensing: a new wavefront sensing method", Proc. SPIE, **976**, San Diego, (1988)
- Shellan, J.B., Private communication, (1987)
- Southwell, W.H., "Wavefront estimation from wavefront slope measurements", J. Opt. Soc. Am., **70**, pp998-1006, (1980)
- Steinhaus, E. and Lipson, S.G., "Bimorph piezoelectric flexible mirror", J. Opt. Soc. Am., **69**, pp478-481, (1979)
- Wallner, E.P., "Optimal wave-front correction using slope measurements", J. Opt. Soc. Am., **73**, pp1771-1776, (1983)
- Young, D.M., *Iterative Solutions of Large Linear Systems*, (Academic, New-York), (1972)
- Zernike, F., "Beugungstheorie des Schneidenverfahrens und seiner verbesserten Form, der Phasenkontrastmethode", (Diffraction theory of the intersection method and its improved form, the phase contrast method), Physica, **1**, p689, (1934)