

# Travail de diplôme

---

*Optimisation a posteriori et en ligne du régulateur  
multi-variables d'un système d'optique adaptative*

---

Auteur : B. Girardet  
Prof. responsable : F. Wildi  
Date : 20 décembre 2004



---

Center for Astronomical Adaptive Optics  
STEWART OBSERVATORY  THE UNIVERSITY OF ARIZONA



# Résumé

Ce travail est destiné à l'optimisation de la boucle d'optique adaptative d'un télescope. L'effort est concentré sur l'ajustement du gain du régulateur, en fonction des conditions d'observation.

Lors du processus de correction du déphasage du front d'onde incident, ledit front d'ondes est décomposé en une série finie de modes que l'on suppose indépendants les uns des autres. L'objectif est donc de corriger chacun d'entre-eux de manière optimale, en appliquant un gain au régulateur qui soit différent pour chaque mode.

La boucle d'optique adaptative corrige en temps réel le déphasage du front d'ondes incident, se basant sur les mesures du senseur de front d'onde. Avec des mesures de très bonne qualité, le gain optimal correspond au gain maximal (en regard de la stabilité du système) que l'on peut appliquer au régulateur. En revanche, lorsque les conditions d'observation se dégradent, il arrive que le gain maximal ne soit pas le meilleur.

Le cas le plus fréquent est celui où l'on observe une étoile de magnitude élevée. En effet, cette dernière étant peu brillante, le flux de photons diminue et le rapport signal sur bruit également. Dans ces conditions, un gain trop élevé du régulateur s'accompagne d'une forte propagation du bruit dans le système. La correction appliquée par le système peut s'avérer plus faible que le bruit ajouté, il faut donc diminuer le gain du régulateur.

L'objectif de ce travail est de développer un algorithme permettant d'ajuster automatiquement le gain du régulateur à partir de données mesurées sur le télescope. Cette optimisation peut s'appliquer a posteriori, c'est-à-dire après l'acquisition de mesure, et en fermant ensuite la boucle, ou bien en ligne, c'est-à-dire régulièrement, pendant une acquisition en boucle fermée.

Cet algorithme a été appliqué sur l'outil de simulation développé par François Wildi. L'amélioration des performances a été quantifiée. On remarque que c'est lors de l'observation d'étoiles peu brillantes que les performances peuvent être améliorées par optimisation modale.

C'est ensuite à partir de données prises sur le ciel, avec le télescope MMT que l'on a cherché les gains optimaux. Ces derniers n'ont pas encore pu être testés car aucun *run* d'optique adaptative n'a eu lieu jusqu'à maintenant. En revanche, une analyse approfondie des gains optimaux obtenus a été faite de manière à comprendre l'allure de ces derniers.

Cette analyse a permis de détecter quelques surprises sur les mesures utilisées. On y trouve des bruits inconnus qui perturbent le calcul d'optimisation modale. La conclusion de cette découverte est qu'il est nécessaire de traiter les données avant d'appliquer l'optimisation

modale pour obtenir les meilleures résultats.

# Remerciements

Je tiens tout d'abord à remercier ma famille pour m'avoir soutenu et accompagné durant toutes mes études. Grâce à eux, et particulièrement mes parents, j'ai pu accomplir mon cursus en toute sérénité.

Vient ensuite un remerciement particulier à mon professeur, M. François Wildi. Ceci pour m'avoir donné l'occasion de réaliser ce travail de diplôme dans un domaine très intéressant. Le le remercie également de m'avoir offert l'opportunité de venir effectuer ce travail de diplôme au Steward Observatory, à l'Université d'Arizona. Cette expérience fut véritablement enrichissante et je ne manquerais pas de mentionner l'accueil chaleureux des membres de l'observatoire.

Parmis eux, Guido Brusa m'a beaucoup aidé dans le traitement des données du télescope. Il m'a aidé à me poser les bonnes questions et m'a toujours donné les renseignements techniques nécessaires à la bonne marche de ce travail.

Bien entendu, j'aimerais également remercier l'Ecole d'Ingénieur du Canton de Vaud (EIVD), université dans laquelle j'ai acquis toutes les connaissances pré-requises pour la réalisation d'un travail de diplôme de ce type.

Alier des domaines aussi varier que le traitement de signal, l'automatique, la programmation et les connaissances techniques d'un système d'optique adaptative fut une très belle façon de conclure mes études d'ingénieur en Microtechniques.



# Table des matières

<b>Résumé</b>	<b>I</b>
<b>Introduction</b>	<b>1</b>
<b>1 L'optique adaptative</b>	<b>3</b>
1.1 Résolution d'un télescope	3
1.1.1 La limite de diffraction	3
1.1.2 La turbulence atmosphérique	4
1.1.3 Le senseur de front d'ondes	5
1.2 La reconstruction	7
1.3 Régulation	9
1.4 Chute de performance	10
1.4.1 Critères de performance	10
1.4.2 Sources de perturbations	11
1.4.3 Observations expérimentales obtenues en simulation	12
<b>2 Optimisation modale</b>	<b>13</b>
2.1 Généralités	13
2.2 Développement	13
2.3 Implémentation pratique	14
2.3.1 Les fonctions de transfert du système	15
2.3.2 Les termes $z_i(f)$ et $m_i(f)$	17
2.3.3 Considération sur le bruit de lecture	20
2.3.4 L'algorithme	24
2.3.5 La stabilité	30
2.4 Simulations et mesures	31
2.5 Conclusion du chapitre	32
<b>3 Optimisation en boucle fermée</b>	<b>35</b>
3.1 Introduction	35
3.2 De la boucle ouverte à la boucle fermée	35
3.3 Optimisation en ligne : mise à jour régulière des gains	35
3.4 Les données utilisées	36
3.4.1 Modélisation du système réel	37
3.4.2 La turbulence en entrée du système	40
3.5 Le nombre d'échantillons	40
3.6 Le temps de calcul	41
3.6.1 Recherche du minimum par dichotomie	41
3.7 Les gains optimaux depuis les données du MMT	43
3.7.1 Le bruit ajouté au signal	43

---

3.7.2	Les résultats et leur analyse . . . . .	45
3.8	L'optimisation modale pour un <i>run</i> . . . . .	51
3.9	Conclusion du chapitre . . . . .	52
<b>A</b>	<b>Remarques et considérations</b>	<b>55</b>
A.1	Normalisation de la FFT de Matlab . . . . .	55
A.2	Problèmes sur les amplitudes des spectres des modes . . . . .	56
A.3	Considérations sur le repliement spatial et temporel . . . . .	57
A.4	Les fonctions de transferts avec approximation par Pade . . . . .	58
<b>B</b>	<b>Description des routines Matlab</b>	<b>59</b>
B.1	Les fichiers d'optimisation . . . . .	59
B.1.1	Optimisation en simulation . . . . .	59
B.1.2	Optimisation depuis des données réelles du MMT . . . . .	60
B.2	Utiliser ces routines dans un autre environnement . . . . .	61
B.3	La fonction de batch . . . . .	61
	<b>Bibliographie</b>	<b>66</b>



# Table des figures

1.1	Figure d'Airy de l'image d'un point rencontrant une ouverture circulaire. Avec une ouverture circulaire, le 84% de l'énergie se trouve dans le pic central. Comme les pupilles des télescopes sont obstruées au centre, une partie de l'énergie est transmise depuis le pic central dans les anneaux. . . . .	4
1.2	On mesure la largeur <i>angulaire</i> à mi-hauteur du pic central pour quantifier la qualité du ciel lors d'une observation astronomique. . . . .	4
1.3	Schéma du WFS. Le front d'ondes rencontre la trame de micro-lentilles pour former des spots sur le capteur CCD. Ce dernier mesure les pixels illuminés et permet le calcul de la position des centres des spots. . . . .	6
1.4	Image du spot sur les quatre quadrants du CCD sous chaque sous-pupille. .	7
1.5	Le senseur de front d'ondes mesure les pentes d'un front d'ondes. Ce front d'ondes est la somme de tous les modes dont il est formé. Avec les pentes, le reconstruteur permet de retrouver le front d'ondes. Le front d'ondes reconstruit est une approximation du front d'ondes entré dans le système. . .	8
1.6	Représentation schématique de la boucle de régulation d'optique adaptative.	9
1.7	Rapport de Strehl . . . . .	11
1.8	Détection et mesure par l'électronique de lecture d'un pixel non-éclairé du WFS avec un read-out de 8 $e^-$ . . . . .	12
1.9	Chute de performance en fonction de la magnitude de l'étoile . . . . .	12
2.1	Représentation schématique de la boucle de régulation d'optique adaptative.	14
2.2	Boucle de régulation pour la modélisation mathématique du système. . . .	15
2.3	Fonctions de transfert du système modélisé (bleu) et du système identifié (rouge). . . . .	18
2.4	Illustration de l'estimation de la variance du bruit par auto-corrélation avec l'approximation d'une courbe par minimisation des moindres carrés. La mesure du bruit correspond à la valeur séparant le premier point rouge et le point noir. . . . .	20
2.5	Binning 3x3 des pixels du capteur CCD utilisé avec le WFS du MMT. Le groupe de 3x3 pixels correspond à un quadrant. . . . .	21
2.6	Disposition des quatre quadrants sous chaque sous-pupille du Shack-Hartman. On voit les directions $x$ et $y$ ainsi que le binning 3x3. . . . .	21
2.7	Mesure du signal d'un quadrant avec et sans seuillage. Le signal bruité est chaque fois comparé au signal non-bruité. Le quadrant choisi est celui qui possédait le signal non-bruité avec la variance la plus élevée. . . . .	22

2.8	Mesure du mode Tip reconstruit après les pentes mesurées par le WFS. En noir : l'amplitude du mode issue d'un signal non-bruité, en rouge : le signal est bruité, mais n'est pas seuillé et sur le signal bleu, les conditions sont identiques, mais on effectue un seuillage pour s'affranchir de la composante continue ajoutée par le bruit de lecture. . . . .	23
2.9	Ce graphique a été obtenu avec l'évaluation d'une simulation respectant les paramètres décrits au tableau 2.2, en boucle ouverte avec une étoile de magnitude 12. . . . .	25
2.10	Densité spectrale de puissance des modes <i>tip</i> et <i>tilt</i> . Les conditions atmosphériques sont celles décrites dans le tableau 2.2. . . . .	26
2.11	Les fonctions de transfert du système, calculées pour différents gains. Ce graphique peut être demandé lors du calcul des gains optimaux. Les fonctions de transfert sont modélisées à partir des paramètres trouvés dans les paramètres de la simulation. . . . .	28
2.12	Affichage lors de l'étape de la recherche du minimum pour le mode 16 sur une étoile de magnitude 11. . . . .	29
2.13	Graphique des gains optimaux obtenus pour une étoile de magnitude 11. Les gains qui sont véritablement appliqués au système sont représentés en bleu. Les gains originaux calculés par l'algorithme d'optimisation sont donnés en rouge s'ils dépassent la ligne verte qui représente le gain limite selon le critère de stabilité. . . . .	30
2.14	Evolution du rapport de Strehl - utilisé comme critère de performance - en fonction de la magnitude de l'étoile. En bleu, la courbe du Strehl pour un système avec un gain plat, et en rouge pour un système avec des gains modaux optimisés. . . . .	33
3.1	Représentation des blocs de la boucle d'optique adaptative du MMT. . . . .	37
3.2	Le rapport de Strehl obtenu lors de simulation dont les gains modaux du contrôleur utilisé ont été calculés à partir d'une acquisition de données en boucle ouverte du temps indiqué en abscisse. . . . .	41
3.3	Représentation de la recherche d'un minimum d'une fonction par dichotomie. . . . .	42
3.4	Valeurs singulières de chaque mode du reconstituteur. . . . .	43
3.5	Illustration de l'addition de bruit sur le signal modal. En bleu : le signal non bruité; en rouge : le signal bruité. . . . .	44
3.6	Les gains optimaux calculés à partir de données prises sur le ciel. En (a) : le signal non-bruité (magnitude 5.4). De (b) à (i) : du bruit a été ajouté artificiellement sur le signal de base. Il a été ajouté linéairement entre chaque figure. Le tableau 3.3 donne une idée des magnitudes balayées correspondantes au niveau de bruit entre les figures. . . . .	45
3.7	Distribution fréquentielle du mode #1. . . . .	47
3.8	Distribution fréquentielle modale. En bleu : mode #3; en rouge : mode #30. . . . .	48
3.9	Comparaison de deux différents set de données pour le même mode (#3). En bleu : le set de données utilisées jusqu'ici, magnitude 5.4; en rouge : données acquises sur la même étoile mais avec un filtre optique permettant de diminuer le nombre de photons incidents et ainsi <i>simuler</i> une étoile de magnitude plus élevée, ici correspondante à 8.75. . . . .	48

3.10	Les gains optimaux calculés à partir de données prises sur le ciel avec un filtre optique monté sur le télescope pour <i>simuler</i> une étoile moins brillante. La magnitude correspondante de l'étoile est 8.75. En (a) : le signal non-bruité (magnitude 5.4). De (b) à (i) : du bruit a été ajouté artificiellement sur le signal de base. Il a été ajouté linéairement entre chaque figure (même niveau que sur la fig 3.10). Le tableau 3.3 donne une idée des magnitudes balayées correspondantes au niveau de bruit entre les figures. . . . .	49
3.11	Comparaison de la distribution fréquentielle des modes #54 (bleu) et #55 (rouge). En (a), les conditions sont les mêmes que pour la fig. 3.6(g) et en (b), les mêmes que pour la figure 3.6(h). . . . .	50
3.12	Comparaison de la distribution fréquentielle des modes #1 (bleu) et #2 (rouge). Le mode #1 possède un pic à 20 [Hz] bien plus prononcé que le mode #2. . . . .	51
3.13	Filtrage du pic à 50 [Hz] avec un filtre non causal appliqué lors d'un pré-traitement à l'optimisation modale. . . . .	52
A.1	Densité spectrale de puissance du <i>tip</i> dans l'atmosphère. Mesures faites en boucle ouverte selon les caractéristiques du tableau 2.2. La courbe bleue est issue de mesures sur une étoile de magnitude 0, la courbe rouge avec une étoile de magnitude 14. . . . .	56
B.1	Un bouton a été ajouté au panneau de contrôle de simulation pour lancer le processus de batch. . . . .	62



# Liste des tableaux

2.1	Variables utilisées pour la génération d'un écran de phases à partir des modes ainsi que pour sa décomposition. . . . .	17
2.2	Caractéristiques générales des simulations. . . . .	32
2.3	Les cas étudiés pour la comparaison des performances. . . . .	33
3.1	Les fonctions de transfert des blocs du MMT. Les fonctions de transfert avec les approximations des retards purs par Pade sont données en annexe dans le tableau A.1 . . . . .	38
3.2	Les paramètres du système d'optique adaptative monté sur le télescope MMT. . . . .	39
3.3	Evaluation du nombre de magnitudes balayées entre les sous-figures de la figure 3.6 . . . . .	46
A.1	Les fonctions de transfert des blocs du MMT dont les retards purs ont été approximés par Pade (ordre 2) . . . . .	58
B.1	Description des paramètres du fichier <i>parameters.m</i> . . . . .	60
B.2	Description des paramètres du fichier <i>parameters.m</i> . . . . .	63



# Introduction

Ce travail s'inscrit dans le domaine de l'optique adaptative. Une technologie moderne appliquée dans la science de l'astronomie. Depuis toujours, l'être humain est fasciné par les lumières provenant du ciel, ces étoiles qu'il observe, auxquelles il donne un nom. Les astronomes vont beaucoup plus loin, équipés d'instruments d'observation aussi sensibles que possible, ils scrutent le ciel pour découvrir des étoiles invisibles à l'oeil nu. Parfois même, des étoiles brillantes en cachent d'autres toutes proches qu'il est difficile de distinguer.

Les télescopes astronomiques possèdent un diamètre de plusieurs mètres. Ils ne donnent cependant pas forcément des images de qualité supérieure aux lunettes astronomiques utilisées par les amateurs dont le diamètre n'est que de quelques centimètres.

La grande surface des télescopes permet de récolter beaucoup de photons, donc beaucoup de lumière et de distinguer plus d'étoiles qu'une lunette, mais leurs grandes dimensions n'améliorent pas la qualité de l'image au sens de la diffraction car un autre phénomène intervient ; celui de perturbation atmosphérique. En effet, l'observation d'objets stellaires depuis le sol se fait au travers de multiples couches d'air en mouvement et de températures différentes. On observe donc une modification de l'indice de réfraction tout au long du chemin parcouru par un rayon lumineux provenant de l'objet pour arriver sur la pupille du télescope. Cela introduit des déphasages importants des rayons les uns par rapport aux autres et donc une déformation de l'image finalement observée.

La nature nous fournit des exemples concrets et visibles tous les jours. Le scintillement des étoiles, la vision au travers de l'air chaud d'un feu de bois ou sur nos routes de bitume bouillant en été. Le *flou* de ces images est dû à des modifications du trajet des rayons lumineux par des variations de l'indice de réfraction. L'astronome doit également composer avec cette caractéristique pour son observation.

Le principe de l'optique adaptative est de mesurer les différences de phases au moyen d'un capteur de front d'onde et d'appliquer une correction par un miroir déformable. Le miroir déformable est contrôlé par une boucle de réglage automatique dont le but est d'annuler l'erreur de phase résiduelle sur le front d'onde.

Dans le cas de faibles déformations et avec une étoile très lumineuse, la correction est très efficace. Cependant, l'observation d'étoiles dont la magnitude est élevée n'est guère améliorée par l'optique adaptative car le bruit de lecture et le bruit de photons viennent perturber la boucle de réglage de manière non négligeable. Une étude de la chute de performance associée a été faite lors du travail de semestre[1].

L'objectif est d'optimiser la boucle de réglage pour s'affranchir au maximum de problèmes liés au bruit de lecture et au bruit de photons. Cette approche a déjà été développée par

plusieurs auteurs [2, 7, 10, 11, 12, 13, 15, 14, 18]. Elle consiste à décomposer le front d'onde en une série de modes, et d'appliquer des paramètres différents au régulateur en fonction du mode contrôlé. On appelle cette technique le contrôle modal.

On peut classer les optimisations comme étant *a posteriori* ou *en ligne*. La première solution est évidemment la plus simple à mettre en oeuvre car elle se base sur des données pré-enregistrées qu'il suffit de traiter pour obtenir un résultat. Dans le second cas, il faut tenir compte de l'évolution en temps réel des conditions d'observation et calculer les paramètres d'optimisation sans perturber le bon fonctionnement du système par des temps de calculs importants.

Le premier chapitre de ce document contient une introduction plus détaillée à l'optique adaptative. On s'intéresse brièvement au problème de la turbulence et au besoin de correction au moyen d'un miroir adaptatif. On étudie les blocs importants composant la boucle de régulation. Puis on évalue les performances du système dans différentes conditions d'observation, en prenant en compte le bruit de photon et le bruit de lecture introduits dans le système. On comprend le besoin de considérer ces effets perturbateurs pour l'optimisation du système.

Dans le second chapitre, nous mettons en pratique la méthode d'optimisation modale développée par Gendron [16]. Nous décrivons en détail les étapes nécessaires à cette optimisation comme la mesure du bruit sur le signal par auto-corrélation, les fonctions de transfert du système et la transformée du Fourier du signal modal. Cette optimisation est appliquée sur un système d'optique adaptative en simulation ; les performances sont quantifiées et comparées au système non-optimisé.

Le troisième chapitre porte sur l'application de cette optimisation sur les données réelles mesurées sur le télescope MMT. Les contraintes pratiques sont étudiées et les résultats obtenus sont étudiés. Les performances de cette optimisation seront évaluées lors des prochains *runs* d'optique adaptative au télescope grâce à une boîte à outil d'optimisation modale développée à la fin de ce travail.



# Chapitre 1

## L'optique adaptative

### 1.1 Résolution d'un télescope

Les astronomes ont toujours cherché à augmenter la résolution des instruments d'observation de manière à obtenir des images plus précises et riches d'informations. Les progrès dans ce domaine ont permis l'étude et la découverte de plus en plus d'objets stellaires.

#### 1.1.1 La limite de diffraction

Un télescope parfait, observant une image ne subissant pas de perturbation est caractérisé par sa limite de diffraction. C'est une limite physique qu'il n'est pas possible de dépasser.

On connaît la formule définissant l'angle de diffraction du premier disque d'Airy (fig. 1.1) de la figure de diffraction lorsqu'un front d'ondes est limité par une ouverture circulaire [24] :

$$\theta = \frac{2.44 \cdot \lambda}{D} \quad (1.1)$$

où  $D$  représente le diamètre de l'ouverture, i.e. le diamètre du télescope et  $\lambda$  la longueur d'onde considérée.

L'angle de diffraction est proportionnel au diamètre du disque d'Airy, à un facteur de distance près. On remarque donc que l'on peut diminuer le diamètre de la tache d'Airy en augmentant le diamètre du télescope, ou plutôt de son miroir primaire.

Le critère de Rayleigh énonce que si la séparation de deux points objets est telle que le maximum de la figure de diffraction de l'un tombe sur le premier minimum de la figure de diffraction de l'autre, alors les deux objets sont résolus.

Cela implique que la résolution angulaire du télescope correspond au diamètre du disque d'Airy (éq. 1.1). On écrit :

$$\theta_{res} = \frac{1.22 \cdot \lambda}{D} \quad (1.2)$$

où  $\theta_{res}$  correspond à la résolution du télescope.

Calculée ainsi, la résolution est exprimée en radians. Les astronomes utilisent en général une autre unité qui est la seconde d'arc :

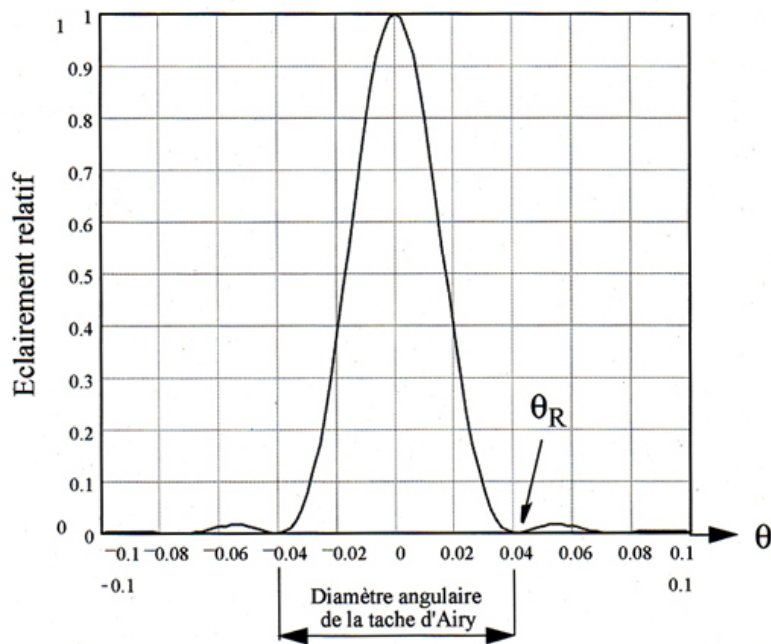


FIG. 1.1: Figure d'Airy de l'image d'un point rencontrant une ouverture circulaire. Avec une ouverture circulaire, le 84% de l'énergie se trouve dans le pic central. Comme les pupilles des télescopes sont obstruées au centre, une partie de l'énergie est transmise depuis le pic central dans les anneaux.

$$1[rad] = 206264.8'' \quad (1.3)$$

### 1.1.2 La turbulence atmosphérique

L'observation astronomique depuis le sol se fait au travers de diverses couches d'air en mouvement. On appelle cela les perturbations atmosphériques. A partir d'un certain diamètre du télescope, ce n'est plus la limite de diffraction qui définit la résolution de ce dernier, mais plutôt la qualité du ciel, le *seeing* qui est fonction des perturbations.

Le seeing indique la largeur angulaire à mi-hauteur de l'image d'une source ponctuelle d'une longueur d'onde donnée. Il correspond au FWHM : Full Width at Half Maximum (fig. 1.2).

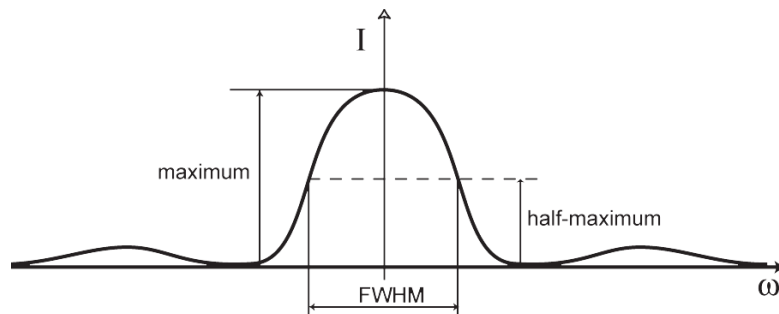


FIG. 1.2: On mesure la largeur *angulaire* à mi-hauteur du pic central pour quantifier la qualité du ciel lors d'une observation astronomique.

Malgré le fait que les grands diamètres des télescopes n'apportent pas d'amélioration de la résolution, on continue de construire des télescopes dont les diamètres mesurent plusieurs mètres pour en augmenter la sensibilité. Cela permet de capter plus de photons provenant de l'objet observé.

Il faut également savoir que les effets de la turbulence sont inversement proportionnels à la longueur d'onde. Ceci se comprend en étudiant le modèle de l'atmosphère décrit par Kolmogorov [9]. Sans entrer dans les détails, on peut décrire ce modèle comme une structure semi-fractale de masses d'air d'échelles décroissantes depuis l'échelle externe correspondant aux plus grands phénomènes macroscopiques (couches d'air, vents, perturbations météorologiques) et transmettant leur énergie cinétique d'une échelle à l'autre par tourbillonnement jusqu'à la plus petite échelle où l'énergie se dissipe en chaleur par frottements visqueux. Ce modèle permet de mettre en évidence différents paramètres dont le principal est le diamètre de Fried  $r_0$ .

La définition de  $r_0$  est la suivante [9] :

Le  $r_0$  pour un site, une turbulence et une longueur d'onde d'observation donnés, est égal au diamètre d'un télescope qui ne subirait pas la turbulence et dont la qualité d'image est équivalente au diamètre d'un télescope infini qui lui la subirait.

De ce modèle on tire également la relation suivante :

$$seeing \approx \frac{\lambda}{r_0(\lambda)} \quad (1.4)$$

donnant la largeur à mi-hauteur d'une image obtenue avec un télescope parfait.

Des considérations sur l'hypothèse de Kolmogorov permettent de déduire la loi suivante :

$$r_0 \propto \lambda^{6/5} \quad (1.5)$$

Cela implique que les grandes longueurs d'ondes (infrarouge) sont nettement moins affectées par la turbulence. Ce qui justifie également la construction des grands télescopes car le diamètre limite est plus élevé lors d'observation dans le domaine de l'infrarouge.

La convention actuelle pour comparer la performance des systèmes d'optique adaptative est d'utiliser la longueur d'onde de  $2.2\mu m$  (Bande K). C'est à cette longueur d'onde que l'on va optimiser le MMT.

L'optique adaptative consiste donc à mesurer la perturbation atmosphérique et de la corriger en temps réel pour ramener le télescope aussi près que possible de la limite de diffraction qui est une limite physique infranchissable. La mesure de la perturbation, en terme de déphasage, est effectuée au moyen d'un senseur de front d'ondes.

### 1.1.3 Le senseur de front d'ondes

Vue de la terre, nous pouvons considérer une étoile comme une source lumineuse ponctuelle émettant des ondes électromagnétiques sphériques. Autour de l'étoile, les fronts d'onde sont donc sphériques et séparés d'une longueur d'onde.

Lorsque ce front d'ondes arrive dans l'atmosphère, nous considérons que la portion de sphère observée est suffisamment petite par rapport à la dimension de la sphère elle-même pour dire que le front d'ondes est *plat* localement. Lors de son parcours au travers des différentes couches atmosphériques, ce front d'ondes se verra déformé car les rayons originalement paraxiaux sont déviés selon les indices de réfraction des couches d'air qu'ils rencontrent. L'image finale est donc déformée. Dans la suite de ce travail nous parlerons en terme de déphasage, mais on peut garder en tête l'idée d'un front d'ondes plat déformé pendant son parcours que l'on cherchera à *aplatir* autant que possible pour qu'il corresponde au mieux à l'image de l'objet observé.

Pour mesurer ce déphasage, il est nécessaire d'utiliser un senseur de front d'ondes (Wave Front Sensor : WFS). Dans notre cas nous utilisons un Shack-Hartman. Il est composé d'une trame de micro-lentilles placées perpendiculairement au flux lumineux incident. Chaque micro-lentille se comporte comme une sous-pupille et va focaliser une portion du front d'ondes sur une portion correspondante d'un capteur CCD (fig. 1.3). L'image de chaque sous-pupille doit se trouver au centre de l'axe optique de ces dernières. Si ce n'est pas le cas, c'est que le front d'ondes est localement déphasé. Et l'on mesure le déphasage sous chaque sous-pupille avec le capteur CCD.

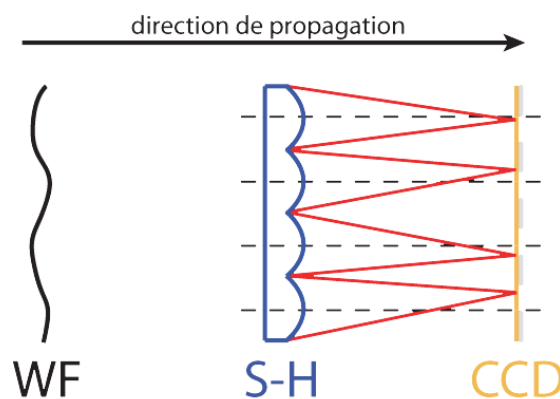
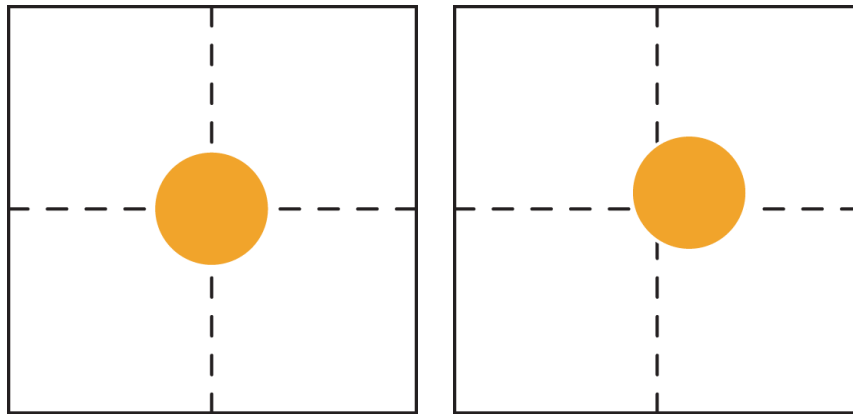


FIG. 1.3: Schéma du WFS. Le front d'ondes rencontre la trame de micro-lentilles pour former des spots sur le capteur CCD. Ce dernier mesure les pixels illuminés et permet le calcul de la position des centres des spots.

En aval de chaque sous-pupille se trouvent quatre pixels du capteur CCD correspondant à quatre quadrants. Ils permettent de mesurer l'énergie transmise au travers de la sous-pupille dans les deux directions (gauche-droite et haut-bas). Ainsi, on peut déterminer les pentes du déphasage moyen du front incident à la sous-pupille (fig. 1.4).

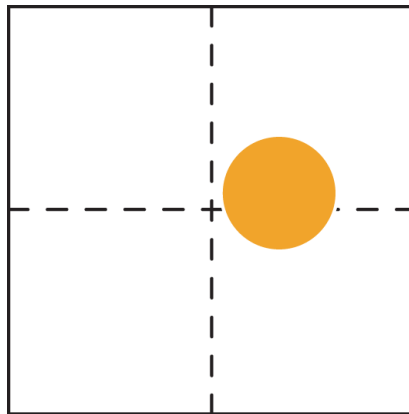
Ce que l'on mesure en réalité, c'est la proportion de photons illuminant une paire de quadrants par rapport à l'illumination totale des quatre quadrants. Ces valeurs ainsi calculées sont ensuite passées dans une table de conversion (Look-Up Table : LUT), dont la grandeur de sortie correspondra à la grandeur que l'on désire mesurer, dans la bonne unité.

Comme le montre la figure 1.4(c), il arrive parfois, surtout lorsque le signal est fortement bruité, que le spot n'illumine plus du tout l'un ou l'autre des quadrants. Dès lors, il n'est plus possible de connaître la proportion de photons incidents un côté plutôt que l'autre. On arrive en-dehors de l'étendue de mesure du capteur.



(a) Le spot est centré dans l'axe de la sous-pupille. Cela indique que le déphasage moyen sur toute la surface de la sous-pupille est nul.

(b) Le spot est décentré. La position du centre du spot peut être calculée en faisant le rapport du nombre de photons mesurés d'un côté par rapport à l'autre.



(c) Si le spot est trop décentré, il n'est plus possible de mesurer la position de son centre car certains quadrants ne sont plus illuminés.

FIG. 1.4: Image du spot sur les quatre quadrants du CCD sous chaque sous-pupille.

Cette opération est effectuée pour toutes les sous-pupilles et ainsi on obtient une matrice des pentes de déphasage sous chaque sous-pupille.

## 1.2 La reconstruction

On détermine le déphasage avec un senseur de front d'ondes qui mesure les pentes locales du déphasage. Il est ensuite très intéressant d'analyser ces informations sous un autre regard. A l'instar de la transformée de Fourier qui permet d'observer un signal de manière fréquentielle, on peut décomposer le signal du WFS en modes pour en faire une analyse ou un traitement modal.

Il existe différentes bases de modes qui peuvent être utilisées. Il est important de bien choisir la base dans laquelle nous désirons effectuer les calculs (la régulation par exemple) car celles-ci possèdent toutes des propriétés particulières. Il faut donc savoir tirer profit de ces particularités pour obtenir le meilleur résultat.

Une propriété indispensable pour faire du *contrôle modal* est d'avoir une base de modes orthogonale. Ainsi, une opération effectuée sur un mode n'affecte pas les autres. Cela permet de traiter chaque mode indépendamment les uns des autres. Pour un contrôle modal parfait, la base de mode doit être orthogonale dans l'espace des pentes comme dans l'espace des actuateurs du miroir. Pour nos simulations, nous allons utiliser la base de mode de Karhunen-Loeve qui possède cette propriété d'orthogonalité si l'on néglige l'oscuration centrale du télescope. La base de mode utilisée sur le télescope MMT a été calculée de manière à être orthogonale dans l'espace des pentes et du miroir pour le cas particulier du télescope.

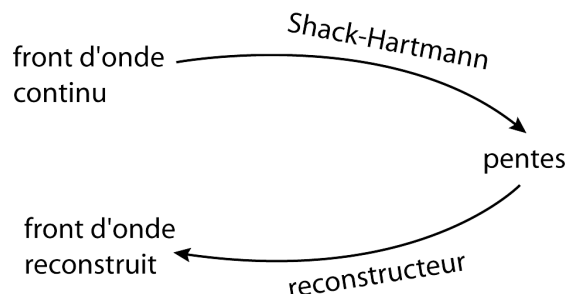


FIG. 1.5: Le senseur de front d'ondes mesure les pentes d'un front d'ondes. Ce front d'ondes est la somme de tous les modes dont il est formé. Avec les pentes, le reconstructeur permet de retrouver le front d'ondes. Le front d'ondes reconstruit est une approximation du front d'ondes entré dans le système.

La reconstruction est le calcul des modes à partir des pentes (fig. 1.5). Il est évident que ces opérations ne peuvent pas rendre avec exactitude la forme du front d'ondes original. Le WFS échantillonne spatialement le front d'ondes, il moyenne donc les informations de déphasage sur toute la surface des sous-pupilles respectives. Cet échantillonnage implique également un repliement spectral des fréquences trop élevées. Et nous allons voir que le reconstructeur n'est capable de reconstruire qu'un nombre fini de modes alors qu'un front d'ondes "naturel" en comprend une infinité.

Le passage du front d'ondes continu original au front d'ondes reconstruit s'appelle la décomposition modale. Le front d'ondes est ainsi décrit dans un espace qui correspond à la base de mode choisie.

Le principe du contrôle modal est de corriger l'erreur de déphasage sur chaque mode. Il est évident que les modes qui ne sont pas pris en compte par le reconstructeur ne seront pas corrigés. De plus, les modes dont les fréquences spatiales sont plus élevées que la fréquence spatiale du senseur de front d'ondes se replient et n'apportent donc que des perturbations supplémentaires.

La reconstruction est, mathématiquement, du calcul matriciel linéaire [16]. Nous considérons que la sortie du WFS est un vecteur de pente  $\vec{s}$  de dimension  $q$ . Nous supposons qu'il existe

une relation linéaire entre l'espace des modes et le vecteur  $\vec{s}$ . Nous allons donc construire une matrice d'interaction entre l'espace  $\vec{z}$  et  $\vec{s}$  où  $\vec{z}$  correspond au vecteur des amplitudes modales du front d'ondes.

Pour faire une décomposition modale complète il faut une infinité de modes. La matrice d'interaction possède donc un nombre de colonnes infini :

$$D_{\infty} \cdot \vec{z} = \vec{s} \quad (1.6)$$

Du point de vue pratique, il est évident que nous n'allons pas décomposer le front d'ondes en une infinité de modes, mais nous allons effectuer une approximation en ne prenant que  $n$  d'entre-eux. Nous obtenons donc une matrice  $D$  de  $n$  colonnes et  $q$  lignes.

$$D \cdot \vec{z} = \vec{s} \quad (1.7)$$

A priori, nous ne connaissons pas la matrice  $D$ , mais nous pouvons la construire expérimentalement [26]. Pour cela, il faut injecter un mode à la fois dans le système (sur le miroir déformable) et de mesurer les pentes données par le senseur de front d'ondes. En répétant l'opération pour chaque mode, nous pouvons construire la totalité de la matrice d'interaction  $D$ .

L'équation 1.7 décrit la réponse du WFS en fonction des amplitudes modales. Pour reconstruire un front d'ondes, c'est l'opération inverse que nous souhaitons effectuer. Mathématiquement, l'inverse d'une matrice n'est possible que pour des matrices carrées dont le déterminant est non nul. Il est cependant possible de calculer la matrice pseudo-inverse si des conditions particulières sont respectées [16].

$$D^+ = (D^t \cdot D)^{-1} \cdot D^t \quad (1.8)$$

Utilisée dans la relation :

$$\vec{m} = D^+ \cdot \vec{s} \quad (1.9)$$

### 1.3 Régulation

Faire de l'optique adaptative consiste à asservir un miroir déformable de manière à annuler le déphasage du front d'ondes incident pour que le front d'ondes illuminant le détecteur soit aussi faiblement déphasé que possible. Le schéma de la boucle de réglage correspond à la fig. 1.6.

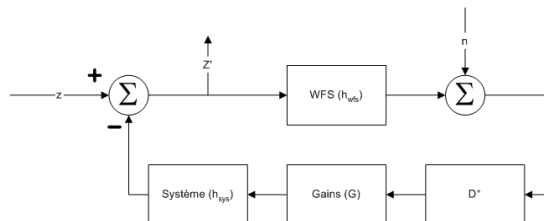


FIG. 1.6: Représentation schématique de la boucle de régulation d'optique adaptative.

Le front d'ondes provenant de l'objet observé  $z$  entre dans le système. Il est corrigé par le miroir déformable. C'est cette grandeur corrigée qui est utilisée comme mesure astronomique. C'est également cette grandeur qui est mesurée par le WFS pour connaître le déphasage résiduel. En passant dans le WFS et le détecteur (CCD), le front d'ondes est perturbé par plusieurs sources de bruit (ex : bruit de lecture, bruit de photons). C'est ensuite un vecteur de pentes  $\vec{s}$  qui entre dans le reconstruteur  $D^+$ .

On obtient donc le front d'ondes mesuré et bruité dans la base de mode choisie. C'est dans cet espace que l'on applique le régulateur et ses gains. Et c'est là qu'intervient la notion de contrôle modal. On peut appliquer un régulateur différent pour chaque mode dans cet espace car les modes sont indépendants les uns des autres - de part l'orthogonalité de la base modale. La difficulté consiste à déterminer le régulateur optimal pour chaque mode de manière à obtenir une erreur de phase résiduelle aussi faible que possible.

Le régulateur fournit une commande de correction modale qui est ensuite transmise au système qui la convertit dans l'espace du miroir déformable. C'est la commande finale qui pourra être appliquée sur les actuateurs qui déforment la membrane du miroir.

## 1.4 Chute de performance

L'évaluation de la performance du système d'optique adaptative en fonction du bruit injecté dans le système permet de cibler le travail en vue de l'optimisation désirée.

### 1.4.1 Critères de performance

Un premier moyen de caractériser la qualité d'un système d'optique adaptative est de calculer la valeur *rms* de l'erreur de phase résiduelle. L'erreur de phase résiduelle correspond à la différence de phase entre le front d'ondes incident sur le miroir déformable et la position de ce dernier.

Ces calculs se font dans la base modale. C'est-à-dire que la mesure des pentes résiduelles est multipliée par le reconstruteur et la position du miroir correspond à la commande modale fournie par le régulateur. Cette valeur *rms* est exprimée en amplitude modale. Comme l'amplitude des modes de la base est normalisée à 1 radian *rms*, la valeur *rms* de l'amplitude modale est donnée en radians.

Lors de prises de vues de bonne qualité, l'erreur *rms* est rapidement proche de zéro. Ce critère de performance n'est donc pas très *sensible*. On parle plus souvent du rapport de Strehl (Strehl Ratio). Il correspond au rapport entre l'intensité maximale de la figure de diffraction de l'image obtenue par rapport à l'intensité maximale de la figure de diffraction donnée par une image parfaite (fig. 1.7).

La figure de diffraction de l'image parfaite n'est pas accessible dans le cas d'une prise de vue réelle. Pour effectuer un calcul précis du rapport de Strehl, il faut donc estimer la forme de la figure de diffraction de l'image parfaite avec les données recueillies dans le système réel. Il est possible de le faire en utilisant la forme du pic central de la figure de diffraction de l'image partiellement corrigée. Ces calculs sont fastidieux.

On définit un autre critère appelé *énergie cohérente* [19] :



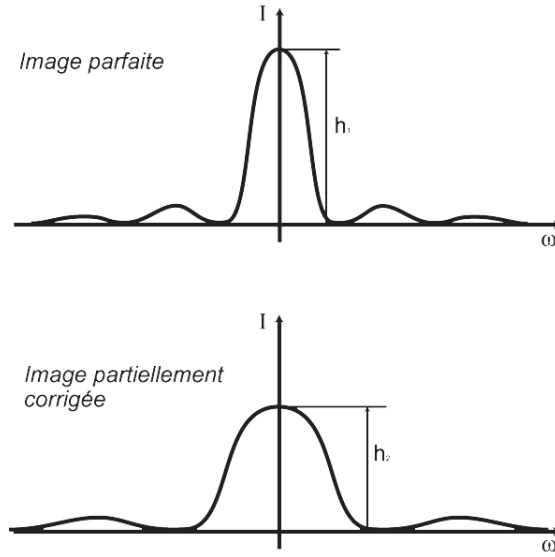


FIG. 1.7: Rapport de Strehl

$$E_c = e^{-\sigma_\phi^2} \quad (1.10)$$

où  $\sigma_\phi$  correspond à la variance de l'erreur de phase résiduelle, i.e la valeur *rms* de l'erreur de phase résiduelle. Ce critère présente l'avantage d'être directement accessible dans la simulation où la phase est définie numériquement sur toute la pupille. Par ailleurs, on peut montrer que si la variance de phase résiduelle est faible, alors :

$$S \approx E_c = e^{-\sigma_\phi^2} \quad (1.11)$$

Mais attention, cette approximation n'est bonne que lorsque  $E_c > 0.5$ . Elle reste cependant acceptable lorsque  $E_c > 0.2$ .

#### 1.4.2 Sources de perturbations

Parmi les nombreuses sources de perturbations, nous allons nous intéresser plus particulièrement au bruit de lecture et au bruit de photons[6].

Le bruit de lecture est directement lié à la conception du capteur CCD utilisé comme détecteur ainsi que de l'électronique qui le commande et conditionne le signal. C'est lors de la lecture de la charge produite par la lumière incidente sur le capteur que ce bruit vient s'ajouter au signal. Il est distribué selon une statistique de Poisson dont l'écart type  $\sigma_r$  varie entre :

$$\sigma_r = 5 \dots 9 e_- \quad (1.12)$$

Cette valeur dépend de la qualité du CCD et de son électronique de lecture. La fig 1.8 montre la détection de photo-électrons par le CCD pour un signal lumineux d'entrée nul. Ce graphique est tiré de la simulation de la détection d'un pixel non-éclairé du WFS et de sa mesure par l'électronique de lecture. On remarque très bien que la moyenne du signal mesuré est de  $8 e_-$ , la valeur du read-out pour ce cas.

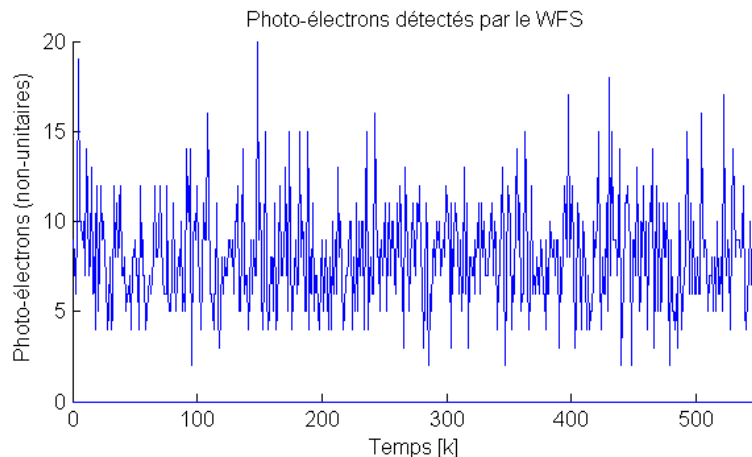


FIG. 1.8: Détection et mesure par l'électronique de lecture d'un pixel non-éclairé du WFS avec un read-out de  $8 e_-$ .

Le bruit de photons est une source de perturbation liée à un phénomène physique. Il n'est par conséquent pas possible de le modifier. Ce bruit suit également une statistique de Poisson de variance égale au nombre de photons moyen arrivant sur la surface du détecteur. On note donc son écart-type :

$$\sigma_p = \sqrt{N} e_- \quad (1.13)$$

où  $N$  correspond au nombre de photons moyen incidents sur la surface.

### 1.4.3 Observations expérimentales obtenues en simulation

Nous avons mesuré la chute de performance liée au bruit de photons et au bruit de lecture en fonction de la magnitude de l'étoile. Ces mesures ont été effectuées en simulation. Le critère de performance utilisé est le rapport de Strehl évalué avec la formule de l'énergie cohérente (éq. 1.11). La configuration de tous les paramètres est déterminée et l'on ne fait varier que la magnitude de l'étoile.

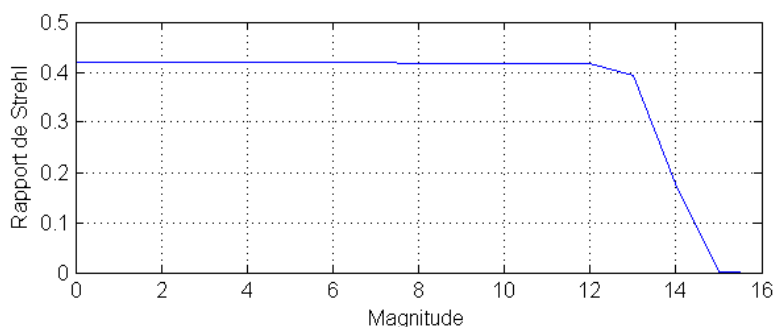


FIG. 1.9: Chute de performance en fonction de la magnitude de l'étoile

La figure 1.9 montre bien qu'une chute de performance intervient très brusquement lors d'observation d'étoiles à magnitude élevée. Dans la suite de ce travail, nous allons chercher des solutions pour retarder cette chute de performance. L'objectif étant de pouvoir utiliser l'optique adaptative pour des étoiles toujours plus faibles.

## Chapitre 2

# L'optimisation modale par minimisation de l'erreur de phase résiduelle

### 2.1 Généralités

Dans sa thèse[15], Gendron développe un raisonnement mathématique permettant d'optimiser le régulateur multi-variable d'un système d'optique adaptative. Cette optimisation s'applique à un régulateur de type intégrateur pur. Le terme d'optimisation modale n'est pas réellement adapté car il laisse le lecteur supposer que l'on peut l'appliquer de manière générale aux contrôleurs d'optique adaptative alors que cette idée se limite à l'intégrateur pur cité plus haut. D'autres auteurs, comme Dessenne [7], décrivent des raisonnements mis en oeuvre pour d'autres types de régulateurs.

Comme nous l'avons vu, l'optique adaptative permet d'augmenter sensiblement la résolution d'un télescope, ceci en diminuant au maximum l'erreur de phase résiduelle causé par les turbulences atmosphériques. Cependant, nous avons pu mettre en évidence une chute de performance notoire pour les étoiles peu brillantes.

L'idée d'appliquer un régulateur à contrôle modal a été suggérée par Léna en 1990. Ceci en optimisant les paramètres internes du régulateur pour chaque mode individuel. Autrement dit, le système est adaptatif en fonction de la turbulence atmosphérique, mais il modifie également ses paramètres internes en fonction de l'objet observé.

Le système corrige l'effet de la turbulence atmosphérique en boucle fermée tel que décrit dans le chapitre précédent. Les paramètres d'optimisation sont définis *a priori* par une acquisition de données en boucle ouverte sur l'objet que l'on désire observer.

### 2.2 Développement

Observons un peu plus en détail le système en boucle fermée (fig. 2.1). Nous cherchons à minimiser l'erreur résiduelle  $\vec{z}$  au cours du temps. On tire de la fig. 2.1 la fonction de transfert en boucle ouverte :

$$h_{ol}(f) = h_{wfs}(f) \cdot h_{sys}(f) \quad (2.1)$$

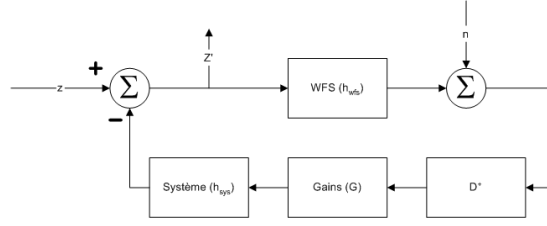


FIG. 2.1: Représentation schématique de la boucle de régulation d'optique adaptative.

On en déduit :

$$\vec{z}' = \vec{z} - h_{ol}(f) \cdot G \cdot D^+ \cdot D_\infty \cdot \vec{z}' - h_{sys}(f) \cdot G \cdot D^+ \cdot \vec{n} \quad (2.2)$$

Supposons que l'on observe une étoile faiblement brillante, cela implique que le signal fourni pas le WFS est fortement bruité. Cette hypothèse nous permet de négliger le repliement spectral des modes à fréquences spatiales élevées sur le WFS dont l'échantillonnage spatial est trop faible [16]. Ces considérations nous permettent d'écrire la relation suivante, écrite pour le  $i^{eme}$  mode. Ainsi, l'on remarque qu'il est possible d'ajuster le gain du régulateur indépendamment pour chacun d'entre-eux sans influencer les autres. Les gains relatifs sont notés  $g_i$  :

$$z_i' = h_{cor}(f, g_i) \cdot z_i - h_n(f, g_i) \sum_j D_{ij}^+ \cdot n_j \quad (2.3)$$

où  $h_{cor}$  est la fonction de transfert de réjection, i.e. la fonction de transfert de correction et  $h_n$  est la fonction de transfert du bruit transmis aux composants actifs tel que le miroir au travers du reconstructeur. Ce bruit a été injecté après le WFS dans la boucle, tenant compte du fond de ciel, du bruit de photons et du bruit de lecture. Sachant que le bruit n'est pas corrélé avec le signal, nous pouvons intégrer la relation 2.3 pour calculer la variance de la phase résiduelle :

$$\langle z_i'^2 \rangle = \int H_{cor}(f, g_i) \cdot \|z_i(f)\|^2 df + \int H_n(f, g_i) \cdot \|m_i(f)\|^2 df \quad (2.4)$$

où  $H_{cor}$  et  $H_n$  représentent le carré de l'amplitude des fonctions de transfert respectives  $h_{cor}$  et  $h_n$  et les fonctions  $z_i(f)$  et  $m_i(f)$  sont les transformées de Fourier respectivement de l'amplitude modale du signal corrigé, non bruité et du signal du bruit sur chaque mode.

Nous allons chercher à minimiser cette variance indépendamment pour chaque mode de manière à optimiser les performances du système.

## 2.3 Implémentation pratique

Pour effectuer l'optimisation, on a besoin de calculer chacun des termes présents dans l'équation 2.4. Les fonctions  $H_{cor}$  et  $H_n$  dépendent directement des caractéristiques de la boucle du système d'optique adaptative. On peut en déterminer un modèle mathématique.

Les termes  $z_i(f)$  et  $m_i(f)$  ne sont pas directement accessibles, il va falloir les remplacer dans l'équation 2.4 par des éléments mesurables.

### 2.3.1 Description des fonctions de transfert $h_{cor}$ et $h_n$

Commençons par modéliser le système par ses fonctions de transfert. La boucle de régulation est composée principalement des éléments suivants : le WFS, un intégrateur et le miroir déformable (fig.2.2). Dans le bloc *FILTER*, nous avons regroupé les opérations d'algèbre linéaire telles que le reconstructeur et les gains. L'amplitude de la fonction de transfert du bloc *FILTER* est unitaire pour tout le spectre de fréquence.

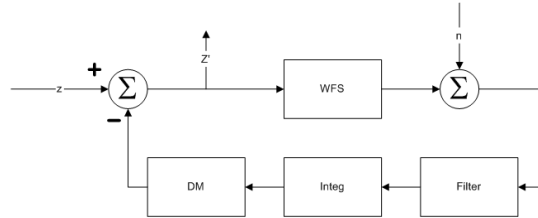


FIG. 2.2: Boucle de régulation pour la modélisation mathématique du système.

Etant donné que ces fonctions de transfert sont utilisées pour une simulation dans Matlab, nous nous intéressons à leurs modèles numériques en  $z$ .

La fonction de transfert en  $s$  du WFS est la suivante :

$$WFS(s) = \frac{1 - e^{-T \cdot s}}{T \cdot s} \quad (2.5)$$

Le détecteur intègre le flux de photons, et donc moyenne le signal durant une période  $T$ . Pour l'implémenter dans l'environnement Matlab, nous allons séparer ce modèle en deux blocs :

$$WFS_1(s) = \frac{1}{T \cdot s} \quad (2.6)$$

et

$$WFS_2(s) = \frac{1}{T \cdot s} \cdot e^{-T \cdot s} \quad (2.7)$$

ce qui donne :

$$WFS(s) = WFS_1(s) - WFS_2(s) \quad (2.8)$$

Le code Matlab, avec la conversion du modèle analogique en modèle numérique par la fonction *c2d* donne :

```
WFS1=tf(1,[general_struct.delta_t 0]);
WFS2=WFS1;
set(WFS2,'inputdelay',general_struct.delta_t);
WFS1_z=c2d(WFS1,general_struct.delta_t,'zoh');
WFS2_z=c2d(WFS2,general_struct.delta_t,'zoh');
WFS_model_z=WFS1_z-WFS2_z;
```

où `general_struct.delta_t` correspond à la période d'échantillonnage du système, i.e. la période d'intégration.

La fonction de transfert `integ_z` de l'intégrateur est évidente :

$$I(z) = \frac{1}{1 - z^{-1}} \quad (2.9)$$

On le modélise sous Matlab de la façon suivante :

```
integ_z = filt([1],[1 -1])
```

On modélise le miroir déformable par la configuration de ses pôles et zéros. Dans la simulation, ces informations sont stockées dans les paramètres du *contrôleur*. Il y a également un retard de commande pour le miroir.

```
DM_model_s = tf(controller_struct.zeros,controller_struct.poles,...
    'InputDelay',T_acd);
DM_model_z = c2d(DM_model_s,general_struct.delta_t,'zoh');
```

où  $T_{acd}$  correspond au délai de commande, et *controller\_struct* contient la configuration des pôles et des zéros.

Selon le schéma de la figure 2.1, on calcule les fonctions de transfert  $h_{sys}$  et  $h_{wfs}$  tels que :

```
h_sys = integ_z * DM_model_z;
h_wfs = WFS_model_z;
```

Et la fonction de transfert en boucle ouverte (éq 2.1) :

```
h_ol = h_wfs * h_sys;
```

Les fonctions de transfert  $h_{cor}$  et  $h_n$  peuvent ensuite être calculées telles que :

$$h_{cor} = \frac{1}{1 + h_{ol} \cdot g_i} \quad (2.10)$$

$$h_n = \frac{h_{sys} \cdot g_i}{1 + h_{ol} \cdot g_i} \quad (2.11)$$

### Identification du système

Nous avons effectué une modélisation mathématique du système. Nous pouvons maintenant le comparer avec le système en simulation. Pour cela, nous allons identifier le système en simulation.

Pour identifier un système, il faut l'exciter avec un signal d'entrée aussi riche que possible en fréquences. Nous allons le faire avec la fonction `randn()` de Matlab. Cette fonction génère un signal pseudo aléatoire tiré d'une distribution normale à moyenne nulle et de variance 1. Le signal ainsi généré sera relativement riche en fréquences. Un nombre infini d'échantillons aurait un spectre de fréquence plat. Nous allons cependant limiter notre identification à 1000 échantillons.

Le signal d'entrée de la simulation est un écran de phases. Il correspond à une matrice carrée décrivant géométriquement le déphasage du front d'ondes incident sur la pupille du télescope. Le signal de sortie est bien entendu dans les mêmes unités que le signal d'entrée, étant donné qu'il est de même nature et qu'il correspond au signal corrigé. Pour effectuer

l'identification, nous n'allons pas le faire avec des données bi-dimensionnelles, mais plutôt calculer la fonction de transfert d'un mode. C'est donc les amplitudes modales d'entrée et de sortie que nous allons utiliser.

Le signal d'excitation du système sera donc un écran de phases généré à partir d'une amplitude modale pseudo aléatoire, et dont tous les autres modes seront nuls. A la sortie du système, nous allons décomposer le front d'ondes en modes pour mesurer l'amplitude modale particulière du mode injecté dans le système. Nous utilisons pour ceci une base de mode canonique, décrivant l'espace des modes du miroir. Dans Matlab, cette base de mode correspond à une variable appelée `canonic_space`. Les dimensions des variables utilisées sont les suivantes :

<code>canonic_space</code>	(32400x49)
<code>wavefront</code>	(180x180)
<code>modes</code>	(49x1)

TAB. 2.1: Variables utilisées pour la génération d'un écran de phases à partir des modes ainsi que pour sa décomposition.

La variable `canonic_space` décrivant la base modale du miroir permet de convertir les modes en écran de phases et vice-versa. Premièrement, l'objectif est de générer un écran de phases à partir d'un vecteur de modes. Comme nous allons effectuer l'identification sur le mode numéro 1, l'amplitude de tous les autres modes est nulle. On obtient l'écran de phases en multipliant la base modale avec le vecteur de mode. On comprend bien que cette opération rendra un vecteur de 32000 colonnes. Il est nécessaire de re-ordonner les éléments du vecteur pour recomposer la matrice de (180x180). On le fait avec la fonction `reshape()` :

```
wavefront = reshape(canonic_space * modes, 180, 180);
```

L'opération inverse est effectuée au moyen de l'opérateur Matlab `\`. Cet opérateur permet d'effectuer une *multiplication inverse* de matrices :

```
modes = canonic_space \ reshape(wavefront, [], 32400);
```

Il est temps de comparer les fonctions de transfert obtenues par la modélisation mathématique et l'identification du système :

La fig 2.3 permet de se convaincre que le modèle mathématique représente bien le système. Attention, cette modélisation est celle de l'outil de simulation. Dès lors nous savons que la simulation est bien modélisée par les fonctions de transferts calculées plus haut. Ces données ne nous permettent pas de dire que l'outil de simulation représente bien le système réel. Pour cela, il faudrait identifier le système réel et le comparer avec le modèle. Cependant, nous pouvons accorder un bon degré de confiance sur ce modèle car il correspond au modèle attendu selon les expérimentations déjà effectuées et les informations que nous pouvons trouver dans la littérature.

### 2.3.2 Les termes $z_i(f)$ et $m_i(f)$

Reprenons l'équation 2.4. Il faut maintenant déterminer les termes  $z_i(f)$  et  $m_i(f)$ . Ils représentent l'énergie de l'amplitude modale contenue respectivement dans le signal non-

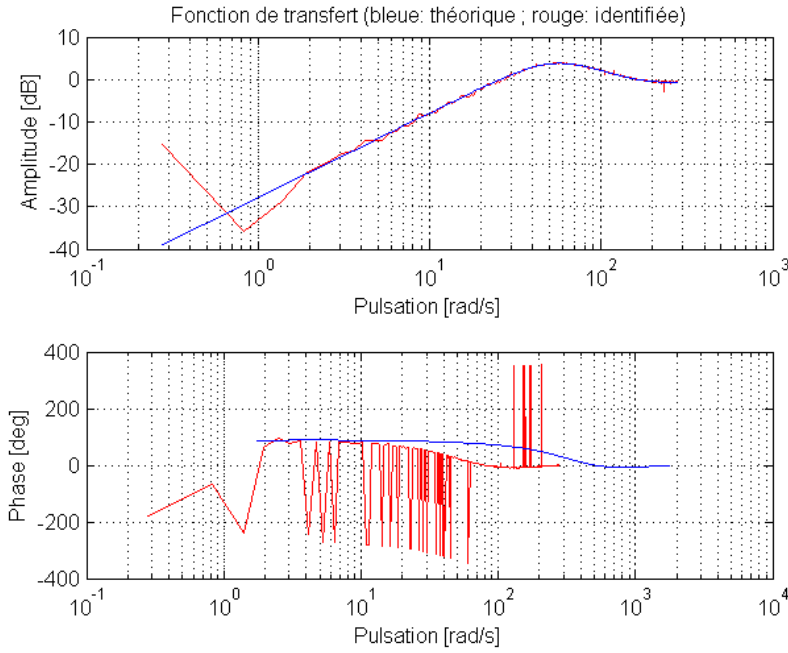


FIG. 2.3: Fonctions de transfert du système modélisé (bleu) et du système identifié (rouge).

bruité et dans le signal du bruit seul. Ces deux grandeurs ne sont pas directement accessibles car nous ne pouvons mesurer que le signal bruité  $\hat{z}_i(f)$ . Ce que l'on sait, c'est que  $\|\hat{z}_i(f)\|^2 = \|z_i(f)\|^2 + \|m_i(f)\|^2$ .

Gendron propose une approximation de l'équation 2.4 où tous les termes sont connus. L'hypothèse de base de son approximation réside dans le fait, qu'à basse fréquence le signal de la turbulence domine sur celui du bruit, et qu'en contre-partie, à haute fréquence c'est l'effet du bruit qui est le plus significatif. On peut écrire [16] :

$$\langle z_i^2 \rangle = \int H_{cor}(f, g_i) \cdot \left( \|\hat{z}_i(f)\|^2 - p(i) \cdot \sigma_n^2 \right) df + p(i) \cdot \sigma_n^2 \cdot \int H_n(f, g_i) df \quad (2.12)$$

où  $p(i)$  est le coefficient de propagation de chaque mode entre l'entrée du bruit et la base modale et  $\sigma_n^2$  correspond à la variance de l'erreur de phase résiduelle mesurée sur les pentes fournies par le WFS. Le terme  $p(i) \cdot \sigma_n^2$  n'est pas dépendant de la fréquence car nous supposons que c'est un bruit blanc dont le spectre de fréquence est plat.

L'approximation apportée à l'équation 2.4 ne signifie pas que  $\|\hat{z}_i(f)\|^2 - p(i) \cdot \sigma_n^2$  est une approximation de  $\|z_i\|^2$ , car ce serait faux. Ce sont les *intégrales* qui sont très proches l'une de l'autre. Cette hypothèse a été démontrée dans l'article de Gendron[16].

Les coefficients de propagation  $p(i)$  dépendent du reconstructeur. En effet, ils représentent le mode de propagation du bruit au travers de ce dernier pour passer de l'espace des pentes à l'espace des modes. Ils sont égaux aux éléments diagonaux de  $(D^t \cdot D)^{-1}$  :

$$p(i) = (D^t \cdot D)_{ii}^{-1} \quad (2.13)$$

Le terme  $\sigma_n$  correspond à la valeur *rms* du bruit de mesure. L'utilisation du terme  $p(i) \cdot \sigma_n^2$



permet de mesurer le bruit  $\sigma_n$  général sur la mesure des pentes. En évaluant le bruit par auto-corrélation, on le fait sur chaque pente. On prend donc ensuite la moyenne de la variance du bruit sur toutes les pentes. On peut également mesurer le terme  $p(i) \cdot \sigma_n^2$  en évaluant le bruit sur le signal modal après reconstruction. Cela évite donc le calcul des coefficient de propagation du bruit.

### Evaluation du bruit par auto-corrélation

Nous supposons que le bruit est totalement décorréolé du signal et de lui-même. Le bruit sur un échantillon à un temps  $k$  ne dépend ni du bruit sur  $k - 1$  ni du bruit sur  $k + 1$ . Alors que le signal de la turbulence est corrélé avec lui-même car il représente l'évolution continue des perturbations. Nous supposons donc que la perturbation au temps  $k + 1$  est largement dépendente de celle au temps  $k$ .

Prenons  $s_q(t)$ , la pente bruitée  $q$  mesurée par le WFS au cours du temps. Nous pouvons en calculer la fonction d'auto-corrélation :

$$R_s(t) = s_q(t) \star s_q(t)^1 \quad (2.14)$$

Toute l'énergie du bruit, étant donné qu'il n'est corrélé qu'avec lui-même, se retrouvera dans  $R_s(t = 0)$ . Cette énergie sera superposée à l'énergie du signal corrélé avec lui-même. En d'autres termes, si l'on pouvait calculer  $R_{s'}(t = 0)$ , l'énergie au temps  $t = 0$  du signal non bruité corrélé avec lui-même, alors l'énergie, i.e. la variance du bruit correspondrait à :

$$\sigma_q^2 = R_s(t = 0) - R_{s'}(t = 0) \quad (2.15)$$

Bien entendu,  $R_{s'}(t)$  n'est pas calculable. Une autre approche est donc utilisée, c'est d'estimer  $R_{s'}(0)$  à partir de  $R_s(t)$ . Il faut pour cela calculer la pente de la fonction  $R_s(t)$  au temps  $t = 0 + \epsilon$  :

$$m = \left. \frac{dR_s(t)}{dt} \right)_{t=0+\epsilon} \quad (2.16)$$

et on obtient ainsi la valeur estimée :

$$R_{s'}(0) = R_s(0 + \epsilon) - m \cdot \epsilon \quad (2.17)$$

Il s'agit maintenant de discrétiser ces équations de manière à les utiliser dans un environnement de calcul numérique comme Matlab. Le signal des pentes est donc  $s_q[k]$ , pour  $k \in \mathbb{N}$  et  $t = (k + 1) \cdot T$ ,  $T$  étant la période d'échantillonnage. On calcule l'auto-corrélation de ce signal :

$$R_s[k] = s_q[k] \star s_q[k] \quad (2.18)$$

L'estimation du premier ordre donne :

$$R_{s'}[1] = 2 \cdot R_s[2] - R_s[3] \quad (2.19)$$

<sup>1</sup>Le symbole  $\star$  est l'opérateur mathématique pour calculer la corrélation de deux signaux. On calcule l'auto-corrélation d'un signal par la corrélation de ce dernier par lui-même. L'auto-corrélation est définie ainsi :  $R_f(t) = \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T f(\tau) f(t + \tau) d\tau$

<sup>2</sup>Nous affirmons qu'en  $t = 0 + \epsilon$ , la fonction d'auto-corrélation  $R_s(t)$  est celle du signal non bruité, supposant que la fonction d'auto-corrélation du bruit est un dirac.

Pour améliorer l'estimation de  $R_s'[1]$ , nous pouvons utiliser plus d'échantillons de  $R_s[k]_{k>1}$  et calculer une courbe de régression par approximation des moindres carrés. Matlab possède deux fonctions tout à fait adaptées à cela : `polyfit()` et `polyval()`.

La figure 2.4 illustre cette estimation par approximation des moindres carrés dans Matlab. Les points bleus représentent la fonction d'auto-corrélation et la courbe rouge est la courbe de régression approximée sans prendre en compte le premier échantillon. La valeur séparant le premier point de la courbe rouge,  $R_s'[1]$  et le point noir,  $R_s[1]$  est la mesure de la variance du bruit décorréolé sur le signal.

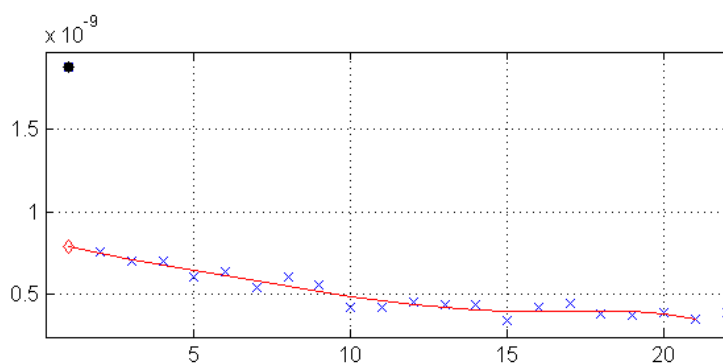


FIG. 2.4: Illustration de l'estimation de la variance du bruit par auto-corrélation avec l'approximation d'une courbe par minimisation des moindres carrés. La mesure du bruit correspond à la valeur séparant le premier point rouge et le point noir.

### 2.3.3 Considération sur le bruit de lecture

Comme cela a déjà été dit, l'optimisation modale est destinée à améliorer les performances de l'optique adaptative pour l'observation d'étoiles faiblement lumineuses. Dans ces conditions, le bruit a une influence considérable. Dans cette section, nous allons nous intéresser plus précisément au bruit de lecture car il ajoute une composante continue aux signaux mesurés.

Comme décrit au paragraphe 1.4.2, le bruit de lecture s'ajoute au signal selon une distribution de Poisson. Du moins c'est un des modèles de description de ce bruit. Il n'est pas absolu, mais c'est le modèle que nous allons utiliser pour nos simulations. En tous les cas, le bruit ajouté est à moyenne non-nulle. La figure 1.8 décrit le bruit détecté pour un signal d'entrée nul. Rappelons ici que ce bruit s'ajoute sur la mesure de chaque pixel (ou comme nous le verrons plus tard, sur chaque quadrant).

Il y a plusieurs méthodes de mesures des pentes avec un WFS de Shack-Hartman, celle que nous utilisons est une méthode dite *quad-cell* où l'on mesure les charges sur des quadrants. Dans le cas du capteur CCD utilisé avec le WFS du MMT<sup>3</sup>, on fait un binning 3x3 (fig. 2.5). C'est un processus qui consiste à combiner les charges de pixels durant la phase de lecture.

<sup>3</sup>Dans quelques mois, un nouveau WFS sera opérationnel sur le MMT. L'électronique de lecture du capteur CCD associé sera bien plus performante et plus rapide. Le binning ne sera plus forcément nécessaire et l'on pourra considérer la mesure des pentes par la méthode *decentroïde*.

Les zones définies par les neuf pixels du binning sont appelées des quadrants.

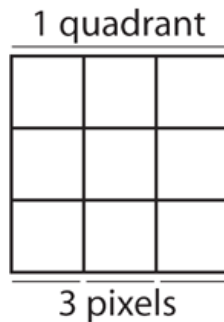


FIG. 2.5: Binning 3x3 des pixels du capteur CCD utilisé avec le WFS du MMT. Le groupe de 3x3 pixels correspond à un quadrant.

Ce sont ensuite quatre quadrants qui sont disposés sous chaque sous-pupille du Shack-Hartman et qui permettent la mesure du centre du spot. Les séparations centrales des quatre quadrants marquent les directions des pentes en  $x$  et en  $y$  (fig. 2.6). On calcule le rapport des photo-électrons détectés d'un côté de l'axe par rapport à ceux détectés de l'autre côté pour connaître la position du centre du spot. On comprend bien que si le nombre de photons détectés est du même ordre de grandeur que le nombre de photons détectés à cause du bruit de lecture, le calcul du rapport entre chaque quadrant sera largement faussé. Comme le bruit est une composante à moyenne non-nulle qui s'ajoute au signal détecté, les pentes mesurées sont en moyennes plus faibles.

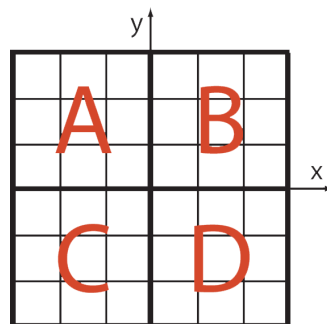
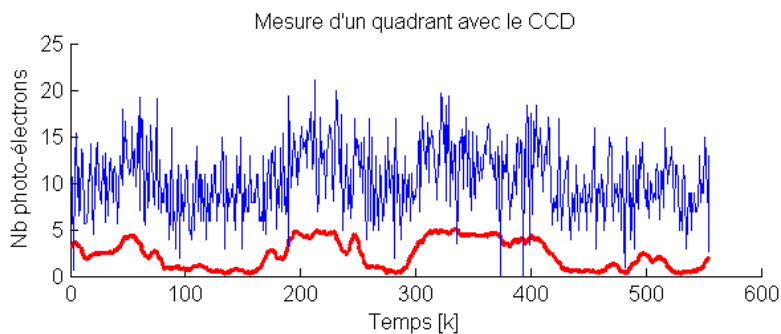


FIG. 2.6: Disposition des quatre quadrants sous chaque sous-pupille du Shack-Hartman. On voit les directions  $x$  et  $y$  ainsi que le binning 3x3.

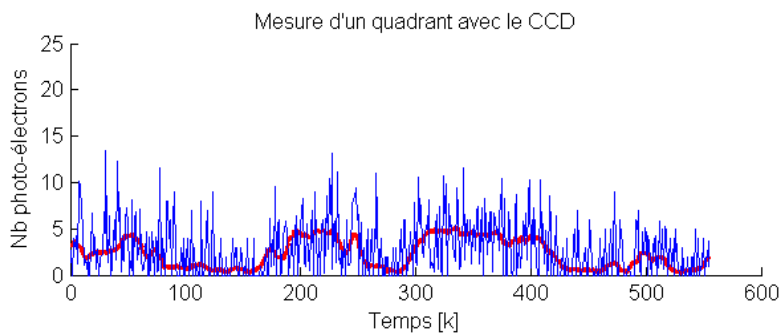
Ceci a pour effet de créer une perte d'amplitude du signal mesuré par le WFS car ce dernier est devenu moins sensible. Comme cette mesure est la mesure de l'erreur dans la boucle de réglage, on a une diminution de la bande passante dans la contre-réaction du système. Pour pallier à ce problème, il faut s'arranger pour effacer la composante continue du bruit. On va le faire par du seuillage. On suppose que l'on connaît la valeur moyenne du bruit de lecture ajouté au signal, et on enlève cette valeur de toutes les mesures du signal. Bien entendu, il n'est pas envisageable de mesurer un nombre de photo-électrons négatifs. Dès lors, après seuillage toutes les mesures sont *clippées* à 0.

Regardons d'un peu plus près les signaux obtenus dans des cas très défavorables comme celui d'une étoile de magnitude 15. La figure 2.7 a été obtenue en simulant la mesure d'un

quadrant du CCD au cours du temps, avec et sans seuillage. Le quadrant choisi est celui dont le signal non-bruité possédait déjà une variance élevée. Cela signifie que beaucoup d'informations sont présentes dans le signal, il est donc important de s'affranchir au maximum des effets du bruit.



(a) En rouge : le signal non-bruité, et en bleu : le signal bruité par le bruit de photons et le bruit de lecture.



(b) En rouge : le signal non-bruité, et en bleu : le signal bruité par le bruit de photons et le bruit de lecture, après seuillage.

FIG. 2.7: Mesure du signal d'un quadrant avec et sans seuillage. Le signal bruité est chaque fois comparé au signal non-bruité. Le quadrant choisi est celui qui possédait le signal non-bruité avec la variance la plus élevée.

On voit que l'effet du seuillage permet de ramener le signal bruité à un niveau correspondant à la bonne mesure. On s'affranchit ainsi de la perte d'amplitude sur le signal des pentes mesurées. On peut observer ceci sur la figure 2.8 où les pentes reconstruites sont comparées dans les conditions étudiées.

Cette figure porte sur l'étude d'un seul mode, le Tip. Mais l'expérience montre que les autres modes se comportent de manière similaire. De plus, c'est un mode comportant beaucoup de signal, il est donc l'un des plus importants. Le signal noir est le mode reconstruit dans des conditions de simulation où aucun bruit n'a été ajouté au signal. On voit très bien la perte d'amplitude sur le signal rouge. Le seuillage permet de ramener la mesure de ce signal à une amplitude correspondant au véritable signal, comme avec le signal bleu.

Nous avons amélioré la sensibilité du capteur, nous avons donc également augmenté sa sensibilité au bruit. Dans l'absolu, le signal ainsi transmis contient beaucoup plus de bruit, mais le rapport signal sur bruit ne varie que très peu. Nous avons donc un signal de *qualité*

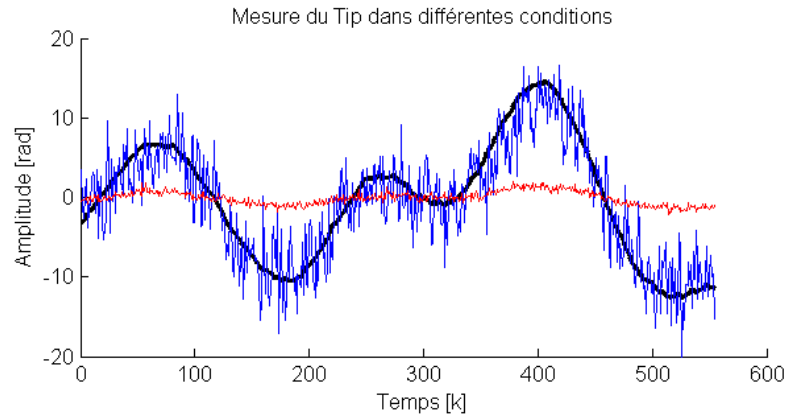


FIG. 2.8: Mesure du mode Tip reconstruit après les pentes mesurées par le WFS. En noir : l'amplitude du mode issue d'un signal non-bruité, en rouge : le signal est bruité, mais n'est pas seuillé et sur le signal bleu, les conditions sont identiques, mais on effectue un seuillage pour s'affranchir de la composante continue ajoutée par le bruit de lecture.

équivalente, mais dont l'amplitude est correspondante au signal que l'on désire mesurer. On retrouve la bande passante nécessaire à une bonne correction dans la boucle de réglage. Il faut cependant garder à l'esprit que ce processus n'améliore pas le rapport signal sur bruit. Les autres problèmes liés au bruit, sa propagation dans le système et les graves erreurs de correction dues à un niveau de bruit trop élevé par rapport au signal restent les mêmes.

### De la simulation à la pratique

Nous avons ici traité du bruit de lecture tel qu'il est modélisé en simulation. Dans ces conditions, on connaît bien ses caractéristiques et l'on peut aisément mesurer ces effets. Dans la pratique, sur un système réel, il est bien plus difficile de quantifier les effets du bruit et ses caractéristiques ne peuvent pas être connues avec autant de précision.

Comme pour tous capteurs CCD, celui qui est utilisé avec le WFS nécessite d'être calibré. Bien d'autres effets physiques ajoutent du bruit et faussent les mesures de ce capteur. On peut séparer les perturbations en deux catégories : les bruits fixes, et les bruits aléatoires.

Les bruits aléatoires, comme le bruit de photons ne peuvent pas être effacés par calibration. En revanche, les bruits fixes que l'on peut mesurer et caractériser le peuvent en partie. La calibration consiste donc à mesurer les effets des bruits fixes dans les conditions d'utilisation et de les effacer le mieux possible avec des artifices tels que des offsets, du seuillage ou même des tables de conversion.

Nous savons donc que le modèle décrit en simulation est optimiste, mais nous allons l'utiliser tel quel. De plus, le WFS du MMT n'est pas bien calibré car les bruits fixes ne sont en réalité pas très constants. C'est donc un capteur très difficile à calibrer. Ce que nous devons supposer pour continuer notre étude, c'est qu'il est possible de mesurer le signal des pentes sans perte d'amplitude.

### 2.3.4 L'algorithme

Nous allons nous intéresser maintenant à l'algorithme Matlab développé pour l'optimisation modale dans l'outil de simulation d'optique adaptative. Pour cela, nous allons porter notre regard sur les signaux exploités et montrer les traitements nécessaires à leur bonne utilisation. Il est basé sur l'implémentation de la relation 2.12 où l'objectif est de rechercher le gain optimal modal qui donnera une variance de phase résiduelle minimale.

L'algorithme décrit ici est très simple et non optimisé. Son implémentation est logique et tout est calculé de manière successive. Par exemple, les fonctions de transfert sont calculées dans une boucle indépendante de la boucle de recherche du minimum de la variance de phase. De plus, la recherche du minimum se fait après avoir calculé toute la courbe de la variance de phase résiduelle en fonction du gain pour tous les gains plutôt que d'utiliser un algorithme par dichotomie.

Cet algorithme a été étudié jusqu'à l'obtention de bons résultats, et comme son fonctionnement est très simple, il est aisé d'y faire des modifications. Il devra être optimisé par la suite.

#### Les paramètres

Des paramètres généraux sont ajustables. Ils sont surtout pratiques dans l'étude de l'algorithme et des signaux utilisés et ne sont pas destinés à être ajustés lors de l'utilisation régulière de la routine. On note par exemple la possibilité de choisir l'affichage des graphiques représentant les grandeurs clés utilisées comme par exemple le terme  $p(i) \cdot \sigma_n^2$ , les fonctions de transfert ou les figures relatives au calcul des gains optimaux.

Un autre paramètre important est celui qui définit les gains pour lesquels l'algorithme va calculer la variance de phase résiduelle. Dans cette version, c'est un vecteur contenant tous les gains qui vont être utilisés. Il définit donc la résolution désirée dans l'espace des gains, ainsi que les valeurs minimales et maximales pour lesquelles le système va chercher l'optimum.

#### Chargement

L'algorithme a besoin d'un fichier de simulation. On le lui fournit en paramètre. Il s'occupe ensuite de charger les fichiers nécessaires, tels que le fichier de simulation, les paramètres de la simulation et le reconstruteur. Il a par exemple besoin de connaître la période d'échantillonnage qu'il va trouver dans les paramètres généraux.

#### Le signal des pentes

Le signal principal utilisé, et celui fourni par le senseur de front d'ondes, c'est-à-dire le signal des pentes. Comme nous traitons les données en *post-processing*, toutes les pentes sont déjà mesurées. Elles proviennent d'une simulation en boucle ouverte et correspondent donc à la mesure directe de la perturbation non-corrigée. Elles sont rendues sous forme d'une matrice de 216 lignes et  $k$  colonnes,  $k$  étant le nombre d'échantillons temporels. Les 216 lignes proviennent des  $2 \cdot 108$  pentes en  $x$  et  $y$  dont l'éclairement est suffisant (au moins 50% du quadrant éclairé).

De ce signal de pentes, nous calculons le terme  $\sigma_n^2$  comme étant la moyenne de la variance des pentes, cette dernière étant évaluée par auto-corrélation (§ 2.3.2). Avec les coefficients

de propagation obtenus selon l'équation 2.13, nous pouvons calculer le terme  $p(i) \cdot \sigma_n^2$ .

Si l'on demande l'affichage des  $p(i) \cdot \sigma_n^2$ , on obtient le graphique de la figure 2.9. Cette figure nous donne des informations utiles sur les modes de propagation du bruit dans le système. On voit rapidement quels sont les modes qui vont poser problème.

On remarque que le bruit se propage beaucoup sur les premiers modes, mais comme ces modes contiennent également beaucoup de signal, le bruit aura moins d'importance. En revanche, dans les modes élevés on trouve des pics où le bruit est très présent, l'influence de ces modes dont le signal non-bruité est faible devra être diminuée dans la contre-réaction du système.

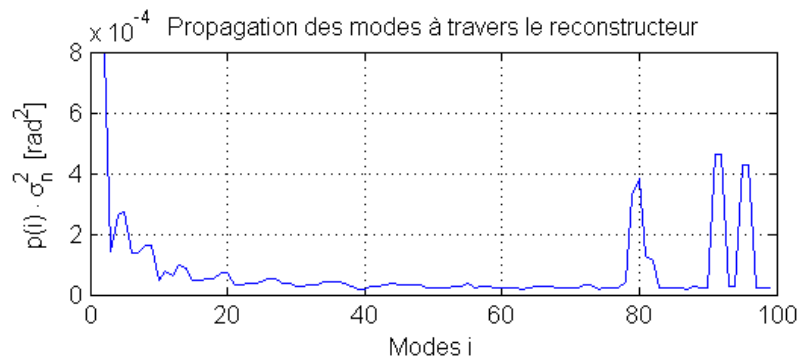


FIG. 2.9: Ce graphique a été obtenu avec l'évaluation d'une simulation respectant les paramètres décrits au tableau 2.2, en boucle ouverte avec une étoile de magnitude 12.

### Le signal reconstruit

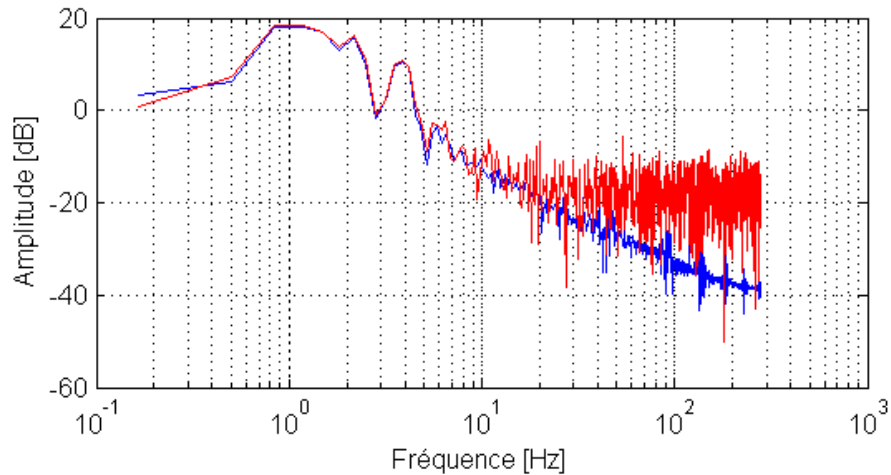
Du signal des pentes, on tire le signal reconstruit en le passant dans le reconstructeur. Il représente l'erreur modale non-corrigée pour la simulation en boucle ouverte. On a donc l'évolution temporelle modale de la turbulence atmosphérique. On obtient le terme  $\|\hat{z}_i(f)\|^2$  par transformée de Fourier.

Avec la transformée de Fourier, il faut également calculer l'axe des fréquences. Dans les paramètres généraux de simulation, on va trouver la période d'échantillonnage  $T_e$  du système, et selon le théorème de Shannon, on sait que les fréquences maximales que le système va pouvoir mesurer sont données par la relation :

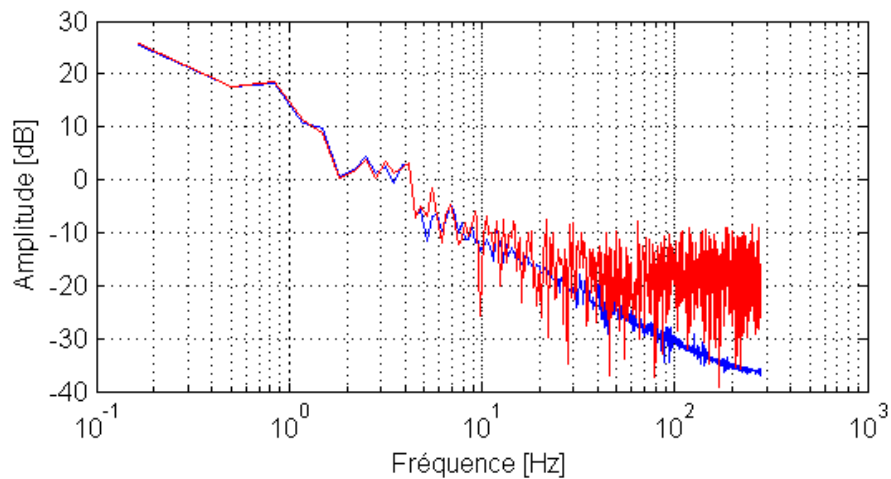
$$f_{max} = \frac{1}{2 \cdot T_e} \quad (2.20)$$

Comme la FFT calculée par Matlab est donnée sur tout le spectre de fréquence, elle varie de  $-f_{max}$  à  $+f_{max}$ . Pour garantir une bonne synchronisation de l'axe des fréquences avec les valeurs calculées par la fonction `fft()` de Matlab, nous avons décrit l'axe de fréquence comme un vecteur variant linéairement de  $-f_{max}$  à  $+f_{max}$ . Etant donné que nous n'utilisons que les fréquences positives, nous tronquons l'axe de fréquences pour ne garder que les valeurs supérieures à zéro et nous ne gardons que les valeurs de la FFT correspondantes à cet axe de fréquences.

Il est bon de se faire une idée de la forme des signaux avec lesquels on travaille. La fig 2.10 représente la densité spectrale de puissance des modes *tip* et *tilt*<sup>4</sup>. On voit que lors de l'observation de magnitude faible, donc pour des étoiles brillantes, le bruit n'est pas vraiment visible sur le spectre. Nous savons que c'est un bruit blanc qui s'ajoute au signal, et pour une magnitude élevée, on le remarque en haute-fréquence. Comme dans ces conditions le signal est bien plus faible, le bruit a plus d'importance et il commence à *dépasser* de la courbe du signal.



(a) Densité spectrale de puissance du *tip* pour une magnitude 0 (courbe bleue) et magnitude 14 (courbe rouge).



(b) Densité spectrale de puissance du *tilt* pour une magnitude 0 (courbe bleue) et magnitude 14 (courbe rouge).

FIG. 2.10: Densité spectrale de puissance des modes *tip* et *tilt*. Les conditions atmosphériques sont celles décrites dans le tableau 2.2.

<sup>4</sup>Ces spectres ne respectent pas la forme théorique qu'ils devraient avoir tel que décrit dans Conan 95 [8]. Ce problème a été détaillé en annexe A.2



### Les fonctions de transfert

Le système a été modélisé par les fonctions de transfert des blocs (§ 2.3.1). Cette modélisation a été regroupée dans une routine annexe qui rend automatiquement la fonction de transfert en boucle ouverte  $h_{ol}$  et la fonction de transfert des blocs situés après le point d'introduction du bruit appelée  $h_{sys}$ .

Une boucle `for` calcule toutes les fonctions de transfert  $h_{cor}(f, g_i)$  et  $h_n(f, g_i)$ . Comme ce sont les termes  $H_{cor} = \|h_{cor}(f, g_i)\|^2$  et  $H_n = \|h_n(f, g_i)\|^2$  qui sont utilisés dans l'équation, on calcule directement les amplitudes des fonctions de transfert au carré. On garde ces valeurs dans une matrice.

On peut demander à l'algorithme de tracer un graphique des fonctions de transfert modélisées. Il prendra 6 gains répartis sur le vecteur de gains donné en paramètre, espacés linéairement et représentera les fonctions de transfert sur un même graphique. Ceci pour la fonction de transfert de réjection et la fonction de transfert du bruit. Le résultat est donné par la figure 2.3.4.

### La recherche du minimum

Une double boucle `for` parcourant tous les modes, et tous les gains calcule une matrice représentant la variance de phase résiduelle en fonction de ces deux variables. On écrit directement la relation 2.12 en utilisant les grandeurs calculées précédemment. Comme Matlab est un environnement de calcul numérique, on discrétise l'équation 2.12 en :

$$\langle z_i^2 \rangle = \sum \|H_{cor}[f, g_i]\|^2 \cdot \left( \|\hat{z}_i[f]\|^2 - p(i) \cdot \sigma_n^2 \right) \cdot \Delta f + \sum \|H_n[f, g_i]\|^2 \cdot \Delta f \quad (2.21)$$

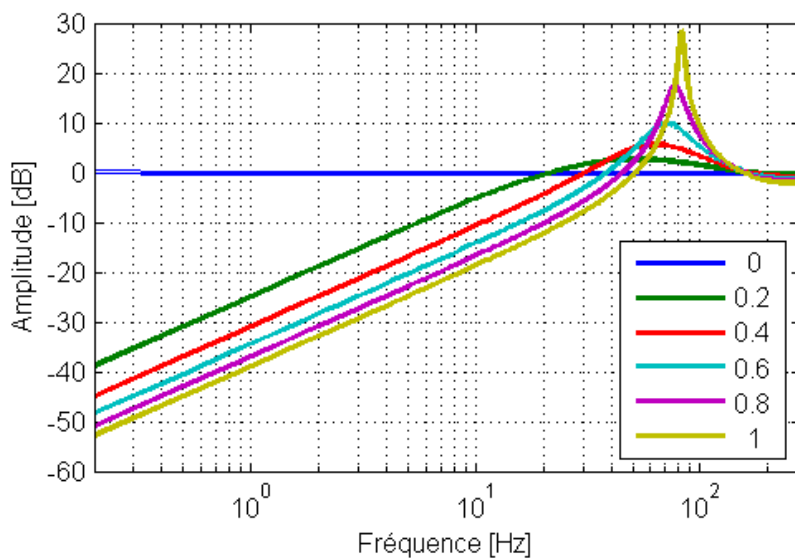
Il suffit ensuite de parcourir la matrice pour chaque mode selon l'axe des gains, et trouver le minimum. Ce sera le gain optimal pour ce mode.

On peut choisir d'afficher quelques graphiques lors de l'étape de la recherche du minimum. La figure 2.12 montre l'état de l'affichage pour l'optimisation de l'observation d'une étoile de magnitude 12 et sur le mode numéro 16. On peut voir le bruit qui a été évalué par auto-corrélation sur le premier graphique de la figure. La ligne rouge représente le niveau  $p(i) \cdot \sigma_n^2$  que l'on peut comparer à l'amplitude fréquentielle du mode. On voit que l'évaluation est correcte. Le graphique central montre les fonctions de transfert correspondantes au gain optimal trouvé. Le dernier graphique montre la variance de phase résiduelle telle que calculée avec la relation 2.21. Cet affichage change pour chaque mode étudié. Ainsi on voit la courbe de variance résiduelle pour chaque mode individuellement.

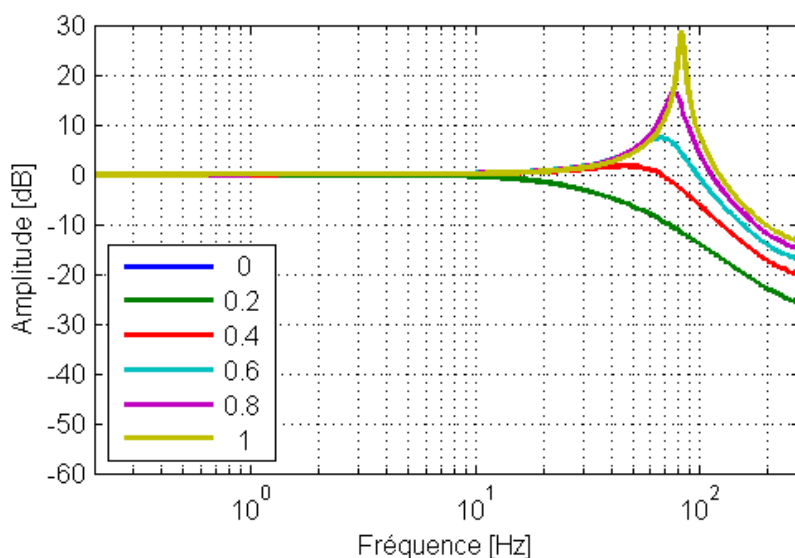
### Des problèmes de normalisation

Dans cet algorithme, nous avons utilisé deux fonctions Matlab que nous déclarons comme *sensibles à une normalisation*. Elles sont la fonction utilisée pour calculer l'auto-corrélation : `xcorr()` et celle pour calculer la transformée de Fourier : `fft()`.

Bien souvent ces fonctions sont utilisées comme grandeurs relatives et ne nécessitent pas une normalisation. Mais dans notre cas elles sont destinées à calculer respectivement le terme  $p(i) \cdot \sigma_n^2$  et  $\|\hat{z}_i(f)\|^2$  dont nous calculons ensuite la différence. Nous devons donc garantir



(a) La fonction de transfert de réjection, i.e. de correction



(b) La fonction de transfert de propagation du bruit.

FIG. 2.11: Les fonctions de transfert du système, calculées pour différents gains. Ce graphique peut être demandé lors du calcul des gains optimaux. Les fonctions de transfert sont modélisées à partir des paramètres trouvés dans les paramètres de la simulation.

que ces deux grandeurs sont normalisées correctement.

Considérant le théorème de Parseval, nous pouvons comparer le signal fréquentiel tiré de la transformée de Fourier dans Matlab du signal temporel correspondant. Cela nous permet de déterminer le facteur de normalisation pour la fonction `fft()`. Le raisonnement étant relativement long, il est détaillé en annexe A.1. On trouve qu'il faut multiplier le résultat de la `fft()` par la période d'échantillonnage du système. La transformée de Fourier d'une

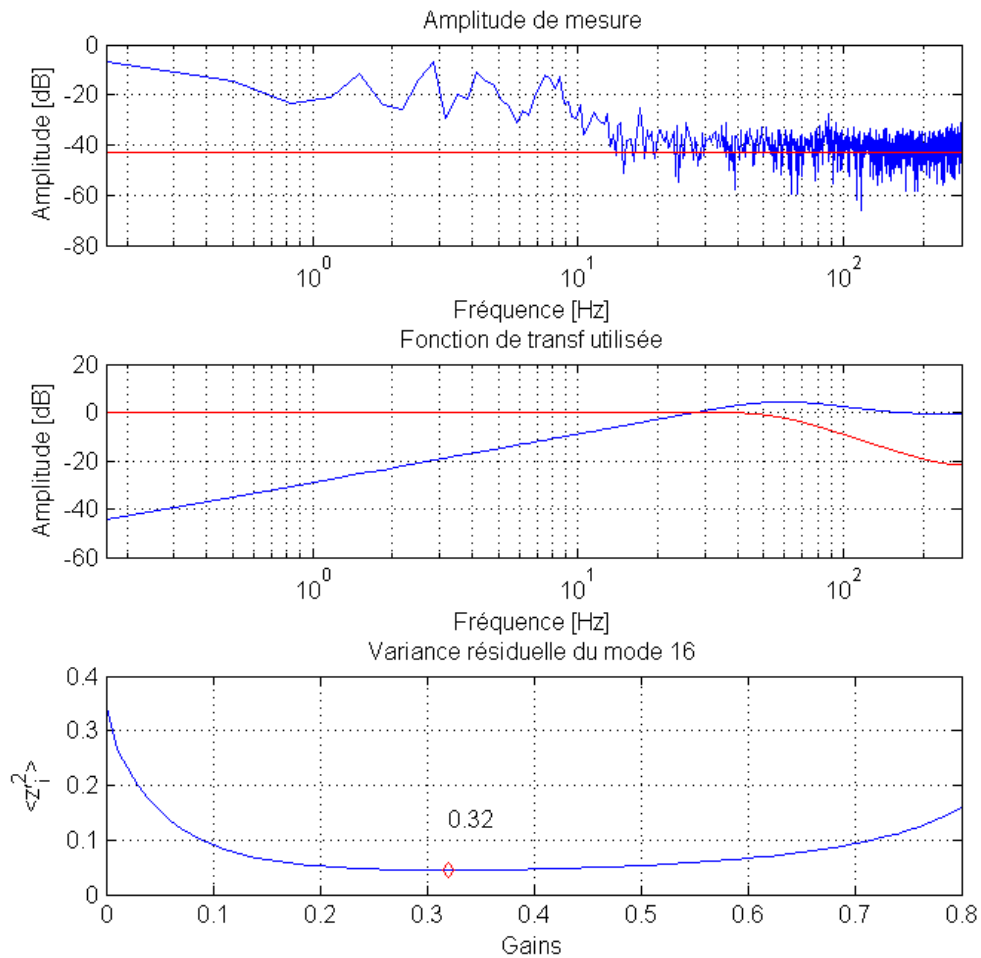


FIG. 2.12: Affichage lors de l'étape de la recherche du minimum pour le mode 16 sur une étoile de magnitude 11.

fonction  $x$  échantillonnée toutes les  $t_e$  secondes s'écrit :

$\text{fft}(x) * t_e$

De plus, des théorèmes de Parseval et de Wiener-Khintchine on peut déduire :

$$\mathcal{F}\{f(t) \star f(t)\} = \|\mathcal{F}\{f(t)\}\|^2 \quad (2.22)$$

qui énonce que la transformée de Fourier de la fonction d'auto-corrélation d'une fonction temporelle est égale à la densité spectrale de puissance de cette dernière. On en déduit que le facteur de normalisation de la fonction d'auto-corrélation est le même que pour la transformée de Fourier, élevé au carré pour correspondre à la densité spectrale de puissance. L'expérience sur Matlab montre que cette dernière déduction est correcte. L'auto-corrélation d'une fonction décrite par le vecteur  $x$  échantillonnée toutes les  $t_e$  secondes s'écrit :

$\text{xcorr}(x) * t_e^2$

### 2.3.5 La stabilité

Tel que décrit plus haut, l'algorithme ne rend que des gains calculés selon un critère d'optimisation tenant compte du résultat obtenu. On cherche à minimiser la variance de phase, mais on ne tourne pas les yeux vers la stabilité du système. Le critère de stabilité d'un système numérique demande à ce que tous les pôles du système soient compris dans le cercle unité. Ce critère, appelé aussi le critère de stabilité absolu n'est pas tolérant face aux problèmes de modélisation du système. Pour le rendre plus sûr on définit les notions de marge de phase et de marge de gain. Sans détailler ces concepts ici, nous disons simplement que nous désirons garantir une marge de phase de  $45^\circ$ . Ce dernier paramètre est souple dans l'algorithme, on peut le modifier aisément en fonction de l'expérience pratique que l'on remarquera lors de l'utilisation des gains optimisés.

Comme l'algorithme de calcul des gains modaux calcule la fonction de transfert en boucle ouverte du système, il se charge également de calculer le gain maximal que le système peut supporter pour respecter cette marge de phase. Comme ce calcul se fait par un ajustement avec le diagramme de Bode, l'algorithme indique à l'utilisateur la marge de phase et la marge de gain définitives correspondantes au gain maximum.

Il est donc ensuite tout à fait possible de saturer les gains modaux dépassant le gain maximal à cette dernière valeur. Cela permet de garantir qu'aucun mode ne deviendra instable dans le système. De plus, on sait que la base modale n'est jamais parfaitement orthogonale, et donc un mode instable, avec le temps, va commencer à destabiliser les modes voisins, ce qui complique encore les problèmes de stabilité. Une fois que l'algorithme d'optimisation a rendu les gains optimaux et la valeur du gain maximum en raison de la stabilité, on peut les exploiter. Ceci est fait dans une routine de plus haut niveau qui a été chargée de sélectionner le fichier et qui va écrire le nouveau contrôleur pour la prochaine simulation. On peut lui demander d'afficher un graphique des gains ainsi obtenus.

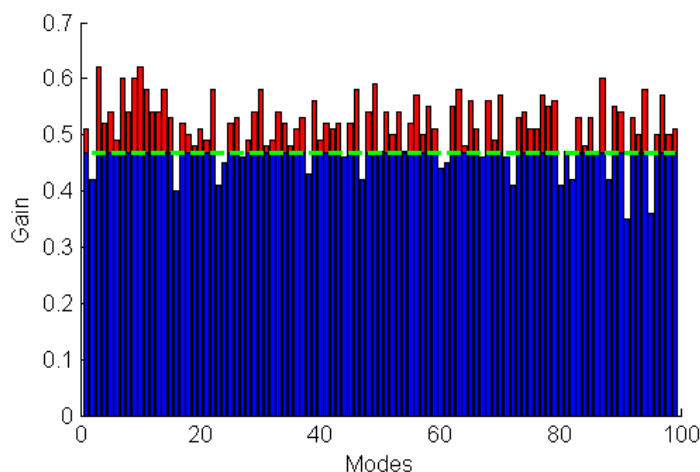


FIG. 2.13: Graphique des gains optimaux obtenus pour une étoile de magnitude 11. Les gains qui sont véritablement appliqués au système sont représentés en bleu. Les gains originaux calculés par l'algorithme d'optimisation sont donnés en rouge s'ils dépassent la ligne verte qui représente le gain limite selon le critère de stabilité.

La figure 2.13 montre les gains optimaux pour une étoile de magnitude 11. On voit que l'on a dû saturer les gains (rouges) rendus par l'algorithme à la valeur du gain maximum respectant le critère de stabilité demandé (vert). Ce sont finalement les gains représentés en bleu qui sont enregistrés dans le nouveau contrôleur qui sera utilisé pour la prochaine simulation.

## 2.4 Simulations et mesures

L'outil de simulation, sans optimisation modale ajustait le gain du régulateur par la synthèse de ce dernier. C'est un *gain plat*<sup>5</sup> qui était donc appliqué sur tous les modes du système. La synthèse du régulateur garantissait une très bonne stabilité du système. Le calcul de ce gain s'appuyait donc uniquement sur les caractéristiques du système alors que le but de l'optimisation modale est de permettre au système de s'adapter aux conditions extérieures pour tirer le meilleur profit des signaux traversant le système.

Les performances du système avec un gain plat ont déjà été étudiées (§ 1.4.3), cependant nous allons représenter ces performances pour un cas plus précisément étudié dans ce chapitre. Les caractéristiques du système<sup>6</sup> sont résumées dans le tableau 2.2.

### Comparaison des performances

Le critère de performance utilisé est le rapport de Strehl. Il a été calculé avec l'éq. 1.11 avec la mesure de l'erreur de phase résiduelle. La figure 2.14 montre le rapport de Strehl obtenu pour des simulations de 3 [s] en fonction de la magnitude de l'étoile observée. Les trois cas décrits dans le tableau 2.3 sont comparés sur ce graphique.

On obtient bien les résultats attendus. Plusieurs remarques importantes sont à signaler :

1. Pour les étoiles brillantes, on remarque que plus le gain est élevé, plus les performances sont bonnes. Le rapport signal sur bruit étant bon dans ces conditions, on ne note pas de problème lié au bruit. Le critère qui limite le gain est donc la stabilité du système. Comme le critère de stabilité a été calculé avec certaines tolérances (marge de phase ou marge de gain), il serait possible d'améliorer les performances avec un gain encore plus élevé tout en prenant le risque de voir le système diverger à cause d'une mauvaise excitation.
2. Pour les étoiles faiblement brillantes, la chute de performance est plus rapide avec un gain élevé. Il faut savoir que la chute de performance apparaît lorsque le bruit commence à *dépasser* (fig. 2.10) du signal et qu'il influence sensiblement la commande de correction. Dans cette situation, plus le gain est élevé, plus on donne d'importance aux erreurs de correction liées au bruit.
3. Dès que la chute de performance liée au bruit intervient, l'optimisation modale est capable d'améliorer les performances. En effet, comme l'objectif de cette optimisation est de diminuer la variance de l'erreur de phase résiduelle pour chaque mode en tenant compte de la propagation du bruit dans le système, la bande passante des *mauvais*

<sup>5</sup>Le terme gain plat est utilisé dans tout ce document pour parler d'un gain constant pour chaque mode :  $g(i) = cste$  avec  $i = 1 \dots n_{mode}$ .

<sup>6</sup>La vitesse du vent et la dimension de l'échelle interne ont été choisis pour éviter des problèmes de repliement spatial et temporel lors des simulations. Plus de détails sont donnés en annexe (A.3) à ce sujet.

Atmosphère	Composée d'une couche se déplaçant à 25 [m/s]. L'échelle externe est de 100 [m] alors que l'échelle interne est de 9 [cm]. L'atmosphère est exactement la même pour toutes les simulations. Cela permet d'éviter les effets statistiques liés à la génération d'un écran de phases.
Senseur de front d'ondes	Ecart-type du bruit de lecture : $\sigma_r = 8 e_-$
Look-up Table	Elle a été définie pour une atmosphère de $r_0 = 16$ [cm]. Elle convertit la mesure du WFS en une mesure du déphasage en radians sur le ciel.
Reconstructeur et base de mode	Une base de 100 modes de Karhunen-Loeve a été utilisée. Le reconstructeur correspond donc à cette base de mode
Contrôleur	Le contrôleur est un intégrateur pur. Pour les simulations sans optimisation modale, son gain est constant pour chaque mode et il est calculé par synthèse du régulateur. Avec le système utilisé il est de 0.2818. Lors de l'optimisation modale, les gains du régulateur sont recalculés pour chaque cas.
Etoile et ciel	Nous faisons varier la magnitude de l'étoile entre 0 et 16 pour mesurer les performances du système dans des conditions de flux différentes. Nous considérons le bruit de photons du signal lumineux et le fond de ciel est fixé à une magnitude 18.
Période d'échantillonnage	Elle est fixée à 1.8 [ms]
Simulation	La durée des simulations est de 3 [s]
Remarques	Toutes ces caractéristiques tendent à simuler le plus précisément le télescope MMT du Mont Hoptkins en Arizona. Certaines autres données, telles que le diamètre du miroir principal n'ont pas été indiquées ici. Elles sont fixes, et caractérisent au mieux le MMT.

TAB. 2.2: Caractéristiques générales des simulations.

modes a été diminuée de manière à limiter l'effet du bruit dans la contre-réaction. En réalité ce sont tous les modes qui voient leur bande passante diminuer car ils sont tous bruités. L'optimisation modale permet simplement de trouver le meilleur gain pour chaque mode.

4. La chute de performance intervient tard. Cette constatation est due au problème du spectre de l'atmosphère dont nous discutons en annexe A.2. Comme ces spectres ont trop d'énergie en haute fréquence, le bruit tarde à se faire *ressentir*. Avec les spectres que Conan décrits [8], la chute de performance interviendrait plus tôt, et l'optimisation modale serait plus rapidement nécessaire.

## 2.5 Conclusion du chapitre

Nous avons vu comment mettre en oeuvre l'optimisation par Gendron[16] en simulation. Nous avons obtenu les résultats attendus avec une amélioration du Strehl lors d'observation

Courbe verte	Gains plats à 0.2818 C'est le système original, avec un gain calculé par une synthèse du régulateur.
Courbe rouge	Gains plats à 0.467 Le gain plat a été poussé à la valeur maximale respectant le critère de stabilité (§ 2.3.5).
Courbe bleue	Gains optimisés L'algorithme d'optimisation des gains a été appliqué. Les gains obtenus par cet algorithme dépassant la limite de stabilité sont saturés à 0.467, ce qui correspond au gain plat de la courbe rouge.

TAB. 2.3: Les cas étudiés pour la comparaison des performances.

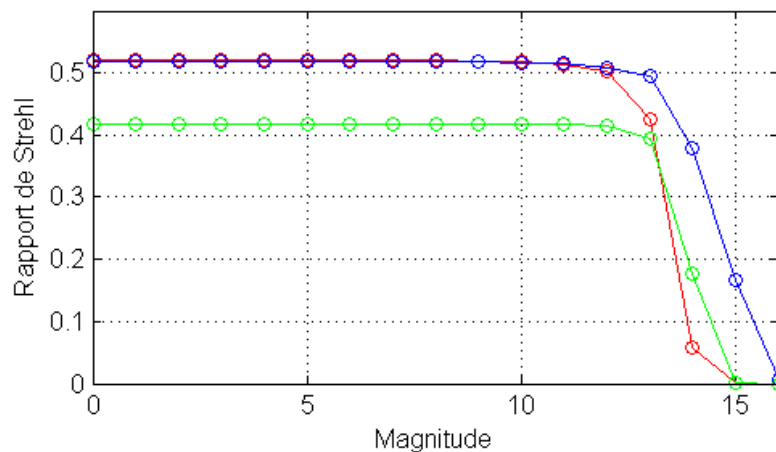


FIG. 2.14: Evolution du rapport de Strehl - utilisé comme critère de performance - en fonction de la magnitude de l'étoile. En bleu, la courbe du Strehl pour un système avec un gain plat, et en rouge pour un système avec des gains modaux optimisés.

d'étoiles faibles où le rapport signal sur bruit devient faible.

Nous avons relevé un problème lors de la mesure avec le CCD sur des signaux fortement bruités par le bruit de lecture. Une perte de sensibilité du capteur diminue sensiblement la bande passante de la contre-réaction. Ceci peut être corrigé par *seuillage*. Cette opération est très facile en simulation où le bruit est très caractérisé, mais bien plus difficile sur un système réel.

Le bruit sur le signal a été mesuré par auto-corrélation. On obtient des résultats très précis lorsque le rapport signal sur bruit est faible car le pic du dirac dû au bruit sur la fonction d'auto-corrélation est mis en évidence. En utilisant Matlab, il faut être sensible aux problèmes de normalisation des signaux obtenus avec les fonctions internes du programme.

L'algorithme développé dans ce chapitre est simple et fonctionnel. En revanche, on peut lui reprocher d'être relativement lent. Dans le chapitre suivant, nous allons étudier comment calculer ces gains optimaux lorsque la boucle est fermée et dans ces conditions il est impor-

tant d'être rapide. Une optimisation temporelle de l'algorithme est donc nécessaire. Comme le principe du calcul des gains optimaux est basé sur des données obtenues en boucle ouverte, nous verrons également comment reproduire ces données lorsque la boucle est fermée.



## Chapitre 3

# Optimisation en boucle fermée

### 3.1 Introduction

Dans le chapitre précédent, nous avons utilisé des données issues de simulations en boucle ouverte. Un gain nul dans la boucle permet une mesure directe de cette grandeur.

Dans la pratique, lorsque la boucle n'est pas fermée, le senseur de front d'ondes sature très rapidement et oscille d'un extrême à l'autre de l'étendue de mesure. Il faut donc adapter l'algorithme pour qu'il puisse utiliser les mesures obtenues après correction du système en boucle fermée.

Cette étape est également un pas de plus vers une régulation en ligne où les gains sont mis à jour régulièrement. Nous allons donc également nous intéresser au temps de calcul de l'algorithme et optimiser les opérations de manière à limiter au maximum la charge de calcul pour le processeur dédié ou non à la tâche de l'optimisation modale. Pour ceci nous allons également déterminer le nombre d'échantillons nécessaires pour que l'optimisation modale soit efficace.

### 3.2 De la boucle ouverte à la boucle fermée

Le passage de la boucle ouverte à la boucle fermée est nécessaire pour deux raisons principales :

1. Le senseur de front d'ondes du télescope n'est pas capable de mesurer des signaux en boucle ouverte car il sature d'un extrême à l'autre de l'étendue de mesure. Ce n'est que lorsqu'une correction est appliquée par la fermeture de la boucle que les mesures restent à l'intérieur de l'étendue de mesure.
2. Pour permettre d'optimiser les gains du régulateur en ligne, et non *a posteriori*, il faut le faire lorsque la boucle est fermée et donc utiliser les données à disposition dans cette configuration.

### 3.3 Optimisation en ligne : mise à jour régulière des gains

Pour effectuer une mise à jour en ligne des gains optimaux, il faut utiliser des données fournies par le calculateur temps réel du système d'optique adaptative lorsque la boucle est fermée. Par exemple, il peut fournir le signal de mesure des pentes et la mesure de la position du miroir et d'après ces données il est possible de déduire la turbulence atmosphérique à

l'entrée du système.

Le calcul des gains optimaux doit être effectué par un processus parallèle au système en boucle fermée et mettre à jour régulièrement les gains modaux, ou bien il peut également être effectué pendant les temps de calcul latents du système. Certaines opérations de calcul de la boucle AO doivent être effectuées à des instants précis, répondant ainsi à des interruptions, mais il est possible qu'il reste assez de temps de calcul latent pour y faire l'optimisation modale.

Un nouveau reconstruteur temps réel est en développement actuellement pour le MMT-AO. Cependant le projet n'est pas suffisamment avancé pour connaître les temps de calculs des différents points-clés de la boucle d'optique adaptative, il n'est donc pas possible de développer le système de calcul des gains modaux pour le moment. Si deux processus différents travaillant en parallèles sont nécessaires, il faudra prévoir la communication entre l'un et l'autre. Le calculateur des gains doit pouvoir acquérir les données des pentes, et éventuellement des données du miroir et être à même de mettre à jour les gains dans le reconstruteur.

De plus, les données *brutes* fournies par le calculateur temps réel nécessitent parfois un pré-traitement avant de pouvoir être utilisées pour un calcul des gains optimaux. Cela est dû à des incidents qui détériorent les mesures sporadiquement. Une analyse de ces incidents permettrait de savoir si ce pré-traitement peut être effectué automatiquement, par filtrage ou détection d'incidents. Ceci sort du cadre de ce travail et nécessite de nouvelles acquisitions de données brutes lors d'un *run* AO.

Nous allons concentrer les efforts de ce travail sur le calcul des gains optimaux avec des données obtenues en boucle fermée et *a posteriori*. De cette manière les gains peuvent être ajustés avant les *runs* en opérant une courte acquisition en boucle fermée.

### 3.4 Les données utilisées

L'objectif est de lire un fichier contenant les pentes mesurées par le WFS. Ces données sont enregistrées dans le format FITS (FITS file). Il est également nécessaire de connaître le reconstruteur utilisé avec le masque des sous-ouvertures illuminées correspondant au reconstruteur. Ces deux autres données sont également utilisables au format FITS.

Pour connaître le signal de la turbulence à l'entrée de la boucle, deux méthodes sont envisageables :

1. Utiliser le signal des pentes et la mesure de la position du miroir. Le signal des pentes correspond à l'erreur résiduelle et la position du miroir à la correction appliquée. La somme de ces deux quantités, dans l'espace des modes, correspond à la turbulence perturbant le système reconstruite sur les modes visibles du système.
2. Au lieu d'utiliser la mesure de la position du miroir, il est possible de la déterminer en connaissant les fonctions de transfert du système. Avec une modélisation mathématique des blocs séparant la mesure des pentes et le miroir déformable, on peut calculer la position de ce dernier.

La première méthode a l'avantage de ne pas faire d'approximation par méconnaissance du système. On mesure directement la position réelle du miroir. Cependant les données ainsi obtenues sont entachées du bruit de mesure. Dans le second cas, les données utilisées sont calculées *aveuglément*, mais sans ajouter de bruit de lecture. De plus, cette méthode a

l'avantage de ne nécessiter qu'une mesure sur le système, et demande donc moins de communication avec le calculateur temps réel. Lié à cela, le DDS (diagnostic data storage) du MMT-AO est limité dans le nombre de données qu'il peut enregistrer. En ne demandant au système de n'enregistrer que les pentes, on augmente le temps d'acquisition possible par le DDS.[3]

### 3.4.1 Modélisation du système réel

Le système réel est modélisé légèrement différemment que le système simulé[4]. Nous allons nous baser sur la représentation de la fig. 3.1 dont les fonctions de transfert sont regroupées dans le tableau 3.1. Le tableau 3.2 résume les paramètres du télescope MMT.

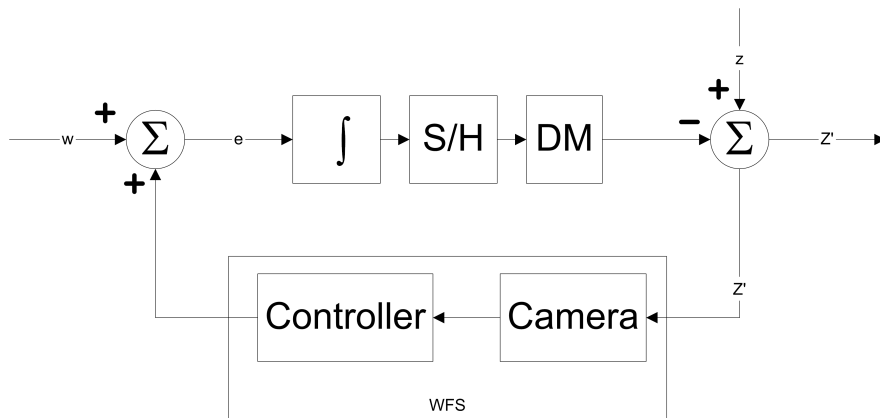


FIG. 3.1: Représentation des blocs de la boucle d'optique adaptative du MMT.

#### Note sur les retards purs

Nous allons modéliser ce système de manière analogique (en  $s$ ). Pour le modèle de la simulation, nous avons choisi une solution numérique (en  $z$ ) car le système est simulé sur Matlab qui est un environnement numérique. Pour caractériser le système réel, le domaine analogique est plus approprié.

Ce système est un mélange analogique-numérique, il faudrait donc modéliser chaque composant dans le domaine qui convient, cependant pour être utilisable dans un environnement comme Matlab, une certaine uniformité est requise. Pour les composants qui sont purement numériques, tel que l'intégrateur, nous allons donc utiliser la conversion dans le domaine analogique.

Un problème plus important se pose concernant les retards purs. Matlab n'est pas capable de fermer la boucle d'un système analogique comportant un retard pur modélisé comme :

$$e^{-sT} \quad (3.1)$$

Les retards purs doivent donc être approximés, et nous le faisons par Pade. Nous avons évalué la précision de cette approximation à la fréquence critique (position du pic sur les fonctions de transfert). Cette fréquence critique a été estimée à environ 60 [Hz]. Une approximation d'ordre 1 donne 15% d'erreur à cette fréquence. Nous avons alors choisi une

Bloc	Modèle	Description
Senseur de front d'ondes (WFS)	$g' \cdot \frac{1-e^{-s \cdot m \cdot T}}{s \cdot m \cdot T} \cdot e^{-s \cdot T}$	Le senseur de front d'ondes intègre le flux pendant $m \cdot T$ [s]. Un retard pur d'une période d'échantillonnage $T$ est ajouté pour terminer le calcul des pentes. Le paramètre $m$ permet d'ajuster le temps d'intégration en fonction de la période d'échantillonnage et le gain $g'$ est un gain de calibration du senseur de front d'ondes dont nous allons parler plus tard.
Intégrateur (Integ)	$g_i \cdot \frac{1}{s}$	La fonction de transfert de l'intégrateur avec son gain $g_i$ .
Convertisseur D/A (Sample & Hold S/H)	$\frac{1-e^{s \cdot T}}{s \cdot T}$	La conversion digitale analogique retarde le système d'environ une demi période d'échantillonnage. Une fois la valeur convertie, cette dernière est maintenue jusqu'à la conversion suivante.
Miroir déformable (DM)	$e^{-s \cdot T_m(i)}$	En première approximation, la fonction de transfert du miroir est modélisée par un retard pur de $T_m(i)$ [s] dont $i$ est le numéro du mode.

TAB. 3.1: Les fonctions de transfert des blocs du MMT. Les fonctions de transfert avec les approximations des retards purs par Pade sont données en annexe dans le tableau A.1

approximation d'ordre 2 avec seulement 5% d'erreur.

On écrit donc :

$$e^{-sT} \approx \frac{1 - \frac{T}{2} \cdot s + \frac{T^2}{16} \cdot s^2}{1 + \frac{T}{2} \cdot s + \frac{T^2}{16} \cdot s^2} \quad (3.2)$$

### Comportement modal du miroir déformable

Il est à signaler tout d'abord que le miroir n'a pas la même réponse pour tous les modes. Il est possible de s'en convaincre aisément en considérant la dynamique mécanique de ce dernier. Par ailleurs, les mesures le confirment.

En première approximation, nous allons modéliser le miroir déformable comme un retard pur. Cette modélisation est effectuée par identification.

Les mesures effectuées sur le miroir sont les réponses indicelles des 80 premiers modes du miroir. Ce sont les modes mécaniques propres à ce dernier, qui sont une combinaison linéaire des modes optiques utilisés dans la boucle de régulation. Comme première approximation,

Paramètre	Variable	Valeur
Période d'échantillonnage	$T$	1.8 [ms]
Facteur de temps d'intégration du WFS	$m$	$\geq 1 \in \mathbb{N}$
Gain de calibration du WFS	$g'$	Dépend du <i>seeing</i> , en général inférieur à 0.5.
Retard pur du miroir déformable	$T_m(i)$	Les valeurs modales sont approximées plus loin dans ce chapitre.

TAB. 3.2: Les paramètres du système d'optique adaptative monté sur le télescope MMT.

nous allons modéliser ces fonctions de transfert comme des retards purs. Il est évident que cette approximation est largement éronnée, mais elle est jugée satisfaisante dans le spectre de fréquences utiles.

Ces mesures sont appelées des *réponses statiques* car elles sont faites en mesurant le mouvement d'un mode mécanique en appliquant une commande sur ce mode unique. Cependant, malgré cette excitation, les autres modes répondent aussi. Ceci est dû au fait que tous les modes du miroir sont en fait déplacés en même temps pour obtenir le mode désiré, et car tous les actionneurs du miroir sont liés mécaniquement par la membrane. Nous supposons cependant que l'effet de l'excitation des autres modes est négligeable. Deux idées nous mènent à cela. Premièrement, cette réponse non désirée est très rapide, beaucoup plus rapide que la période d'intégration du senseur de front d'ondes. Et deuxièmement, la réponse de ces modes est nulle en très basses fréquences. Elle augmente pour avoir un effet sensible en hautes fréquences, dans des fréquences plus élevées que celles où le système est capable d'appliquer une correction.

### Ajustement d'un retard pur sur les réponses indicielles des modes optiques

Ces réponses indicielles sont données sous la forme de mesures numériques de la position du miroir commandé par un saut unité, mode par mode. La méthode utilisée pour ajuster un retard pur sur ces signaux est de convoluer ces mesures avec un peigne de dirac modulé par une fonction sinusoïdale et échantillonné à la même fréquence que les mesures des réponses indicielles. Après convolution, on obtient une sinusoïde déphasée. Ce déphasage est mesurable, en ne tenant pas compte des valeurs transitoires en début et fin de convolution. Il correspond au retard pur cherché.

Le fait d'utiliser des fonctions de transfert différentes pour chaque mode charge considérablement l'algorithme. Bien que les fonctions de transfert ne soient calculées qu'une seule fois puis enregistrées dans des fichiers, lesdits fichiers deviennent volumineux, longs à charger et demandent plus de calculs de la part du processeur. Pour toutes ces raisons, l'utilisation de fonctions de transfert modales est une option dans l'algorithme, qui ne doit être utilisée que si une grande précision est nécessaire. Les retards purs modélisés pour chaque mode varient de 0.82 [ms] à 1,12 [ms]. La variation du gain optimal trouvé est négligeable dans ces conditions.

### La calibration du senseur de front d'ondes

Issu de la méthode de mesure des pentes, un défaut de calibration intervient dans le senseur de front d'ondes[3]. En effet, la taille du spot éclairant les quadrants du capteur CCD dépend du *seeing*. La méthode de calcul des pentes dite *quad-cell* perd de la sensibilité avec la dégradation du ciel. Comme le senseur de front d'ondes du MMT a été calibré une fois pour toute avec un *seeing* à la limite de diffraction, lors des mesures sur le ciel il faut tenir compte d'un gain de calibration qui décrit cette perte de sensibilité. Ce gain est appelé  $g_l$ . Par ailleurs, il est relativement difficile de déterminer sa valeur. Une des méthodes utilisée à cette fin est d'injecter un signal de pente modulé à une certaine fréquence juste après les valeurs des pentes rendues par le senseur de front d'ondes[3, 5]. Supposant que tous les blocs du système sont connus, mis à part  $g_l$ , on peut déterminer ce dernier à partir des mesures du signal modulé à travers la boucle.

### 3.4.2 La turbulence en entrée du système

Les mesures accessibles sont les pentes en boucle fermée. Pour calculer les gains optimaux, nous avons besoin de connaître la turbulence à l'entrée du système. Pour ceci, nous allons déterminer la turbulence *modale*.

La reconstruction des pentes nous donne la mesure de l'erreur résiduelle modale limitée aux modes reconstruits après correction. Négligeons également le repliement spatial des modes à fréquences spatiales élevées sur le senseur de front d'ondes. Nous connaissons les fonctions de transfert qui mènent ces erreurs modales aux commandes modales du miroir. Prenant en compte encore la dynamique du miroir, nous pouvons évaluer la position modale du miroir. Les deux grandeurs ainsi obtenues, l'erreur résiduelle modale et la position modale du miroir, toutes deux additionnées, donnent une approximation de la perturbation modale excitant le système limitée aux modes reconstruits.

Pour cela, il faut connaître le gain de l'intégrateur  $g_i$  qui a été appliqué au système lors de l'acquisition des mesures en boucle fermée. Dans ce cas nous avons supposé un gain constant en fonction des modes.

## 3.5 Le nombre d'échantillons

Une des influences du nombre d'échantillons utilisés dans le calcul des gains optimaux se retrouve dans la résolution fréquentielle de la FFT et des fonctions de transfert ainsi que sur la résolution de la fonction d'auto-corrélation. Une autre influence est la population statistique du signal spatial d'entrée. Nous savons que l'écran de phases a une certaine distribution spatiale définie par son échelle interne et son échelle externe.

Il est difficile de trouver une expression mathématique exacte permettant de déterminer le nombre d'échantillons nécessaires. Considérant l'effet statistique de l'atmosphère, avec un vent de 25 [m/s] et une échelle externe de 100 [m], on peut imaginer un temps d'acquisition de 3 à 5 [s]. Comme la fréquence d'échantillonnage du système est de 1.8 [ms], cela conduit à une résolution fréquentielle meilleure que 1 [Hz].

La figure 3.2 montre des mesures obtenues par simulations pour connaître le nombre d'échantillons nécessaires. La courbe représente le rapport de Strehl obtenu lors d'une simulation utilisant un contrôleur dont les gains modaux ont été calculés à partir d'une acquisition en

boucle ouverte du temps indiqué en abscisse du graphique. On remarque qu'une acquisition de 2.5 [ms] est suffisante dans ce cas pour obtenir les meilleures performances. Pour toutes ces simulations, c'est le même écran de phases qui a été utilisé, donc les informations que l'on peut en tirer ne sont pas totalement générales. Pour les simulations nous allons toujours utiliser un temps d'acquisition de 3 [s].

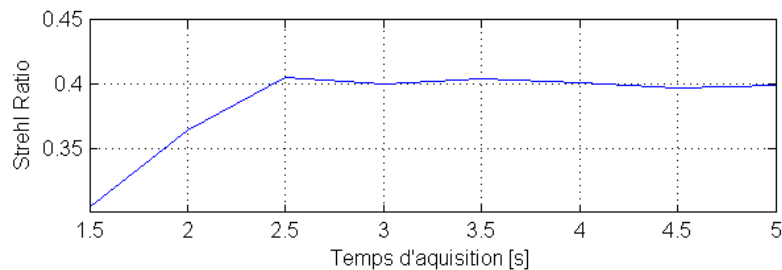


FIG. 3.2: Le rapport de Strehl obtenu lors de simulation dont les gains modaux du contrôleur utilisé ont été calculés à partir d'une acquisition de données en boucle ouverte du temps indiqué en abscisse.

## 3.6 Le temps de calcul

Pour intégrer le calcul des gains optimaux dans le processus d'optique adaptative, il est bon d'optimiser le temps de calcul. Le but en est double : diminuer la charge de calcul dédiée à cette tâche pour l'unité de calcul qui va s'en charger, et pouvoir mettre à jour les gains optimaux le plus souvent possible. Bien entendu, cette optimisation n'est pas envisageable en *temps réel*, c'est-à-dire pour chaque coup d'horloge. Cependant, le but est de mettre à jour les gains *en-ligne*, pendant que la boucle est fermée.

L'algorithme présenté au chapitre précédent recherche les gains en calculant la variance de l'erreur de phase résiduelle pour chaque gain considéré. Pour éviter ces calculs inutiles, nous allons utiliser un processus de recherche du minimum par *dichotomie*<sup>1</sup>.

### 3.6.1 Recherche du minimum par dichotomie

La dichotomie est un processus itératif de recherche où à chaque étape l'espace de recherche est restreint à l'une des deux parties. Pour cela, il est nécessaire de connaître un test relativement simple permettant de déterminer laquelle des deux parties contient la solution cherchée. Il faut également définir un critère de sortie qui indique que le résultat trouvé est satisfaisant. Nous allons procéder ainsi (voir fig. 3.3) :

1. Définir les bornes de l'espace de recherche des gains (points 1 et 2 de la fig. 3.3).
2. Définir un critère de sortie qui sera la précision désirée dans l'espace des gains.
3. Couper l'espace des gains en deux à chaque itération (on obtient le point 3 de la fig. 3.3).
4. Déterminer si la pente est croissante ou décroissante à ce point.

<sup>1</sup>Le terme dichotomie est tiré du grec : "coupé en deux"

5. Rejeter l'espace des gains compris entre le gain médiant et les gains situés où la courbe de la variance de phase résiduelle est croissante (partie droite de la courbe sur la fig. 3.3). Ainsi le point numéro 3 devient le numéro 2 et un nouvel espace de recherche est défini.
6. Recommencer au point 3 jusqu'à ce que les bornes extérieures de l'espace de recherche des gains soient égales au double du critère de précision désiré, correspondant au critère de sortie du processus de recherche par dichotomie.

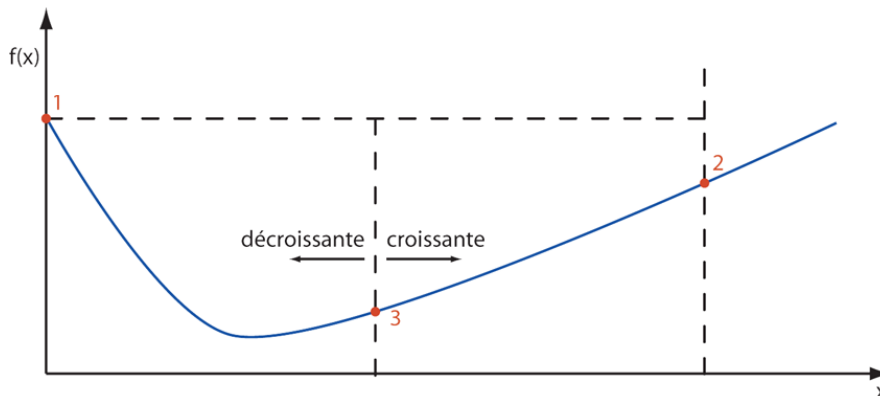


FIG. 3.3: Représentation de la recherche d'un minimum d'une fonction par dichotomie.

### Critères et paramètres utilisés

Les bornes de l'espace de recherche ainsi que la précision désirée sur le gain cherché sont des paramètres modifiables au début de l'algorithme. Généralement, on définit le gain minimum comme étant un gain nul et le gain maximal à la limite de stabilité du système. Ceci permet de balayer largement l'espace des gains tout en ne faisant pas de calculs inutiles. Cependant, il est envisageable de restreindre cet espace de recherche pour des situations particulières. Pour cela il est nécessaire de très bien connaître le système. Un exemple est de ne pas chercher de minimum en-dessous d'une valeur comme 0.2 pour une étoile très brillante.

Une précision de l'ordre d'un pour cent sur les gains recherchés est un critère satisfaisant. On peut facilement demander une précision d'un pour mille sans effectuer beaucoup de calculs supplémentaires lors de la recherche du gain optimal, mais c'est lors du calcul des fonctions de transferts que cela va prendre plus de temps. Cela n'a pas beaucoup d'importance car le calcul des fonctions de transferts ne doit être fait qu'une fois, donc il n'influence pas le comportement de l'algorithme lorsqu'il est en ligne.

L'espace des gains est coupé en deux parties égales à chaque itération.

Pour déterminer si la pente est croissante ou décroissante c'est en général la dérivée de la fonction qui est utilisée. Pour éviter le calcul de la dérivée de la fonction exprimant la variance de l'erreur de phase résiduelle, nous calculons deux fois cette variance. Une première fois pour le point considéré, et une seconde fois pour le gain situé un index en aval du vecteur de gain. Nous testons ensuite si la première variance est inférieure ou supérieure à la seconde.



### Gain de temps avec la dichotomie

Pour comparer le temps de calcul nécessaire par dichotomie par rapport à la première méthode où tous les gains sont calculés, nous avons mesuré le temps de calcul nécessaire sur Matlab avec le même ordinateur. Le temps de calcul mesuré est celui qui est nécessaire pour mesurer le bruit sur les pentes par auto-corrélation, effectuer la FFT du signal modal et la recherche du gain optimal. Les autres opérations comme le calcul des coefficients de propagation du bruit, l'axe de fréquence ou les fonctions de transfert ne sont pas prises en compte car elles ne sont nécessaires que pour initialiser l'algorithme.

Le calcul des gains optimaux a été effectué dans les deux cas sur le même fichier de simulation qui dure 3 [s] avec une étoile de magnitude 14 et un reconstruteur de 100 modes en Karuhen-Loeve. L'algorithme de base a pris 4.6 [s], alors que l'algorithme par dichotomie a eu besoin 1.1 [s] pour calculer les gains optimaux.

## 3.7 Les gains optimaux depuis les données du MMT

### 3.7.1 Le bruit ajouté au signal

Un seul signal a été utilisé pour tester l'algorithme d'optimisation modale. Ce signal contient les pentes mesurées par le senseur de front d'ondes lors de l'observation d'une étoile de magnitude 5.4 en boucle fermée. Ce dernier est considéré comme *non-bruité* car le rapport signal sur bruit est suffisant. Connaissant les coefficients de propagation du bruit dans le système, tirés du reconstruteur (éq. 2.13 et fig. 3.4), nous allons artificiellement et progressivement ajouter du bruit sur ce signal pour observer l'évolution des gains optimaux.

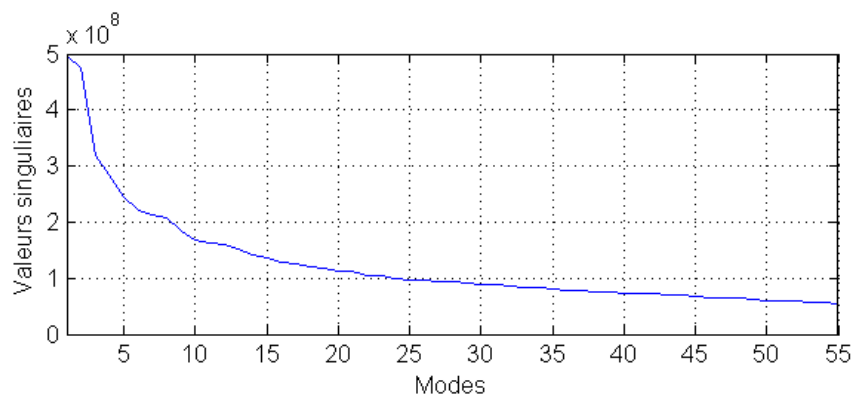


FIG. 3.4: Valeurs singulières de chaque mode du reconstruteur.

La figure 3.5 illustre le bruit qui est ajouté au signal. La courbe bleue est le spectre de la turbulence reconstituée en mesurant une étoile de magnitude 5.4. En rouge, on voit le spectre du même signal où le bruit a été ajouté. Ce dernier est ajouté sur le signal temporel modal comme un bruit blanc de moyenne nul et de variance proportionnelle aux coefficients de propagation du bruit.

Ce bruit n'est pas calibré comme une fonction de la magnitude de l'étoile guide. Il est ajouté de manière arbitraire pour observer des conditions de plus en plus sévères.

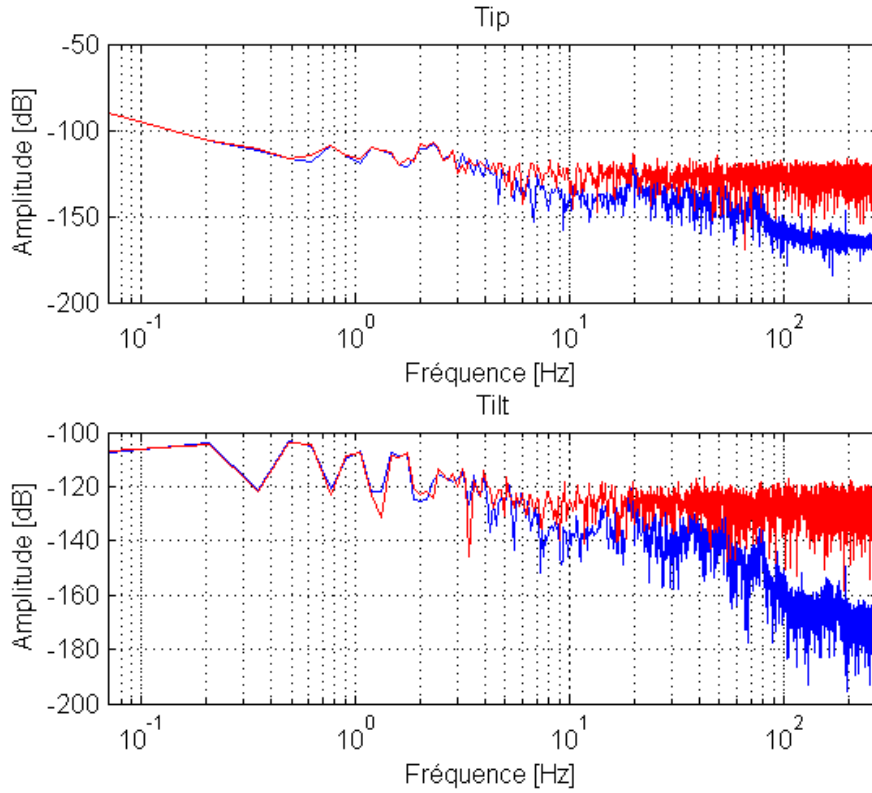


FIG. 3.5: Illustration de l'addition de bruit sur le signal modal. En bleu : le signal non bruité ; en rouge : le signal bruité.

### Note sur les coefficients de propagation du bruit

Il est envisageable de ne pas considérer les coefficients de propagation du bruit dans le système. Ceux-ci définissent la propagation du bruit entre l'espace des pentes et l'espace des commandes des actuateurs du miroir déformable. La matrice de reconstruction totale  $R$  effectue le passage entre ces deux espaces qui ne sont pas normalisés au même niveau. En regard de l'équation 1.9, écrivons :

$$\vec{c} = R \cdot \vec{s} = A \cdot \vec{p} \cdot D^+ \cdot \vec{s} \quad (3.3)$$

Le vecteur de pente  $\vec{s}$  est passé dans l'espace des modes par la matrice  $D^+$ . On obtient l'espace des pentes modales. Le vecteur de propagation  $\vec{p}$ , équivalent au vecteur de propagation du bruit amplifie le signal de pentes modales pour obtenir le modal signal de commande du miroir. La matrice  $A$  convertit ce signal modal dans l'espace des actuateurs du miroir déformable.

Pour effectuer l'optimisation modale, il suffit de rester dans l'espace des pentes modales et ainsi n'utiliser que la matrice  $D^+$  dont les valeurs singulières sont unitaires. Comme l'on traite chaque mode pour lui-même, le fait de ne pas considérer les coefficients de propagation du bruit n'affecte le résultat de l'intégrale (éq. 2.12) que d'une amplification proportionnelle à ces derniers. La position du minimum de la variance reste donc inchangée.

### 3.7.2 Les résultats et leur analyse

On voit sur la figure 3.6 l'évolution des gains optimaux dans des conditions de plus en plus bruitées. Comme décrit plus haut, ce bruit n'est pas calibré en fonction de la magnitude de l'étoile. Cependant, il est possible d'évaluer le nombre de magnitudes balayées entre plusieurs niveaux de bruit. Ne prenant en compte que le bruit de photons, la variance de ce dernier est équivalent au nombre de photons incidents. Sachant qu'un facteur 2.5 sépare le nombre de photons correspondant à une différence d'une unité de magnitude, il est possible d'évaluer le nombre de magnitudes séparant deux niveaux de bruit. Le tableau 3.3 donne une idée de ces grandeurs correspondantes à la figure 3.6.

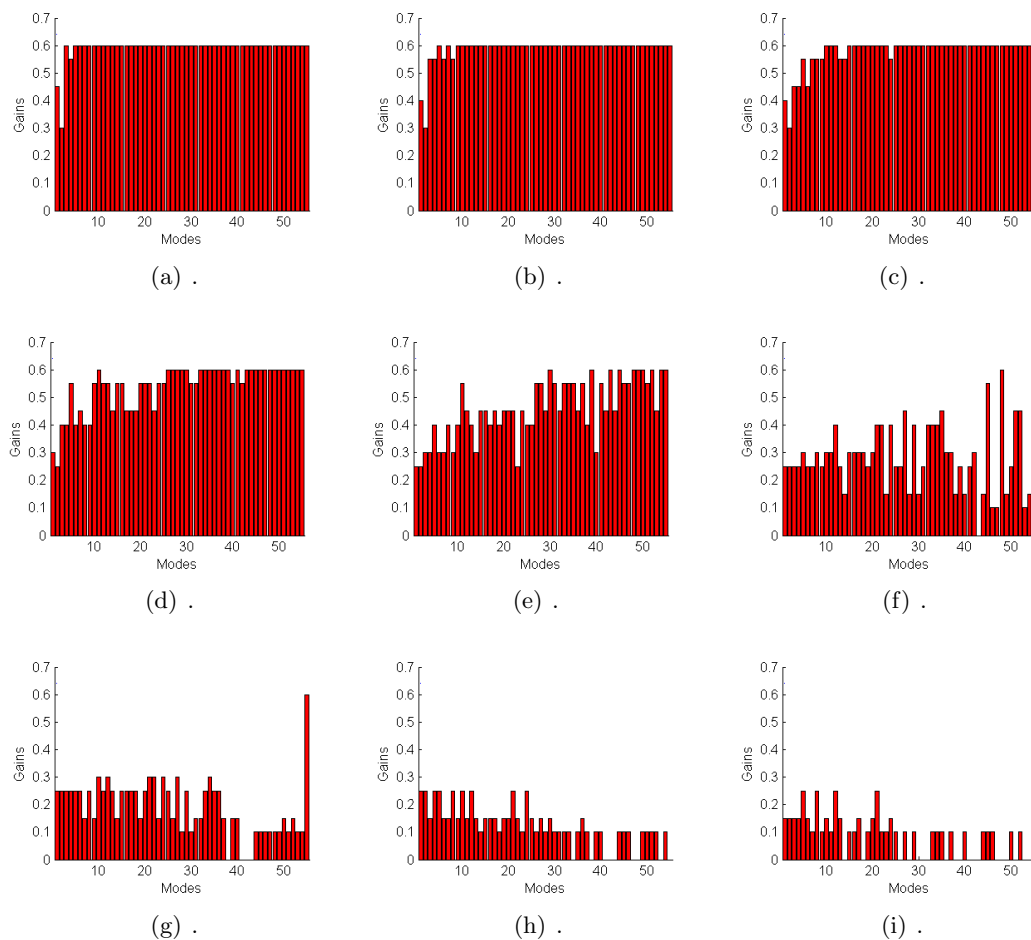


FIG. 3.6: Les gains optimaux calculés à partir de données prises sur le ciel. En (a) : le signal non-bruité (magnitude 5.4). De (b) à (i) : du bruit a été ajouté artificiellement sur le signal de base. Il a été ajouté linéairement entre chaque figure. Le tableau 3.3 donne une idée des magnitudes balayées correspondantes au niveau de bruit entre les figures.

#### Qu'est-ce qui fixe la bande passante ?

La bande passante est principalement fixée par trois paramètres :

1. Le spectre temporel du mode .

De	A	Magnitudes balayées
b)	d)	1.2
b)	g)	2.4
b)	i)	3.2
d)	i)	1

TAB. 3.3: Evaluation du nombre de magnitudes balayées entre les sous-figures de la figure 3.6

2. Le dépassement des fonctions de transfert (de réjection et de bruit) dépendant du gain.
3. Le niveau de bruit sur le mode .

Ce sont les paramètres à analyser pour comprendre la forme des gains optimaux obtenus. De plus, la comparaison de deux modes ne dépend plus que des paramètres 1 et 3 car les fonctions de transfert sont modélisées de la même manière pour le système.

Des données intéressantes à analyser sont celles issues d'une étoile brillante, dont le bruit sur le signal est négligeable. Dans ce cas, l'équation 2.12 devient :

$$\langle z_i^2 \rangle = \int H_{cor}(f, g_i) \cdot \|\hat{z}_i(f)\|^2 df \quad (3.4)$$

où la comparaison de deux modes ne se fait plus que par le premier paramètre cité plus haut ; le spectre respectif de chacun des modes à analyser. Il faut cependant rester conscient que cette méthode d'optimisation modale est destinée à améliorer les performances lors d'observation d'étoile faible, et donc bruitée. L'équation 2.12 est par ailleurs écrite en supposant que le repliement spectral est négligeable en regard du bruit dans le système. Ceci ne reste pas forcément vrai lors d'un bon rapport signal sur bruit et n'a pas été étudié dans ce travail.

Pour analyser la forme des modes avec des signaux modaux bruités en entrée, il faut également considérer le niveau de bruit du signal. Il faut savoir que le reconstruteur utilisé dans la boucle d'optique adaptative du MMT est dimensionné de manière à ordonner les modes selon la valeur singulière de ce dernier dans le reconstruteur. Les coefficients de propagation du bruit suivent donc une loi monotone décroissante. Les valeurs singulières du reconstruteur sont représentées sur la figure 3.4.

Comme le bruit sur chaque mode est proportionnel au coefficient de propagation du bruit correspondant à ce dernier, le bruit sera également monotoniquement décroissant. Ce seul paramètre aurait tendance à engendrer une augmentation du gain en fonction du mode.

### Les pics de puissance

Sur la figure 3.7, nous observons la distribution fréquentielle de la puissance du signal du mode #1. Nous y trouvons deux pics :

1. Un pic à 20 [Hz] qui est dû à des vibrations mécaniques de la structure soutenant le miroir déformable[3]. Ces oscillations produisent du *tip* et du *tilt* essentiellement. On retrouve donc ces pics sur ces modes en particulier. Une solution envisagée actuellement pour diminuer ce pic est d'utiliser une commande en boucle ouverte. Des

accéléromètres mesurent les vibrations de la structure, et une correction en boucle ouverte peut être appliquée sur les modes *tip* et *tilt*.

- Un pic à 49 [Hz] dont la raison n'est pas connue actuellement. Ce pic apparaît pour une très faible bande de fréquences ( $< 0.5$  [Hz]). Il est présent principalement sur les modes #1, #2 et #4. Ces considérations laissent supposer qu'il n'est pas issu de vibrations mécaniques. Nous allons le considérer comme un bruit s'ajoutant au signal et ne correspondant pas à une réelle turbulence à corriger. Il est vraisemblablement de source électrique.

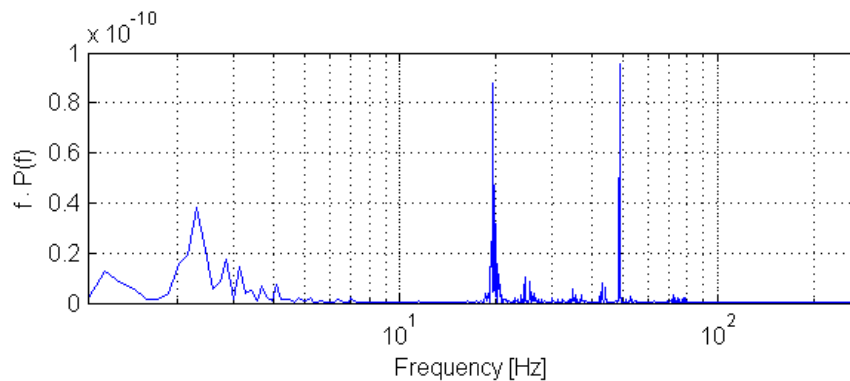


FIG. 3.7: Distribution fréquentielle du mode #1.

D'autres modes possèdent parfois d'autres pics. On en trouve à 30 [Hz] par exemple. Comme leur présence n'est pas systématique, il est difficile d'en déduire l'origine. Le manque de données ne permet pas d'étudier cela suffisamment profondément pour en tirer des conclusions fiables. Il est évident que ces pics ne proviennent pas de la turbulence atmosphérique. Ce qu'il faut déterminer, ce sont leurs conséquences sur la qualité de l'image finale. Par exemple, les oscillations de la structure à 20 [Hz] dégradent l'image finale, alors que vraisemblablement celles à 50 [Hz] proviennent d'un bruit qui ne détériore que les mesures.

### La forme des gains optimaux

La figure 3.8 compare la distribution fréquentielle des modes #3 et #30 dans des conditions de bruit similaires à la figure 3.6(d). On remarque que pour les fréquences 10-20 [Hz], le mode #30 est celui qui contient le plus de puissance. Cette plage de fréquence se situe avant le dépassement de la fonction de transfert de réjection, il est donc positif d'augmenter le gain du système pour ce mode. Cependant, ce mode contient aussi plus de puissance dans la zone de dépassement des fonctions de transfert et cela à tendance à diminuer le gain optimal. L'algorithme cherche le meilleur compromis considérant le bruit et le signal. En l'occurrence le gain pour le mode #30 est plus élevé que pour le mode #3.

Nous comparons également deux set de données acquis dans des conditions différentes (fig. 3.9). Le mode #3 a été choisi pour cette comparaison car il ne contient pas de pic à des fréquences particulières. Le spectre bleu est celui de l'étoile de magnitude 5.4 utilisée jusqu'ici dans ce travail. Le spectre de la courbe rouge est obtenu à partir de la même étoile, mais un filtre optique a été ajouté pour réduire le nombre de photons incidents. De cette manière on peut *simuler* une étoile de magnitude plus élevée. Ces mesures représentent une étoile de magnitude 8.75.

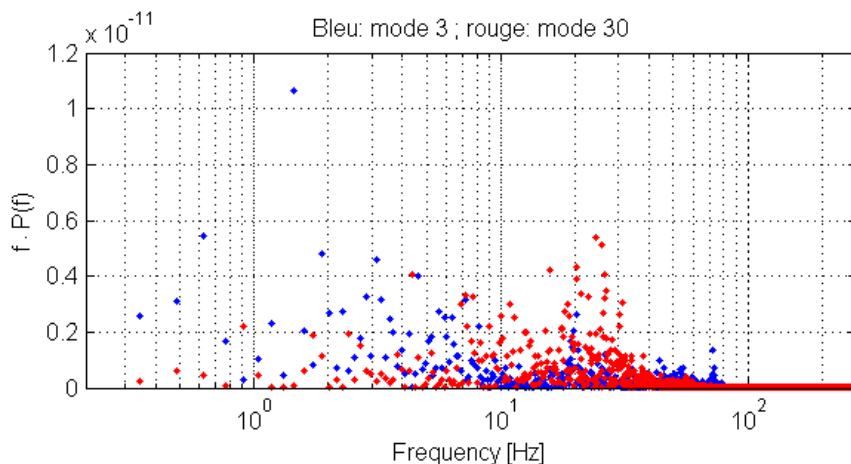


FIG. 3.8: Distribution fréquentielle modale. En bleu : mode #3 ; en rouge : mode #30.

Dans la zone de 30 à 80 [Hz], on voit bien que le spectre *non-filtré* contient plus de puissance. Une étude sur d'autres données montre que cette puissance supplémentaire ne dépend ni du nombre de photons incidents, ni du mode étudié (du moins pas directement). Vraisemblablement, cette puissance n'est pas due à la turbulence, mais vient d'un bruit ajouté au signal lors de la mesure.

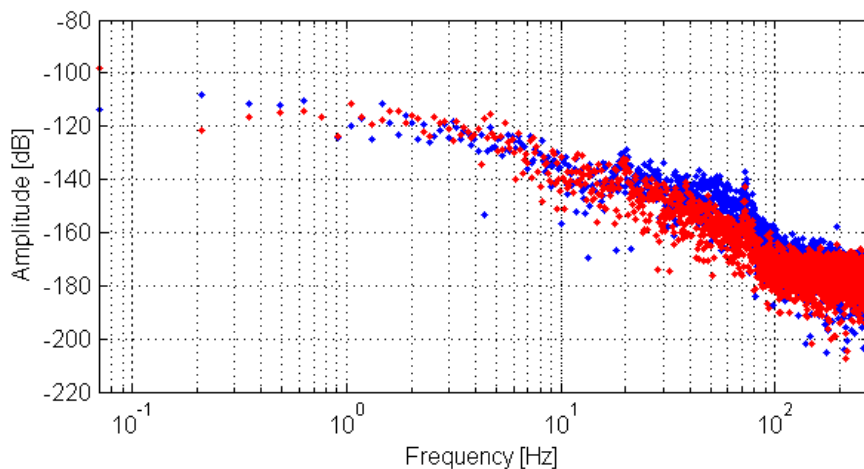


FIG. 3.9: Comparaison de deux différents set de données pour le même mode (#3). En bleu : le set de données utilisées jusqu'ici, magnitude 5.4 ; en rouge : données acquises sur la même étoile mais avec un filtre optique permettant de diminuer le nombre de photons incidents et ainsi *simuler* une étoile de magnitude plus élevée, ici correspondante à 8.75.

Regardons maintenant l'allure des gains optimaux pour le second set de données (fig. 3.10). Les conditions sont les mêmes que pour la figure 3.6, elles sont reportées dans le tableau 3.3.

La principale différence de ce set de données, mis à part le rapport signal sur bruit plus faible à l'origine (puisque du bruit artificiel a ensuite été ajouté), réside dans le fait que dans la zone du spectre de 30 [Hz] à 80 [Hz], le signal contient moins d'énergie. Comme l'énergie

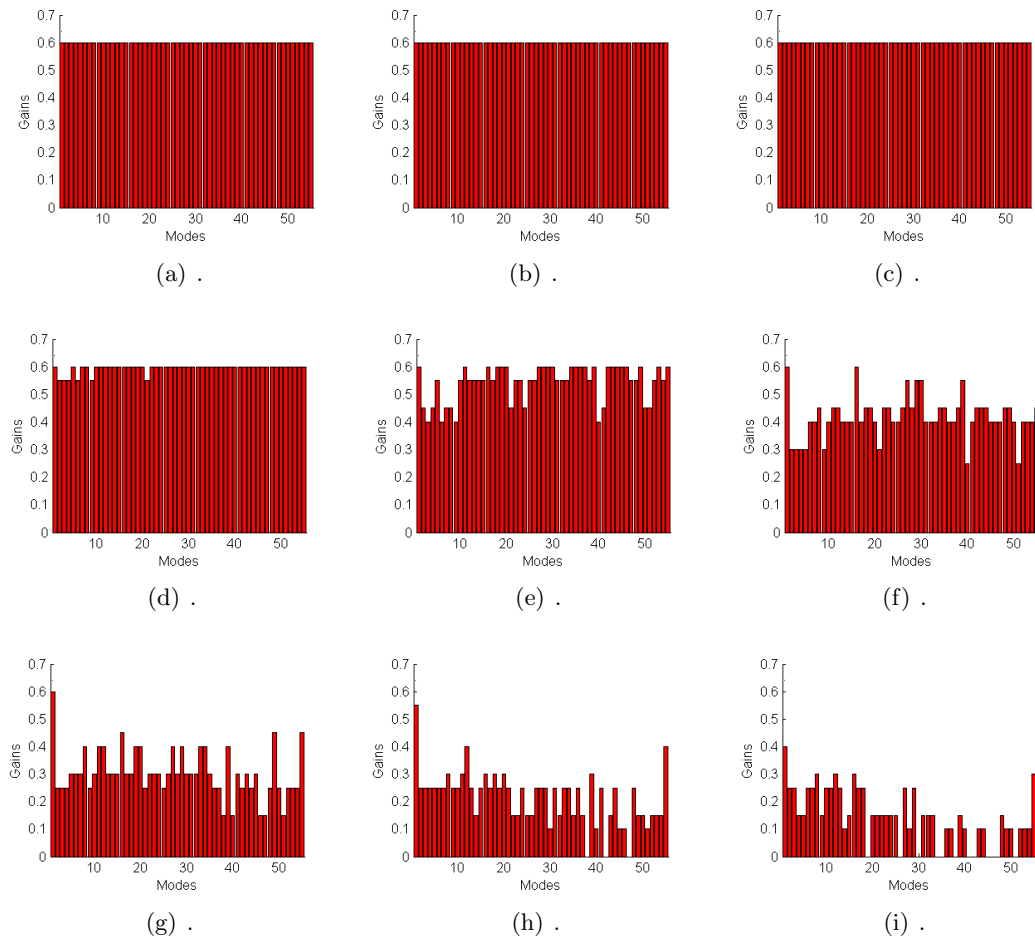


FIG. 3.10: Les gains optimaux calculés à partir de données prises sur le ciel avec un filtre optique monté sur le télescope pour *simuler* une étoile moins brillante. La magnitude correspondante de l'étoile est 8.75. En (a) : le signal non-bruité (magnitude 5.4). De (b) à (i) : du bruit a été ajouté artificiellement sur le signal de base. Il a été ajouté linéairement entre chaque figure (même niveau que sur la fig 3.10). Le tableau 3.3 donne une idée des magnitudes balayées correspondantes au niveau de bruit entre les figures.

manquante est considérée comme un bruit qui s'ajoute au signal dans certaines conditions, ce deuxième set de données peut être qualifié de *meilleure qualité* que le précédent.

### Comportement du dernier mode

Les figures 3.6 et 3.10 montrent que le dernier mode a un comportement très particulier. Le gain optimal trouvé pour ce mode a tendance à rester plus élevé que les autres jusqu'à ce qu'il chute à zéro pour un niveau de bruit trop élevé. Observons la figure 3.11 qui permet de comparer l'allure du gain pour le dernier mode, une fois dans les conditions de la figure 3.6(g) et une fois dans les conditions de 3.6(h) respectivement, les sous-figures 3.11(a) et 3.11(b) de la figure 3.11. Un niveau de bruit égal a été artificiellement ajouté au signal, les coefficients de propagation du bruit ne sont pas pris en compte (§ 3.7.1). Attention au changement d'échelle verticale entre les deux sous-figures. Ce que l'on remarque, c'est que

le dernier mode (#55) possède plus d'énergie entre 8 [Hz] et 30 [Hz] dans les deux cas. Mais de manière relative au niveau de bruit, on constate que cette énergie supplémentaire perd tout son effet lorsque le niveau de bruit augmente.

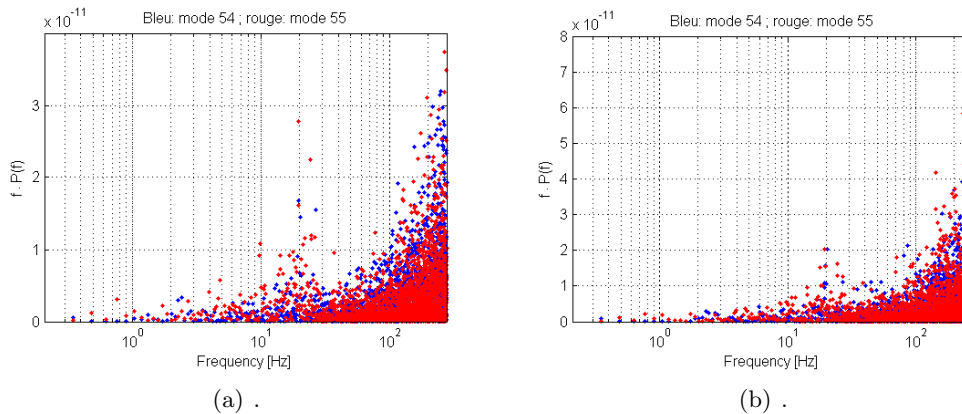


FIG. 3.11: Comparaison de la distribution fréquentielle des modes #54 (bleu) et #55 (rouge). En (a), les conditions sont les mêmes que pour la fig. 3.6(g) et en (b), les mêmes que pour la figure 3.6(h).

### Comportement du premier mode

Dans le second set de données, on remarque que le premier mode a également un comportement particulier. Avec la base de modes utilisée, les modes #1 et #2 sont proches de modes *tip* et *tilt*. Par conséquent, leur comportement est usuellement très similaire, or dans notre cas on remarque une différence flagrante. Comparons la distribution fréquentielle entre les modes #1 et #2 pour le second set de données (fig 3.12). Les signaux utilisés sont les mêmes que pour la figure 3.10(g). Le mode #1 possède le pic à 20 [Hz] dont nous avons parlé (§ 3.7.2) et qui correspond aux oscillations de la structure portant le miroir déformable. En l'occurrence, ces vibrations ont largement affecté le mode #1, mais pas le #2. Comme cette fréquence est située avant les fréquences de dépassement des fonctions de transfert, le système peut les corriger. L'algorithme d'optimisation cherche le gain le plus élevé possible pour corriger cette composante.

### La nécessité d'un pré-traitement

Après avoir étudié l'allure des gains optimaux ainsi que les causes de certaines particularités, on en conclut qu'un pré-traitement des données est nécessaire.

Idéalement, ce traitement doit pouvoir être effectué de manière automatique au sein de l'algorithme d'optimisation modale. Il ne peut être basé que sur des raisonnements mathématiques, mais doit se baser principalement sur des observations précédentes et des connaissances approfondies des caractéristiques du système.

D'autres auteurs ont tiré ces conclusions également. Gendron [17], par exemple, parle de détection d'incidents et de pré-traitement nécessaire pour les données utilisées dans *Come-On*



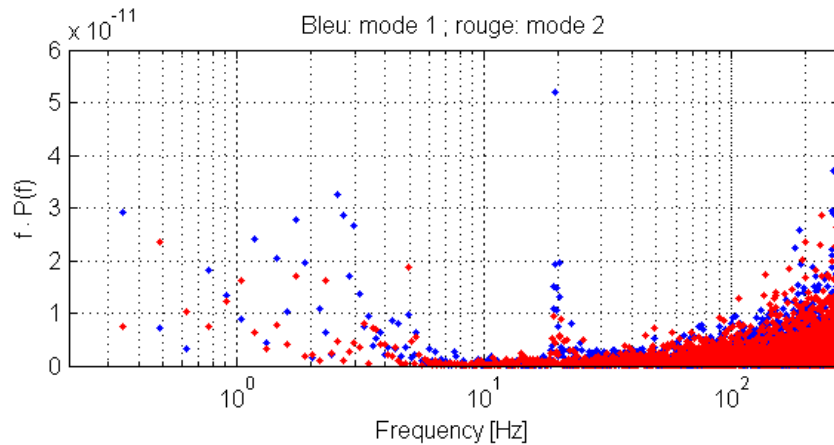


FIG. 3.12: Comparaison de la distribution fréquentielle des modes #1 (bleu) et #2 (rouge). Le mode #1 possède un pic à 20 [Hz] bien plus prononcé que le mode #2.

*Plus.*

Il faut identifier les perturbations, détecter leurs sources et évaluer leurs conséquences. Si une perturbation agit sur le système et dégrade autant de la même manière l'image que les mesures, alors cette perturbation doit rester telle quelle dans les données utilisées pour l'optimisation modale. Comme son effet se retrouve sur l'image finale, l'algorithme d'optimisation doit pouvoir en tenir compte pour minimiser l'erreur de phase résiduelle. D'un autre côté, si cette perturbation n'agit que sur les mesures, il faut s'en débarrasser. Pour cela il faut étudier son comportement pour être à même d'appliquer un filtre.

Le terme pré-traitement indique que les données doivent être modifiées *avant* l'optimisation modale, mais elles représentent un *post-traitement* par rapport à l'acquisition des données. Dans cette situation, des filtres non-causals peuvent être utilisés. C'est ainsi que le pic à 50 [Hz] (§ 3.7.2) peut être retiré du spectre de puissance de manière très précise (fig. 3.13)

### 3.8 L'optimisation modale pour un run

Actuellement et pour le MMT, cette optimisation n'a été testée qu'en simulation. Les résultats sont encourageants et motivent une application concrète de l'algorithme pour être utilisée lors d'un *run* d'optique adaptative. Comme décrit plus haut dans ce chapitre, un nouveau reconstruteur temps réel est en développement en ce moment. Ce dernier permettra l'implémentation d'un algorithme d'optimisation modale automatisé, mais au vu de l'avancement actuel des travaux cette option n'a pas été envisagée dans ce travail.

En revanche, une boîte à outils a été développée pour rendre la tâche de l'optimisation modale aussi aisée que possible. Celle-ci peut être exécutée sur un environnement Matlab, par exemple sur un ordinateur portable embarqué au télescope. Comme le reconstruteur est en général toujours le même et que la modélisation du système ne change pas, la seule entrée que ce programme requiert est un fichier de pentes. Ce dernier contient les mesures du senseur de front d'ondes en boucle fermée.

Un fichier de paramètres permet d'ajuster quelques options pour cette boîte à outils. On y

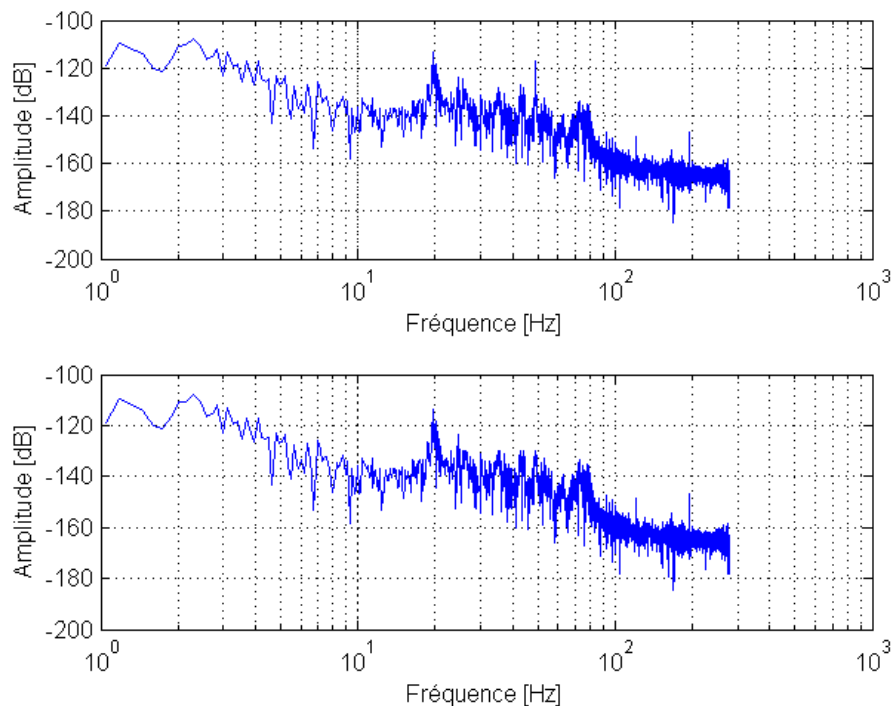


FIG. 3.13: Filtrage du pic à 50 [Hz] avec un filtre non causal appliqué lors d'un pré-traitement à l'optimisation modale.

trouve par exemple la possibilité d'ajuster la gamme des gains dans laquelle on va chercher l'optimum ou le gain appliqué à la boucle d'optique adaptative en boucle fermée lors de la mesure destinée à l'optimisation modale. Plus de détails techniques concernant l'utilisation de ce programme sont donnés en annexe B.1.2.

Ce programme génère un fichier de gains à appliquer au régulateur

### 3.9 Conclusion du chapitre

Le passage de la boucle ouverte à la boucle fermée est nécessaire car le système d'optique adaptative n'est pas capable de prendre des mesures sans que la perturbation ne soit au moins partiellement corrigée. De plus, ce passage est une étape de plus vers une optimisation modale en ligne.

Le modèle mathématique du système a été revu pour correspondre au mieux aux composants du télescope et du système d'optique adaptative. Connaissant les fonctions de transferts de la boucle du système, il est possible, à partir de la mesure des pentes en boucle fermée de déterminer le signal d'entrée correspondant à la turbulence atmosphérique. C'est le signal nécessaire pour effectuer l'optimisation modale.

Comme les gains optimaux ne sont valables que pour la turbulence atmosphérique mesurée *a priori*, il est important de les déterminer le plus rapidement possible pour être à même de faire l'ajustement du système *a posteriori* dans des conditions semblables. De plus, une optimisation temporelle de l'algorithme est également un pas de plus vers une optimisation en ligne. Pour cela, un algorithme de recherche du gain optimal par dichotomie a été implé-

menté.

Nous nous attendions à obtenir une courbe des gains en fonctions des modes de la même allure que celle observée au chapitre précédent ??, c'est-à-dire une courbe généralement décroissante. Ce comportement n'apparaît pas dans tous les cas lors du traitement des données prises sur le ciel. L'analyse de ces résultats montre que l'allure inattendue a un sens par rapport aux données utilisées, mais que les gains obtenus ne sont vraisemblablement pas les meilleurs.

En effet avant l'optimisation modale, un pré-traitement des données est nécessaire. Nous avons identifié plusieurs sources de bruits déformant l'allure du spectre des données modales. Plusieurs pics de fréquences ont été détectés. Le premier, à 20 [Hz] est dû à une oscillation mécanique de la structure du télescope et ajoute donc une composante à cette fréquence sur les deux premiers modes. Cette fréquence doit être corrigée, et fait partie du signal *perturbateur* que l'optique adaptative cherche à annuler, même s'il n'est pas produit par la turbulence atmosphérique. D'autres pics, à différentes fréquences ont été identifiés comme étant vraisemblablement de source électrique. Ces signaux ne perturbent que les mesures et doivent donc être retirés avant d'appliquer l'optimisation modale. Un filtrage non causal peut être appliqué sur ces signaux car ces opérations de pré-processing se font après l'acquisition des données sur le télescope.



# Annexe A

## Remarques et considérations

### A.1 Normalisation de la FFT de Matlab

Nous allons effectuer un bilan d'énergie entre le domaine temporel et fréquentiel d'un signal  $x(t)$  dont nous calculons la FFT avec Matlab avec la fonction `fft()`.

Pour les notations, nous allons utiliser une lettre minuscule pour parler d'un signal temporel, et sa majuscule correspondante dans le domaine fréquentiel. Comme nous cherchons précisément le coefficient de normalisation de la FFT dans Matlab qui est un environnement numérique, nous n'utiliserons que des signaux discrets. Nous noterons cependant  $x[t]$  pour rappeler l'idée que c'est un signal temporel. Par  $x[t]$  il faut bien comprendre que c'est un signal discret calculé pour chaque temps  $t = k \cdot T$ ,  $T$  étant la période d'échantillonnage du signal discret.

Le théorème de Parseval indique que :

$$\sum \|X[f]\|^2 \cdot \Delta f = \sum \|x[t]\|^2 \cdot \Delta t \quad (\text{A.1})$$

Notons  $F[f]$  le résultat de la FFT de Matlab sur le signal  $x[t]$  tel que

`F=fft(x)` ;

Nous savons que  $F[f]$  n'est pas normalisée comme  $X[f]$ . Nous savons par expérience que :

$$\sum \|x[t]\|^2 = \frac{1}{N} \cdot \sum \|F[f]\|^2 \quad (\text{A.2})$$

où  $N$  représente le nombre d'échantillons du signal  $x[t]$ . Ecrivons :

$$\sum \|x[t]\|^2 \cdot \Delta t = c^2 \sum \|F[f]\|^2 \cdot \Delta f \quad (\text{A.3})$$

où  $c$  est le facteur de normalisation cherché. Il en vient :

$$c = \sqrt{\frac{\sum \|x[t]\|^2 \cdot \Delta t}{\sum \|F[f]\|^2 \cdot \Delta f}} \quad (\text{A.4})$$

De A.2 et A.4, on tire :

$$c = \sqrt{\frac{\Delta t}{N \cdot \Delta f}} = \sqrt{\frac{\Delta t}{N \cdot \frac{1}{N \cdot \Delta t}}} = \sqrt{(\Delta t)^2} = \Delta t \quad (\text{A.5})$$

En conclusion, la transformée de Fourier d'une fonction  $x$  échantillonnée toutes les  $t_e$  secondes s'écrit :

$$\text{fft}(x)*t_e$$

## A.2 Problèmes sur les amplitudes des spectres des modes

Observons attentivement la densité spectrale de puissance d'un mode dans l'atmosphère. Pour cette étude, nous allons utiliser le mode *tip* qui est bien présent dans l'atmosphère.

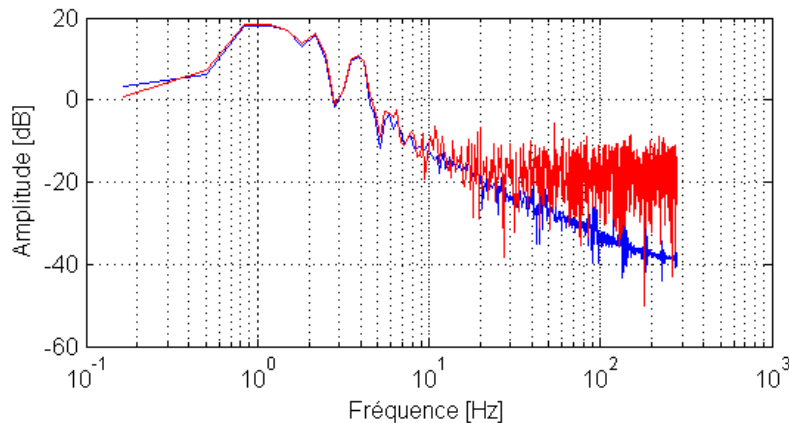


FIG. A.1: Densité spectrale de puissance du *tip* dans l'atmosphère. Mesures faites en boucle ouverte selon les caractéristiques du tableau 2.2. La courbe bleue est issue de mesures sur une étoile de magnitude 0, la courbe rouge avec une étoile de magnitude 14.

La figure A.1 représente la densité spectrale de puissance du mode étudié, qui est issu d'une mesure par le senseur de front d'ondes et reconstruction. C'est donc une mesure de l'amplitude modale contenue dans l'écran de phases envoyé à l'entrée du système. Conan et Noll [8, 21] donne la forme théorique des modes de Zernike dans l'atmosphère. Cette forme est d'amplitude constante jusqu'à un *cutoff*, et ensuite l'amplitude décroît selon la loi :  $f^{-\frac{17}{3}}$ .

On remarque que le mode étudié n'a pas cette allure. La pente après le *cutoff* est bien plus faible, il y a donc trop d'énergie en haute fréquence. Avec ce genre de signal, le bruit prend moins d'importance. En effet, le *vrai* signal couvre plus le bruit que dans la réalité. Sur la figure A.1, on remarque bien l'effet du bruit pour une étoile très faiblement brillante, mais on comprend que le système devrait être *sensible au bruit* plus rapidement. Ce phénomène se fait ressentir sur la chute de performance comme on peut le voir sur la figure 2.14. On voit que cette chute n'intervient qu'à partir des étoiles de magnitude 12, mais vraisemblablement, avec un spectre tel que décrit par Conan, cette chute de performance interviendra plus rapidement. L'optimisation modale ne prend que plus d'importance avec cette constatation. Effectivement on sait que les performances vont chuter plus rapidement, il faudra donc appliquer un régulateur optimisé plus tôt.

Ce problème sur les spectres de la turbulence n'est pas totalement nouveaux. D'autres auteurs ont déjà soulevé des problèmes similaires, tel que Gendron [?, Gendron1] ou Rao [22]. Etant donné que ce travail porte précisément sur l'optimization modale et vu le peu de temps à disposition, aucune étude plus approfondie n'a été menée à ce sujet.

### A.3 Considérations sur le repliement spatial et temporel

Dans cette section nous utilisons les termes *repliement spatial* et *repliement temporel* pour parler respectivement du repliement spectral spatial et spectral temporel. Le but est de confronter les variables et signaux utilisés au théorème de l'échantillonnage.

Le signal d'entrée injecté dans le système est un écran de phases représentant un front d'ondes à chaque instant  $t = k \cdot T_e$ ,  $T_e$  étant la fréquence d'échantillonnage temporelle. Cet écran de phases est défini par une matrice de  $N \cdot N$  point,  $N$  étant le nombre de points par côté. Cet écran de phases couvre le diamètre  $D$  du télescope. La période spatiale de ce front d'ondes est donc de :

$$T_s = \frac{D}{N} \quad (\text{A.6})$$

Le théorème de Shannon nous dit que pour éviter du repliement spatial, il faut que le signal décrivant le front d'ondes ne possède pas de périodes spatiales inférieures à :

$$T = 2 \cdot T_s \quad (\text{A.7})$$

Ce qui se définit avec l'échelle interne du modèle de la turbulence utilisé lors de la génération de l'écran de phases. Dans le cas du télescope MMT, on doit donc utiliser un écran de phases dont l'échelle interne respecte le critère suivant :

$$l_0 > 2 \cdot \frac{6.5}{180} [m] \cong 7.2 [cm] \quad (\text{A.8})$$

Ce critère est *absolu* et permet d'utiliser des signaux corrects en simulation. Ce problème intervient de part l'utilisation de signaux numériques et donc échantillonnés spatialement. Il est bien entendu qu'un cas réel possède des fréquences spatiales plus faibles, mais le front d'ondes n'entre pas échantillonné spatialement dans le système.

En revanche, autant le système réel que la simulation échantillonnent spatialement le front d'ondes lors de sa mesure sur le senseur de front d'ondes. Il est évident que les fréquences spatiales vont se replier dans ce cas, mais nous ne chercherons pas à nous en protéger en simulation car le système réel n'y est pas protégé.

Le système est temporellement échantillonné à  $T_e$  secondes. Nous pouvons calculer la vitesse du vent maximale  $v_{max}$  avant qu'un objet fixe, de petite dimension se déplaçant à  $v_{max} [m/s]$  ne produise du repliement temporel. La fréquence d'échantillonnage est :

$$f_e = \frac{1}{T_e} \quad (\text{A.9})$$

La vitesse maximale du vent pour un objet de  $d [m]$  est de :

$$v_{max} = \frac{f_e}{2} \cdot d \quad (\text{A.10})$$

Le MMT a une période d'échantillonnage  $T_e = 0.0018 [s]$ , et nous prenons un  $l_o$  de  $9 [cm]$  de manière à respecter le critère donné en A.8. Cela nous conduit à une vitesse de vent :

$$v_{max} = \frac{1}{2 \cdot 0.0018} \cdot 0.09 = 25 [m/s] \quad (\text{A.11})$$

Ceci pour éviter le repliement spatial de l'objet de 9[cm] poussé par le vent. Mais ce critère n'est pas *absolu* dans le sens où le système réel y est aussi sensible. Nous n'avons pas besoin de nous protéger en simulation contre ce repliement.

#### A.4 Les fonctions de transferts avec approximation par Pade

Comme décrit au §3.4.1, les retards purs utilisés dans la modélisation du système analogique ont été approximés par Pade (éq. 3.2).

Bloc	Modèle
WFS	$\frac{1 - \frac{T}{2} \cdot s + \frac{T^2}{16} \cdot s^2}{1 + (m+1) \frac{T}{2} \cdot s + (\frac{1}{4} + m + \frac{m^2}{4}) \frac{T^2}{4} \cdot s^2 + (1+m) \frac{m \cdot T^3}{32} \cdot s^3 + \frac{m^2 \cdot T^4}{256} \cdot s^4}$
Integ	$\frac{1}{T \cdot s}$
S/H	$\frac{1}{1 + \frac{T}{2} \cdot s + \frac{T^2}{16} \cdot s^2}$
DM	$\frac{1 - \frac{T_m(i)}{2} \cdot s + \frac{T_m(i)^2}{16} \cdot s^2}{1 + \frac{T_m(i)}{2} \cdot s + \frac{T_m(i)^2}{16} \cdot s^2}$

TAB. A.1: Les fonctions de transfert des blocs du MMT dont les retards purs ont été approximés par Pade (ordre 2)



## Annexe B

# Description des routines Matlab

Durant tout ce travail, plusieurs routines Matlab ont été développées dans le but de fournir une boîte à outils. Les fonctions les plus importantes sont décrites ici avec un mode d'emploi permettant leur utilisation. Le code source Matlab contient en général suffisamment de commentaires pour être compris par un utilisateur habitué à cet environnement.

### B.1 Les fichiers d'optimisation

#### B.1.1 Optimisation en simulation

Nous décrivons ici comment utiliser l'algorithme d'optimisation modale des gains à partir de fichiers obtenus en simulation avec l'outil de simulation Matlab développé par F. Wildi.

L'optimisation est lancée par la commande `optimization`. Cette dernière va donc charger le fichier `optimization.m`. La hiérarchie des fichiers est la suivante :

- `optimization.m` (charge les fichiers et sauve les données)
- `calc_relatives_gains.m` (calcule les gains optimaux)
  - `parameters.m` (contient tous les paramètres)
  - `tf_for_gain_opt.m` (modélise le système)
  - `modal_Noise.m` (gère le calcul du bruit)
    - `calc_noise_by_ac.m` (calcul le bruit par auto-corrélation)
    - `calc_noise_by_av.m` (calcul le bruit par moyennage des hautes-fréquences)

La syntaxe d'utilisation est la suivante :

```
optimization(dir,file)
```

Les paramètres `dir` et `file` sont respectivement le chemin vers le fichier et le fichier de simulation en boucle ouverte qui va être utilisé pour effectuer l'optimisation modale.

#### Les paramètres

Le fichier de paramètres `parameters.m` permet les ajustements décrits dans le tableau B.1.

Variable Matlab	Description
gain_min, gain_max	Plage de recherche du gain optimal
gain_precision	La précision désirée sur le vecteur des gains. Cela définit la résolution de l'axe des gains optimaux.
phase_margin	Marge de phase désirée pour la saturation des gains optimaux. Cela permet de garantir la stabilité du système

TAB. B.1: Description des paramètres du fichier *parameters.m*

### Le format des données

Le fichier fourni par le paramètre `file` est un fichier obtenu par l'outil de simulation Matlab développé par F. Wildi. Ces fichiers portent en général un nom comme : *closed\_\*.mat*. A ceux-ci, un fichier de paramètre de la forme *param\_closed\_\*.mat* et lié et permet de déterminer les caractéristiques de la simulation. L'algorithme se chargera automatique de charger ce fichier et les autres fichiers liés à la simulation tel que le reconstruteur par exemple.

### B.1.2 Optimisation depuis des données réelles du MMT

L'optimisation est lancée par la commande `optimization`. Cette dernière va donc charger le fichier *optimization.m*. La hierarchie des fichiers est la suivante :

- *optimization.m* (charge les fichiers et sauve les données)
- *calc\_modal\_gains.m* (calcule les gains optimaux)
- *parameters.m* (contient tous les paramètres)
- *s\_modal\_modelization.m* (modélise le système)
- *modal\_Noise.m* (gère le calcul du bruit)
  - *calc\_noise\_by\_ac.m* (calcul le bruit par auto-corrélation)
  - *calc\_noise\_by\_av.m* (calcul le bruit par moyennage des hautes-fréquences)

La syntaxe d'utilisation est la suivante :

```
optimization(do_save_file,dir)
```

Avec les deux paramètres facultatifs suivants :

- *do\_save\_file* : flag indiquant si on doit enregistré un fichier (=1) avec les gains optimaux ou pas (=0). Le fichier est enregistré au format ASCII avec une ligne par gain optimal
- *dir* : pour indiquer le répertoire où la fenêtre de sauvegarde doit proposer l'enregistrement du fichier de gains optimaux.

### Les paramètres

Le fichier de paramètres *parameters.m* permet les ajustements décrits dans le tableau B.2.

### Les formats des données

Toutes les données utilisées sont au format FITS. Les pentes sont enregistrées comme une matrice où chaque ligne correspond à un échantillon temporel, et les colonnes correspondent

aux 288 pentes (d'un WFS 12x12). Ces 288 pentes sont ordonnées premièrement en  $x$  (144) et ensuite les pentes en  $y$  (144). Les 144 sous-ouvertures du senseur de front d'ondes sont ordonnées d'abord selon les colonnes (axe  $x$ ) et ensuite selon les lignes (axe  $y$ ).

Les 288 pentes ne sont pas toutes utilisées lors de la reconstruction. Un fichier de *masque* définit les sous-ouvertures utilisées. Ce fichier contient une matrice 12x12 où chaque élément correspond à une sous-ouverture. Cette matrice est remplie de 1 pour les sous-ouvertures utilisées et 0 dans le cas contraire.

Le reconstituteur est également une matrice de dimension  $n \times q$  où  $n$  correspond au nombre de modes corrigés et  $q$  au nombre de pentes réellement utilisées pour la reconstruction (somme des pentes utilisées en  $x$  et en  $y$ ). D'abord les pentes en  $x$  et ensuite les pentes en  $y$ .

## B.2 Utiliser ces routines dans un autre environnement

Ces routines sont écrites pour l'environnement Matlab. Il est donc nécessaire de posséder un ordinateur avec ce programme installé pour faire les faire tourner. Cependant, Matlab possède un interpréteur *C* qui permet de convertir les routines Matlab en fichier \*.c. Il est donc ensuite possible de compiler ces fichiers pour obtenir une version exécutable des routines dans l'environnement correspondant au compilateur. Ceci n'a pas été testé dans ce travail.

## B.3 La fonction de batch

Parmi les nombreuses routines écrites pour gagner du temps dans les opérations régulières de ce travail, il est en une qu'il est intéressant de décrire ici ; la fonction permettant de programmer plusieurs simulations qui vont se succéder sans interaction avec l'utilisateur. Cela permet de dédier un ordinateur pour effectuer plusieurs simulations pendant plusieurs heures sans s'en occuper.

Le fichier qui gère le lancement des simulations est *general\_sequencer.m*. Il répond à des *call-back* venant des actions sur les boutons du panneau de contrôle. Un nouveau bouton  $y$  a donc été ajouté (fig. B.1). Le nouveau fichier *general\_sequencer.m* contient donc un nouveau cas (un nouveau *call-back*) à traiter.

Pour utiliser cette fonction, il faut créer un répertoire

*batch* à la racine du répertoire principal contenant la simulation. Ce répertoire contiendra des fichiers appelés *batch1.mat*, *batch2.mat*, ..., *batchn.mat*. Ces fichiers contiennent la sauvegarde d'une structure Matlab (*batch\_struct*) avec les propriétés suivantes :

```
g_run_graph_flag
g_run_decomp_flag
result_directory
wfs_model
atmos_directory
WFS_file
WFS_directory
WFS_LUT_directory
WFS_LUT_file
```

```
reconst_directory
atmos_file
reconst_file
general_file
controller_file
result_file
```

La routine *batch\_builder.m* (située également dans le répertoire *batch*) permet de générer facilement ces fichiers de structure pour programmer les simulations à effectuer. Le processus de batch lance toutes les simulations les unes après les autres en incrémentant le nombre à la fin du nom de fichier. Lorsqu'il ne trouve plus de fichier il s'arrête.

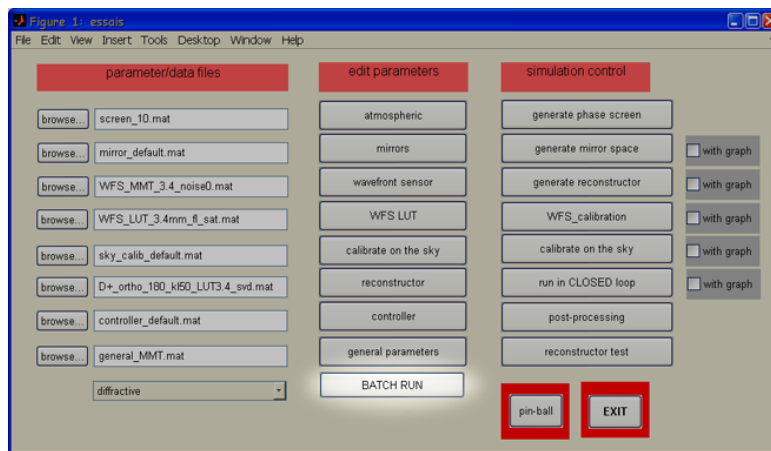


FIG. B.1: Un bouton a été ajouté au panneau de contrôle de simulation pour lancer le processus de batch.

Variable Matlab	Description
gain_min, gain_max gain_precision	Plage de recherche du gain optimal La précision désirée sur le vecteur des gains. Cela définit la résolution de l'axe des gains optimaux.
phase_margin	Marge de phase désirée pour la saturation des gains optimaux. Cela permet de garantir la stabilité du système
t_e m_t_e	Période d'échantillonnage du système Facteur de temps d'intégration du senseur de front d'onde. Permet d'ajuster un temps d'intégration différent de la période d'échantillonnage.
g_prime closed_loop_gain	Le gain $g'$ de calibration du senseur de front d'onde. Le gain de la boucle utilisée lors de l'acquisition des mesures en boucle fermé
modal_dm	Flag permettant de définir si l'on veut utiliser des fonctions de transfert différentes pour chaque mode (=1) ou pas (=0) pour le miroir déformable.
dm_modal_delay_file	Fichier contenant les valeurs du retard pur estimé pour chaque mode du miroir déformable. Le fichier est de type ASCII avec un ligne pour chaque mode. Si le flag <code>modal_dm</code> est à zéro (fonction de transfert identique pour chaque mode), alors la valeur moyenne de ces retards purs est utilisée comme valeur globale.
use_freq_from, use_freq_to type_noise_evaluation	Plage de fréquence utilisée pour effectuer l'optimisation modale. Définit la méthode utilisée pour évaluer le niveau de bruit. Les possibilités sont l'auto-corrélation (=1) ou effectuer une moyenne sur les 100 derniers échantillons de fréquence du spectre (=2). La seconde méthode est sensiblement plus rapide, mais elle n'est utilisable qu'avec un niveau de bruit relativement élevé.
force_tf	Lorsque ce flag est à 1, les fonctions de transfert sont de toute manière re-calculées pour chaque optimisation modale. En le mettant à 0, les fonctions de transfert ne sont calculées que si les paramètres utilisés dans ces calculs sont modifiés (dans le fichier <i>parameters.m</i> ).
sat_stability_limit	Lorsque ce flag est à 1, aucun gain n'est recherché en dehors de la limite de stabilité du système. Dans le cas contraire, les gains finaux sont bel et bien saturés (à l'enregistrement), mais le graphique montre les gains non-saturés. Cela permet de mieux étudier l'allure des gains optimaux mais ralentis le processus de minimization.
use_default_reconst	Un nouveau reconstituteur n'est demandé que si ce flag est à 0. Dans le cas contraire, le reconstituteur donné par le paramètre <code>default_reconst_file</code> est utilisé avec son masque correspondant <code>default_mask_file</code> .
default_reconst_file	Le reconstituteur utilisé lorsque le flag <code>use_default_reconst</code> est à 1.
default_mask_file	Le mask correspondant au reconstituteur par défaut.

TAB. B.2: Description des paramètres du fichier *parameters.m*



# Bibliographie

- [1] Girardet B. *Travail de semestre : optimisation à posteriori et en ligne d'un régulateur multi-variables d'un système d'optique adaptative*. 07 2004.
- [2] Robert T. Brigantic, Michael C. Roggermann, Bryon M Welsh, and Kenneth W. Bauer. Optimization of adaptive-optics systems closed-loop bandwidth settings to maximize imagine-system performance. *Applied Optics*, 37(5) :848–855, February 1998.
- [3] Guido Brusa, Douglas L. Miller, Matt A. Kenworthy, Donald L. Fisher, and Armando Riccardi. Mmt-ao : two years of operation with the first adaptive secondary. volume 5490, pages 23–33. SPIE, 2004.
- [4] Guido Brusa, Armando Riccardi, Piero Salinari, Francois P. Wildi, Michael Lloyd-Hart, Hubert M. Martin, Richard Allen, Donald Fisher, Doug L. Miller, Roberto Biasi, Daniele Gallieni, and Fabio Zocchi. Mmt adaptive secondary : performance evaluation and field testing. volume 4839, pages 691–702. SPIE, 2003.
- [5] Guido Brusa, Armando Riccardi, Francois P. Wildi, Michael Lloyd-Hart, Hubert M. Martin, Richard Allen, Donald L. Fisher, Douglas L. Miller, Roberto Biasi, Daniele Gallieni, and Fabio Zocchi. Mmt adaptive secondary : first ao closed-loop results. volume 5169, pages 26–36. SPIE, 2003.
- [6] Michael W. Burke. *Image Acquisition*, volume 1, chapter 5, pages 539–679. Chapman & Hall, 1996.
- [7] Dessenne Caroline. *Commande modale et prédictive en optique adaptative*. PhD thesis, ONERA, 1999.
- [8] Jean-Marc Conan, Gérard Rousset, and Pierre-Yves Madec. Wave-front temporal spectra in high-resolution imaging through turbulence. *Optical Society of America*, pages 1559–1570, 1995.
- [9] L. Demailly. *Contrôle distribué en optique adaptative. Vers un système multi-agents intelligent*. PhD thesis, Université de Caen, 02 1996.
- [10] C. Dessenne, P.Y. Madec, and G. Rousset. Modeal prediction for closed-loop adaptive optics. *Optics Letters*, 22 :1535–1537, 1997.
- [11] C. Dessenne, P.Y. Madec, and G. Rousset. Optimization of a predictive controller for closed-loop adaptive optics. *Applied Optics*, 21 :4623–4633, 1998.
- [12] C. Dessenne, P.Y. Madec, and G. Rousset. Sky implementation of modal predictive control in adaptive optics. *Optics Letters*, 5 :339–341, 1998.
- [13] B.L. Ellerbroek and T.A. Rhoadarmer. Adaptive wavefront control algorithms for closed loop adaptive optics. *Mathematical and computer modelling*, page 33, 2001.
- [14] Brent L. Ellerbroek, Charles Van Loan, Nikos P. Pitsianis, and Robert J. Plemmons. Optimizing closed-loop adaptive-optics performance with use of multiple control bandwidths. *Optical Society of America*, 11(11) :2871–2886, November 1994.

- [15] E. Gendron. *Optimisation de la commande modale en optique adaptative : application à l'astronomie*. PhD thesis, Université de Paris VII, March 1995.
- [16] E. Gendron and P. Léna. Astronomical adaptive optics i. modal control optimization. *Astronomy and Astrophysics*, 291 :337–347, 1994.
- [17] E. Gendron and P. Léna. Astronomical adaptive optics ii. experimental results of an optimized modal control. *Astronomy and Astrophysics. Suppl. Ser.*, 111 :153–167, 1995.
- [18] James Steven Gibson, Chi-Chao Chang, and Brent L. Ellerbroek. Adaptive optics : wavefront correction by use of adaptive filtering and control. *Optical Society of America*, 39(16) :2525–2538, June 2000.
- [19] Brice Le Roux. *Commande optimale en Optique Adaptative Classique et Multiconjugée*. PhD thesis, Université de Nice, 10 2003.
- [20] Roland Longchamp. *Commande numérique de systèmes dynamiques*. Presses polytechniques et universitaires romandes, 1st edition, 1995.
- [21] Robert J. Noll. Zernike polynomials and atmospheric turbulence. *Optical Society of America*, pages 207–211, 1976.
- [22] Ruizhong Rao, Shipeng Wang, Xiaochun Liu, and Zhiben Gong. Turbulence spectrum effect on wave temporal-frequency spectra for light propagating through the atmosphere. *Optical Society of America*, 16(11) :2755–2762, November 1999.
- [23] Armando Riccardi, Guido Brusa-Zappellini, Valdemaro Biliotti, Ciro Del Vecchio, Piero Salinari, Paolo Stefanini, Paolo Mantegazza, Roberto Biasi, Mario Andrighettoni, Claudio Franchini, Daniele Gallieni, Michael Lloyd-Hart, Patrick C. McGuire, Stephen M. Miller III, and Hubert M. Martin. Adaptive secondary mirror for the 6.5-m conversion of the multiple mirror telescope : latest laboratory test results of the p36 prototype. volume 4007, pages 524–531. SPIE, 2000.
- [24] Horst Stöcker, Francis Jundt, and Georges Guillaume. *Toute la physique*, volume 1, chapter 12, pages 343–424. Chapman & Hall, 1999.
- [25] Robert K. Tyson. *Principles of Adaptive Optics*. Academic Press, 2nd edition, 1998.
- [26] Francois P. Wildi and Guido Brusa. Determining the interaction matrix using starlight. volume 5490, pages 164–173. SPIE, 2004.