École polytechnique de Louvain

Low-rank Models for Exoplanets Detection by Angular and Spectral Differential Imaging

Author: **Nicolas Mil-Homens Cavaco**
Supervisors: **Pierre-Antoine Absil, Laurent Jacques**
Reader: **Valentin Christiaens**
Academic year 2023–2024
Master [120] in Mathematical Engineering

Abstract

Differential Imaging is currently a widespread technique to post-process images of a remote planetary system, captured by ground-based telescopes during an observation campaign, so that exoplanets around the host star are directly visible after post-processing. To extract crucial information about the spectrum of the exoplanets, the light is further decomposed into several spectral channels during the data acquisition process, yielding a 4-D data cube of spectral frames and temporal frames. By combining those images cleverly, Angular and Spectral Differential Imaging (ASDI) allows to remove the so-called quasi-static speckles that dramatically affect our detection capabilities, since they have the same shape as an exoplanet. This Master's thesis has multiple objectives. Firstly, a great importance is attached to the modeling of the 4-D data cubes used in ASDI. Then, we design new ASDI post-processing algorithms based on low-rank models, which take two forms : inverse problem-based approaches and greedy algorithms. Both are finally compared with an ASDI version of the classical PCA algorithm by massively injecting fake companions in a data cube without exoplanets from the VLT. It turns out that these new algorithms correctly detect the injected fake companions as well as PCA and even outperform the latter at small angular separation.

Acknowledgments

Je souhaite sincèrement remercier Pierre-Antoine Absil et Laurent Jacques, mes promoteurs de mémoire, pour leur soutien, leurs conseils de qualité et leur grande disponibilité tout au long de l'année.

J'aimerais également remercier Valentin Christiaens, Olivier Absil et Hazan Daglayan pour leur aide précieuse dans ce TFE. En particulier, je les remercie pour le partage de leur expertise, des cubes de données, de codes informatiques et pour leur disponibilité lors des échanges par courriel. En outre, je remercie Valentin Christiaens d'avoir accepté d'être lecteur de ce mémoire.

Merci à mes amis pour l'ambiance chaleureuse et la motivation que vous m'avez apportées lorsque nous travaillions sur nos mémoires respectifs au bâtiment Euler.

Enfin, j'aimerais exprimer ma reconnaissance envers mes parents et ma famille pour leur écoute et leurs encouragements.

Notation

$\mathbf{1}_n$	Vector of size n full of ones
$\boldsymbol{\delta}_{i,j}$	Matrix of size $m \times n$ full of zeros except in (i,j) where it is equal to one
$\|.\|_0$	ℓ_0-pseudo-norm counting the number of non-zero entries of a vector or a matrix
$\|.\|_\infty$	ℓ_∞-norm : $$\|\boldsymbol{a}\|_\infty = \max_{i=1,\ldots,n} \|a_i\| \qquad \forall \boldsymbol{a} \in \mathbb{R}^n$$
$\|.\|_p$	Componentwise ℓ_p-norm $(p = 1, 2)$: $$\|\boldsymbol{a}\|_p = \left(\sum_{i=1}^{n} a_i^p\right)^{1/p} \qquad \forall \boldsymbol{a} \in \mathbb{R}^n$$ $$\|\boldsymbol{A}\|_p = \left(\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^p\right)^{1/p} \qquad \forall \boldsymbol{A} \in \mathbb{R}^{m\times n}$$
$\|.\|_{p,q}$	Mixed-norm $(p, q = 0, 1, 2)$: $$\|\boldsymbol{A}\|_{p,q} = \left\| \left[\|\boldsymbol{a}_i\|_p \right]_{i=1}^{n} \right\|_q \qquad \forall \boldsymbol{A} = [\boldsymbol{a}_1\ \boldsymbol{a}_2\ \cdots\ \boldsymbol{a}_n] \in \mathbb{R}^{m\times n}$$
$\langle ., . \rangle$	ℓ_2 inner-product : $$\langle \boldsymbol{a}, \boldsymbol{b} \rangle = \sum_{i=1}^{n} a_i b_i \qquad \forall \boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$$ $$\langle \boldsymbol{A}, \boldsymbol{B} \rangle = \sum_{i=1}^{m}\sum_{i=j}^{n} a_{ij} b_{ij} \qquad \forall \boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{m\times n}$$
$\boldsymbol{\Sigma}_r$	Diagonal matrix of the r first singular values in decreasing order of a matrix
$\boldsymbol{U}_r, \boldsymbol{V}_r$	Matrices containing the r first left and right singular vectors of a matrix respectively
$\mathcal{H}_r^{\mathrm{SVD}}$	Truncated SVD operator projecting on the low-rank manifold: $$\begin{aligned}\mathcal{H}_r^{\mathrm{SVD}}(\boldsymbol{X}) &:= \arg\min_{\boldsymbol{L}} \|\boldsymbol{X} - \boldsymbol{L}\|_2^2 \quad \text{s.t. } \operatorname{rank}(\boldsymbol{L}) \le r \\ &= \boldsymbol{U}_r \boldsymbol{\Sigma}_r \boldsymbol{V}_r^\top\end{aligned}$$
$\mathcal{I}$	Identity operator
prox_f	Proximal operator

$\mathcal{A}^*$ Adjoint of a linear operator $\mathcal{A} : \mathcal{X} \subseteq \mathbb{R}^m \mapsto \mathcal{Y} \subseteq \mathbb{R}^n$ satisfying :

$$\langle \mathcal{A}\boldsymbol{x}, \boldsymbol{y} \rangle = \langle \boldsymbol{x}, \mathcal{A}^* \boldsymbol{y} \rangle \qquad \forall \boldsymbol{x} \in \mathcal{X}, \forall \boldsymbol{y} \in \mathcal{Y}$$

$\mathcal{L}$ Loss function

$\mathcal{R}$ Regularization function

α Regularization parameter

ε Tolerance

μ, μ' Stepsize

L Lipschitz constant

$\boldsymbol{\eta}$ Noise

$N \times N$ Size of a frame

$n \times n$ Size of a preprocessed frame

T Number of temporal frames of a data cube

W Number of spectral frames of a data cube

$\boldsymbol{p}$ Position vector in a continuous 2D image

t Time or time index

λ Wavelength

l, λ_l Wavelength index of a spectral frame and its associated wavelength

D Diameter of the aperture

λ/D Diameter of a resolution element

$\bar{\lambda}/D$ Mean diameter of a resolution element

$\boldsymbol{Y}$ Light intensity field (data cube)

$\boldsymbol{Z}$ Rescaled light intensity field according to the spatial dimension (preprocessed data cube)

$\boldsymbol{Y}_\star, \boldsymbol{Z}_\star$ Speckle pattern component of $\boldsymbol{Y}$ and $\boldsymbol{Z}$

$\boldsymbol{\varphi}$ Point spread function (PSF)

$\mathcal{Q}_{\boldsymbol{\alpha}}$ Rotation operator of parallactic angle $\boldsymbol{\alpha} \in \mathbb{R}^T$ and associated 2×2 rotation matrices $\{\boldsymbol{Q}_{\alpha(t)}\}_{t=1}^T$:

$$\mathcal{Q}_{\boldsymbol{\alpha}} : \mathbb{R}^{W \times N^2} \mapsto \mathbb{R}^{WT \times N^2} \equiv \text{ discrete version of } F(\lambda, \boldsymbol{p}) \mapsto F(\lambda, \boldsymbol{Q}_{\alpha(t)}\boldsymbol{p})$$

$\mathcal{D}_{\boldsymbol{s}}$ Scaling operator of scale factor $\boldsymbol{s} \in \mathbb{R}^W$:

$$\mathcal{D}_{\boldsymbol{s}} : \mathbb{R}^{WT \times N^2} \mapsto \mathbb{R}^{WT \times n^2} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto F(\lambda, t, s(\lambda)^{-1}\boldsymbol{p})$$

$\mathcal{T}_{\boldsymbol{\varphi}}$ Convolution operator with the PSF $\boldsymbol{\varphi}$:

$$\mathcal{T}_{\boldsymbol{\varphi}} : \mathbb{R}^{W \times N^2} \mapsto \mathbb{R}^{W \times N^2} \equiv \text{ discrete version of } F(\lambda, \boldsymbol{p}) \mapsto [F(\lambda, .) * \varphi(\lambda, .)](\boldsymbol{p})$$

$\mathcal{P}'_\Omega$	Partial restriction operator on the set of spatial coordinates $\Omega \subset \mathbb{R}^2$: $$\mathcal{P}'_\Omega : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{WT\times\lvert\Omega\rvert} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto \{F(\lambda, t, \boldsymbol{p})\}_{\boldsymbol{p}\in\Omega}$$
$\mathcal{P}_\Omega$	Restriction operator on the set of coordinates $\Omega \subset \mathbb{R}\times\mathbb{R}\times\mathbb{R}^2$: $$\mathcal{P}_\Omega : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{\lvert\Omega\rvert} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto \{F(\lambda, t, \boldsymbol{p})\}_{(\lambda,t,\boldsymbol{p})\in\Omega}$$
$\mathcal{M}_\Omega$	Mask operator defined on the set of coordinates $\Omega \subset \mathbb{R}\times\mathbb{R}\times\mathbb{R}^2$. The mask and the restriction operators are linked by $\mathcal{M}_\Omega = \mathcal{P}^*_\Omega\mathcal{P}_\Omega$: $$\mathcal{M}_\Omega : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{WT\times n^2} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto \begin{cases} F(\lambda, t, \boldsymbol{p}) & \text{if } (\lambda, t, \boldsymbol{p}) \in \Omega \\ 0 & \text{otherwise.} \end{cases}$$
Ψ	Trajectorlet operator : $\Psi(.) = \mathcal{D}_{\boldsymbol{s}}\mathcal{Q}_{\boldsymbol{\alpha}}\mathcal{T}_{\boldsymbol{\varphi}}(.)$
$\boldsymbol{L}$	Low-rank matrix modeling the background
$\boldsymbol{S}$	Circumstellar signal
$\boldsymbol{R}$	Residual map
$\boldsymbol{M}$	Detection map
g	Trajectory described by an exoplanet in a data cube or starting pixel of that trajectory
$\boldsymbol{P}_{l,g}$	Planet signature defined as a PSF measured at wavelength λ_l that moves along trajectory g in the data cube
$a_{l,g}$	Planet flux along trajectory g measured at wavelength λ_l
B	Number of batches
b	Batch index
Λ_b	b^{th} batch

Abbreviation

ADI	Angular Differential Imaging
c-ADI	Classical ADI
AMAT	Alternating Minimization Algorithm with Trajectories
AO	Adaptive Optics
ASDI	Angular and Spectral Differential Imaging
ELT	Extremely Large Telescope
FBAT	Forward-Backward Algorithm with Trajectories
FN	False Negative
FP	False Positive
FPR	False Positive Rate
FWHM	Full Width of Half Maximum
HCI	High Contrast Imaging
IFS	Integral Field Spectrograph
JWST	James Webb Space Telescope
LLSG	Local Low-rank plus Sparse Gaussian
LRPT	Low-Rank Plus sparse Trajectory decomposition
PCA	Principal Component Analysis
PSF	Point Spread Function
ROC	Receiver Operating Characteristics
SDI	Spectral Differential Imaging
c-SDI	Classical SDI
SNR	Signal-to-Noise Ratio
SVD	Singular Value Decomposition
SPHERE	Spectro-Polarimetric High-contrast Exoplanet REsearch imager
TN	True Negative
TP	True Positive
TPR	True Positive Rate

VIP	Vortex Image Processing (Python toolbox)
VLT	Very Large Telescope

Contents

Introduction

Since the first exoplanet detected in 1992 by Wolszczan and Frail, detecting new exoplanets has become nowadays a common event. While in 2022 astronomers were celebrating the 5000th exoplanet discovery, in comparison today there are already more than 5600 populating the sky[1]. However, among them, only 1.4% were directly imaged, meaning that most of the exoplanets were detected via indirect methods such as transit (74.2%) and radial velocity (19.4%)[2].

On the one hand, the leading method of transit is based on measurements of the brightness of a star. The presence of a planet is deduced from a periodic decrease of luminosity of the star. When this occurs, it can be the sign of a planet transiting in front of the star and hence masking a part of the light arriving at our telescopes. On the other hand, radial velocity exploits the Doppler effect of light. In the case of a massive planet, both the star and the planet will act on each other according to the gravity law, leading to circular movements of both massive bodies. Consequently, if this movement is large enough, a periodic Doppler shift can be measured in the star light captured by our telescopes, revealing the presence of an exoplanet around that star. Overviews of those two techniques can be found for instance in [1] and [2] respectively.

Despite the success of indirect methods, only limited information about the planet can be extracted from those measurements. In contrast, direct imaging combined with spectroscopy allows to know several chemico-physical quantities characterizing the circumstellar body and therefore remains of great interest in the search of Earth-like exoplanets. With the emergence of new hardware in the next decades, direct imaging is even presented as the most promising technique for the incoming years [3].

In this master thesis, we will investigate direct methods through angular and spectral differential imaging that consist in combining several pictures of a star imaged during a complete observation campaign, where each picture is simultaneously captured at different wavelengths in the near-infrared.

As mentioned previously, only a small fraction of exoplanets were directly imaged, and the reason is simply that this task is very hard.

The first challenge is the high contrast between the planet and the host star. An analogy to illustrate the phenomenon is to see the planet as a firefly in front of a lighthouse representing the star, both located very far away from our eyes [4, 5]. With this picture in mind, one can understand how difficult it is to distinguish the faint firefly of its very bright environment. That is why direct imaging is commonly called High Contrast Imaging (HCI).

The second challenge comes from the large distance between the remote planetary system and the Earth, but also from the tiny angular separation between the exoplanet and the host star. In order to image the exoplanet, we therefore need highly sensitive telescopes with very large aperture to reach high resolution [6]. Otherwise, both the exoplanet and the star would

[1] All planets found until today are available at https://exoplanet.eu/catalog/. Last access on 27/05/24.
[2] https://exoplanets.nasa.gov/. Last access on 27/05/24

completely overlap on the images we capture. Until recently [7], only ground-based telescopes were able to reach such a resolution. While they remain the most powerful instruments in terms of achievable angular resolution, these induce another challenge: the atmospheric disturbance on the wavefront arriving from the space to our sensors.

Combining coronagraphs, advanced optics, deformable mirrors, wavefront control systems and image processing algorithms, we are able today to partially overcome those challenges. Despite all those advances in the field, only bright, young massive exoplanets at high angular separation from the star have been directly imaged to date.

In this context, we propose new image processing algorithms inspired by previous works, such as [8], [9] and [10], in order to enhance the contrasts in coronographic images coming from ground-based telescopes. This work involves several image processing and machine learning concepts such as low-rank models, sparsity principle, proximal algorithms, ... Moreover, we will follow an inverse problem procedure by formulating a forward model of data cubes for Angular and Spectral Differential Imaging (ASDI).

This document is composed of six chapters. The first chapter introduces the context and the state-of-the-art of High Contrast Imaging and Differential Imaging. Then, in the second chapter, several details about the Python toolbox VIP (for Vortex Image Processing) and the VLT/SPHERE-IFS datasets used in this master thesis are provided. The third chapter describes some elements of Inverse Problem theory as well as the optimization algorithms used to solve the inverse problems we will encounter. Chapter 4 is the core of the document where all the information provided in the previous chapters is gathered to design new ASDI post-processing algorithms. Chapter 5 recalls and explains some metrics widely used for exoplanets detection. Finally, numerical results can be found in Chapter 6 where we assess and discuss the performance of each ASDI algorithms covered in this work using those metrics. To conclude this document, several ideas for improvements and potential research directions are given.

Chapter 1

High contrast imaging (HCI)

High contrast imaging, also called direct imaging, consists in capturing the light of exoplanets via powerful telescopes located at the surface of the Earth (ground-based) or inside satellite (space-based). The name "high contrast" comes from the large gap in terms of brightness between the exoplanet we want to detect and the host star, making HCI a difficult task.

This first chapter introduces the fundamental background needed to understand the content of the next chapters. Firstly, we introduce some limits on the performance of HCI that we cannot overcome because of physics. Then, more details are given about the challenges of HCI mentioned in the introduction. Secondly, we introduce several differential imaging techniques and briefly review the literature. The main references consulted to write this chapter are [5, 11, 12].

1.1 Intrinsic physical limits

Diffraction and Point Spread Function (PSF)

The most fundamental effect that limits the resolution of any optical device is the light diffraction. More precisely, when a point source at a long distance of the telescope is observed, Fraunhofer's approximation holds. The light arriving in the focal plane can thus be seen as the result of a superposition of an infinite number of coherent point sources that interfere with each other, yielding the so-called Point Spread Function (PSF) on the image [13]. The PSF thus corresponds to the response of the instrument to a Dirac's delta (the point source), and when a whole planetary system is imaged, the PSF acts as a convolution kernel that blurs the real image that we would have obtained without diffraction (see figure 1.1).

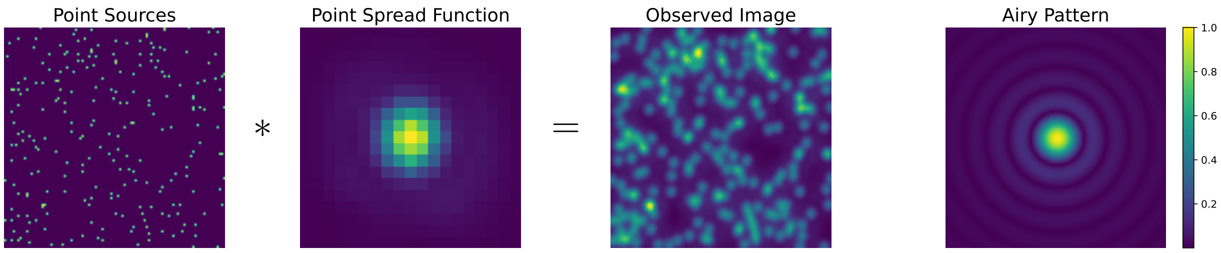

Figure 1.1: Left: Convolving the point sources, for instance representing planets in the sky, with the instrumental PSF yields the final image. Right: Airy pattern obtained as the modulus of the Fourier Transform of a circular aperture. The larger the aperture, the smaller the Airy pattern and the higher the resolution.

Regarding the shape of the PSF, the latter is directly related to the Fourier Transform of the aperture's shape. For instance, for a perfectly circular aperture, the PSF draws the so-called Airy pattern that is composed of a very bright spot at its center surrounded by successive rings of light of decreasing intensity as we move away from the center. A more complete mathematical description of the Airy pattern is provided in chapter 2.

From the previous picture, we can understand what happens when two point sources are close to each other. If their distance is lower than the diameter of the bright spot in the center, then they overlap each other and become indistinguishable. In optics theory, this is called the Rayleigh criterion. The diameter of the bright spot, generally referred to angular resolution θ, is measured in radian. It depends both on the diameter D of the circular aperture and the wavelength of the light λ, and roughly amounts to

$$\theta \approx 1.22\frac{\lambda}{D}. \tag{1.1}$$

In view of this expression, the more we ask a high resolution θ of the telescope, the more D has to increase. For example, to image β Pictoris c[1] at 19.4 pc (i.e 63.4 light-years) at angular separation of ~ 100 mas (i.e $4.8 \cdot 10^{-7}$ rad), in the K-band ($\lambda \sim 2.2$ µm) [14], we need a telescope with a diameter of at least $D \approx 5.5$ m. In the past, the Hubble Space Telescope aperture of 2.4 m was not convenient to directly detect exoplanets with space-based telescope, but since the recent launch of the James Webb Space Telescope (JWST) with an aperture of ~ 6.5 m, space-based direct imaging is now a reality and has the potential, thanks to the combination of near-infrared and mid-infrared images, to outperform old ground-based telescopes such as the Very Large Telescope (VLT) with its aperture of 8 m [7]. However, with the recent installation of the Extremely Large Telescope (ELT) equipped with a huge aperture (~ 39 m) that will be operational in the coming years[6], ground-based telescopes remain the leaders in terms of resolution power.

Photon noise

The second effect limiting HCI is called photon noise, which designates the uncertainty associated with the measurement of the light. This comes from the quantized nature of the light and the independence of photon arrivals at the detector. Those last can be modeled as a classical Poisson process $N(t)$ counting the number of photons we measure on a time interval of length t, corresponding to the exposure time. The arrival rate κ of the process is proportional to the flux of the point source [15, 11], which is defined as the amount of energy received from a celestial object per unit of area and unit of time.

In the context of exoplanet detection, there are generally two sources of light, the exoplanet and the star, associated to Poisson processes $N_P(t)$ and $N_S(t)$ respectively. A common metric to evaluate the performance of an optical device is the Signal to Noise Ratio (SNR) defined as the ratio between the signal of interest and the standard deviation of the noise. Hence, a high SNR means good performance. Mathematically, in the ideal case where only the photon noise holds, the SNR is given by :

$$\mathrm{SNR}(t) = \frac{\mathbb{E}[N_P(t)]}{\sqrt{\mathrm{Var}[N_P(t) + N_S(t)]}} = \frac{\kappa_P t}{\sqrt{\kappa_P t + \kappa_S t}} = \mathrm{SNR}_P(t)\left(1 + \frac{\kappa_S}{\kappa_P}\right)^{-1/2}, \tag{1.2}$$

where $\mathrm{SNR}_P(t) = \sqrt{\kappa_P t}$ is the best SNR we can achieve when no star lies in the neighborhood of the exoplanet. We therefore see that the presence of the star highly deteriorates the SNR as

[1] β Pictoris c is the first planet detected by the radial velocity indirect method that was confirmed by direct imaging (in 2020).

the ratio κ_S/κ_P is typically of order 10^6 if no additional optical device, such as a coronagraph, is used. In summary, equation (1.2) provides an upper bound on the best SNR we can achieve and highlights how decreasing the contrast ratio is crucial to improve detectability of a planet. Note that here we have made a simplified analysis, and interested readers can consult [11, 12, 16] for more information.

1.2 Challenge one: High Contrast

The previous section illustrates mathematically how the high contrast deteriorates the SNR. We therefore would like to lower the ratio κ_S/κ_P as much as possible, which can be achieved by using a coronagraph.

A coronagraph is an instrument that blocks out the light from a star so that nearby objects, such as exoplanets, can be observed. In simple words, one can see this device as producing an artificial eclipse, even if this analogy has limitations as it is not totally correct [17, Chapter 1].

The first coronagraph, invented by Lyot in the 1930s, is represented in figure 1.2. This is composed of four planes, alternating pupil plane and focal plane successively, which filter and mask the starlight thanks to the apodizer, the focal plane mask and the Lyot stop. Those components are optimized so that the on-axis light from the star is attenuated while the off-axis light from the planet is transmitted through the coronagraph.

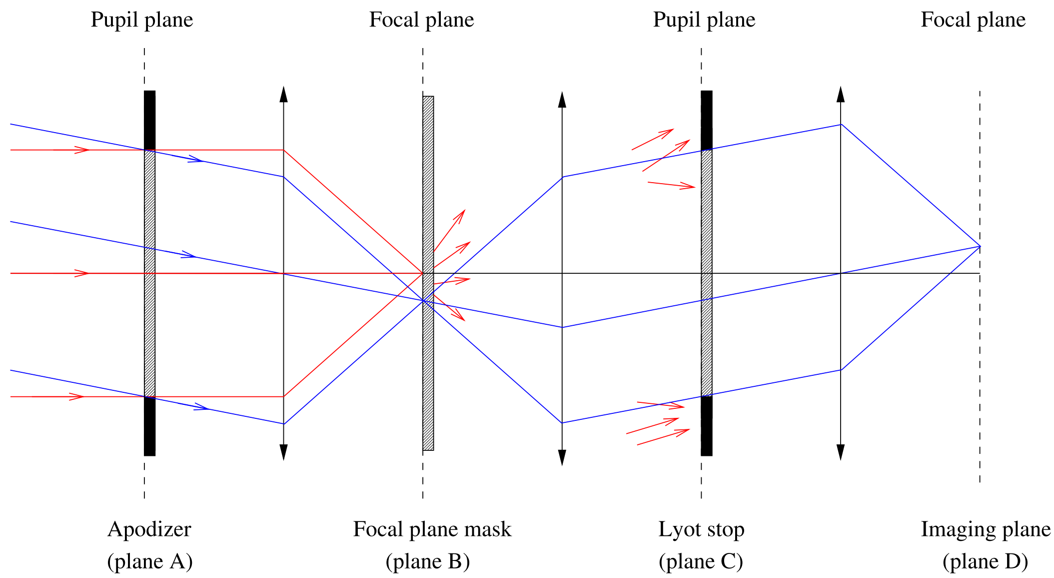

Figure 1.2: Lyot coronagraph from [11]. The starlight (red) is filtered while the exoplanet light (blue) arrives in the imaging plane D without alteration. Black arrows between a pupil plane and a focal plane represent a lens so that an optical Fourier Transform is performed on the signal that passes through this lens according to Fraunhofer's approximation. The inverse transform is made when going from a focal plane to a pupil plane.

It is important to mention that the whole optical instrument can be summarized in an operator $\mathcal{C}$ applied on the incoming wave, but it is no longer shift invariant. Therefore, we cannot define a PSF as in the previous section [12, 18]. However, since only on-axis light is impacted by the coronagraph, at sufficiently large angular separation, the operator $\mathcal{C}$ can still be modeled by a convolution with the non-cornographic PSF, that is, the impulse response observed before the focal plane mask. This is the approximation commonly adopted in the HCI community that we will also adopt in this document.

1.3 Challenge two: Atmospheric turbulence

In the previous section, we considered an ideal case where both the star and the exoplanet emit perfectly flat wavefront of light. In practice, with ground-based telescope, the light passes through the atmosphere yielding aberrations that deteriorate the quality of the image. Those aberrations mainly come from the dependency of the refractive index of the atmosphere with respect to the continuously fluctuating temperature and pressure [19, Chapter 3]. As a consequence, wavefronts arriving at the entry of the telescope are not perfectly flat and other techniques like Adaptive Optics (AO) must be used to decrease the effects of aberrations (see figure 1.3).

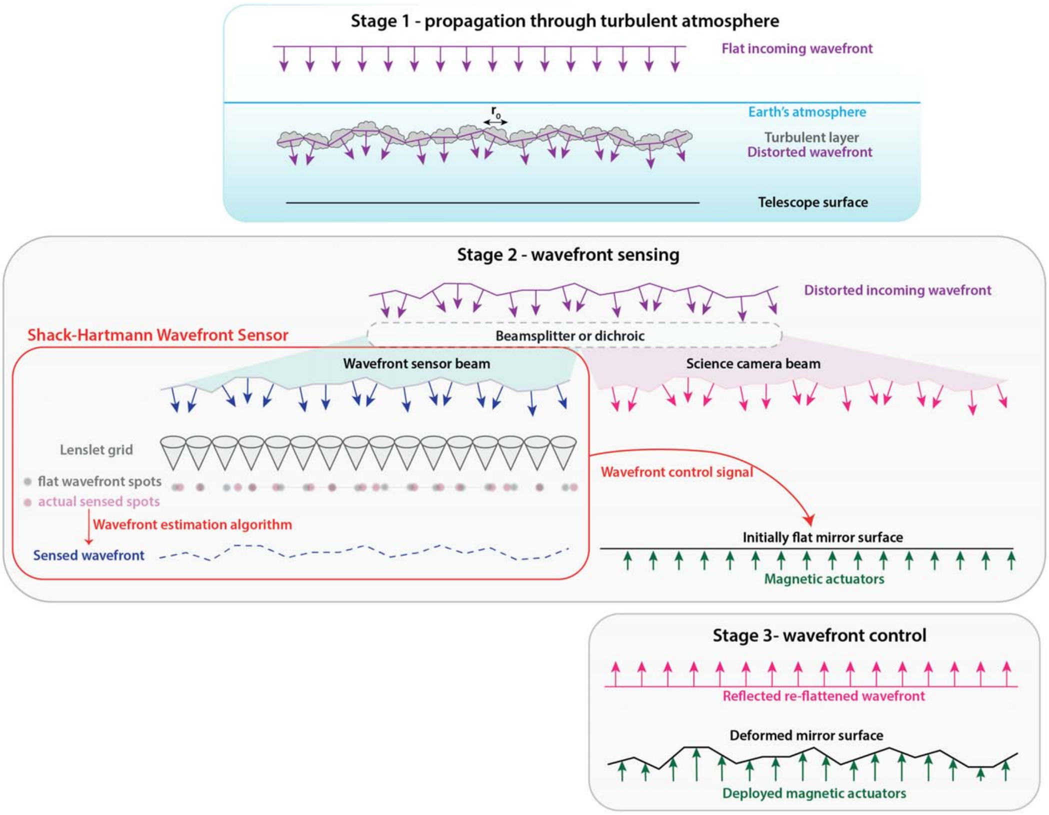

Figure 1.3: Simplified picture of the AO pipeline [5]. Stage 1 : the wavefront of light coming from space is distorted when passing through the atmosphere. Stage 2 : the wavefront shape is estimated by sampling the latter with a lenslet grid that extracts information about the slope of the wavefront. Stage 3: the wavefront is flattened thanks to deformable mirrors that are calibrated to fit the estimated wavefront shape.

However, due to the imperfection of optical devices, some residual turbulence remains even after those additional corrections, which induces an instrumental PSF shape that deviates from an Airy pattern. This involves the presence of the so-called speckles on the AO-corrected image that come from both the uncorrected atmospheric aberrations and the inner instrumental aberrations [5, 11, 12]. The first source of speckles rapidly evolves with the pressure and the temperature of the atmosphere and creates a diffuse halo of starlight around the coronagraph mask. The second source yields the so-called quasi-static speckles that have a time life from some minutes to several hours and a shape of the same order as an exoplanet (see figure 1.4). It is thus crucial to remove them to avoid false detection. This requires to use advanced post-processing techniques as presented in the next section.

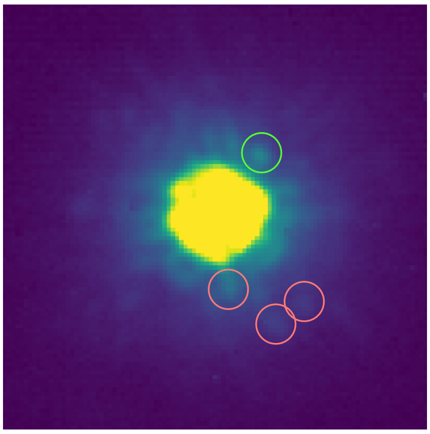

Figure 1.4: Example of quasi-static speckles (in light red) in an AO-corrected image of beta Pictoris by the VLT/NaCo [20]. The famous β Pic b is encircled in green.

1.4 Differential Imaging

Because of limitation of AO to remove atmospheric aberrations, post-processing of AO-corrected images is needed to enhance the contrast ratio between the host star and the exoplanet and thus the detectability of the latter. The most widespread class of post-processing methods is called differential imaging. This technique exploits the difference in terms of geometric variety with respect to a certain physical quantity between the signal of interest and the background containing the speckles.

Typically, this uses data cubes $\boldsymbol{Y} \in \mathbb{R}^{K \times N^2}$ with K frames of $N \times N$ pixels coming from AO-corrected images that are stacked and flattened in rows. In particular, we will cover Angular Differential Imaging (ADI) and Spectral Differential Imaging (SDI) which both allow to dissociate the circumstellar signal from the speckles by using the fact that one of them "moves through frames" while the other one remains quasi-static (see figure 1.5).

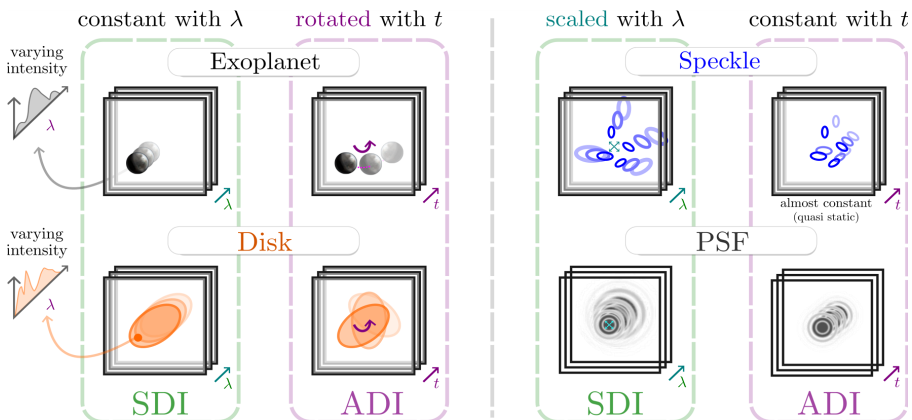

Figure 1.5: Geometric transformations experienced by different components of the data cube with respect to the wavelength λ and the time t. In ADI, exoplanets and disks both rotate through temporal frames while they remain static along the spectral axis. In contrast, quasi-static speckles and the PSF both remain static along the time axis while they scale with the wavelength.

Remark: For the rest of the document, disks will be no longer considered. We will thus use data cubes $\boldsymbol{Y}$ free of circumstellar disks. Consequently, a circumstellar signal will designate a set of exoplanets.

1.4.1 Angular Differential Imaging (ADI)

The first type of differential imaging, called Angular Differential Imaging, was proposed for the first time in 2006 by Marois et al. [21]. Given a data cube $\boldsymbol{Y} \in \mathbb{R}^{T\times N^2}$ constituting of T temporal frames captured during one observation campaign, ADI leverages the Earth rotation to make the circumstellar signal rotating around the host star, while maintaining the background of speckles fixed. The geometric transformation allowing to dissociate the signal from the background is thus a rotation of each frame with respect to a certain angle called parallactic angle. We denote it Rotate(.) in this section. Its formal definition will be given in section 2.1.2 of chapter 2.

Given a circumstellar signal $\boldsymbol{S} \in \mathbb{R}^{T\times N^2}$ (neglecting effects of the instrument) of constant flux and constant position along the time axis, the data cube $\boldsymbol{Y}$ is then modeled as the contribution of three terms : the speckle pattern $\boldsymbol{Y}_\star$ containing speckles from both the instrumental and non-corrected atmospheric aberrations, the rotating signal Rotate($\boldsymbol{S}$) and some instrumental noise $\boldsymbol{\eta}$ (see figure 1.6).

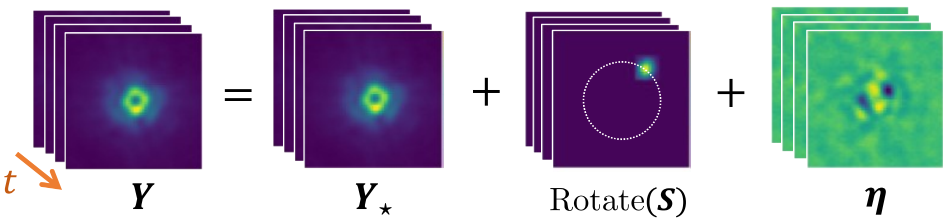

Figure 1.6: ADI model : background + signal + noise (images of VLT/NaCo beta Pictoris from [8]).

Since the speckle pattern is almost static, the simplest method to extract the signal of interest is called classical ADI (c-ADI) and consists in averaging each frame of the data cube to obtain the background $\boldsymbol{Y}_\star$. Then, one can reveal the exoplanets by subtracting the data cube with the estimated background, derotating each frame by the corresponding parallactic angle and finally averaging the result (or taking the median).

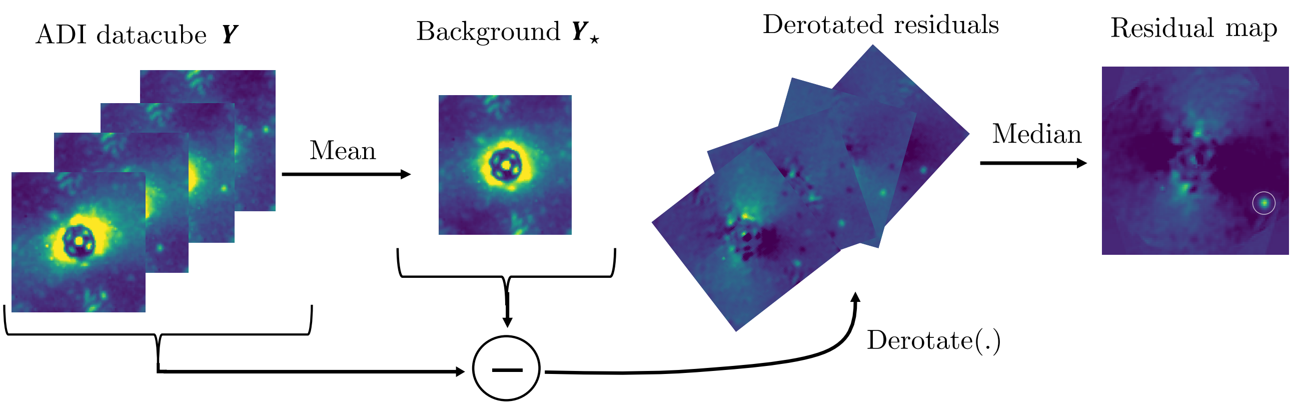

Figure 1.7: Classical ADI (c-ADI) applied on four temporal frames of the VLT/SPHERE-IFS-2 data cube of section 2.2 with a fake planet.

More precisely, c-ADI follows three steps illustrated in figure 1.7 from left to right :

(i) Background model: $\boldsymbol{Y}_\star \approx \text{c-ADI}(\boldsymbol{Y}) = \frac{1}{T}\mathbf{1}_T\mathbf{1}_T^\top\boldsymbol{Y}$.

(ii) Subtraction and derotation: $\widetilde{\boldsymbol{Y}} = \text{Derotate}(\boldsymbol{Y} - \boldsymbol{Y}_\star)$

(iii) Residual map (revealing the planet): $\boldsymbol{R} = \frac{1}{T}\mathbf{1}_T^\top \widetilde{\boldsymbol{Y}}$ (or median along the time axis).

1.4.2 Spectral Differential Imaging (SDI)

The second technique, called Spectral Differential Imaging, provides data cubes $\boldsymbol{Y} \in \mathbb{R}^{W\times N^2}$ of W spectral frames. In SDI, observation diversity is achieved through wavelength variations: the spatial configuration of the circumstellar signal is constant, while speckles are dilated by a wavelength-dependent factor (the physical quantity being now the wavelength λ) [5, 22]. The geometric transformation inducing diversity is then formalized as a scaling operator, denoted by Scale(.), which applies a wavelength-dependent scale factor on each frame of a data cube. This operator will be formally defined in section 2.1.2 of chapter 2.

At the origin, SDI was called Simultaneous Differential Imaging and was first introduced in the work of Racine et al. in 1999 [23]. This name highlights an interesting property of spectral images. Those are taken simultaneously meaning that the speckle pattern $\boldsymbol{Y}_\star$ is almost identical over all the spectral channels up to a scale factor and a flux factor. Hence, contrary to ADI, $\boldsymbol{Y}_\star$ is no longer static but speckles are highly correlated along the spectral axis. On the other hand, the circumstellar signal $\boldsymbol{S} \in \mathbb{R}^{W\times N^2}$ remains fixed through frames.

In order to recover the same setup as in c-ADI, we generally apply a scaling operator on $\boldsymbol{Y}$ to get $\boldsymbol{Z} := \mathrm{Scale}(\boldsymbol{Y})$, leading to the following model :

$$\boldsymbol{Z} = \boldsymbol{Z}_\star + \mathrm{Scale}(\boldsymbol{S}) + \boldsymbol{\eta}. \tag{1.3}$$

As we compensate the dilation of the speckle pattern by scaling the data cube, the resulting speckle pattern $\boldsymbol{Z}_\star$ will become static while the planet will move in the radial direction. This yields a model that is almost identical as ADI, where the rotation operator has been replaced by the scaling.

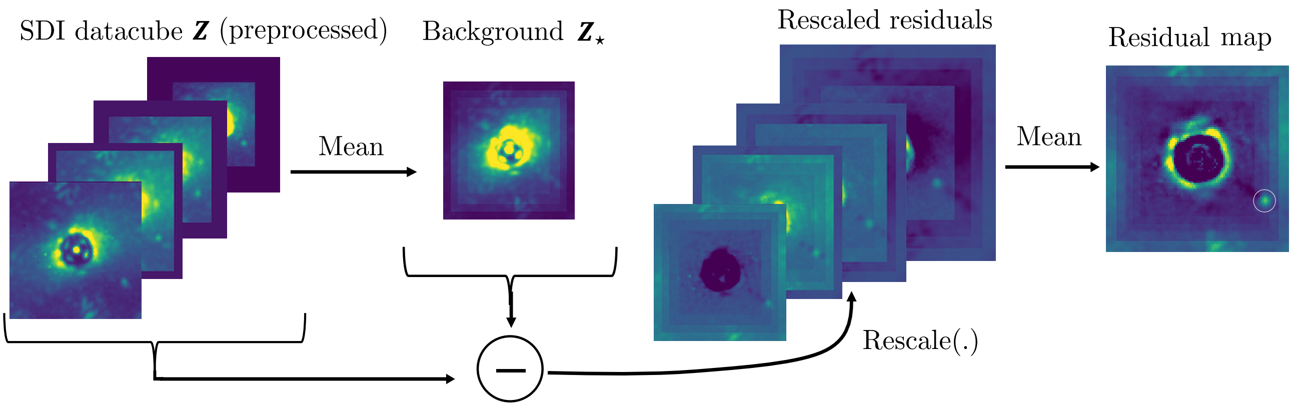

Figure 1.8: Classical SDI (c-SDI) applied on four spectral frames of the preprocessed VLT/SPHERE-IFS-2 data cube of section 2.2 with a fake planet.

Hence, a similar method to classical ADI, called classical SDI (c-SDI), can be applied to recover the signal of interest (see figure 1.8) :

(i) Preprocessing: $\boldsymbol{Z} = \mathrm{Scale}(\boldsymbol{Y})$

(ii) Background model: $\boldsymbol{Z}_\star \approx \text{c-SDI}(\boldsymbol{Z}) = \frac{1}{W}\mathbf{1}_W\mathbf{1}_W^\top \boldsymbol{Z}$

(iii) Subtraction and rescaling: $\widetilde{\boldsymbol{Y}} = \mathrm{Rescale}(\boldsymbol{Z} - \boldsymbol{Z}_\star)$

(iv) Residual map (revealing the planet): $\boldsymbol{R} = \frac{1}{W}\mathbf{1}_W^\top \widetilde{\boldsymbol{Y}}$

Note that contrary to ADI, we do not take the median at the end of c-SDI since the flux of a circumstellar signal may be different from one frame to another.

1.4.3 Angular and Spectral differential imaging (ASDI)

In order to exploit both varieties coming from ADI and SDI data, it is possible to combine both methods in an unique model using 4-D data cube $W \times T \times N \times N$ so that each spectral frame is preprocessed with the same scale factor at each time. This gives the picture of figure 1.9.

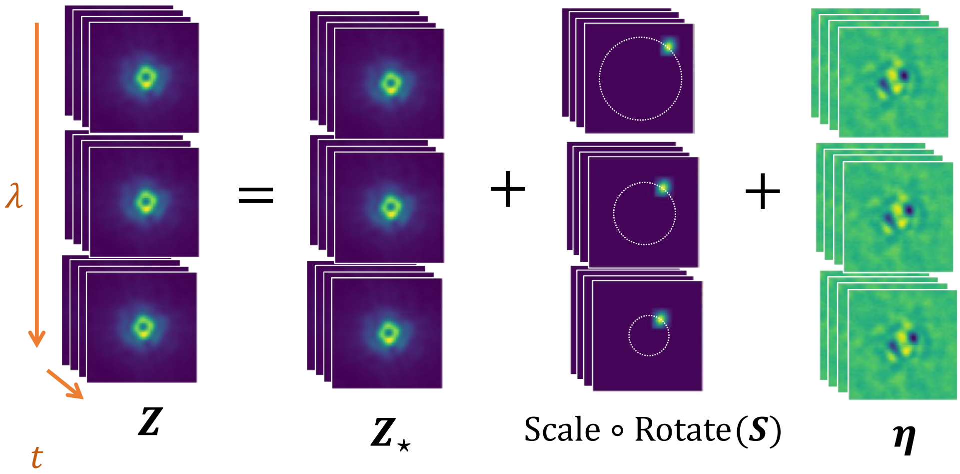

Figure 1.9: ASDI model. Definitions of Rotate(.) and Scale(.) are extended to ASDI data cubes according to section 2.1.2 of chapter 2.

The idea is to apply successively an ADI algorithm for each wavelength and a SDI algorithm for each time (or conversely), or even a complete ASDI algorithm to obtain a background model that we subtract from the original preprocessed data cube [24]. Then, after rescaling and derotation, both time and spectral frames are combined and averaged together to reveal the signal of interest. More precisely, storing the data cube in a matrix $\boldsymbol{Y} \in \mathbb{R}^{WT \times N^2}$ and given an ASDI algorithm building a background model, one can apply the following steps :

(i) Preprocessing: $\boldsymbol{Z} = \text{Scale}(\boldsymbol{Y})$

(ii) Background model: $\boldsymbol{Z}_\star = \text{ASDI}(\boldsymbol{Z})$

(iii) Subtraction and rescaling: $\widetilde{\boldsymbol{Y}} = \text{Derotate} \circ \text{Rescale}(\boldsymbol{Z} - \boldsymbol{Z}_\star)$

(iv) Residual map (revealing the planet): $\boldsymbol{R} = \frac{1}{WT} \mathbf{1}_{WT}^\top \widetilde{\boldsymbol{Y}}$

where ASDI($\boldsymbol{Z}$) can be decomposed in two steps (or not) depending on the user's wishes by applying successively an ADI (resp. SDI) algorithm on each spectral (resp. temporal) channel and then directly a SDI (resp. ADI) algorithm on each temporal (resp. spectral) channel on the output of the first pass.

1.5 State of the art

While c-ADI and c-SDI allow to enhance the contrast in a AO-corrected image, they still suffer from limited performance. The goal of this section is to describe more advanced techniques mainly designed for ADI that try to reach a better contrast on post-processed images than the one obtained with c-ADI and c-SDI.

1.5.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) designates a method, typically used in data sciences, that aims at reducing the number of dimensions of a dataset, by projecting the datapoints on a certain number of orthogonal principal axis obtained by maximizing the variance of the data along those axes. PCA is directly related to the Singular Value Decomposition (SVD) and low-rank approximation of matrices through the Eckart-Young-Mirsky theorem [25, 26].

Historically, the use of this method for ADI was introduced under two names : PCA [27] and KLIP [28]. These can be interpreted as modeling the speckle pattern $\boldsymbol{Y}_\star$ by a low-rank matrix $\boldsymbol{L}$ obtained by projecting the data cube $\boldsymbol{Y}$ on the r first principal axes of PCA, or equivalently by doing a truncated SVD.

Following this idea, one can replace the background model of c-ADI by

$$\begin{aligned} \boldsymbol{Y}_\star \approx \mathcal{H}_r^{\mathrm{SVD}}(\boldsymbol{Y}) \;&:=\; \arg\min_{\boldsymbol{L}} \|\boldsymbol{Y}-\boldsymbol{L}\|_2^2 \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r \\ &=\; \boldsymbol{U}_r \boldsymbol{\Sigma}_r \boldsymbol{V}_r^\top . \end{aligned} \tag{1.4}$$

The matrices $\boldsymbol{U}_r$ and $\boldsymbol{V}_r$ are the left and right r first singular vectors while $\boldsymbol{\Sigma}_r$ is the diagonal matrix with the r first singular values in decreasing order. Intuitively, since the speckle pattern is almost static through frames, we expect that its row space will be low-dimensional. In particular, if the speckles were perfectly static, the row space would be spanned by only one frame since they are all identical. To illustrate this intuition, we provide in figure 1.10 a comparison between a true speckle pattern and its rank-one approximation.

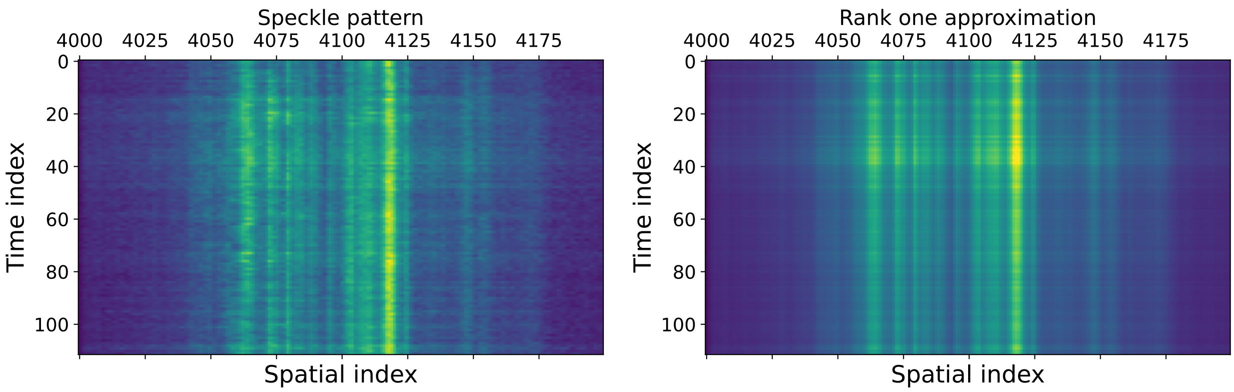

Figure 1.10: Left: speckle pattern matrix obtained as a data cube without planet. This is here the empty cube SPHERE-IFS-2 of chapter 2 section 2.2 with $W = 1$, $T = 112$ and $N = 200$. Right: best rank-one approximation of the speckle pattern matrix. The rank one truncated SVD is performed over the whole matrix even if we only show here 200 spatial indexes among the 40000.

To extend PCA to SDI datacubes, one can replace above $\boldsymbol{Y}$ by the preprocessed data cube $\boldsymbol{Z}$ and $\boldsymbol{Y}_\star$ by $\boldsymbol{Z}_\star$. Regarding ASDI, several variants are possible that we will call single step ASDI and double step ASDI. While double ASDI consists in applying successively PCA-ADI and PCA-SDI (or conversely) as explained in 1.4.3, single step ASDI concatenates spectral and temporal dimensions in a single dimension and applies a truncated SVD to obtain the background $\boldsymbol{Z}_\star = \mathcal{H}_r^{\mathrm{SVD}}(\boldsymbol{Z})$.

PCA-ASDI was explored by both [22] and [24]. However, conclusions seem to be contradictory: the first authors recommend to use single step ASDI whereas the second ones conclude that double ASDI performs better. According to [22], the difference might be due to the implementation of those methods, the datases used, the integration time as well as the available spectral channels.

While PCA is simple to apply and still provides a good model of the speckle pattern, it has limited performance. Indeed, in practice, a large proportion of the circumstellar signal also stays in the r first components, leading to underestimation of the flux called "auto-subtractions" in the literature.

1.5.2 Low-rank-plus-sparse models

In order to overcome limitations of PCA, a family of methods, called low-rank-plus-sparse models, tends to take into account the signal component in the resolution of the signal detection problem following a greedy approach or using the sparsity principle. The latter assumes that the signal of interest has a specific structure that can be explained with a few coefficients in a well-chosen basis of functions, such as the Fourier basis, a wavelet basis, or a combination of bases yielding a dictionary of basis functions called atoms. Applying this principle generally gives a sparsity constraint involving a ℓ_0 pseudo-norm on the parameters explaining the signal.

Low-rank-plus-sparse methods have been developed for exoplanet detection in ADI data cubes and yield models of the form

$$\boldsymbol{Y} = \boldsymbol{L} + \Phi(\boldsymbol{S}) + \boldsymbol{\eta} \quad \text{s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r, \tag{1.5}$$

with possibly additional constraints on $\boldsymbol{S}$.

Among them, one can cite LLSG [29], LRPT [8], AMAT [9] and MAYO [10]. The differences between those methods are the meaning of the parameter $\boldsymbol{S}$, the operator Φ and the noise term $\boldsymbol{\eta}$, but also the optimization algorithm used to recover the parameters of (1.5) and the priors on $\boldsymbol{S}$ (i.e the basis chosen in the sparsity principle). Table 1.1 provides information about how each method differs from each other.

Method	Parameter $\boldsymbol{S}$	Operator Φ	Noise model $\boldsymbol{\eta}$	Category
LLSG	convolved rotating signal	$\mathcal{I}$	gaussian	Sparsity principle + Non-convex
LRPT	exoplanets only	Convolve $\circ$ Rotate(.)	gaussian	Sparsity principle + Non-convex
AMAT	one single exoplanet	Convolve $\circ$ Rotate(.)	gaussian or laplacian	Greedy + Non-convex
MAYO	exoplanets + disks	Convolve $\circ$ Rotate(.)	gaussian + laplacian	Sparsity principle + Convex relaxation

Table 1.1: Comparison between low-rank-plus-sparse models. Convolve(.) denotes here the convolution with the (non-coronographic) instrumental PSF with respect to the spatial dimension. $\mathcal{I}$ is the identity operator.

Local Low-rank plus Sparse Gaussian-noise model (LLSG)

The first method called LLSG was inspired by robust PCA which has been proposed in the computer vision literature to model a video as the superposition of a low-rank background and a sparse foreground. However, when transposing this idea to ADI, [29] had to make robust PCA more flexible. Indeed, while the sparse component captures the rotating circumstellar signal

$\boldsymbol{S} \in \mathbb{R}^{T \times N^2}$, the low-rank component cannot completely capture the background $\boldsymbol{Y}_\star \in \mathbb{R}^{T \times N^2}$ since speckles are quasi-static, meaning that $\boldsymbol{Y}_\star$ is never exactly low-rank.

It was thus proposed to model the data cube as the sum of three terms: $\boldsymbol{Y} = \boldsymbol{L} + \boldsymbol{S} + \boldsymbol{\eta}$, where $\boldsymbol{\eta}$ is a gaussian noise. The algorithm allowing to recover the parameters $\boldsymbol{L}$ and $\boldsymbol{S}$, called GoDec, consists in minimizing the squared ℓ_2-norm of the residuals, i.e. solving

$$\min_{\boldsymbol{L} \in \mathbb{R}^{T \times N^2}, \boldsymbol{S} \in \mathbb{R}^{T \times N^2}} \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{L} - \boldsymbol{S}\|_2^2 \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r, \|\boldsymbol{S}\|_0 \leq k \tag{1.6}$$

by alternating the minimization over $\boldsymbol{L}$ and $\boldsymbol{S}$ successively. An alternative version of GoDec performs a convex relaxation of the sparsity constraint by replacing the ℓ_0 pseudo-norm by a ℓ_1-norm, yielding the following optimization problem :

$$\min_{\boldsymbol{L} \in \mathbb{R}^{T \times N^2}, \boldsymbol{S} \in \mathbb{R}^{T \times N^2}} \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{L} - \boldsymbol{S}\|_2^2 + \alpha \|\boldsymbol{S}\|_1 \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r. \tag{1.7}$$

The constant $\alpha > 0$ is a regularization parameter allowing to weight the sparsity constraint (see chapter 3). Again, alternating minimization is used to solve (1.7). The i^{th} iteration of LLSG is given by

$$\begin{aligned}
\boldsymbol{L}^{(i+1)} &= \arg\min_{\boldsymbol{L}:\operatorname{rank}(\boldsymbol{L}) \leq r} \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{L} - \boldsymbol{S}^{(i)}\|_2^2 + \alpha \|\boldsymbol{S}^{(i)}\|_1 = \mathcal{H}_r^{\text{SVD}}(\boldsymbol{Y} - \boldsymbol{S}^{(i)}) \\
\boldsymbol{S}^{(i+1)} &= \arg\min_{\boldsymbol{S}} \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{L}^{(i+1)} - \boldsymbol{S}\|_2^2 + \alpha \|\boldsymbol{S}\|_1 = \operatorname{prox}_{\alpha\|.\|_1}(\boldsymbol{Y} - \boldsymbol{L}^{(i+1)}),
\end{aligned}$$

where $\operatorname{prox}_{\alpha\|.\|_1}(.)$ is the proximal operator that will be defined in chapter 3.

Finally, to account for the local structure of the data cube, the authors locally split the latter in patches of quadrant annuli of width $\frac{2\lambda}{D}$ and apply the LLSG algorithm on each of them with an adaptive regularization parameter at each iteration. The latter is chosen as the square root of the mean absolute deviation of the entries of $\boldsymbol{S}^{(i)}$ associated to each patch.

Low-Rank Plus sparse Trajectory decomposition (LRPT)

The second method, called LRPT, introduces an additional prior information in the LLSG algorithm. The latter indeed does not take into account both the rotation of the circumstellar signal and the response of the instrument modeled by the PSF. By introducing this information in (1.6), LRPT extends the GoDec algorithm as

$$\min_{\boldsymbol{L} \in \mathbb{R}^{T \times N^2}, \boldsymbol{S} \in \mathbb{R}^{N^2}} \frac{1}{2} \|\boldsymbol{Y} - \boldsymbol{L} - \Phi(\boldsymbol{S})\|_2^2 \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r, \|\boldsymbol{S}\|_0 \leq k, \tag{1.8}$$

where $\Phi(.) = \text{Convolve} \circ \text{Rotate}(.)$ applies both the rotation operator and the convolution with the instrumental PSF. The authors proposed to solve (1.8) by using a modified version of the normalized alternating hard thresholding algorithm whose details can be found in [8, 30].

Alternating Minimization Algorithm with Trajectories (AMAT)

Instead of solving the exoplanet detection problem in one run, the authors of AMAT decompose this problem in N^2 subproblems. For each subproblem, a circular trajectory g tracking a single rotating planet is drawn and stored in a matrix $\boldsymbol{P}_g \in \mathbb{R}^{T \times N^2}$. This matrix is called "the planet signature" and is build by placing a PSF along the circular trajectory g. Then, the following minimization problem is solved

$$\min_{\boldsymbol{L} \in \mathbb{R}^{T \times N^2}, a_g \in \mathbb{R}} \|\boldsymbol{Y} - \boldsymbol{L} - a_g \boldsymbol{P}_g\| \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r, \tag{1.9}$$

by an alternating minimization

$$\begin{aligned} \boldsymbol{L}^{(i+1)} &= \arg\min_{\boldsymbol{L}:\text{rank}(\boldsymbol{L})\leq r} \|\boldsymbol{Y} - \boldsymbol{L} - a_g^{(i)}\boldsymbol{P}_g\| \\ a_g^{(i+1)} &= \arg\min_{a_g} \|\boldsymbol{Y} - \boldsymbol{L}^{(i+1)} - a_g\boldsymbol{P}_g\|. \end{aligned}$$

The norm $\|.\|$ can be either a ℓ_2-norm (if gaussian speckle noise) or a ℓ_1-norm (if laplacian speckle noise). At the end of the algorithm, the circumstellar signal is reconstructed by gathering all the flux values a_g computed in each subproblem.

While a closed form of $\boldsymbol{L}^{(i+1)}$ and $a_g^{(i+1)}$ is available for the ℓ_2-norm, it was observed that the ℓ_1-norm minimization provides better results. In fact, [31] has shown that the non-static part of the speckle pattern is better modeled by a heavy tail distribution similar to the one of a laplacian distribution (see chapter 4 for more information), which can explain why ℓ_1 minimization performs better.

Contrary to LLSG and LRPT, AMAT is a greedy algorithm since planets are found one by one by solving N^2 instances of (1.9), that is, one instance by possible pixel locations of an exoplanet in the data cube. Consequently, applying this algorithm can take several hours, even if parallelization is used.

Morphological components analysis pipeline for circumstellar discs and exoplanets imaging in the near-infrared (MAYO)

The main motivation of MAYO is to overcome several bad effects of PCA such as auto-substraction creating negative flux in some area of the cube and underestimation of the flux of the circumstellar signal. It combines greedy and inverse problem approaches to solve the detection problem using ADI for both disks and exoplanets. While this algorithm is too general for our purpose of exoplanets detection, it introduces several interesting ideas that are important to mention.

Firstly, contrary to the previous low-rank-plus-sparse models, MAYO performs a convex relaxation of the low-rank constraint on the background model $\boldsymbol{L}$, avoiding to rely on alternating non-convex optimization. In the signal processing community, it is common to replace the rank operator by a nuclear norm, which is defined as the ℓ_1-norm of the vector storing the singular values of $\boldsymbol{L}$. However, in MAYO, they proceed differently. Instead, they build a low-rank matrix $\hat{\boldsymbol{L}}$ applying a fixed-point iterative algorithm named GreeDS that was shown to still provide good background models, especially when disks lie in the data cube [10]. Then, they replace the low-rank constraint $\text{rank}(\boldsymbol{L}) \leq r$ by imposing $\boldsymbol{L}$ to lie in the span of temporal eigenvectors of $\hat{\boldsymbol{L}}$, that is the column space of $\hat{\boldsymbol{L}}$. More precisely, they first compute its truncated SVD $\mathcal{H}_r^{\text{SVD}}(\hat{\boldsymbol{L}}) = \hat{\boldsymbol{U}}_r\hat{\boldsymbol{\Sigma}}_r\hat{\boldsymbol{V}}_r^\top$ and then build the set of matrices $\hat{\mathcal{P}}^{(r)} = \{\boldsymbol{A} \in \mathbb{R}^{T\times N^2} : \hat{\boldsymbol{U}}_r\hat{\boldsymbol{U}}_r^\top\boldsymbol{A} = \boldsymbol{A}\}$. Finally, they impose that $\boldsymbol{L} \in \hat{\mathcal{P}}^{(r)}$.

Secondly, MAYO explicitly drops from the cost function the close neighborhood of the star at small angular separation, where the instrument response becomes non-shift invariant. This is done by inserting a mask operator $\mathcal{M}_\Omega(.)$ in the cost function.

Those two ideas will inspire us to write our ASDI models in chapter 4.

Chapter 2

Numerical tools and datasets

This chapter is composed of two parts. First, useful commands provided by the Vortex Image Processing (VIP) package in Python [32] are described. Subsequently, we provide details about the VLT/SPHERE-IFS datasets that were used to test the post-processing algorithms.

2.1 The VIP toolbox

2.1.1 Normalize PSF

When we model the behavior of an instrument as a convolution between its PSF and the point source, we would like to avoid that this convolution modifies the flux of the source. Hence, given a PSF provided with the data cube, we always should first normalize the PSF so that its ℓ_1-norm restricted to a resolution element, defined as a ball of diameter λ/D, is equal to 1. This enables to be unchanged the intensity integrated on a resolution element before and after convolution. The method `vip.fm.normalize_psf` of the VIP toolbox allows us to perform this ℓ_1-normalization.

A second functionality of `vip.fm.normalize_psf` is to fit a PSF model (e.g. gaussian, Airy disk) on the computed normalized PSF, so that we can estimate the full width of half maximum (FWHM) of the instrumental PSF. In practice, it is this quantity that is used to define the diameter of a resolution element since it is roughly equal to λ/D. Moreover, it is convenient to define the diameter of a resolution element independently of the wavelength. In this case, we will denote it as $\bar{\lambda}/D$, which designates the mean FWHM of the PSF over the spectral channels of the ASDI data cube.

2.1.2 The geometric operators

Rotation operator

In chapter 1, we introduced the rotation operator Rotate(.) which rotates each temporal frame of an ASDI data cube $\boldsymbol{Y} \in \mathbb{R}^{WT \times N^2}$ with respect to the parallactic angles. We here formalize it via the linear operator $\mathcal{Q}_{\boldsymbol{\alpha}}(.)$. Given a continuous light intensity field $F : \mathbb{R} \times \mathbb{R}^2 \mapsto \mathbb{R}$, we define

$$\mathcal{Q}_{\boldsymbol{\alpha}} : \mathbb{R}^{W \times N^2} \mapsto \mathbb{R}^{WT \times N^2} \equiv \text{ discrete version of } F(\lambda, \boldsymbol{p}) \mapsto F(\lambda, \boldsymbol{Q}_{\alpha(t)}\boldsymbol{p}), \tag{2.1}$$

where $\boldsymbol{p} \in \mathbb{R}^2$ is the position of a pixel (in a "continuous" image) and $\boldsymbol{Q}_{\alpha(t)}$ is a 2×2 rotation matrix of angle $\alpha(t)$. The vector $\boldsymbol{\alpha} = [\alpha(t)]_{t=1}^{T}$ are the parallactic angles associated to each temporal frame. Hence, $\mathcal{Q}_{\boldsymbol{\alpha}}(.)$ is equivalent to duplicate T times a SDI data cube and apply the rotation operator Rotate(.) on the resulting ASDI data cube. The inverse operator is defined as

$$\mathcal{Q}_{\boldsymbol{\alpha}}^{-1} : \mathbb{R}^{WT\times N^2} \mapsto \mathbb{R}^{W\times N^2} \equiv \text{ discrete version of } F(\lambda, \boldsymbol{p}) \mapsto \frac{1}{T}\sum_{t'=1}^{T} F(\lambda, \boldsymbol{Q}_{-\alpha(t')}\boldsymbol{p}). \tag{2.2}$$

In practice, the command `vip.preproc.cube_derotate` is used to evaluate the rotation operator (once per spectral channel). This performs the rotation with respect to the center of the frame with the function `cv2.warpAffine` of `opencv`[1]. This function allows to apply an affine transformation (here the rotation matrix) and to interpolate the grid of pixels via a Lanczos interpolation [33].

Scaling operator

The second operator previously denoted by Scale(.) is also a linear operator, which we denote by $\mathcal{D}_{\boldsymbol{s}}(.)$. For a certain integer $n > N$, we define

$$\mathcal{D}_{\boldsymbol{s}} : \mathbb{R}^{WT\times N^2} \mapsto \mathbb{R}^{WT\times n^2} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto F(\lambda, t, s(\lambda)^{-1}\boldsymbol{p}). \tag{2.3}$$

The inverse operator, previously denoted by Rescale(.), is defined as

$$\mathcal{D}_{\boldsymbol{s}}^{-1} : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{WT\times N^2} \equiv \text{ discrete version of } F(\lambda, t, \boldsymbol{p}) \mapsto F(\lambda, t, s(\lambda)\boldsymbol{p}). \tag{2.4}$$

The vector $\boldsymbol{s} = [s(\lambda_l)]_{l=1}^{W}$ contains the scale factors with respect to each spectral frame l captured at wavelength λ_l (see section 2.1.4).

In practice, the command `vip.preproc.cube_rescaling_wavelength` is used to evaluate $\mathcal{D}_{\boldsymbol{s}}(.)$ (once per temporal frame). Similarly to the rotation operator, VIP here again performs a Lanczos interpolation via `opencv`. By convention, $s(\lambda) \geq 1$ and $n > N$ in order to apply an up-scaling. This prevents the loss of information due to downsampling when downscaling images. More precisely, a zero-padding of size $(n - N)/2$ is applied at each border of the image so that, by successively applying $\mathcal{D}_{\boldsymbol{s}}(.)$ and $\mathcal{D}_{\boldsymbol{s}}^{-1}(.)$ on the image, we recover exactly the initial frame (up to some numerical errors).

2.1.3 Inject companion

To assess performance of a new ASDI post-processing algorithm, we need to have access to a ground truth, that is, a data cube that contains planets at known locations and with known fluxes. To this end, it is common to inject an artificial planet in an empty ASDI data cube, and then to post-process it. Repeating this procedure several times allows to build a ground truth library of data cubes that can be used to make statistics about the post-processing algorithm performance.

In real dataset, the flux of a planet varies with the wavelength and is influenced by several physical properties such as its spectrum, its airmass and the transmission of the atmosphere [34]. To simplify our approach, we limit ourselves to planets with constant flux that we inject via the command `vip.fm.cube_inject_companions` of VIP. More precisely, given an empty ASDI data cube $\boldsymbol{Y}_{\text{empty}} \in \mathbb{R}^{WT\times N^2}$, this function injects a planet by convolving a point-like

[1]Recently, a more accurate version of this operator using Fourier interpolation was added in VIP, but it also takes more time to perform the interpolation. Since this operator is evaluated a large number of times in the algorithms designed in this Master's thesis, we will not use this interpolation mode.

source (here of constant flux $a \in \mathbb{R}$) with the instrumental PSF and then applies the rotation operator :

$$\boldsymbol{Y} = \boldsymbol{Y}_{\text{empty}} + a\mathcal{Q}_{\boldsymbol{\alpha}}(\boldsymbol{\varphi}_p). \tag{2.5}$$

The matrix $\boldsymbol{\varphi}_p \in \mathbb{R}^{W \times N^2}$ denotes the instrumental PSF centered in pixel $p \in \{1, \ldots, N^2\}$ of the data cube, corresponding to the position of the planet.

2.1.4 Find scale vector

In order to apply the scaling operator to preprocess an ASDI data cube, one has to determine according to which law the PSF and the speckle pattern scale with λ. To illustrate this scaling behavior, we provide in figure 2.1 an example of PSF measured at different wavelengths.

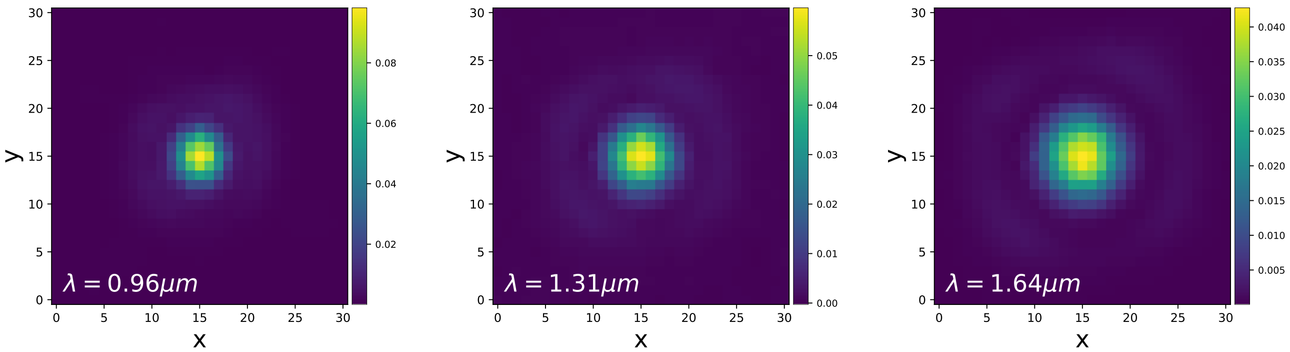

Figure 2.1: Normalized PSF of the VLT/SPHERE-IFS measured at wavelength $\lambda = 0.96$µm, 1.31mµ, 1.64µm.

Let $\varphi(\lambda, \boldsymbol{p})$ be the continuous version of the instrumental PSF evaluated at wavelength $\lambda \in \mathbb{R}$ and at position $\boldsymbol{p} \in \mathbb{R}^2$ in the focal plane. Given an arbitrary wavelength of reference λ_0 and another wavelength λ, the scale factor $s(\lambda)$ relative to λ_0 is formally defined by the following equality :

$$\varphi(\lambda, \boldsymbol{p}) = \varphi(\lambda_0, s(\lambda)\boldsymbol{p}) \quad \forall \boldsymbol{p} \in \mathbb{R}^2. \tag{2.6}$$

In the ASDI literature, it is common to define the scale factor as $\propto 1/\lambda$ [5, 12, 22, 24]. This comes from an important result in diffraction's theory. For a perfectly circular aperture without central obscuration, the PSF draws an Airy pattern that has a close form[2] given by :

$$\varphi_{\text{Airy}}(\lambda, \boldsymbol{p}) = I_0 \left(\frac{2J_1\left(\frac{\pi D}{\lambda}\frac{\|\boldsymbol{p}\|}{R}\right)}{\frac{\pi D}{\lambda}\frac{\|\boldsymbol{p}\|}{R}} \right)^2, \tag{2.7}$$

where $J_1(.)$ is the Bessel function of first kind, R is the distance between the center of the aperture and the point light in the focal plane (see figure 2.2) and $I_0 > 0$ is the maximum value of the PSF attained in $\boldsymbol{p} = \boldsymbol{0}$. Assuming that R is approximately constant and equal to the focal length, one can insert φ_{Airy} in (2.6) and notice that the scale factor satisfies

$$s(\lambda) = \frac{\lambda_0}{\lambda}. \tag{2.8}$$

[2] It is common to define φ_{Airy} using the angular separation θ instead, and to replace $\frac{\|\boldsymbol{p}\|}{R}$ by $\sin\theta$.

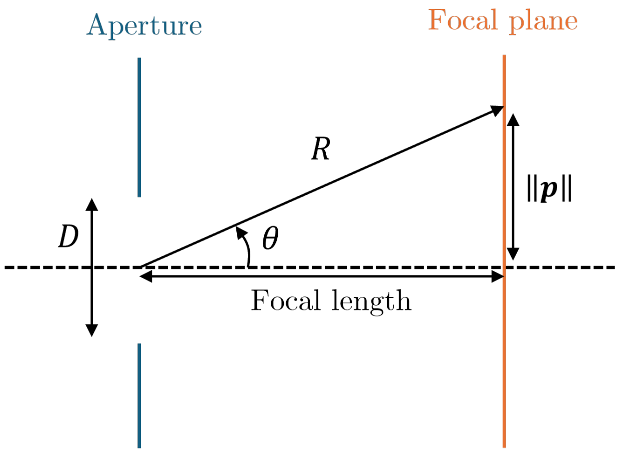

Figure 2.2: Illustration of the quantities involved in (2.7).

In practice, we observe that the scale factor $s(\lambda)$ does not always follow this law. This might be due to imperfections of the adaptive optics (quantified by the so-called Strehl Ratio[3]) or other atmospheric effects, such as the absorption band of atmospheric particles.

A first approach to find $s(\lambda)$ would be to use (2.6). Since the equality may not hold, we would like to find the scale factor minimizing the residuals. However, preprocessing the data aims to freeze the speckle pattern w.r.t λ while moving the planet in the radial direction. Therefore, assuming that $\boldsymbol{Y}_\star$ is the dominant contribution in $\boldsymbol{Y}$, the idea is to fit the scale factor as

$$s^*(\lambda_l) = \arg\min_{s(\lambda_l)} \mathcal{L}(\mathcal{D}_{s(\lambda_W)}\boldsymbol{Y}_{W,:}, \mathcal{D}_{s(\lambda_l)}\boldsymbol{Y}_{l,:}), \quad l = 1, \ldots, W, \tag{2.9}$$

where $\mathcal{D}_{s(\lambda_l)}\boldsymbol{Y}_{l,:} \in \mathbb{R}^{1\times n^2}$ is the scaled data cube $\boldsymbol{Y}_{l,:} \in \mathbb{R}^{1\times N^2}$ with one single spectral frame corresponding to wavelength λ_l and $\mathcal{L}(.,.)$ is a loss function defined by the user such as the ℓ_2-norm, the ℓ_1-norm or the Hubert loss [35]. By convention, we impose that $s^*(\lambda_W) = 1$ for all $l \in \{1, \ldots, W-1\}$. This means that the wavelength of reference is $\lambda_0 = \lambda_W$, which allows to have $n > N$.

Looking at the objective function in (2.9), a close form of its gradient cannot be found since the variable is the parameter of the scaling operator that involves interpolation. This prompts the use of zero-order optimization method to perform the regression.

In VIP, the function `vip.preproc.rescaling.find_scal_vector` is implemented and uses the Nelder-Mead simplex algorithm to solve (2.9). The main issue of this algorithm lies in the lack of convergence guarantees. It was indeed shown that, for a family of 2D functions that contains strictly convex functions, we can ensure that Nelder-Mead converges to a non-stationary point [36].

Another algorithm, called BOBYQA, was thus adopted. It was originally introduced by Powell [37]. It follows a trust region approach and locally interpolates the objective function by using a quadratic model. At each iteration, this model is updated and minimized to give the next iterate. Using this algorithm, available in the Python library py-BOBYQA [38], gives a certain speed up compared to Nelder-Mead implemented in VIP. For the four data cubes in figure 2.3 with $W = 39$ spectral frames (and $T = 1$), py-BOBYQA indeed finds the scale factor in $\sim$8 seconds[4] against $\sim$14 seconds for Nelder-Mead.

[3]The Strehl Ratio is defined as the ratio of a star's observed peak intensity relative to that of its theoretical diffraction-limited peak intensity [5].

[4]The time was measured with a Intel(R) Core(TM) i7-8750H 2.20GHz processor of 8 Go RAM.

Figure 2.3 shows the gap between the theoretical law λ_0/λ and the optimized scale factor $s^*(\lambda)$ found by py-BOBYQA with several loss functions and Nelder-Mead in VIP. We notice that, for the majority of the wavelengths, the scale factor found by each method is almost identical. Moreover, for the data cube SPHERE-IFS-2, we see that Nelder-Mead largely deviates from the law λ_0/λ in the range $[1\mu m, \ldots, 1.1\mu m]$, with a scale factor of almost twice larger than the latter. A such abrupt deviation should not be observed in practice, which suggests that Nelder-Mead did not converge to the optimal solution.

Due to the similarities between the optimized law and the law λ_0/λ (except for the data cube SPHERE-IFS-3), the latter was adopted in this master thesis. Furthermore, this choice allowed to speed up the preprocessing of the data cube, especially during massive injections of fake companions in chapter 6.

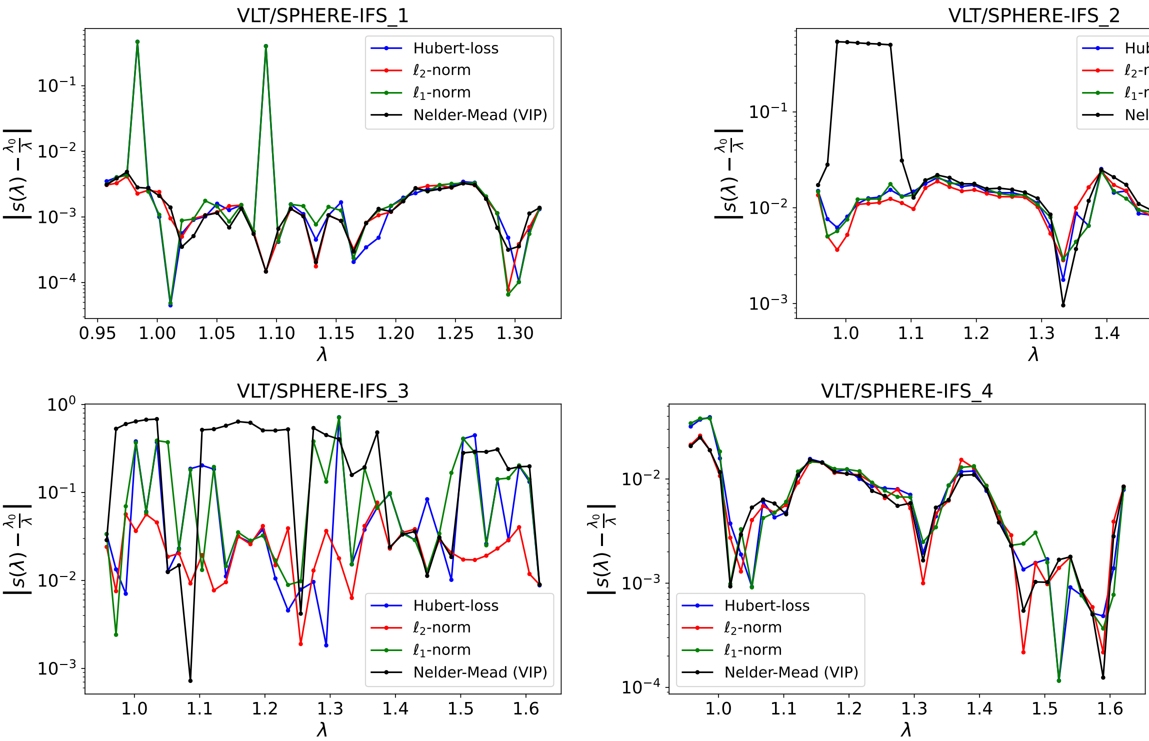

Figure 2.3: Application of py-BOBYQA and Nelder-Mead of VIP on four data cubes of the data challenge (see next section). For each wavelength, the absolute difference between the optimized scale factor and the law λ_0/λ is shown. The wavelength of reference λ_0 is here the one associated to the last frame of the data cube (not shown in those plots).

2.2 The datasets

The datasets used in this master thesis come from the Very Large Telescope (VLT) equipped of the apodized Lyot coronagraph, the Spectro-Polarimetric High-contrast Exoplanet REsearch (SPHERE) imager and an Integral Field Spectrograph (IFS). We refer readers to [39, 40, 41] for detailed descriptions. A data cube coming from those instruments is called in short a VLT/SPHERE-IFS data cube, whose typical shape is $W \times T \times N \times N$ with $W = 39$, $T = \mathcal{O}(100)$ and $N = 200$. The IFS decomposes the light into 39 monochromatic channels in the spectral band $0.96 - 1.64$ µm, corresponding to the Y-band and the J-band of the Near-Infrared.

Figure 2.4 shows a frame of the dataset `sphere_ifs_cube_2.fits` from the first phase of the

Exoplanet Imaging Data Challenge[5], where circumstellar signals have been removed. In this image, we identify several elements that were not modeled in the previous section. In fact, only the small blobs (e) represent some examples of (instrumental) quasi-static speckles. Following the analysis of [42], we will describe the different components constituting the VLT/SPHERE-IFS image.

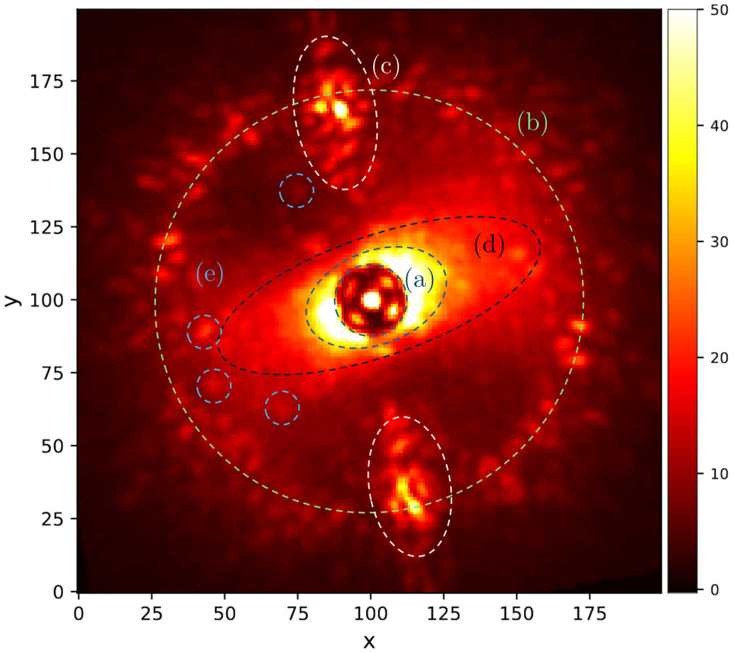

Figure 2.4: One frame of the VLT/SPHERE-IFS at $\lambda = 0.99$ [μm]. Both x and y-axes are in pixel scale, where 4.36 pixels correspond to one resolution element λ/D. Areas (a), (b), (c) and (d) are described in section 2.2.1, 2.2.2 and 2.2.3.

2.2.1 Area (a): coronagraph mask and AO halo

The first area (a) represents the close neighborhood around the star. The dark annulus drawing a cross-like pattern comes from both the starlight masked by the Lyot coronagraph and the cross shape of the VLT, commonly called "the spider". An important characteristic of this area is that the response of the instrument cannot be modeled by a space invariant PSF. This is due to the non-shift invariant effects introduced by the coronagraph. On the other hand, the bright annulus that surrounds the dark area comes from atmospheric aberrations that were not corrected by AO [12, 5] and is thus called the AO halo.

This last introduces difficulties for SDI post-processing algorithms. To illustrate this claim, one can apply c-ADI (with the first spectral channel) and c-SDI (with the first temporal frame) on this VLT/SPHERE-IFS empty cube, which gives the results of figure 2.5. In this figure, we see how the AO halo was not captured by the background model $\boldsymbol{Z}_\star$ provided by c-SDI since it completely stays in the mean residual frame.

In view of this observation, one can formulate three hypotheses: (i) the AO halo does not scale with λ and therefore exactly behaves as a circumstellar signal, (ii) either the wavelength factor is not the same in this specific area as the rest of the data cube, or (iii) the flux variation of the light with respect to the wavelength introduces a bias in the mean residual frame.

In order to validate or reject some of these hypotheses, I examined the variation of the flux integrated in an annulus of width $\frac{\lambda}{D}$ and of varying inner radius that grows linearly with the

[5] https://exoplanet-imaging-challenge.github.io/datasets1/

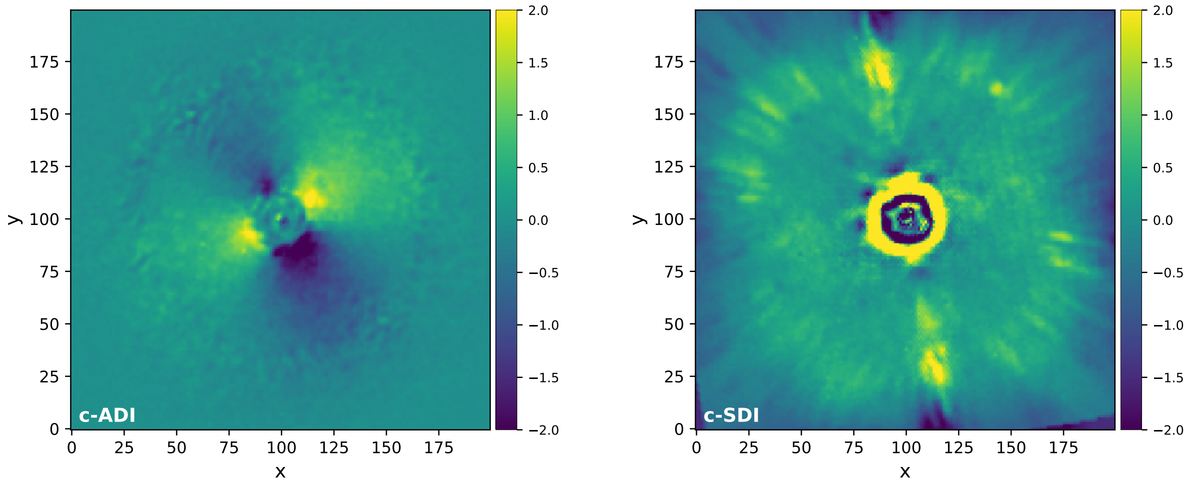

Figure 2.5: Mean residual frame obtained after applying c-ADI (left) and c-SDI (right) on the VLT/SPHERE-IFS data cube. A linear scale factor is used for c-SDI. To make a fair comparison between both methods, I set $T = 20$ for the ADI data cube and $W = 20$ for the SDI data cube, by selecting frames uniformly spaced in time and in wavelength respectively.

wavelength. This construction implicitly assumes a linear law for the scale factor. Interestingly, we observe in figure 2.6 that the flux of each annulus follows the same law with the wavelength, up to a normalizing constant, except in the close neighborhood of the star. In particular, we see on the right plot a certain deviation of the blue curve (coronagraph mask) and the orange curve (AO halo) from the general law, which can explain why a bias is introduced in the background model obtained by c-SDI. The left plot allows to highlight the large difference in terms of flux between the AO halo and the rest of the data cube.

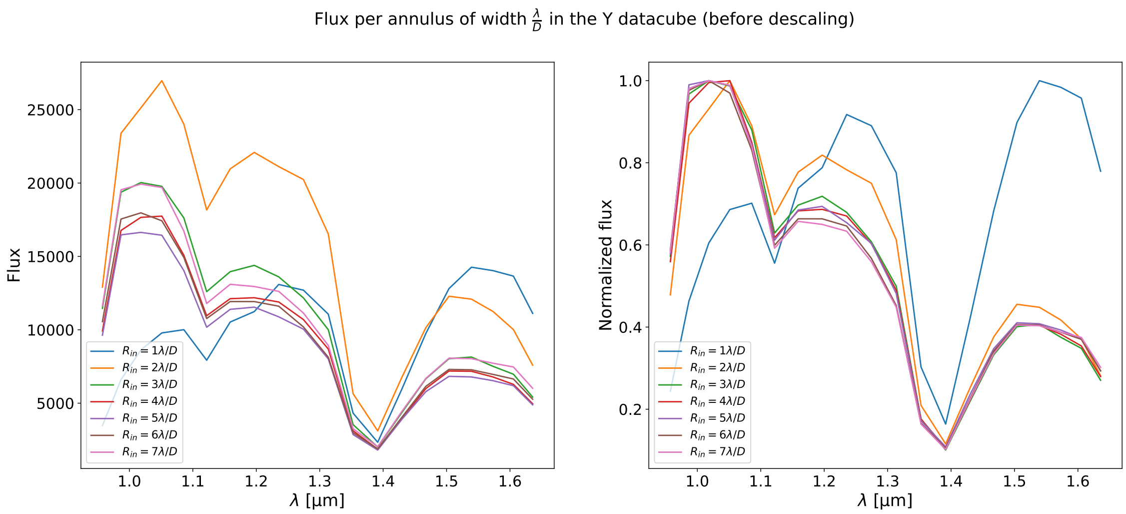

Figure 2.6: Left: flux of the SDI data cube $\boldsymbol{Y}$ summed over an annulus of inner radius R_{in} and of width $\frac{\lambda}{D}$. Right: Normalized flux of an annulus of $\boldsymbol{Y}$ computed, separately for each wavelength, as the sum of all the light intensities in the annulus divided by the maximum value reached by this sum along the spectral axis.

These observations seem to support the third hypothesis, i.e. the introduction of a bias in the flux around the star because of the wavelength dependence of the light intensity. We indeed expect that the background model of c-SDI will be able to capture the normalizing constant used in the right plot of figure 2.6. This normalizing constant allows to overlap each curve, except for the orange and the blue curves that do not fit with this law.

Figure 2.7 indeed confirms this intuition by showing the flux per annulus of the residual cube, once the background model of c-SDI has been removed.

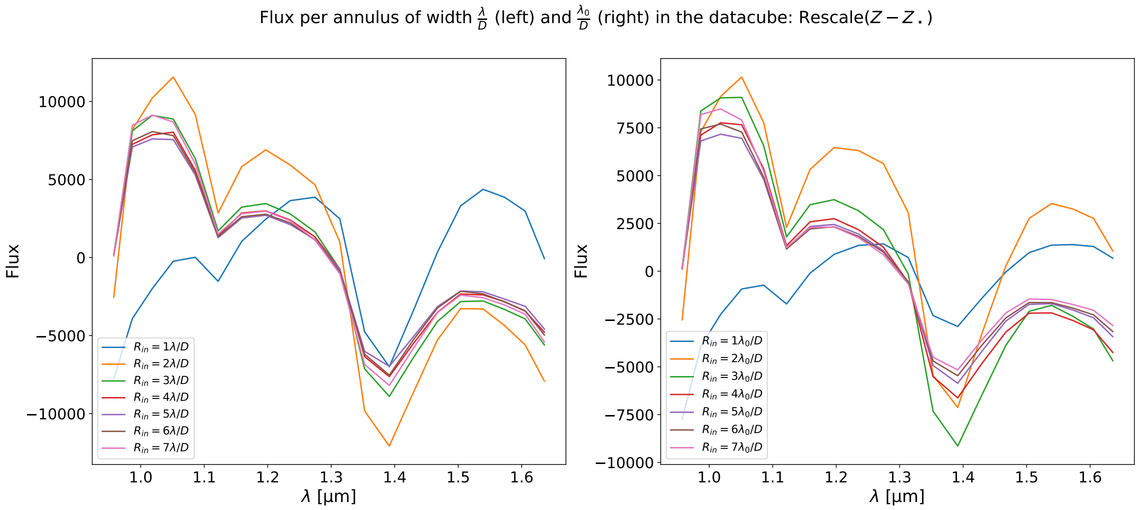

Figure 2.7: Flux of the residual data cube Rescale($\boldsymbol{Z} - \boldsymbol{Z}_\star$) summed over an annulus of inner radius R_{in} of width $\frac{\lambda}{D}$ (left) and width $\frac{\lambda_0}{D}$ (right), where $\lambda_0 = 1.64$ [μm] corresponds to the last channel (convention of VIP).

The left plot was build exactly like the left plot of the previous figure. We see that removing the background model from the data cube has two effects: it centers the flux curves (zero-mean) and accentuates the similarities between them.

On the right, on can observe the flux just before being averaged at the last step of c-SDI. This time, the curves are obtained integrating the flux over an annulus that do not scale with the wavelength. Intuitively, it is like jumping between the curves of the left plot. Since the orange curve is clearly not zero-mean, this explains why the AO halo was not suppressed by c-SDI in figure 2.5.

While this analysis was made for c-SDI, it was observed that this area (a) involves similar difficulties for more advanced SDI post-processing algorithms such as PCA-SDI as shown in figure 2.8.

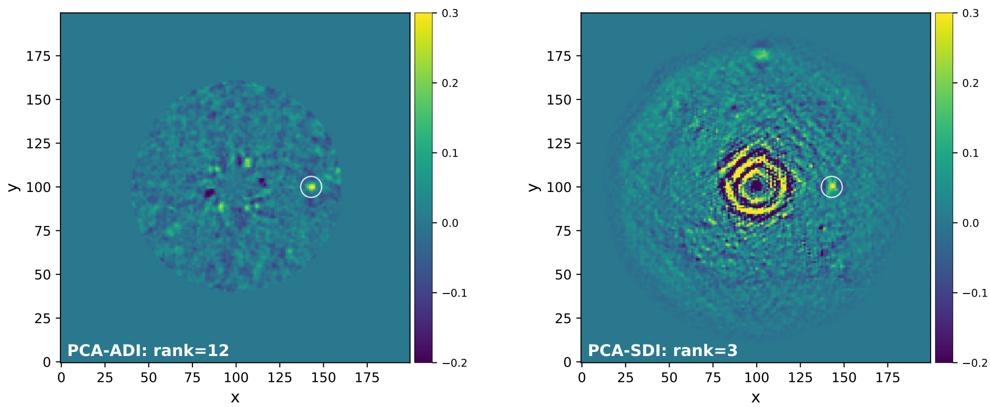

Figure 2.8: Mean residual frame obtained after applying PCA-ADI (left) and PCA-SDI (right) on the VLT/SPHERE-IFS data cube with an injected companion at position (143, 100) with a flux of 10. Ranks were determined so that the planet flux is maximized. A circular mask was used to remove the non-corrected part of the data cube (see next section).

Finally, when proceeding by visual inspection of the data cube, it seems that (i) does not hold. Regarding (ii), it is possible, by examining visually the descaled data cube, to check that this part of the speckle component also describes a radial trajectory, while the rest of the data cube remains static. This suggests that (ii) holds.

2.2.2 Area (b) and (c): correction ring

We now focus on the second area (b). This draws the limit of correction by Adaptive Optics (AO), commonly called the correction ring. Intuitively, the AO acts as a low-pass filter on the spatial frequencies in the focal plane, meaning that all the spatial frequencies outside the correction ring were not corrected by AO and thus should not be taken into account during the post-processing step. This low-pass filter behavior can be explained by the finite number of high-order deformable mirrors and the limited spatial sampling capacity of the wavefront sensing in the adaptive optics.

On the other hand, the correction ring has a particular pattern. This comes from space frequency aliasing errors that are introduced by the imperfection of AO. Consequently, most of the uncorrected high frequencies tend to aggregate around the correction ring (see [42] for more details). The particular pattern of area (c) is also due to aliasing error and comes from the imprint of the high-order deformable mirrors actuator grid.

In order to highlight how the light behaves differently outside the correction ring, one can draw similar flux curves as in the previous section. We see that there is a clear qualitative change of behavior around 10–12 λ/D, which exactly corresponds to the radius of the correction ring (that scales with the wavelength).

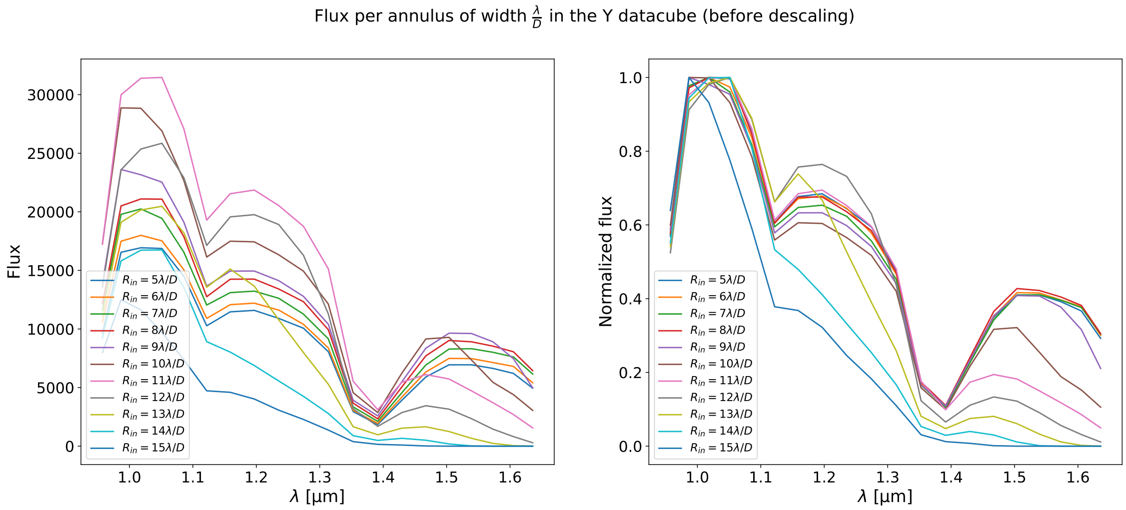

Figure 2.9: Left: flux of the SDI data cube $\boldsymbol{Y}$ summed over an annulus of inner radius R_{in} and of width $\frac{\lambda}{D}$. Right: Normalized flux of an annulus of $\boldsymbol{Y}$ computed as in figure 2.6.

2.2.3 Area (d): other atmospheric effects

The last area (d) is not present in every data cubes. This pattern has typically an extended shape (here an ellipsoid) that comes from additional atmospheric effects. Among them, one can cite the wind-driven halo, the low-order residuals and the low wind effect. Those effects will not be modeled in this paper.

Chapter 3

Inverse problem theory

This chapter introduces in a general framework the theory of inverse problem as well as some elements of proximal optimization that will be useful for the rest of this document. It was written consulting [43, 44, 45].

3.1 What is an inverse problem ?

We talk about inverse problems as soon as we are interested in identifying and characterizing objects for which we only have access to indirect measurements. Objects of interest are reconstructed by fitting a parametric model (often called the *forward model*) on the data, that is, to find parameters minimizing a well-chosen loss function. "Objects" should be read here in a general sense as it can represent vectors, functions, parameters of a probability density function [46], of a transfer function [47], of a partial differential equation [48], or even of an optimization model [49]. Hence, inverse problems is a very large field that has many applications not only in physics (e.g. X-ray computed tomography, astronomy), in engineering (e.g. system identification, medical imaging, computer vision, signal processing), but also in statistics and in finance.

In the scheme that will interest us, we define an inverse problem by three components :

- *Measurement operator* $\mathcal{A}$: This operator maps the objects of interest, denoted by $\boldsymbol{x}$, to observations $\boldsymbol{y}$, often called *measurement* or *data* that contain information about those objects. In the context of exoplanets detection, measurements $\boldsymbol{y}$ will be a (flattened) ASDI data cube, while objects $\boldsymbol{x}$ will contain both the exoplanets and the speckle pattern.
- *Statistical model of the noise* $\boldsymbol{\eta}$: In real applications, measurements are not perfect, leading to the acquisition of noisy data. We identify two sources of errors : *detector* noise introduced by errors during the measurement with instruments, and *modeling* errors coming from simplified and imperfect character of $\mathcal{A}$ to model reality. In ASDI, the *modeling* error will typically represent the non-static part of the speckle pattern.
- *Prior model and regularization* : Inverting the measurement operator $\mathcal{A}$ to recover the parameter $\boldsymbol{x}$ is generally an ill-posed problem, meaning that either the solution does not exist, is not unique or does not depend continuously of the data. In the best case, the problem might be well-posed but remains ill-conditioned, that is, a small perturbation on the data affects dramatically the value of the solution. Therefore, to enhance the well-posedness of the inverse problem, it is crucial to introduce prior information on the parameter $\boldsymbol{x}$, reducing the size of parameters space $\mathcal{X}$ by adding explicit constraints or

adding regularization terms.

Given the three components described above, an inverse problem is defined as follows :

Definition 3.1.1. *Let $\mathcal{X} \subseteq \mathbb{R}^m$ be the set of parameters and $\mathcal{Y} \subseteq \mathbb{R}^n$ be the set of measurements. An inverse problem consists in estimating the parameter $\boldsymbol{x} \in \mathcal{X}$ such that*

$$\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}) + \boldsymbol{\eta} \qquad (\text{Forward model}) \tag{3.1}$$

for a given $\boldsymbol{y} \in \mathcal{Y}$, by solving the optimization problem

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}\in\mathcal{X}} \mathcal{L}(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x})) + \alpha\mathcal{R}(\boldsymbol{x}) \qquad (\text{Inverse Problem}) \tag{3.2}$$

where $\alpha > 0$, $\mathcal{L} : \mathcal{Y}\times\mathcal{Y} \mapsto \mathbb{R}_{\geq 0}$ is a well-chosen loss function *and $\mathcal{R} : \mathcal{X} \mapsto \mathbb{R}_{\geq 0}$ is a nonnegative* regularization *function.*

The loss function, also called fidelity function, measures the distance between the model $\mathcal{A}(\boldsymbol{x})$ and the data $\boldsymbol{y}$ while the regularization function promotes a certain structure of the solution such as sparsity or smoothness of the parameters. The balancing between those two terms is controlled by the regularization parameter α that can be seen as a Lagrangian multiplier associated to the constraint $\mathcal{R}(\boldsymbol{x}) = 0$.

3.2 Noise model and loss function

We now show the strong connection between the loss function and the noise model, by interpreting (3.2) as a maximum a posteriori (MAP) estimator. Indeed, if we assume that the likelihood w.r.t. the data $\boldsymbol{y}$ and the parameter $\boldsymbol{x}$ is $p(\boldsymbol{y}|\boldsymbol{x}) \propto e^{-\mathcal{L}(\boldsymbol{y},\mathcal{A}(\boldsymbol{x}))}$ and the prior parameter distribution is $p(\boldsymbol{x}) \propto e^{-\alpha\mathcal{R}(\boldsymbol{x})}$, then, using the Bayes rule, the MAP estimator is given by

$$\begin{aligned}
\hat{\boldsymbol{x}}^{MAP} &= \arg\max_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{x}|\boldsymbol{y}) = \arg\max_{\boldsymbol{x}\in\mathcal{X}} p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}) \\
&= \arg\min_{\boldsymbol{x}\in\mathcal{X}} -\log p(\boldsymbol{y}|\boldsymbol{x}) - \log p(\boldsymbol{x}) \\
&= \arg\min_{\boldsymbol{x}\in\mathcal{X}} \mathcal{L}(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x})) + \alpha\mathcal{R}(\boldsymbol{x}) \\
&= \hat{\boldsymbol{x}}.
\end{aligned}$$

The MAP estimator thus exactly corresponds to the solution of the inverse problem $\hat{\boldsymbol{x}}$. Moreover, if the loss function satisfies $\mathcal{L}(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x})) = \mathcal{L}(\boldsymbol{y} - \mathcal{A}(\boldsymbol{x}), 0)$ for all $\boldsymbol{x} \in \mathcal{X}$ and $\boldsymbol{y} \in \mathcal{Y}$, then the likelihood $p(\boldsymbol{y}|\boldsymbol{x})$ is simply the noise distribution $p(\boldsymbol{\eta}) \propto e^{-\mathcal{L}(\boldsymbol{\eta},0)}$. Hence, the choice of a loss function in an inverse problem is generally determined by the noise model $\boldsymbol{\eta}$.

The most common example is the i.i.d. Gaussian noise with probability distribution $p(\boldsymbol{\eta}) \propto e^{-\frac{\|\boldsymbol{\eta}\|_2^2}{2\sigma^2}}$. In this case, we directly identify the loss function $\mathcal{L}(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x})) \propto \frac{1}{2}\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x})\|_2^2$, corresponding to the squared euclidean distance between the model and the data. Moreover, in the case of an i.i.d. Laplacian noise: $p(\boldsymbol{\eta}) \propto e^{-\frac{\|\boldsymbol{\eta}\|_1}{\sigma}}$, the associated loss function will be the Manhattan's distance: $\mathcal{L}(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x})) \propto \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x})\|_1$. Given the central limit theorem, the Gaussian hypothesis is used most of the time.

3.3 Solving the inverse problem with proximal optimization

Proximal optimization is related to composite convex and non-smooth optimization. More precisely, it solves optimization problems with an objective function that can be divided into two components :

$$\min_{\boldsymbol{x}} F(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x}). \tag{3.3}$$

In the context of inverse problems, the first component will be generally the loss function $f(.) = \mathcal{L}(\boldsymbol{y}, \mathcal{A}(.))$ and is assumed to be smooth and convex, while the second term will capture the regularization term $g(.) = \alpha\mathcal{R}(.)$ that we only require to be convex.

Since $g(.)$ can be non-differentiable, we have to introduce a generalization of the gradient for convex non-smooth functions called subdifferential.

Definition 3.3.1. *The subdifferential of a continuous function $f : \mathbb{R}^n \mapsto \mathbb{R}$ at a point $\boldsymbol{x}$ is defined as*

$$\partial f(\boldsymbol{x}) := \{\boldsymbol{g} \in \mathbb{R}^n : f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \langle \boldsymbol{g}, \boldsymbol{y} - \boldsymbol{x}\rangle,\ \forall \boldsymbol{y} \in \mathbb{R}^n\}, \tag{3.4}$$

where an element $\boldsymbol{g}$ of the set $\partial f(\boldsymbol{x})$ is called a subgradient of f at point $\boldsymbol{x}$.

In particular, when the function is differentiable and convex, the subdifferential at $\boldsymbol{x}$ contains only one element equal to the gradient of the function at $\boldsymbol{x}$. By construction, $\boldsymbol{x}^*$ will be a minimizer of $F(.)$ if $\mathbf{0} \in \partial F(\boldsymbol{x}^*)$.

3.3.1 Proximal gradient method

In order to find this minimizer, an efficient algorithm called proximal gradient method or forward-backward algorithm combines an explicit gradient step according to $f(.)$ with an implicit gradient step according to $g(.)$. Starting from $\boldsymbol{x}^{(0)}$, for some constants $\mu, \mu' > 0$, this algorithm provides the next iterate according to the following rule :

$$\boldsymbol{x}^{(i+1)} \in \boldsymbol{x}^{(i)} - \mu\nabla f(\boldsymbol{x}^{(i)}) - \mu'\partial g(\boldsymbol{x}^{(i+1)}). \tag{3.5}$$

This last expression can also be written as

$$\boldsymbol{x}^{(i+1)} = (\mathcal{I} + \mu'\partial g)^{-1}\left(\boldsymbol{x}^{(i)} - \mu\nabla f(\boldsymbol{x}^{(i)})\right) \tag{3.6}$$

$$=: \operatorname{prox}_{\mu' g}\left(\boldsymbol{x}^{(i)} - \mu\nabla f(\boldsymbol{x}^{(i)})\right). \tag{3.7}$$

The operator $(\mathcal{I}+\mu'\partial g)^{-1}(.)$ is called the proximal operator of $\mu' g(.)$ and is denoted by $\operatorname{prox}_{\mu' g}(.)$. Another equivalent definition involves the following minimization problem :

$$\operatorname{prox}_{\mu' g}(\boldsymbol{x}) := \arg\min_{\boldsymbol{z}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 + \mu' g(\boldsymbol{z}). \tag{3.8}$$

For some functions $g(.)$, one can find a close form for the prox operator (see for instance appendix A.3). Moreover, it has several interesting properties that will not be explored in this document. Interested readers may consult [45] for more information.

In the context of inverse problem, provided that the loss function $\mathcal{L}(\boldsymbol{y}, \mathcal{A}(.))$ is smooth and convex, and the regularizer $\mathcal{R}(.)$ is convex, the proximal gradient method thus allows to solve (3.2) using the update rule :

$$\boldsymbol{x}^{(i+1)} = \operatorname{prox}_{\mu'\alpha\mathcal{R}(.)}\left(\boldsymbol{x}^{(i)} - \mu\nabla_{\boldsymbol{x}}\mathcal{L}\left(\boldsymbol{y}, \mathcal{A}(\boldsymbol{x}^{(i)})\right)\right). \tag{3.9}$$

This method has been shown to converge in $\mathcal{O}(\varepsilon^{-1})$ iterations to reach an ε-precision on the solution $\boldsymbol{x}^*$ for some tolerance $\varepsilon > 0$ [50].

3.3.2 Accelerated proximal gradient method

In order to enhance the rate of convergence of the proximal gradient method, we can adopt the accelerated version of this algorithm [50, 51, 52, 53]. Several implementations exist but I chose to implement the one seen in [53] that gave better performances in terms of "learning curve" than the one proposed in [50] although this last is easier to implement. Starting from $\boldsymbol{v}^{(0)} = \boldsymbol{x}^{(0)}$, $\bar{\boldsymbol{g}}^{(0)} = 0$ and $\Theta^{(0)} = 0$, with $\mu = \mu' = \frac{1}{L}$ and L the Lipschitz constant of $\nabla f(.)$, one iteration $i \geq 0$ of the retained accelerated method is defined by the following steps :

Step 1 : update weights	$\theta^{(i)} = \frac{1+\sqrt{1+4\mu^{-1}\Theta^{(i)}}}{2\mu^{-1}}$ $\Theta^{(i+1)} = \Theta^{(i)} + \theta^{(i)}$ $\gamma^{(i)} = \frac{\theta^{(i)}}{\Theta^{(i+1)}}$
Step 2 : Compute weighted sums	$\boldsymbol{u}^{(i)} = (1-\gamma^{(i)})\boldsymbol{x}^{(i)} + \gamma^{(i)}\boldsymbol{v}^{(i)}$ $\bar{\boldsymbol{g}}^{(i+1)} = (1-\gamma^{(i)})\bar{\boldsymbol{g}}^{(i)} + \gamma^{(i)}\nabla f(\boldsymbol{u}^{(i)})$
Step 3 : update solution	$\boldsymbol{x}^{(i+1)} = \mathrm{prox}_{\mu' g}\left(\boldsymbol{u}^{(i)} - \mu\nabla f(\boldsymbol{u}^{(i)})\right)$ $\boldsymbol{v}^{(i+1)} = \mathrm{prox}_{\Theta^{(i+1)} g}\left(\boldsymbol{x}^{(0)} - \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}\right)$

This method is not simple to interpret. The term $\gamma^{(i)}(\boldsymbol{v}^{(i)} - \boldsymbol{x}^{(i)})$ can be seen as a momentum on $\boldsymbol{x}^{(i)}$ while the definition of $\bar{\boldsymbol{g}}^{(i)}$ and $\boldsymbol{v}^{(i)}$ results from the minimization of a potential function that is build so that the method converges in $\mathcal{O}(\varepsilon^{-1/2})$ iterations. Some additional details are provided in appendix A.4. In order to highlight the superiority of the accelerated method, figure 3.1 shows the "learning curve" of both methods when solving the inverse problem (FBAT$_X$) that we will introduce in section 4.2 of chapter 4.

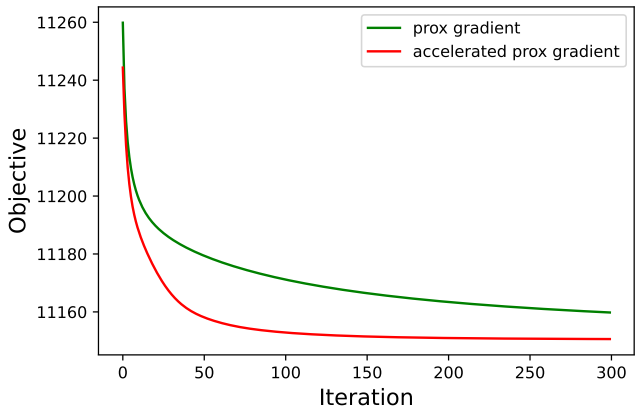

Figure 3.1: 300 iterations of both the proximal gradient method (green) and its accelerated version (red) on the minimization problem (FBAT$_X$) (see next chapter). The data cube used is VLT/SPHERE-IFS-2 with $W = 6$ and $T = 20$, while the parameters of (4.13) that were used are $r = 25$ and $\alpha = 0.25$.

3.3.3 The power iteration algorithm

It is important to note that the Lipschitz constant L of $\nabla f(.)$ is needed to use the accelerated gradient method. This constant is defined by the following inequality :

$$\forall \boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^n : \|\nabla f(\boldsymbol{x}') - \nabla f(\boldsymbol{x})\|_2 \leq L\|\boldsymbol{x}' - \boldsymbol{x}\|_2. \tag{3.10}$$

In particular, when we use a loss function corresponding to the squared ℓ_2-norm, we have $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \mathcal{A}(\boldsymbol{x})\|_2^2$ and consequently :

$$\|\nabla f(\boldsymbol{x}') - \nabla f(\boldsymbol{x})\|_2 = \|\mathcal{A}^*\mathcal{A}(\boldsymbol{x}' - \boldsymbol{x})\|_2 \leq \underbrace{\lambda_{\max}(\mathcal{A}^*\mathcal{A})}_{:=L} \|\boldsymbol{x}' - \boldsymbol{x}\|_2. \tag{3.11}$$

To compute the Lipschitz constant, we thus need to evaluate the maximum eigenvalue of the linear operator $\mathcal{A}^*\mathcal{A}(.)$. This can be done by using the so-called power iteration algorithm [54, p 205] working as follows. Starting from a random vector $\boldsymbol{x}^{(0)} \in \mathbb{R}^n$, we follow two step : (i) apply the linear operator $\mathcal{A}^*\mathcal{A}(.)$ and (ii) normalize the result, and repeat until convergence. Mathematically, the power iteration algorithm can be defined by the following update rule :

$$\boldsymbol{x}^{(i+1)} = \frac{\mathcal{A}^*\mathcal{A}(\boldsymbol{x}^{(i)})}{\|\mathcal{A}^*\mathcal{A}(\boldsymbol{x}^{(i)})\|_2} \tag{3.12}$$

We stop the algorithm when $\|\boldsymbol{x}^{(i+1)} - \boldsymbol{x}^{(i)}\| \leq \varepsilon$ for some $\varepsilon > 0$ and we return the Rayleigh quotient

$$L := \frac{\langle \mathcal{A}^*\mathcal{A}(\boldsymbol{x}^{(i+1)}), \boldsymbol{x}^{(i+1)} \rangle}{\|\boldsymbol{x}^{(i+1)}\|_2^2}. \tag{3.13}$$

Chapter 4

FBAT: Forward-Backward Algorithms with Trajectories to detect exoplanets in ASDI data cubes

In this chapter, we present the main contribution of this Master's thesis. Our aim is to extend to ASDI data cubes the low-rank-plus-sparse models previously introduced in section 1.5.

Starting from an ASDI data cube $\boldsymbol{Y} \in \mathbb{R}^{WT \times N^2}$, we would like to reconstruct a map $\boldsymbol{S} \in \mathbb{R}^{W \times N^2}$ containing the flux, varying with the wavelength, of each exoplanet in the cube. Note that we have concatenated both spectral and temporal dimensions to avoid dealing with tensor of order 3. In order to freeze the speckle pattern along the spectral dimension, we preprocess the data cube by scaling each spectral frame with respect to the scale factor $s(\lambda) = \lambda_W/\lambda$ yielding $\boldsymbol{Z} \in \mathbb{R}^{WT \times n^2}$. By convention, $n > N$ in order to avoid losing information while scaling the frames. From this setup, we now follow an inverse problem approach to solve the signal detection problem.

4.1 The forward model

According to section 1.4.3, we can decompose the data cube $\boldsymbol{Z}$ as the contribution of three additive terms: the speckle pattern, the circumstellar signal and the detector noise. Below, we model each of them one by one.

4.1.1 The circumstellar signal component

If an exoplanet lies in the preprocessed data cube $\boldsymbol{Z}$, then it will be a point source that rotates according to the parallactic angle in the temporal dimension and that moves radially in the spectral dimension since we have preprocessed the data cube. These transformations are equivalent to the application of both $\mathcal{Q}_{\boldsymbol{\alpha}}(.)$ and $\mathcal{D}_{\boldsymbol{s}}(.)$ on the circumstellar signal $\boldsymbol{S}$.

However, since a coronagraph is used to provide ASDI images, the light of the planet is also altered by the instrument. As mentioned in section 1.2, the instrument acts as a non-shift invariant operator that cannot be modeled by a convolution with the instrumental PSF as kernel. Nevertheless, assuming a sufficiently large angular separation of the exoplanet, one can neglect this property of the coronagraph and still model it as a convolution with the non-coronagraphic PSF [11] that we denote by $\boldsymbol{\varphi} \in \mathbb{R}^{W \times N^2}$.

The impact of the instrument on the signal is thus formalized as :

$$\text{(Convolution)} \quad \mathcal{T}_{\boldsymbol{\varphi}} : \mathbb{R}^{W\times N^2} \mapsto \mathbb{R}^{W\times N^2} \equiv \text{ discrete version of } F(\lambda, \boldsymbol{p}) \mapsto [F(\lambda, .) * \varphi(\lambda, .)](\boldsymbol{p}). \quad (4.1)$$

Gathering the three previous operators together, we obtain what we call the trajectorlet operator $\Psi(.)$ that models the circumstellar component of the data cube :

$$\Psi(\boldsymbol{S}) := \mathcal{D}_{\boldsymbol{s}}\mathcal{Q}_{\boldsymbol{\alpha}}\mathcal{T}_{\boldsymbol{\varphi}}(\boldsymbol{S}). \quad (4.2)$$

Finally, as the number of exoplanets in the cube is finite and much lower than N^2, we impose $\boldsymbol{S}$ to be sparse according to the spatial dimension using the ℓ_2/ℓ_0 pseudo-mixed-norm:

$$\|\boldsymbol{S}\|_{2,0} \leq k. \quad (4.3)$$

4.1.2 The speckle pattern

Regarding the speckle pattern $\boldsymbol{Z}_\star \in \mathbb{R}^{WT\times N^2}$, we have seen that it is almost static along both the spectral axis and the temporal axis. Hence, one can model it as the sum of two terms :

$$\boldsymbol{Z}_\star := \boldsymbol{L} + \boldsymbol{\eta}_{\text{ns}} \quad (4.4)$$

where $\boldsymbol{L}$ is a low-rank matrix representing the static part of $\boldsymbol{Z}_\star$ and $\boldsymbol{\eta}_{\text{ns}}$ is a noise term that models the non-static part of the speckle pattern. Our foward model can be written as follows :

$$\begin{aligned} \boldsymbol{Z} &= \boldsymbol{L} + \Psi(\boldsymbol{S}) + \boldsymbol{\eta} \\ \text{s.t.} \quad & \text{rank}(\boldsymbol{L}) \leq r \\ & \|\boldsymbol{S}\|_{2,0} \leq k. \end{aligned} \quad (4.5)$$

The noise term $\boldsymbol{\eta} \in \mathbb{R}^{WT\times n^2}$ gathers all the noise contributions that we will discuss in subsection 4.1.3. This model can be seen as a generalization of LRPT to ASDI data cubes.

However, we are now faced with two difficulties. On the one hand, in the previous model, we should not take into account pixels outside the limit of correction by choronagraphy that scales with λ (as explained in section 2.2). We define $r_{out}(\lambda)$ as the angular separation at which this limit is reached.

On the other hand, close to the star, one can observe some non-modeled phenomena, such as the bright annulus around the center dark disk that does not seem to scale with λ like the rest of the data cube. Moreover, as mentioned previously, too close to the center, the PSF $\boldsymbol{\varphi}$ does not remain invariant with the position in the frame. For those reasons, we would like to avoid data at angular separation lower than some constant r_{in}.

Once the data cube is preprocessed, those two limits become $\tilde{r}_{out} = s(\lambda)r_{out}(\lambda) \approx$ cst and $\tilde{r}_{in}(\lambda) = s(\lambda)r_{in}$ respectively.

However, as the bright annular around the star does not have the same scale factor as the rest of the speckle pattern, it will move radially after scaling exactly as an exoplanet. In order to limit false detection at small angular separation, we thus should keep $\tilde{r}_{in}(\lambda)$ constant. For instance, for the dataset SPHERE-IFS-2, these radii have been set to $\tilde{r}_{in} = \frac{5\bar{\lambda}}{D}$ and $\tilde{r}_{out} = \frac{20\bar{\lambda}}{D}$ respectively.

Now that the domain of validity of our forward model has been established, one can follow a similar approach to [10] and define a binary mask, which is an annular mask of inner-radius $\tilde{r}_{in}$ and outer-radius $\tilde{r}_{out}$. However, since the boundary of the AO-corrected area scales with λ, one can be faced with boundary issues when a part of this area leaves the field of view. We thus should take into account those boundaries in the mask to avoid forcing zero-values in some parts of the data that are in reality unobserved.

Combining all those observations, we would like to keep data included in the set of coordinates Ω as defined below :

$$\Omega := \Omega_{\text{border}} \cap (\mathbb{R} \times \mathbb{R} \times \Omega_{\text{ann}}) \tag{4.6}$$

$$\text{where } \Omega_{\text{ann}} := \{\boldsymbol{p} \in \mathbb{R}^2 : \tilde{r}_{in} \leq \|\boldsymbol{p} - \mathbf{c}\|_2 \leq \tilde{r}_{out}\} \tag{4.7}$$

$$\Omega_{\text{border}} := \{(\lambda, t, \boldsymbol{p}) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2 : \|\boldsymbol{p} - \mathbf{c}\|_\infty \leq s(\lambda)N/2\} \tag{4.8}$$

and $\boldsymbol{c} \in \mathbb{R}^2$ is the center of the frame. In order to perform selection of indexes from those previous sets, it is convenient to define the three linear operators of table 4.1.

Name	Symbol	Definition : discrete version of
(Partial restriction)	$\mathcal{P}'_{\Omega_{\text{ann}}} : \mathbb{R}^{WT \times n^2} \mapsto \mathbb{R}^{WT \times \lvert\Omega_{\text{ann}}\rvert}$	$F(\lambda, t, \boldsymbol{p}) \mapsto \{F(\lambda, t, \boldsymbol{p})\}_{\boldsymbol{p} \in \Omega_{\text{ann}}}$
(Restriction)	$\mathcal{P}_\Omega : \mathbb{R}^{WT \times n^2} \mapsto \mathbb{R}^{\lvert\Omega\rvert}$	$F(\lambda, t, \boldsymbol{p}) \mapsto \{F(\lambda, t, \boldsymbol{p})\}_{(\lambda, t, \boldsymbol{p}) \in \Omega}$
(Mask)	$\mathcal{M}_\Omega : \mathbb{R}^{WT \times n^2} \mapsto \mathbb{R}^{WT \times n^2}$	$F(\lambda, t, \boldsymbol{p}) \mapsto \begin{cases} F(\lambda, t, \boldsymbol{p}) & \text{if } (\lambda, t, \boldsymbol{p}) \in \Omega \\ 0 & \text{otherwise.} \end{cases}$

Table 4.1: Linear operators allowing to perform a selection of indexes in an ASDI data cube.

Using this notation, there are two possibilities to adjust our low-rank constraint for the speckle pattern. First, we can follow the same approach as in MAYO and model $\boldsymbol{Z}_\star$ with a $WT \times n^2$ low-rank matrix $\boldsymbol{L}$ but such that the values outside the mask are not taken into account in the loss function of the inverse problem. Secondly, inspired by annular version of ADI techniques, we can consider that only the data $\mathcal{P}'_{\Omega_{\text{ann}}}(\boldsymbol{Z})$ is available, yielding a low-rank model $\boldsymbol{L}$ of shape $WT \times |\Omega_{\text{ann}}|$. In other terms, given a loss function $\mathcal{L}$, either we optimize $\mathcal{L}(\mathcal{P}_\Omega(\boldsymbol{Z}), \mathcal{P}_\Omega(\boldsymbol{L} + \Psi(\boldsymbol{S})))$, or we optimize $\mathcal{L}(\mathcal{P}'_{\Omega_{\text{ann}}}(\boldsymbol{Z}), \boldsymbol{L} + \mathcal{P}'_{\Omega_{\text{ann}}}(\Psi(\boldsymbol{S})))$, both subject to $\text{rank}(\boldsymbol{L}) \leq r$.

While the latter does not take into account the boundaries of the frames, we observe in figure 4.1 that, thanks to our definition of $\tilde{r}_{in}$ and $\tilde{r}_{out}$, the annular mask encircled in red hardly never leaves the field of view represented by a square, when λ grows. This motivates us to still explore both approaches.

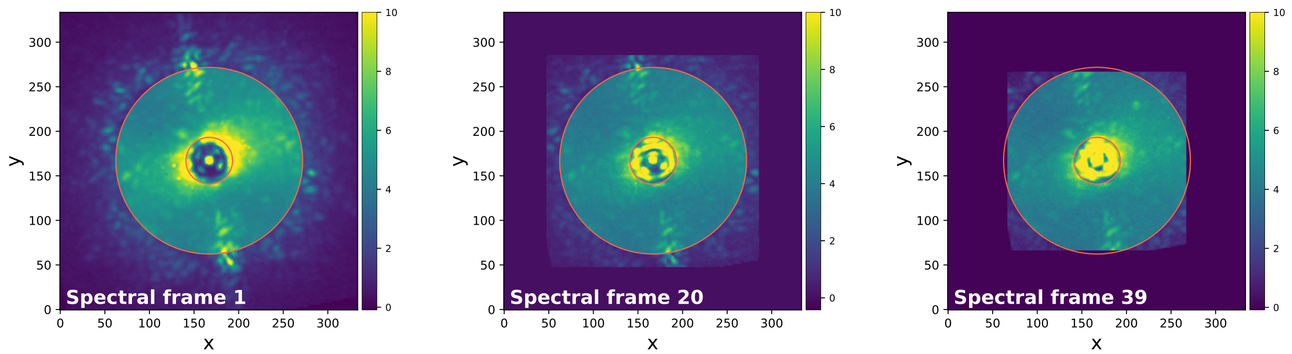

Figure 4.1: Spectral frames number 1, 20 and 39 of the VLT/SPHERE-IFS-2 data cube with $T = 1$ and $W = 39$. The mask Ω_{ann} is is delimited by the red circles while Ω_{border} is the square of decreasing size through frames.

4.1.3 The noise model

Until now, we did not consider the noise term $\boldsymbol{\eta}$. This last will directly provide the loss function we should use in the inverse problem as stated in section 3.2. In fact, several effects have not been modeled yet, such as the non-static part of the speckle pattern $\boldsymbol{\eta}_{\mathrm{ns}}$ as well as other noise effects coming from detector $\boldsymbol{\eta}_{\mathrm{det}}$. [10] suggests a Gaussian model for $\boldsymbol{\eta}_{\mathrm{det}}$, invoking the central limit theorem.

We now characterize the non-static part of $\boldsymbol{Z}_\star$. In the last decades, several studies have shown that the speckle component of the data cube follows a Modified Riccian (MR) probability distribution. Traditionally, we denote by I the mean intensity of an AO-corrected long exposure image and we say that this quantity follows a $\mathrm{MR}(I_c, I_s)$ distribution defined as

$$p_{\mathrm{MR}}(I, I_s, I_c) = \frac{1}{I_c} \exp\left(-\frac{I + I_c}{I_s}\right) \mathcal{I}_0 \left(\frac{2\sqrt{II_c}}{I_s}\right) \tag{4.9}$$

where $\mathcal{I}_0$ is the modified Bessel function of the first kind, I_c is a local time-averaged static PSF due to the diffraction of the starlight and I_s is the quasi-static speckle noise intensity [55, 12]. It is important to note that I_c and I_s are not constant quantities but remain constant at a given angular separation. From the work of [31], it has been shown that the MR distribution is sub-exponential, meaning that the tail of the distribution tends to be fatter than the one of a Gaussian distribution and better fits the tail of Laplacian distribution, particularly for high intensity observation.

Given this result, several works tried to model the non-static speckle noise as a Laplacian random variable in order to enhance the performance of their post-processing algorithms [10, 56, 9]. However, as a first approximation, we will here assume that the residual noise comes from the contribution of several independent random variables so that the resulting noise term follows a Gaussian distribution by the central limit theorem. Hence, our loss function will be the so-called ℓ_2-norm.

4.2 The inverse problem

We are now able to formulate the inverse problem resulting from what we discussed in the previous sections :

$$\begin{array}{cl} \min\limits_{\boldsymbol{L} \in \mathbb{R}^{WT \times n^2}, \boldsymbol{S} \in \mathbb{R}^{W \times N^2}} & \frac{1}{2} \|\boldsymbol{Z} - \boldsymbol{L} - \Psi(\boldsymbol{S})\|_2^2 \\ \text{s.t.} & \operatorname{rank}(\boldsymbol{L}) \leq r \\ & \|\boldsymbol{S}\|_{2,0} \leq k. \end{array} \tag{4.10}$$

Note that we did not add explicitly the mask in (4.10) in order to stay as general as possible. If we use the partial restriction operator, then one should replace $\boldsymbol{Z}$ and $\Psi(\boldsymbol{S})$ by $\mathcal{P}'_{\Omega_{\mathrm{ann}}}(\boldsymbol{Z})$ and $\mathcal{P}'_{\Omega_{\mathrm{ann}}}(\Psi(\boldsymbol{S}))$ respectively. If we use the restriction operator instead, then one should replace $\boldsymbol{Z} - \boldsymbol{L} - \Psi(\boldsymbol{S})$ by $\mathcal{P}_{\Omega}(\boldsymbol{Z} - \boldsymbol{L} - \Psi(\boldsymbol{S}))$. This does not have a major impact on the optimization algorithms we will describe hereafter. More details will be given in section 4.4.

The main drawback of the previous model is that both constraints are non-convex, leading to an optimization problem generally NP-hard. In order to avoid non-convexity, a common strategy is to perform a convex relaxation of (4.10), for instance by replacing the rank operator by a nuclear norm and the pseudo ℓ_0-norm by a ℓ_1-norm.

We here follows another approach to relax the rank operator. Since PCA still provides good background models, we choose to keep the principal singular vectors $\{\boldsymbol{u}_j, \boldsymbol{v}_j\}_{j=1}^r$ letting the singular values $\{\sigma_j\}_{j=1}^r$ as free variables that can be optimized. By relaxing the sparsity constraint on the flux into a ℓ_2/ℓ_1 regularizer, it yields

$$(\text{FBAT}_\Sigma) \min_{\boldsymbol{\Sigma}\in\mathbb{R}^{r\times r}, \boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{\Sigma}\boldsymbol{V}_r^\top - \Psi(\boldsymbol{S})\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}, \tag{4.11}$$

where $\boldsymbol{U}_r = [\boldsymbol{u}_j]_{j=1}^r$, $\boldsymbol{V}_r = [\boldsymbol{v}_j]_{j=1}^r$ are the matrices storing the principal singular vectors and $\boldsymbol{\Sigma}$ is a full matrix of variables that is not imposed to be diagonal in order to maintain enough free variables in the model. The main motivation of this reformulation is that an exoplanet should not have a strong impact on the result of a truncated SVD.

In order to be able to apply the proximal gradient algorithm, we need to compute both the gradient of the loss function and the proximal operator of the regularizer. After calculation (see appendix A.2), it follows that

$$\begin{array}{rcl} \boldsymbol{R}^{(i)} & = & \boldsymbol{U}_r\boldsymbol{\Sigma}^{(i)}\boldsymbol{V}_r^\top + \Psi(\boldsymbol{S}^{(i)}) - \boldsymbol{Z} \\ \boldsymbol{\Sigma}^{(i+1)} & = & \boldsymbol{\Sigma}^{(i)} - \mu\boldsymbol{U}_r^\top\boldsymbol{R}^{(i)}\boldsymbol{V}_r \\ \boldsymbol{S}^{(i+1)} & = & \text{prox}_{\mu\alpha_A\|\cdot\|_{2,1}}\left(\boldsymbol{S}^{(i)} - \mu\Psi^*(\boldsymbol{R}^{(i)})\right), \end{array} \tag{4.12}$$

where $\Psi^*(.)$ is the adjoint of the trajectorlet operator (see appendix A.1 for its mathematical derivation).

Testing this algorithm on a small example provided similar results as PCA-ASDI (cf. chapter 5 section 5.4), which suggests me to adopt a more flexible model where I only keep the spectro-temporal eigenvectors fixed while optimizing both the singular values and the spatial eigenvectors together that I store in a matrix $\boldsymbol{X} \in \mathbb{R}^{r\times n^2}$. This gives the following optimization model :

$$(\text{FBAT}_X) : \min_{\boldsymbol{X}\in\mathbb{R}^{r\times n^2}, \boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{X}^\top - \Psi(\boldsymbol{S})\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}. \tag{4.13}$$

Again, after some calculation, the proximal gradient descent can be written as

$$\begin{array}{rcl} \boldsymbol{R}^{(i)} & = & \boldsymbol{U}_r\left(\boldsymbol{X}^{(i)}\right)^\top + \Psi(\boldsymbol{S}^{(i)}) - \boldsymbol{Z} \\ \boldsymbol{X}^{(i+1)} & = & \boldsymbol{X}^{(i)} - \mu\left(\boldsymbol{U}_r^\top\boldsymbol{R}^{(i)}\right)^\top \\ \boldsymbol{S}^{(i+1)} & = & \text{prox}_{\mu\alpha\|\cdot\|_{2,1}}\left(\boldsymbol{S}^{(i)} - \mu\Psi^*(\boldsymbol{R}^{(i)})\right). \end{array} \tag{4.14}$$

Remark: It is important to mention that, to have a guarantee of convergence of the proximal gradient method (and its accelerated version), it is crucial to have the exact adjoint operator $\Psi^*(.)$. In appendix A.1, we give the continuous version of $\Psi^*(.)$ and sample it to get a discrete operator, but in practice, since we perform an interpolation of the grid, finding the true adjoint is more complex. One can indeed check that the equality $\langle\Psi(\boldsymbol{S}), \boldsymbol{Z}\rangle = \langle\boldsymbol{S}, \Psi^*(\boldsymbol{Z})\rangle$ is not exact for any $\boldsymbol{S}, \boldsymbol{Z}$ randomly generated. An alternative to compute the gradient is thus to use automatic differentiation available in PyTorch. Both approaches will be compared.

4.3 Greedy approaches

In this section, we present another direction that has been explored during this master thesis that adopts a greedy point of view.

4.3.1 Planets seen as trajectories

Instead of solving directly the whole inverse problem, we propose here to decompose the trajectorlet operators as a sum of trajectories drawn by each exoplanets in the data cube. Mathematically, we rewrite $\boldsymbol{S}$ in the canonical basis as :

$$\boldsymbol{S} = \sum_{g=1}^{N^2} \sum_{l=1}^{W} a_{l,g} \boldsymbol{\delta}_{l,g}, \tag{4.15}$$

where $\boldsymbol{\delta}_{l,g} \in \mathbb{R}^{W \times N^2}$ is a matrix full of zero except at position (l, g) where it is equal to one, and $a_{l,g} \in \mathbb{R}$ is the planet flux measured at wavelength λ_l and located in pixel g. This way, one can decompose the trajectorlet operator as :

$$\Psi(\boldsymbol{S}) = \sum_{g=1}^{N^2} \sum_{l=1}^{W} a_{l,g} \Psi(\boldsymbol{\delta}_{l,g}) = \sum_{g=1}^{N^2} \sum_{l=1}^{W} a_{l,g} \boldsymbol{P}_{l,g}. \tag{4.16}$$

According to their definition, matrices $\boldsymbol{P}_{l,g}$ are build first centering in g the PSF, measured at wavelength λ_l, in the row l of a $W \times N^2$ zero matrix, so that only one row contains non-zero values. Then, we apply on those matrices the rotation and scaling operators in order to obtain $WT \times n^2$ matrices.

In (4.16), the matrix sum $\sum_{l=1}^{W} a_{l,g} \boldsymbol{P}_{l,g}$ can be seen as a PSF that draws an helical trajectory in 4-D, but such that the flux is weighted by $a_{l,g}$ along the spectral axis. In particular, when $W = 1$ and assuming that only one planet lies in the data cube, $\boldsymbol{P}_{l,g}$ has the same meaning as in AMAT-ADI.

In practice, matrices $\boldsymbol{P}_{l,g}$ are highly sparse since the spatial shape of the PSF is several orders of magnitude smaller than the size of a data cube. Moreover, one can show that convolution, rotation and scaling commute with each other (up to some numerical error introduced by the interpolation of the grid) provided that we slightly modify their input-output dimensions :

$$\Psi(.) = \mathcal{D}_{\boldsymbol{s}} \mathcal{Q}_{\boldsymbol{\alpha}} \mathcal{T}_{\boldsymbol{\varphi}}(.) = \widetilde{\mathcal{T}}_{\widetilde{\boldsymbol{\varphi}}} \widetilde{\mathcal{Q}}_{\boldsymbol{\alpha}} \widetilde{\mathcal{D}}_{\boldsymbol{s}}(.) \tag{4.17}$$

where $\widetilde{\mathcal{Q}}_{\boldsymbol{\alpha}} : \mathbb{R}^{W \times n^2} \mapsto \mathbb{R}^{WT \times n^2}$, $\widetilde{\mathcal{D}}_{\boldsymbol{s}} : \mathbb{R}^{W \times N^2} \mapsto \mathbb{R}^{W \times n^2}$, $\widetilde{\mathcal{T}}_{\widetilde{\boldsymbol{\varphi}}} : \mathbb{R}^{WT \times n^2} \mapsto \mathbb{R}^{WT \times n^2}$ and $\widetilde{\boldsymbol{\varphi}} = \widetilde{\mathcal{D}}_{\boldsymbol{s}}(\boldsymbol{\varphi})$. The idea is thus to first compute, by hand, the 4-D helical trajectory drawn by one pixel starting in g (see figure 4.2) and then inject the rescaled PSF $\widetilde{\boldsymbol{\varphi}}$ centered at each pixel of the trajectory by only retaining the non-zero values of $\widetilde{\boldsymbol{\varphi}}$. Matrices $\boldsymbol{P}_{l,g}$ can therefore be stored in coordinate format (COO).

4.3.2 Our greedy algorithm

Using this decomposition of the trajectorlet operator, one can directly substitute it in (4.13). However, evaluating $\Psi(.)$ with the N^2 trajectories is in practice intractable as it requires $\mathcal{O}(N^2WTK)$ operations, where K is the number of non-zero values of the rescaled PSF $\widetilde{\boldsymbol{\varphi}} \in \mathbb{R}^{W \times n^2}$. To illustrate this with numbers, let $W = 20$, $T = 6$, $N = 200$ and $n = 342$, then this number of operations amounts to $\mathcal{O}(10^9)$, which is two orders of magnitude higher than the number of entries in an ASDI data cube $\boldsymbol{Z} \in \mathbb{R}^{WT \times n^2}$. This suggests a greedy approach by splitting the set of N^2 trajectories in B batches denoted by Λ_b for $b = 1, \ldots, B$. This way, one solves for each batch b :

$$(\text{FBAT}_X\text{-Greedy}) : \min_{\boldsymbol{X}_b \in \mathbb{R}^{r \times n^2}, \boldsymbol{S}_b \in \mathbb{R}^{W \times N}} \frac{1}{2} \|\boldsymbol{Z} - \boldsymbol{U}_r \boldsymbol{X}_b^\top - \Psi_b(\boldsymbol{S}_b)\|_2^2 + \alpha \|\boldsymbol{S}_b\|_{2,1} \tag{4.18}$$

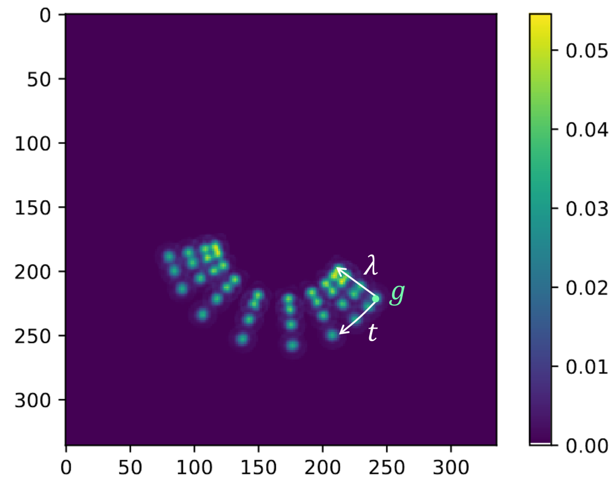

Figure 4.2: Helical trajectory drawn by an exoplanet in the preprocessed data cube $\boldsymbol{Z}$. This picture corresponds to $\sum_{l=1}^{W} \mathbf{1}_{WT}^{\top} \boldsymbol{P}_{l,g}$ that has been reshaped as a $n \times n$ frame. In this example, I set $W = 4$ and $T = 10$.

where $\Psi_b(\boldsymbol{S}_b) := \sum_{g \in \Lambda_b} \sum_{l=1}^{W} a_{l,g} \boldsymbol{P}_{l,g}$ can be seen as a trajectorlet operator restricted to batch b. With this notation, we can still apply the proximal gradient descent in (4.14). See A.1.4 for the definition of $\Psi_b^*(.)$.

Once each subproblem is solved, we gather the output of each of them as a set of background models $\{\boldsymbol{L}_b\}_{b=1}^{B}$ and a composite flux map $\boldsymbol{S}$ such that $\boldsymbol{L}_b = \boldsymbol{U}_r \boldsymbol{X}_b^{\top}$ for $b = 1, \ldots, B$ and $[\boldsymbol{S}]_{l,g} = [\boldsymbol{S}_b]_{l,g}$ if $g \in \Lambda_b$, for $l = 1, \ldots, W$ and $b = 1, \ldots, B$.

An advantage of this division in batches is that we can easily parallelize it since each optimization problem is independent. Increasing the batchsize yields less optimization problems to solve but needs more computational resources and thus cannot be parallelized with many threads (given the limit of the `lemaitre3` CECI server used to run the algorithm). I thus tested it on an example with $N^2 = 10040$ and batches of size 50, 100, 250 and 500 with 20, 10, 4 and 2 threads respectively, and I measured the time taken by each choice of batchsize. From results of table 4.2, I thus chose to retain the batchsize of 50 with 20 threads as it was the fastest.

Batchsize $\lvert\Lambda_b\rvert$	# Batches B	# Threads	Memory per thread (Mo)	Execution time
50	201	20	4000	705.92s
100	101	10	4000	1232.92s
250	41	4	8000	2423.68s
500	21	2	8000	$>$ 4880s

Table 4.2: Comparison of execution time with different batchsizes. Parameters used are $W = 20$, $T = 6$, $N^2 = 10040$, $\alpha = 0.5$ and $r = 30$.

4.3.3 AMAT-ASDI

In the case of batches of size one, we recover the same setup as AMAT. One can thus naturally extend this algorithm to ASDI data cubes as :

$$(\text{AMAT}) : \min_{\boldsymbol{L} \in \mathbb{R}^{WT \times n^2}, a_{g,l} \in \mathbb{R}} \frac{1}{2} \|\boldsymbol{Z} - \boldsymbol{L} - \textstyle\sum_{l=1}^{W} a_{l,g} \boldsymbol{P}_{l,g}\|_2^2 \text{ s.t. } \operatorname{rank}(\boldsymbol{L}) \leq r \tag{4.19}$$

using alternating exact minimization :

$$\begin{aligned}\boldsymbol{L}^{(i+1)} &= \mathcal{H}_r^{\text{SVD}}\left(\boldsymbol{Z} - \sum_{l=1}^{W} a_{l,g}^{(i)}\boldsymbol{P}_{l,g}\right)\\ a_{l,g}^{(i+1)} &= \frac{\langle \boldsymbol{Z} - \boldsymbol{L}^{(i+1)}, \boldsymbol{P}_{l,g}\rangle}{\|\boldsymbol{P}_{l,g}\|_2^2}.\end{aligned}$$

In fact, this model was the starting point of this master thesis and was used as a model of reference to construct the algorithms previously proposed. AMAT-ASDI owns several advantages. For a single trajectory focused on the true position of a planet we inject, it highly outperforms PCA-ASDI (see figure 5.8 at the end of the next chapter). Moreover, it was empirically observed that it converges quickly to a fixed point (~ 50 iterations against ~ 200 for FBAT$_X$-Greedy). However, this algorithm is non-convex because of the low-rank constraint, and furthermore, it takes several hours to run (e.g. > 15 hours for $W = 20$, $T = 6$, $N^2 = 10040$ and 20 threads) as we have to solve N^2 instances of AMAT-ASDI, i.e one per trajectory.

4.4 Dealing with the mask and the initialization

As mentioned in section 4.2, we will provide further details about the insertion of the mask, represented by Ω, in the optimization algorithm provided in (4.13). Both possibilities are developed below: the restriction mode that uses $\mathcal{P}_\Omega(.)$ and the partial restriction mode that uses ${\mathcal{P}'}_{\Omega_{\text{ann}}}(.)$.

Restriction mode

Following the first approach that uses the restriction operator, we modify (4.13) as

$$\min_{\boldsymbol{X}\in\mathbb{R}^{r\times n^2},\boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\mathcal{P}_\Omega(\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{X}^T - \Psi(\boldsymbol{S}))\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}. \tag{4.20}$$

Using the chain rule, this optimization model can be solved by the proximal gradient descent as follows :

$$\begin{array}{rcl}\boldsymbol{R}^{(i)} &=& \mathcal{M}_\Omega\left(\boldsymbol{U}_r\left(\boldsymbol{X}^{(i)}\right)^\top + \Psi(\boldsymbol{S}^{(i)}) - \boldsymbol{Z}\right)\\ \boldsymbol{X}^{(i+1)} &=& \boldsymbol{X}^{(i)} - \mu\left(\boldsymbol{U}_r^\top\boldsymbol{R}^{(i)}\right)^\top\\ \boldsymbol{S}^{(i+1)} &=& \text{prox}_{\mu\alpha\|.\|_{2,1}}\left(\boldsymbol{S}^{(i)} - \mu\Psi^*(\boldsymbol{R}^{(i)})\right)\end{array} \tag{4.21}$$

where we use the property $\mathcal{P}_\Omega^*\mathcal{P}_\Omega = \mathcal{M}_\Omega$.

Partial restriction mode

Following the second approach that uses the partial restriction operator, one can modify (4.13) as

$$\min_{\boldsymbol{X}\in\mathbb{R}^{r\times|\Omega_{\text{ann}}|},\boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\mathcal{P}'_{\Omega_{\text{ann}}}(\boldsymbol{Z}) - \boldsymbol{U}_r\boldsymbol{X}^T - \mathcal{P}'_{\Omega_{\text{ann}}}(\Psi(\boldsymbol{S}))\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}. \tag{4.22}$$

which yields the following proximal algorithm

$$\begin{array}{rcl}\boldsymbol{R}^{(i)} &=& \boldsymbol{U}_r\left(\boldsymbol{X}^{(i)}\right)^\top + \mathcal{P}'_{\Omega_{\text{ann}}}(\Psi(\boldsymbol{S}^{(i)})) - \mathcal{P}'_{\Omega_{\text{ann}}}(\boldsymbol{Z})\\ \boldsymbol{X}^{(i+1)} &=& \boldsymbol{X}^{(i)} - \mu\left(\boldsymbol{U}_r^\top\boldsymbol{R}^{(i)}\right)^\top\\ \boldsymbol{S}^{(i+1)} &=& \text{prox}_{\mu\alpha\|.\|_{2,1}}\left(\boldsymbol{S}^{(i)} - \mu\Psi^*({\mathcal{P}'}^*_{\Omega_{\text{ann}}}(\boldsymbol{R}^{(i)}))\right).\end{array} \tag{4.23}$$

In the equation above, $\mathcal{P}'^{*}_{\Omega_{\text{ann}}}(.)$ is the adjoint of the partial restriction operator, that is defined as

$$\mathcal{P}'^{*}_{\Omega_{\text{ann}}} : \mathbb{R}^{WT\times|\Omega_{\text{ann}}|} \mapsto \mathbb{R}^{WT\times n^2} \equiv \text{ discrete version of } F(\lambda,t,\boldsymbol{p}) \mapsto \begin{cases} F(\lambda,t,\boldsymbol{p}) & \text{if } \boldsymbol{p}\in\Omega_{\text{ann}} \\ 0 & \text{otherwise.}\end{cases}$$

The main advantage of this second mode is that the matrix $\boldsymbol{X}$ is of lower dimension than in the restriction mode since $|\Omega_{\text{ann}}| \ll n^2$. However, as mentioned in section 4.1.2, this mode does not take into account the border of the images.

Initialization

A last important aspect that has not been covered yet is the initialization of the optimization algorithms, that is, the choice of $\boldsymbol{S}^{(0)}$ and $\boldsymbol{X}^{(0)}$ for FBAT_X and FBAT_X-Greedy. For the flux map $\boldsymbol{S}^{(0)}$, we simply set all its entries to zero since we expect this map to be sparse. Regarding the singular vectors $\boldsymbol{U}_r$ and $\boldsymbol{X}^{(0)}$, those are defined as the truncated SVD in the mask associated to Ω_{ann}. The exact definition depends on the mask mode we select :

$$(\boldsymbol{X}^{(0)})^\top := \begin{cases} P'^{*}_{\Omega_{\text{ann}}}(\boldsymbol{\Sigma}_r\boldsymbol{V}_r^\top) & \text{if the mask mode is "restriction"} \\ \boldsymbol{\Sigma}_r\boldsymbol{V}_r^\top & \text{if the mask mode is "partial restriction"}\end{cases}$$
$$\text{where } \boldsymbol{U}_r\boldsymbol{\Sigma}_r\boldsymbol{V}_r^\top := \mathcal{H}_r^{\text{SVD}}(\mathcal{P}'_{\Omega_{\text{ann}}}(\boldsymbol{Z})).$$

Note that the operator $P'^{*}_{\Omega_{\text{ann}}}(.)$ was extended to matrices of shape $r \times |\Omega_{\text{ann}}|$. Since both FBAT_X and FBAT_X-Greedy are convex optimization problems, the initialization has no big influence on the solution provided by those algorithms. However, a good initialization allows to decrease the number of iterations.

Stopping criterion

Since the optimal solution $(\boldsymbol{S}^*, \boldsymbol{X}^*)$ is not known, we cannot use the classical stopping criterion $\sqrt{\|\boldsymbol{S}^{(i)} - \boldsymbol{S}^*\|_2^2 + \|\boldsymbol{X}^{(i)} - \boldsymbol{X}^*\|_2^2} \leq \varepsilon$. Instead, we look at the relative variation of the flux between two successive iterations and stop the algorithm if $\frac{\|\boldsymbol{S}^{(i+1)}-\boldsymbol{S}^{(i)}\|_2}{\|\boldsymbol{S}^{(i+1)}\|_2} \leq \varepsilon$, where the value $\varepsilon = 10^{-3}$ is used. Moreover, a maximal number of iterations of 200 is fixed so that $(\boldsymbol{S}^{(200)}, \boldsymbol{X}^{(200)})$ is returned if the tolerance ε is not reached.

Accelerated proximal gradient method

A last important remark concerns the optimization method used. Although only the formulas of the proximal gradient method have been provided, all of them have been converted to their accelerated version in order to speed up their rate of convergence. Note that the name "Forward-Backward" given to the FBAT algorithms is quite abusive, since this is in fact the accelerated proximal gradient method that is used and not the forward-backward algorithm (i.e the proximal gradient descent).

Chapter 5

Metrics to detect exoplanets

5.1 Signal detection theory

Signal detection theory lies in a statistical formalism through an hypothesis testing framework. Let $\boldsymbol{M} \in \mathbb{R}^{N \times N}$ be a map, called detection map, provided in output of an ASDI post-processing algorithm, for instance the residual map of PCA-ASDI, a SNR map, or a flux map (see section 5.3). For each resolution element of diameter $\bar{\lambda}/D$ in the map, we define the null hypothesis H_0 : no companion, against the research hypothesis H_1 : there is a companion. By construction, small values of $\boldsymbol{M}$ support H_0 while high values support H_1.

We denote by M the ninth decile[1] of one resolution element in $\boldsymbol{M}$ so that, by selecting a threshold τ, we reject H_0 in favor of H_1 if 10% of the values in the resolution element are above the threshold τ, that is, $M \geq \tau$. The correctness of our decision thus strongly depends on the threshold we use. In particular, there are four possible outcomes that are summarized in table 5.1 below.

	A companion (H_1 is true)	No companion (H_0 is true)
Detection (reject H_0)	True Positive (TP)	False Positive (FP)
No detection (do not reject H_0)	False Negative (FN)	True Negative (TN)

Table 5.1

Avoiding False Positive, also called false alarm or error of type I, is the main concern of astronomers, since once an exoplanet is detected by ASDI, we generally perform a second observation campaign to validate the detection, which may be wasteful in case of false detection. The quality of a post-processing algorithm is thus quantified by its ability to minimize the False Positive Rate (FPR) while maximizing the True Positive Rate (TPR) respectively defined by :

$$\mathrm{FPR}(\tau) = \Pr[M \geq \tau | H_0] \approx \frac{\mathrm{FP}(\tau)}{\mathrm{TN}(\tau) + \mathrm{FP}(\tau)} \tag{5.1}$$

$$\mathrm{TPR}(\tau) = \Pr[M \geq \tau | H_1] \approx \frac{\mathrm{TP}(\tau)}{\mathrm{TP}(\tau) + \mathrm{FN}(\tau)}. \tag{5.2}$$

The quantities $\mathrm{TP}(\tau), \mathrm{FP}(\tau), \mathrm{FN}(\tau), \mathrm{TN}(\tau)$ are the empirical numbers of TP, FP, FN and TN obtained by simulating a large number of realizations M under both hypotheses with the threshold τ. More precisely, for a list of empty data cubes, we inject in each of them one single

[1] This choice is motivated by the desire to penalize false detections. It will be more clear in figure 5.1.

fake companion located at a different position for each cube, with the same angular separation and the same flux. Then, we build an overlap of balls of diameter equal to the aperture so that the planet is entirely contained in one of those elements (see figure 5.1). Finally, we evaluate M on each ball for a given threshold τ, which yields a TP, FP, FN and TN for one data cube. Summing up over all the data cubes provides the empirical values of TP(τ),FP(τ),FN(τ) and TN(τ). In fact, there are several alternatives to compute those quantities but we chose here to follow the one proposed by [57] (see next section for more details).

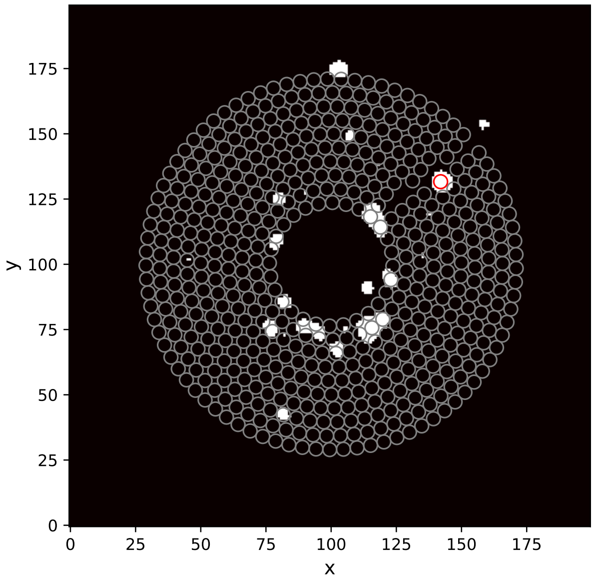

Figure 5.1: Illustration of an overlap of resolution elements in a map $\boldsymbol{M}$. White blobs correspond to positive detection. The red circle contains the true planet at its center and allows to count the TP and FN. Once this resolution element is placed, we draw grey resolution elements from $3.5\bar{\lambda}/D$ to $14\bar{\lambda}/D$ in order to count the FP and TN. By definition of M, we count one FP if 10% of the resolution element is filled of white color. This allows to penalize false alarms and to take into account the case when a grey circle is not centered on a false detection.

5.2 ROC curves

In order to visualize the performance of a classifier, it is convenient to plot what we call a Receiver Operating Characteristics (ROC) curve that shows on a 2D plane the curve (FPR, TPR)(τ) parameterized by the threshold τ. Typically, this curve grows monotonically when decreasing the threshold. Starting from a high threshold, both TPR and FPR will be zero (always returns False) and for a low threshold, those two quantities will be equal to 1 (always returns True). Moreover, a random classifier will draw a diagonal straight line, while a perfect classifier will reach the upper left corner $(0, 1)$. For another given classifier, the closer to the $(0, 1)$ corner it is, the better is the classifier. An illustration of a theoretical ROC curve is given in figure 5.2.

However, this approach has certain limitations. This mainly comes from the imbalance between True and False classes. Indeed, since a data cube contains a low number of exoplanets compared to the number of possible locations, we will generally have TN(τ) $\gg$ FP(τ) yielding a low FPR even if the number of False Positives still remains high.

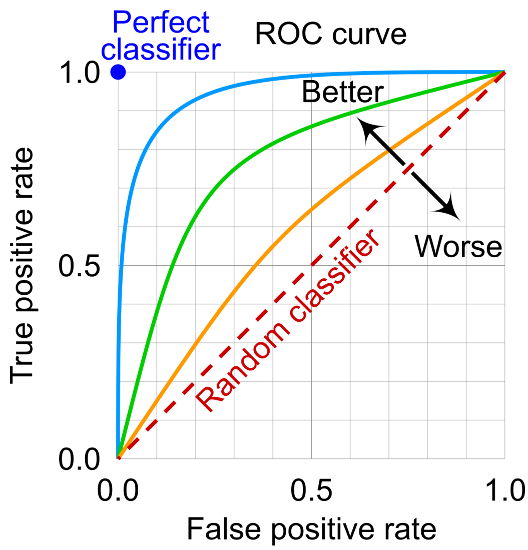

Figure 5.2: Theoretical ROC curve from [58].

To overcome this bad effect, [59] proposed to modify the axes of the ROC curve, replacing the FPR by the average number of False Positives (mean FPs) over all the data cubes (see left plot in figure 5.4). They also slightly changed the manner they compute the FPs to further penalize false detections.

Given a threshold τ, they construct a binary map and set to 1 each pixel (i, j) such that the $[\boldsymbol{M}]_{ij} \geq \tau$ and 0 otherwise. Then, they divide this map into segments of 1. If the size of the segment is smaller than one resolution element, they skip this segment. However, if the size of a segment exceeds twice the size of a resolution element, then they count one false positive by resolution element that fits with the shape of this segment, even if it contains a true planet.

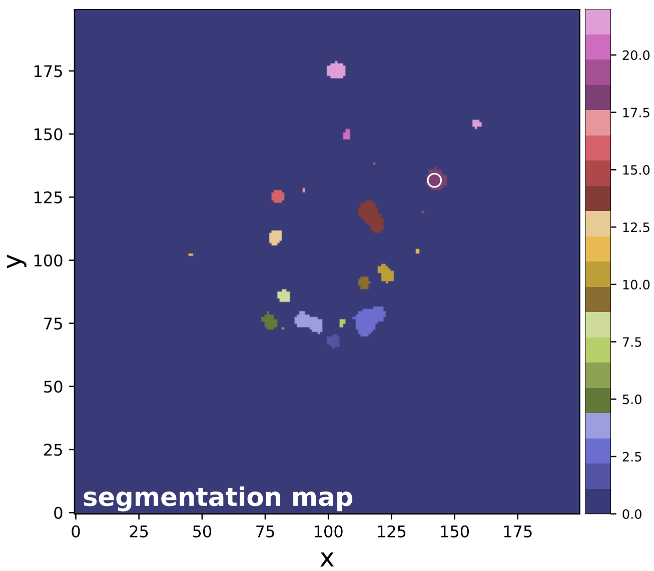

Figure 5.3: Illustration of the division into segments performed in VIP. Here more than 20 segments were found. Both segment number 3 (in blue) and number 15 (in brown) will here generate 2 FPs, while the 11th and 12th in yellow will be skipped.

Those modifications have two important consequences. First, if every pixel in the image is classified as a detection, the TPR will be 0 and the ROC curve will never reach the upper right corner. Secondly, a random classifier will no longer draw a diagonal line. This can be misleading in some context. For example, when we inject a planet with a zero flux, each algorithm should behave as a random classifier and therefore should provide a diagonal curve, which is not the case with this definition of FPs.

It is for those reasons that I chose to follow the approach proposed by [57] to compute the FPR and the TPR. To manage the imbalance of the classes, they instead propose to modify the axes of the ROC by $\sqrt{\text{TPR}}$ and $\sqrt{\text{FPR}}$ respectively, so that it highlights the left part of the curve, that is, the closest from the target corner $(0, 1)$ (see right plot in figure 5.4).

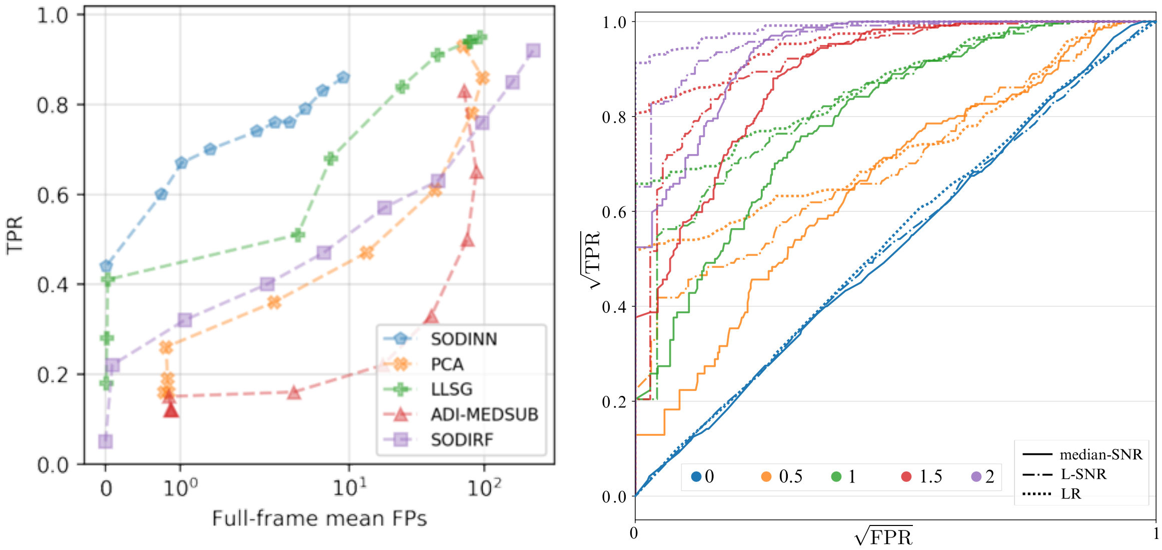

Figure 5.4: Example of ROC curve from [59] (left) and [57] (right). A diagonal line (in blue) is obtained when injecting a planet with zero flux on the left plot.

5.3 Performing detection

The last component that remains before performing exoplanet detection is the definition of a good detection map $\boldsymbol{M}$.

5.3.1 The SNR map

The most widespread metric used in the HCI community to assess the quality of a detection is the Signal to Noise Ratio (SNR or S/N to avoid confusion with the conventional SNR defined in chapter 1). Given a background model $\boldsymbol{L}$, this test works on the residual map

$$\boldsymbol{R} := \frac{1}{W} \mathbf{1}_W^\top \mathcal{Q}_{\boldsymbol{\alpha}}^{-1} \mathcal{D}_{\boldsymbol{s}}^{-1} (\boldsymbol{Z} - \boldsymbol{L}) \tag{5.3}$$

and is based on two assumptions. Firstly, it assumes that the non-static speckle noise follows a Modified Riccian distribution whose statistic is invariant at a given angular separation. Secondly, it considers that ASDI post-processing algorithms perform a whitening of the speckle statistic, yielding i.i.d. gaussian random variables at each angular separation. Consequently, one can apply the so-called two samples t-test in the residual data cube $\boldsymbol{R}$, where the first population corresponds to a single resolution element that encircles the planet ($n_1 = 1$) and the second population is composed of n_2 resolution elements defined such that $n_1 + n_2$ non-overlapping resolution elements cover the whole annulus (see figure 5.5). This test allows to determine whether both populations come from the same distribution or not, assuming they have the same variance.

Let $\bar{x}_1$ and $\bar{x}_2$ be the mean residual flux of each population. A decision between the null hypothesis $H_0 : \bar{x}_1 - \bar{x}_2 = 0$ against $H_1 : \bar{x}_1 - \bar{x}_2 \neq 0$ is made using the test statistic :

$$\mathrm{S/N} = \frac{\bar{x}_1 - \bar{x}_2}{s_2\sqrt{1 + \frac{1}{n_2}}} \tag{5.4}$$

where s_2 is the empirical standard deviation of the fluxes of the noise resolution elements. Evaluating this test for each pixel (successively being population one) of the residual map provides the so-called SNR map.

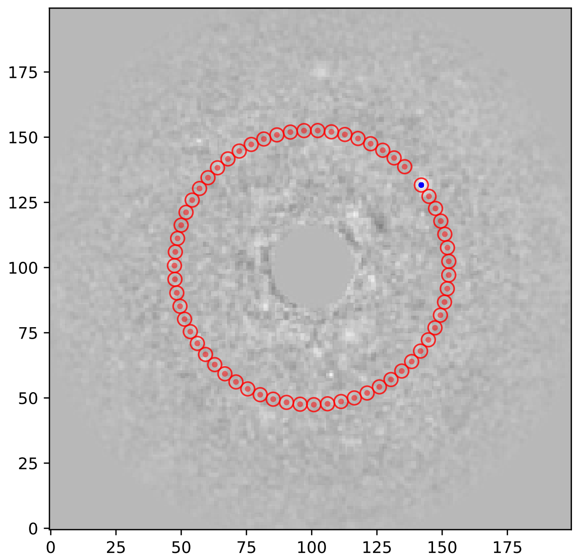

Figure 5.5: Illustration of both populations involved in the t-test. Blue and red balls represent the first and the second population respectively. A high S/N means that a planet is likely to be located inside the blue ball.

A particularity of the t-test is its robustness to small sample statistics, which mainly comes from the correcting factor $\sqrt{1 + \frac{1}{n_2}}$ that takes into account the number of samples in the second population. However, when the gaussian hypothesis does not hold exactly or when at least two planets lie in the same annulus, this test suffers from limited performances to make the right decision.

Remark: For the greedy method FBAT$_X$-Greedy, since there are several background models $\{\boldsymbol{L}_b\}_{b=1}^{B}$, that is, one for each batch, the definition of the SNR map should be adapted. For each batch $b = 1, \dots, B$, we first construct a residual map $\boldsymbol{R}_b$ following definition (5.3) and then a SNR map $\boldsymbol{M}_b$ applying (5.4). Finally, the final SNR map is obtained merging $\{\boldsymbol{M}_b\}_{b=1}^{B}$ by only retaining for each batch b the values associated to the pixels g of the batch, i.e $g \in \Lambda_b$.

5.3.2 The flux map

In the case of FBAT$_X$, another possibility is to directly use the flux map. To apply the same procedure as previously described to compute the TP, FP, FN, TN by resolution elements, one has to convolve the flux map $\boldsymbol{S}$ with the PSF. This way, the detection map can be written as

$$\boldsymbol{M} := \tfrac{1}{W}\mathbf{1}_W^\top \mathcal{T}_\varphi(\boldsymbol{S}). \tag{5.5}$$

5.3.3 Other maps

We quickly mention two other detection maps that tend to overcome limitations of the S/N. Those are the standard trajectory intensity mean (STIM) map and the Likelihood Ratio (LR) map. The STIM map, originally proposed by [31], was designed to be more robust in case of small sample statistics by using temporal statistic, and takes into account an imperfect whitening by post-processing algorithms of the speckle noise Modified Riccian distributed. Furthermore, it is not impacted by the presence of several planets in the same annulus. This property is particularly useful in the case of extended sources [24].

On the other hand, [57] has recently proposed a Likelihood Ratio (LR) test to perform the detection. This test combines all the advantages of STIM with a powerful statistical tool for hypothesis testing.

5.4 Comparison of the methods

To conclude this chapter, we compare all our algorithms introduced in the previous chapter with PCA-ASDI on a toy-example using the SNR map and the flux map. The results are shown in figure 5.6 for PCA and in figure 5.7 and 5.8 for the other methods.

On the one hand, looking at the output of FBAT_X in "restriction" mode (lower right) and in "partial restriction" mode (lower center) in figure 5.7, we see that both mask modes are almost equivalent. This result was expected given that the AO-corrected area hardly never leaves the field of view. One should note that the running time for the "restriction" mode is ~ 1.25 times slower than the "partial restriction" mode (here 243s against 193s).

Regarding the PyTorch implementation of FBAT_X in partial restriction mode, it provides a higher SNR score but also a lower flux score compared to the other FBAT_X algorithms. Since the difference is small, we will no longer consider the PyTorch version of FBAT_X in this paper and will let it to future research. However, we note that automatic differentiation with PyTorch could be interesting to explore since it allows to speedup the running time of FBAT_X and to get the exact gradient via the chain rule.

On the other hand, Greedy methods were used with the mask mode "partial restriction" in figure 5.8. Moreover, they were applied with only the true trajectory of the planet, thus fixing $\alpha = 0$. This means that greedy methods have an advantage over others since they try to fit a model that perfectly corresponds to reality. Consequently, implementing FBAT_Σ in a greedy form allows to highlight its poor performance since it reaches almost the same SNR score as PCA-ASDI. Finally, as mentioned above, AMAT highly outperforms PCA and interestingly FBAT_X-Greedy tends to mimic this behavior whereas we have fixed the spectro-temporal eigenvectors contrary to AMAT.

Those results motivate us to keep the "partial restriction" mode for the rest of this document and to drop the FBAT_Σ algorithm. Finally, we will also drop AMAT for the ROC analysis since it is not tractable to run $N^2 = \mathcal{O}(10^5)$ times the AMAT algorithm for several injections.

Note that, even though one single example is shown below, the decision to drop algorithms in this section is based on several simulations of the same kind, with injections at different angular separation and with different fluxes.

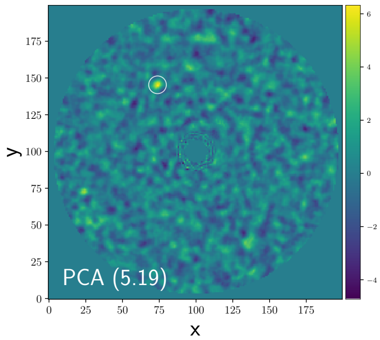

Figure 5.6: SNR map obtained with PCA-ASDI method, setting $r = 40$, applied on the empty version of the data cube VLT/SPHERE-IFS-2 of the data challenge with $W = 7$ and $T = 16$, selecting frames equally spaced in time and in wavelength. A planet was injected with a constant flux of 1 at $\frac{10\lambda}{D}$ of angular separation (encircled in grey). The SNR score evaluated at the position of the planet is given in parentheses.

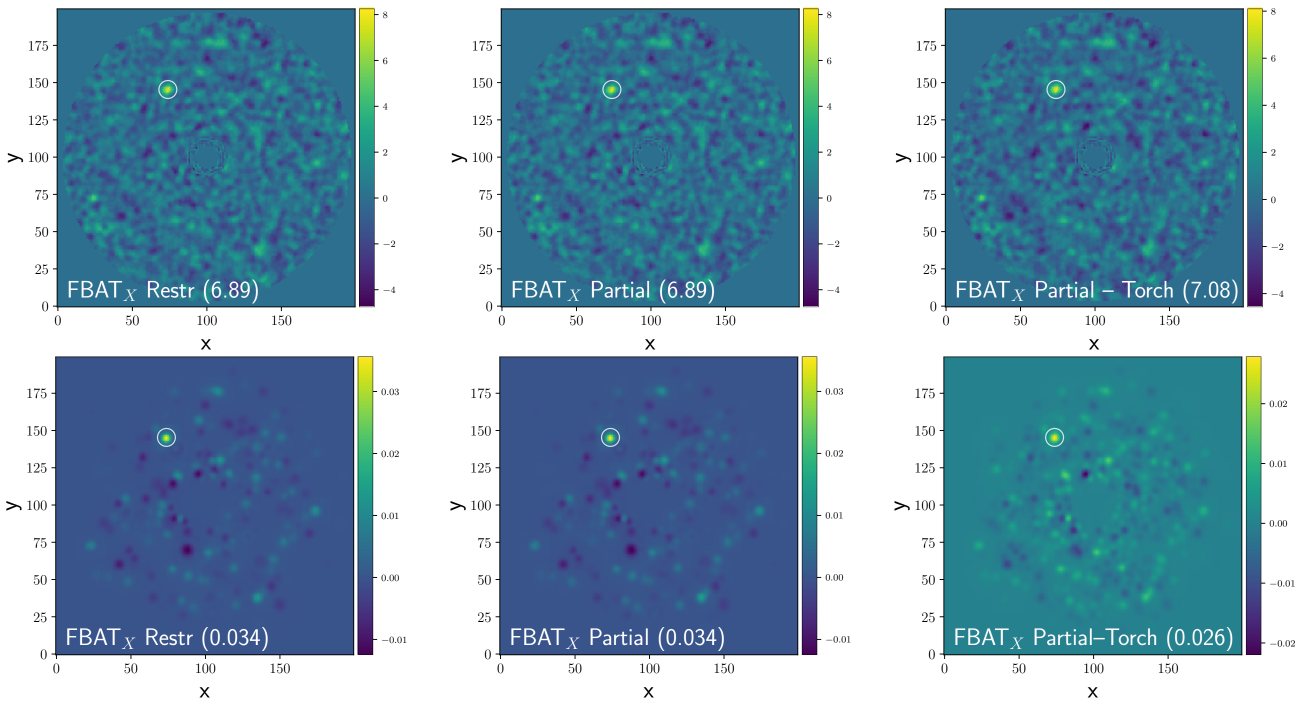

Figure 5.7: SNR map (top) and flux map (bottom) of the FBAT$_X$ inverse problem method with $r = 40$ and $\alpha = 0.5$ applied on the same data cube as PCA-ASDI in figure 5.6. The partial restriction mode, restriction mode and a PyTorch implementation of the partial restriction mode (shown from left to right) are used. In parenthesis is given the SNR score (top) and the flux score (bottom) evaluated at the position of the planet.

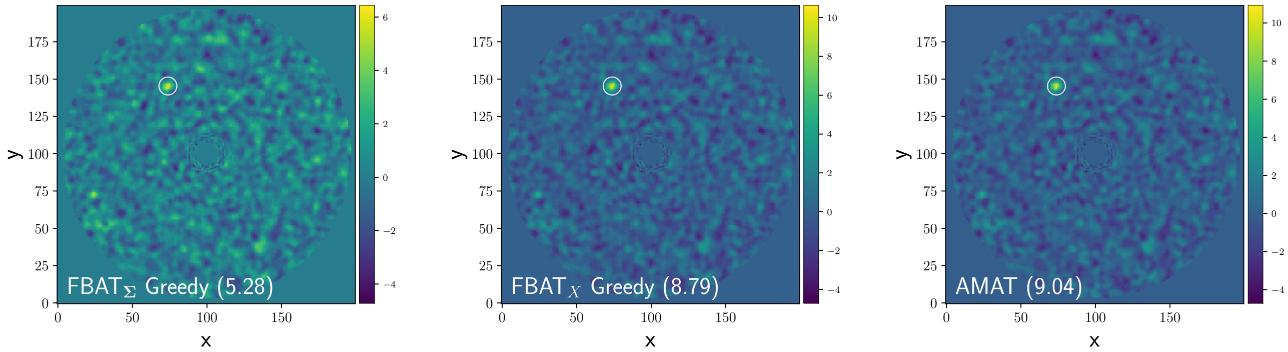

Figure 5.8: SNR map obtained with greedy methods applied on the same data cube as PCA-ASDI in figure 5.6, with $r = 40$ and $\alpha = 0$. The SNR score evaluated at the position of the planet is given in parentheses.

..

Chapter 6

Results and discussion

In this last chapter, we compare the performance of three ASDI algorithms : single step PCA-ASDI, FBAT_X, FBAT_X-Greedy. We will also study the impact of the detection map for FBAT_X, denoting them FBAT_X-Flux and FBAT_X-SNR, for the flux map and the SNR map respectively.

6.1 Preliminaries

The data cube

The data cube used in this chapter will be the empty version of VLT/SPHERE-IFS-2 introduced in section 2.2 of chapter 2. Two selections of spectro-temporal frames will be adopted : $(W, T) = (20, 6)$ or $(W, T) = (7, 16)$ where frames are sampled at indexes equally spaced. We will call them "Configuration $(20, 6)$" and "Configuration $(7, 16)$". Hence, we have 120 frames for the first configuration against 112 frames for the second one. Firstly, this allows to lower the running time of the algorithms and to avoid memory overload. Secondly, it can show whether SDI data are more significant than ADI data to perform the detection or conversely. Thirdly, we select frames equally spaced to limit the phenomenon of "self-subtraction", which occurs when the planet does not move enough through frames to be distinguishable from the background. This phenomenon especially appears at small angular separation.

The mask mode

Regarding the mask, we will use the partial restriction mode involving $\mathcal{P}'_{\Omega_{\text{ann}}}(.)$, that is, working in an annulus of inner radius $\tilde{r}_{\text{in}} = 5\bar{\lambda}/D$ and outer radius $\tilde{r}_{\text{out}} = 20\bar{\lambda}/D$ since it was observed to be the most efficient in terms of computational cost without impacting the results. Consequently, the PCA algorithm can be seen as the initialization of the FBAT_X-like algorithms.

It is important to notice that the radii $\tilde{r}_{\text{in}}$ and $\tilde{r}_{\text{out}}$ define an annular mask in the preprocessed data cube $\boldsymbol{Z}$. Therefore, in the original data cube $\boldsymbol{Y}$, our range of detection goes from $\sim 3\bar{\lambda}/D$ to $\sim 20\bar{\lambda}/D$ since we have $\tilde{r}_{\text{in}}/s(\lambda) \in [3\bar{\lambda}/D, 5\bar{\lambda}/D]$ and $\tilde{r}_{\text{out}}/s(\lambda) \in [12\bar{\lambda}/D, 20\bar{\lambda}/D]$ for the data cube VLT/SPHERE-IFS-2 (with spectral frames of equally spaced indexes). In particular, when the planet is at angular separation $4\bar{\lambda}/D$ or $14\bar{\lambda}/D$ in the original data cube, it sometimes leaves the field of view. Thanks to our definition of the restriction mode, this only makes the detection more difficult, but it does not impose the flux model $\boldsymbol{S}$ to be zero in the spectral channels where the planet leaves the field of view.

6.2 Receiver operating characteristic curves

In order to assess and compare the performance of our ASDI algorithms, we successively generate and post-process with each algorithm 25 data cubes containing a single planet injected at a random azimuthal position, in the empty data cube VLT/SHPERE-IFS-2. Injected planets have a constant flux of $1, 1.5, 2, 2.5$ and an angular separation of $4\bar{\lambda}/D$, $6\bar{\lambda}/D$, $8\bar{\lambda}/D, 10\bar{\lambda}/D, 12\bar{\lambda}/D, 14\bar{\lambda}/D$. The flux value here refers to the one used in the command `vip.fm.cube_inject_companions` of VIP. To better visualize these choices of angular separations, recall that grey circles in figure 5.1 of chapter 5 define the resolution elements for which the TP,FN,FP,TN are evaluated. These are centered from $3.5\bar{\lambda}/D$ to $14\bar{\lambda}/D$.

6.2.1 Tuning the hyperparameters

Before applying our algorithms, we have at most two hyperparameters to tune: (i) the rank r of the low-rank matrix $\boldsymbol{L}$ representing the speckle pattern and (ii) the regularization constant α that promotes sparsity of the circumstellar signal $\boldsymbol{S}$. For PCA, only the rank has to be tuned.

Using the ROC curves given in figure A.1 in the appendix, the best rank for Configuration $(20, 6)$ and Configuration $(7, 16)$ are set to $r = 50$ and $r = 40$ respectively, by selecting the highest TPR with a zero FPR among all the possible ranks. In practice, for Configuration $(20, 6)$, the choice $r = 30$ could also be valid while for Configuration $(7, 16)$, one could set $r = 50$. That is why, to not penalize PCA, we have selected the same ranks for the other algorithms.

Regarding the regularization parameter $\alpha > 0$ of FBAT$_X$, it was observed that $\alpha = 1$ yields almost the same output as PCA (on toy-examples similar to section 5.4 of chapter 5). We thus test several values between 0 and 1, i.e $\alpha \in \{0, 0.1, 0.25, 0.5, 0.75, 0.9\}$. Finally, for FBAT$_X$-Greedy, since we consider batches of size 50, we expect to impose a lighter sparsity constraint than FBAT$_X$. Hence, we try $\alpha \in \{0, 0.1, 0.25, 0.5\}$. The best parameters have been determined using the other ROC curves provided in appendix A.5 and are displayed in table 6.1.

Algorithm	Configuration $(20, 6)$	Configuration $(7, 16)$
PCA	$r = 50$	$r = 40$
FBAT$_X$-Flux	$r = 50$ $\alpha = 0.25$	$r = 40$ $\alpha = 0.5$
FBAT$_X$-SNR	$r = 50$ $\alpha = 0.5$	$r = 40$ $\alpha = 0.5$
FBAT$_X$-Greedy	$r = 50$ $\alpha = 0.25$ # threads $= 20$ batchsize $\lvert\Lambda_b\rvert = 50$	$r = 40$ $\alpha = 0.5$ # threads $= 20$ batchsize $\lvert\Lambda_b\rvert = 50$

Table 6.1: Table of hyperparameters. The number of threads and the batchsize for FBAT$_X$-Greedy comes from table 4.2.

6.2.2 Running time

One should note that each method does not have the same running time. Since PCA is the initialization of FBAT$_X$, it is the fastest. On the other hand, FBAT$_X$-Greedy is the slowest since we have to run the optimization algorithm once for each batch. Table 6.2 gives the mean and the standard deviation of the running time of each algorithm over all the data cubes of Configuration $(20, 6)$.

Algorithm	Mean running time	Standard deviation
PCA	60.33s	1.96s
FBAT$_X$-Flux	437.72s	21.39s
FBAT$_X$-SNR	/	/
FBAT$_X$-Greedy	1022.62s	41.11s

Table 6.2: Mean and standard deviation of the running time over the $4 \cdot 6 \cdot 25 = 600$ data cubes of Configuration $(20, 6)$ on the `lemaitre3` CECI cluster.

In this table, the 60 seconds of PCA mainly come from the computation of the SNR map. Moreover, the time of FBAT$_X$-SNR was not displayed since the SNR map was computed separately using the residual map of FBAT$_X$-Flux, but we expect that the running time would have been approximately the same as FBAT$_X$-Flux plus 60 seconds. Finally, this table also shows that obtaining one single ROC curve of FBAT$_X$-Greedy roughly takes 7 hours on average against 25 minutes for PCA.

6.2.3 Results

The results are displayed in three parts. First, figures 6.1 and 6.2 show the ROC curves of each algorithm at small angular separation $4\bar{\lambda}/D$ with a varying flux. Then, in figure 6.3 and 6.4, we provide the ROC curves for a small flux of 1 and for several angular separations. Finally, in figure 6.5 and 6.6, we show curves with a flux of 1.5. We do not display curves of higher fluxes for angular separations higher than $4\bar{\lambda}/D$ because they are the same as when Flux $= 1.5$. The small numbers in color indicate the threshold used at a certain point of the curve.

Small angular separation

At small angular separation, we expect to make the detection more challenging for the reasons previously mentioned, i.e. the higher self-subtraction and the fact that the planet leaves the field of view.

From figure 6.1, one can observe that both FBAT$_X$-Flux and FBAT$_X$-Greedy outperform PCA. Even if the difference is not large, we indeed see that the green and the blue ROC curves are above the black curve.

Regarding Configuration $(7, 16)$ in figure 6.2, this difference is far more pronounced. FBAT$_X$-Flux largely outperforms all the other algorithms. For example, for flux $= 2$, it reaches a $\sqrt{\text{TPR}}$ of 0.6 with almost a zero $\sqrt{\text{FPR}}$ against $\sqrt{\text{TPR}} \approx 0.35$ for PCA. This behavior could come from the sparsity of the flux model $\boldsymbol{S}$, yielding a possibly sparse flux detection map.

It is interesting to look at the difference between Configuration $(20, 6)$ and Configuration $(7, 16)$. We indeed notice that the second one allows to reach a higher $\sqrt{\text{TPR}}$ with zero $\sqrt{\text{FPR}}$, especially for FBAT$_X$-Flux. In fact, this difference could be expected since the planet leaves the field of view in certain spectral frames. Hence, more ADI frames means more data available while more SDI frames does not necessarily provide relevant frames containing the planet.

Finally, in both configurations, FBAT$_X$-SNR does not perform far better than PCA. Looking at the threshold values, we see that both PCA and FBAT$_X$-SNR achieve a similar SNR at the same location on the ROC curve. These two observations could be explained by the fact that the flux model $\boldsymbol{S}$ (returned by FBAT$_X$-SNR) has very small entries and hence the background model $\boldsymbol{L}$ is almost the same as the one provided by PCA. Residual maps are thus similar for both PCA and FBAT$_X$-SNR, which yields similar SNR maps and ROC curves.

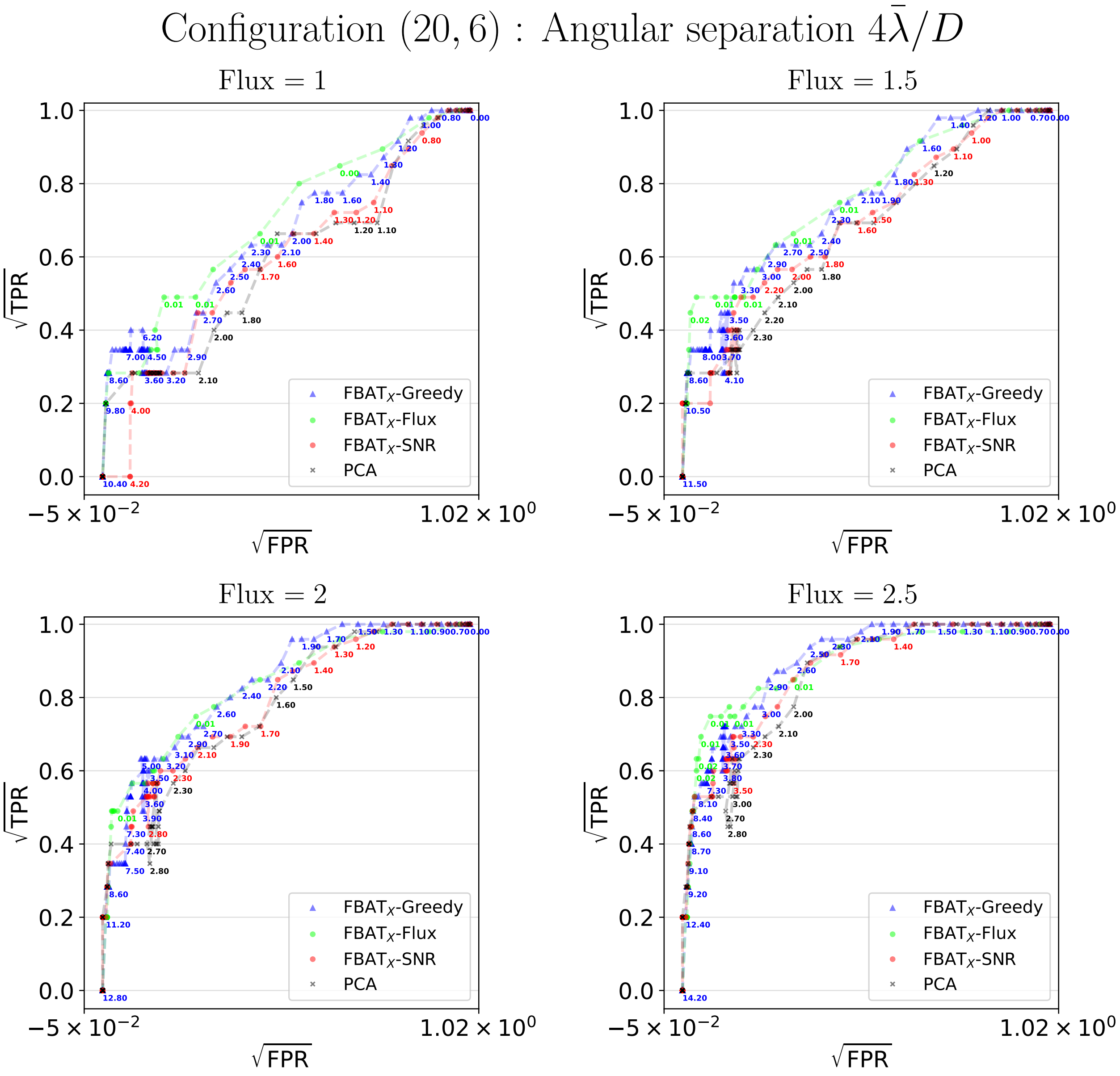

Figure 6.1: PCA, FBAT$_X$-FLux, FBAT$_X$-SNR and FBAT$_X$-Greedy applied on configuration $(20, 6)$ at small angular separation $4\bar{\lambda}/D$ with injected planets of flux $1, 1.5, 2, 2.5$.

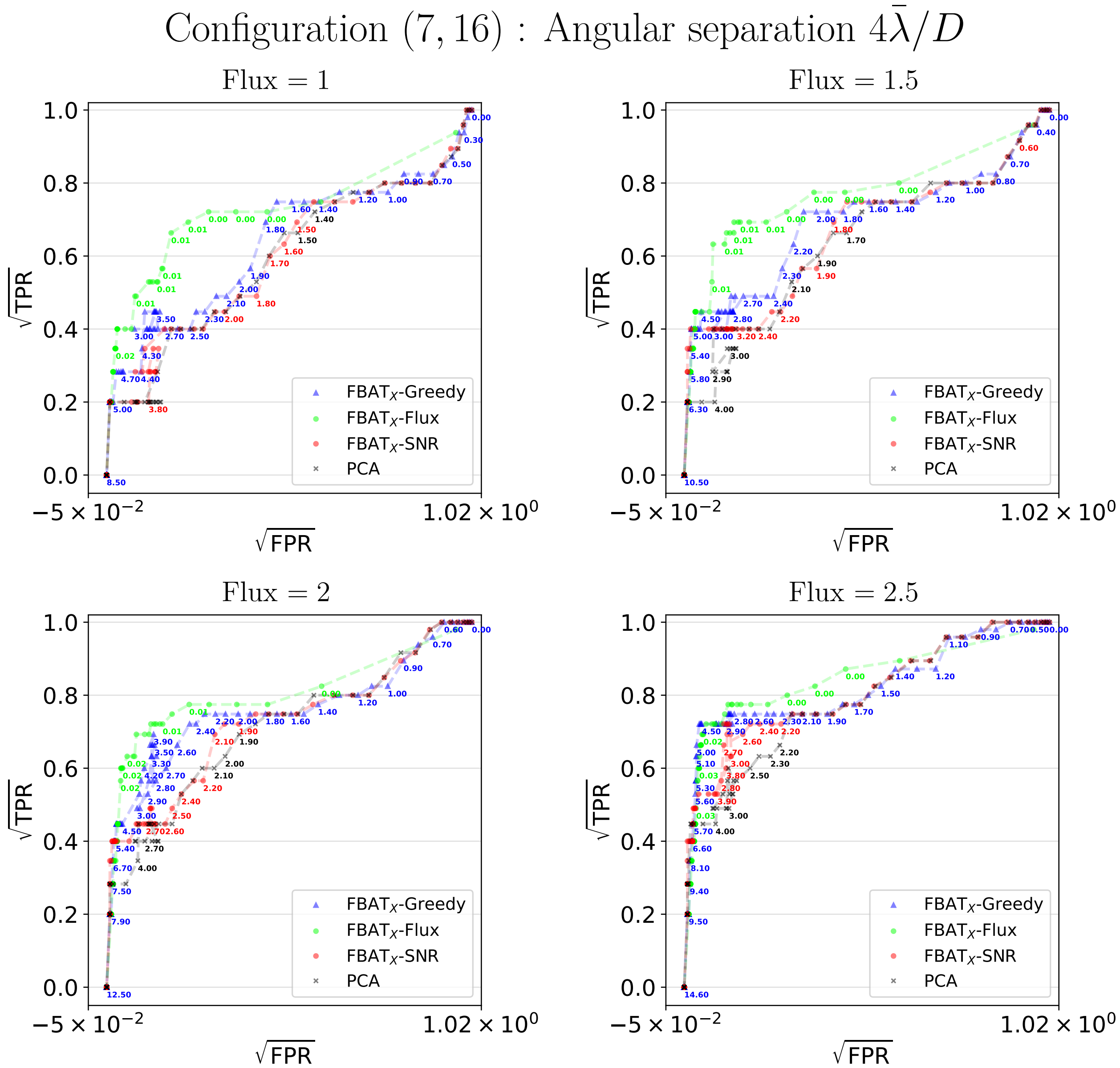

Figure 6.2: PCA, FBAT$_X$-Flux, FBAT$_X$-SNR and FBAT$_X$-Greedy applied on configuration $(7, 16)$ at small angular separation $4\bar{\lambda}/D$ with injected planets of flux $1, 1.5, 2, 2.5$.

Small flux

For angular separations greater than $4\bar{\lambda}/D$ with a small flux, all the methods are almost equivalent, except at $6\bar{\lambda}/D$ and $14\bar{\lambda}/D$. At these angular separations, in Configuration $(20, 6)$, this is FBAT$_X$-Flux that achieves the highest $\sqrt{\text{TPR}}$ score at zero $\sqrt{\text{FPR}}$. For example, it reaches ~ 0.62 againt ~ 0.45 for the other methods at $6\bar{\lambda}/D$. In contrast, for Configuration $(7, 16)$, this is now FBAT$_X$-Greedy that performs better especially at $14\bar{\lambda}/D$.

Another notable difference between FBAT$_X$-Greedy and PCA is the threshold values on the SNR at a given point of the ROC curve. For example, in Configuration $(20, 6)$, when $6\bar{\lambda}/D$ at $\sqrt{\text{TPR}} \approx 0.43$, the threshold on the SNR of FBAT$_X$-Greedy amounts to 7.2 while the one of PCA is only 3.8. As mentioned in the previous chapter, FBAT$_X$-Greedy indeed allows to largely boost the SNR exactly as AMAT, but on the other hand, it also increases the SNR at several positions free of planets. This explains why FBAT$_X$-Greedy does not largely outperform PCA (and is sometimes even slightly worst for some angular separations).

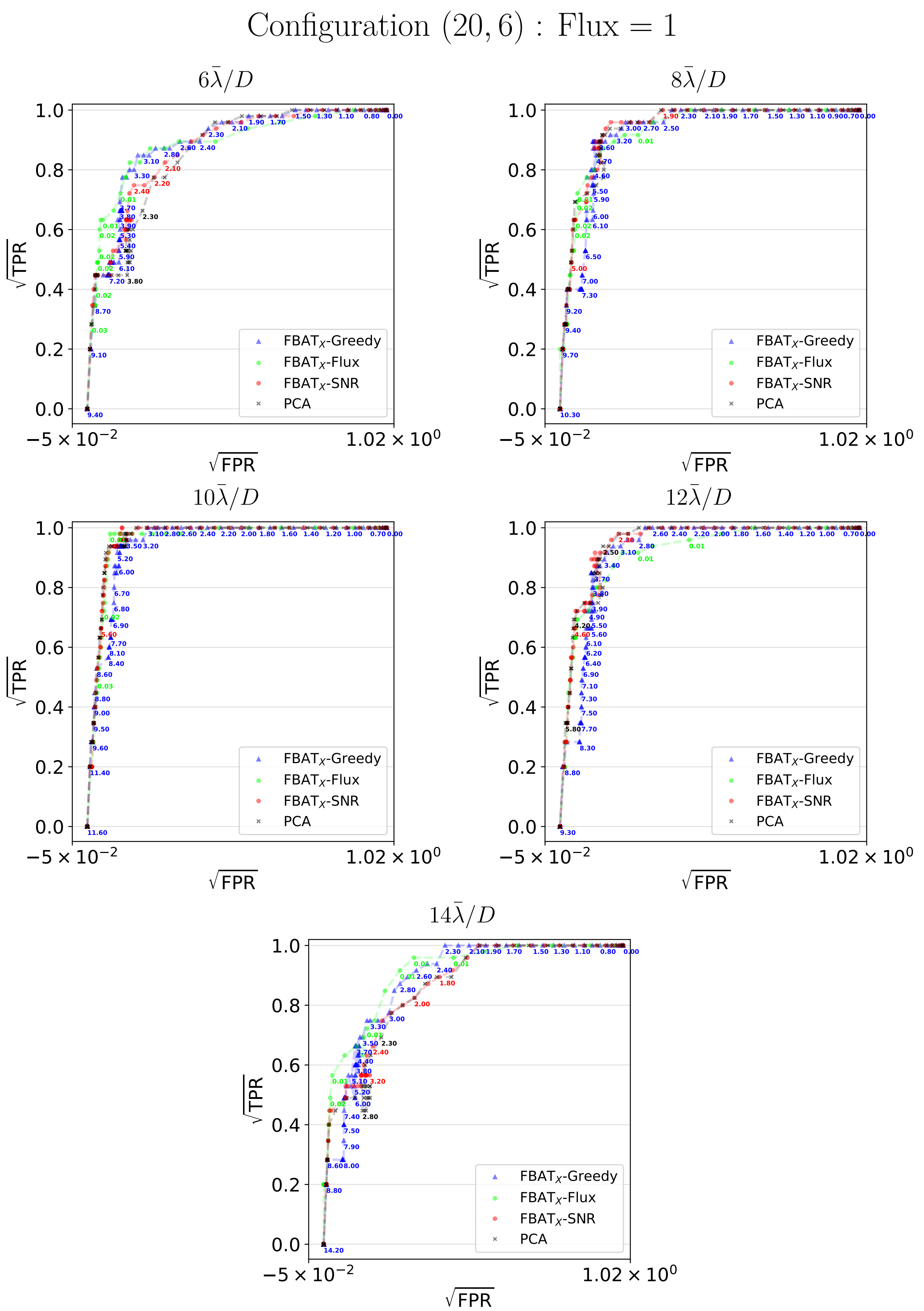

Figure 6.3: PCA, FBAT$_X$-Flux, FBAT$_X$-SNR and FBAT$_X$-Greedy applied on configuration $(20, 6)$ at varying angular separation from $6\bar{\lambda}/D$ to $14\bar{\lambda}/D$ with injected planets of flux 1.

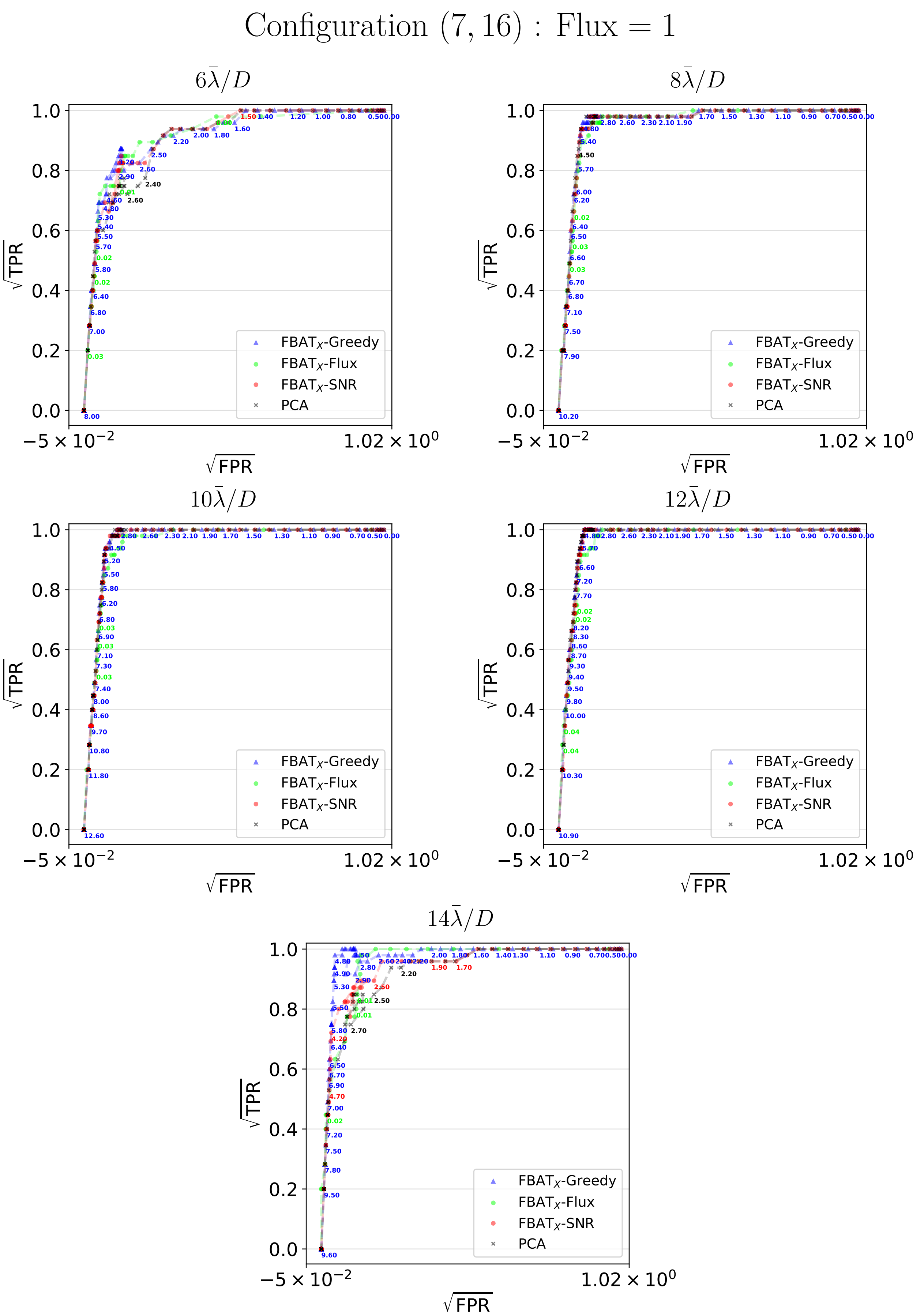

Figure 6.4: PCA, FBAT$_X$-Flux, FBAT$_X$-SNR and FBAT$_X$-Greedy applied on configuration (7, 16) at varying angular separation from $6\bar{\lambda}/D$ to $14\bar{\lambda}/D$ with injected planets of flux 1.

Other angular separations with Flux $= 0.5$

When the flux of the injected planet is greater or equal 1.5, all the ROC curves almost overlap. Moreover, in Configuration $(7, 16)$, they reach a $\sqrt{\text{TPR}} \approx 1$ at small $\sqrt{\text{FPR}}$ and thus all the methods tend to be perfect classifiers. We do not display all the angular separations since the curves are similar. We only show the one at $6\bar{\lambda}/D$ and $14\bar{\lambda}/D$.

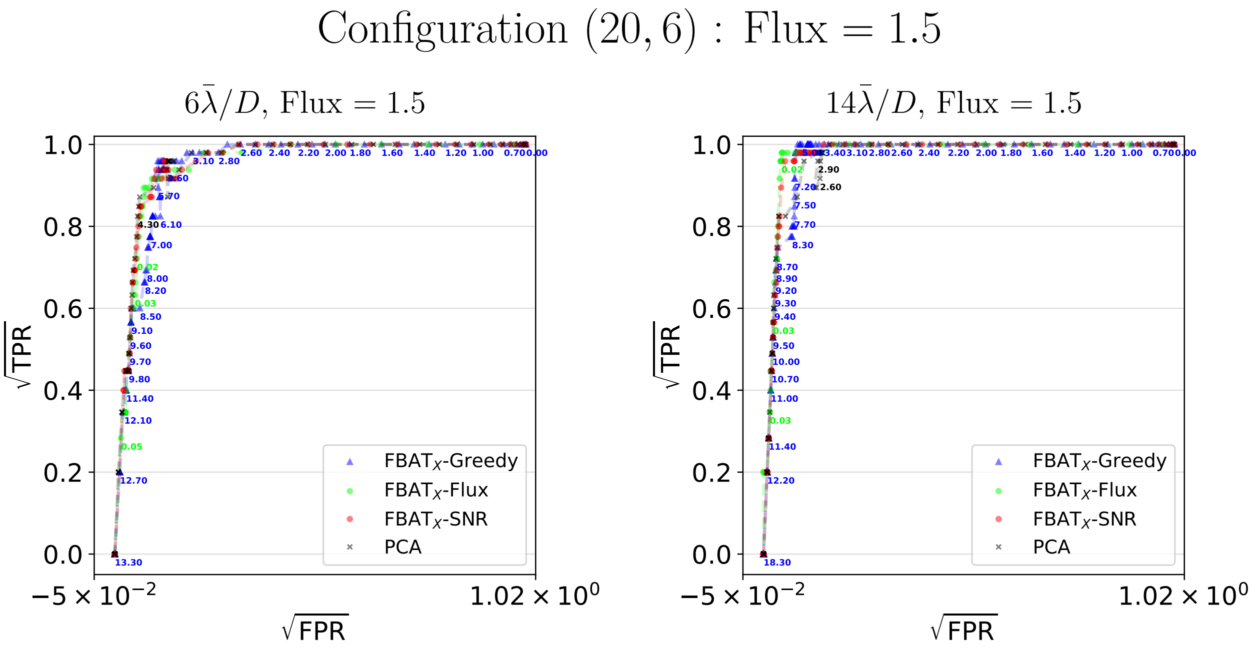

Figure 6.5: PCA, FBAT$_X$-Flux, FBAT$_X$-SNR and FBAT$_X$-Greedy applied on configuration $(20, 6)$ at varying angular separation of $6\bar{\lambda}/D$ and $14\bar{\lambda}/D$ with injected planets of flux 1.

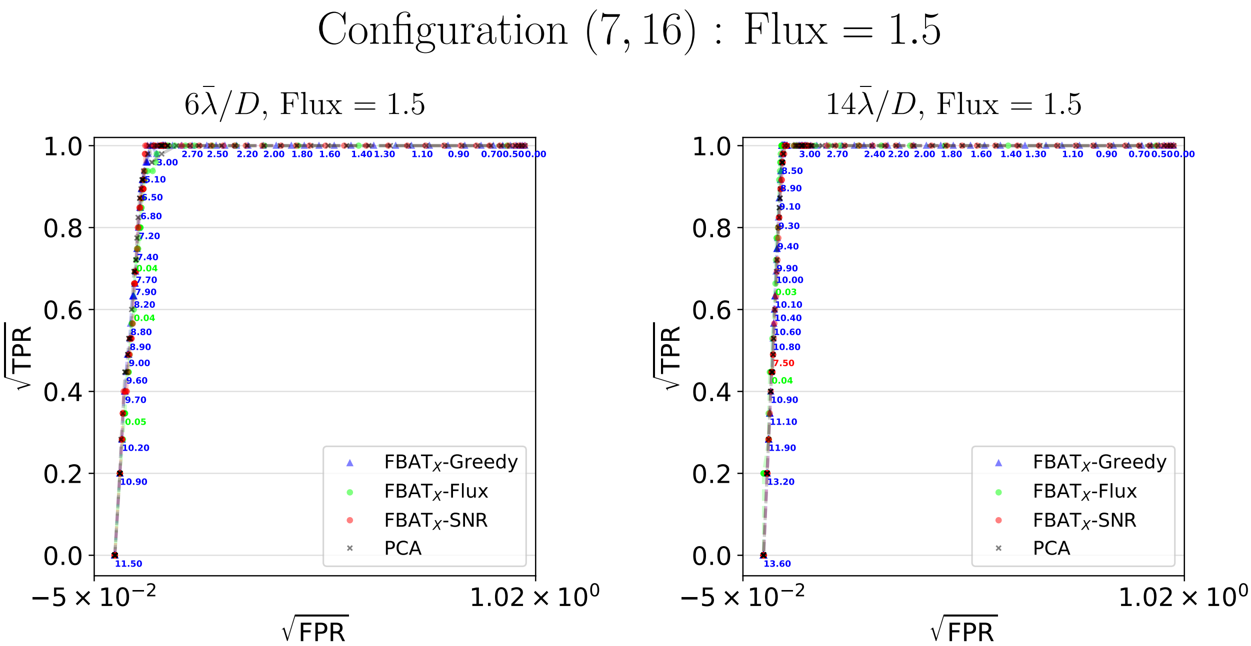

Figure 6.6: PCA, FBAT$_X$-Flux, FBAT$_X$-SNR and FBAT$_X$-Greedy appliedon on configuration $(7, 16)$ at varying angular separation of $6\bar{\lambda}/D$ and $14\bar{\lambda}/D$ with injected planets of flux 1.

Conclusion

Summary

In this Master's thesis, we have investigated the extension of low-rank models to ASDI data cubes. After analyzing the VLT/SPHERE-IFS-2 data cube, we proposed an inverse problem approach to solve the detection problem, which led to the FBAT_X optimization model. Several variants of FBAT_X have been derived, namely the greedy model FBAT_X-Greedy, which relies on a decomposition of the trajectorlet operator in batches. Finally, these algorithms have been compared with the single step PCA-ASDI algorithm as reference through a ROC curves analysis using both the flux map and the SNR map.

It turns out that all the algorithms correctly detect the injected companions and have similar performance at large angular separation for fluxes greater than 1. On the other hand, at small angular separation, both FBAT_X-Flux and FBAT_X-Greedy outperform PCA. While two different configurations have been studied, our analysis has shown that the difference between both was not significant, except at small angular separation. However, a more complete characterization of those algorithms should be made, for example, with smaller injected fluxes to increase the challenge of the detection.

Possible improvements

At several steps of the modeling process of ASDI data cubes, some simplifying assumptions have been adopted, such as the linear scale factor and the gaussian noise model for the non-static part of the speckle pattern. A possible improvement for the scale factor would be to add a regularization prior, such as a law $\propto 1/\lambda^2$ that would replace the initial law in $\propto 1/\lambda$. Regarding the noise model, a more detailed study of the statistics of the quasi-static speckles in ASDI data cubes would be require. While a laplacian noise model has provided better results in ADI, it is not clear that the same behavior will be observed for ASDI. Once an accurate noise model is available, we will be able to adapt the Likelihood Ratio map for ASDI to perform the detection.

Moreover, several areas of the data cube have not been modeled, where the most important part should be the AO-halo. Proposing an accurate model of the latter (for example with a different scale factor) should allow to detect exoplanets at smaller angular separations.

Finally, due to limited memory resources and computational capabilities, only a small portion of the data was effectively used. A possible suggestion to use all the information could be to rely on a Stochastic Gradient Descent with a division in batches (inspired from the training of Neural Networks). Furthermore, only a small subspace of the hyperparameter space has been explored. The Bayesian Optimization framework as proposed in auto-RSM [60] could be helpful to find the best hyperparameters.

Appendix A

A.1 Derivation of the adjoints operators

A.1.1 Rotation

Let $F : \mathbb{R} \times \mathbb{R}^2 \mapsto \mathbb{R}$ and $G : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2 \mapsto \mathbb{R}$ with finite support corresponding to the ASDI datacube domain. Then,

$$\begin{aligned}
\langle \mathcal{Q}_{\boldsymbol{\alpha}} F, G \rangle &= \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda, \boldsymbol{Q}_{\alpha(t)} \boldsymbol{p}) G(\lambda, t, \boldsymbol{p}) \ \mathrm{d}\lambda \, \mathrm{d}t \, \mathrm{d}\boldsymbol{p} \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda, \boldsymbol{p'}) G(\lambda, t', \boldsymbol{Q}_{\alpha(t')}^{-1} \boldsymbol{p'}) \ \mathrm{d}\lambda \left| \frac{\partial(t, \boldsymbol{p})}{\partial(t', \boldsymbol{p'})} \right| \mathrm{d}t' \, \mathrm{d}\boldsymbol{p'} \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda, \boldsymbol{p'}) \left(\int_{\mathbb{R}} G(\lambda, t', \boldsymbol{Q}_{-\alpha(t')} \boldsymbol{p'}) \, \mathrm{d}t' \right) \ \mathrm{d}\lambda \, \mathrm{d}\boldsymbol{p'} \\
&= \langle F, \mathcal{Q}_{\boldsymbol{\alpha}}^* G \rangle
\end{aligned}$$

where we made the change of variable :

$$\begin{cases} t' = t \\ \boldsymbol{p'} = \boldsymbol{Q}_{\alpha(t)} \boldsymbol{p} \end{cases} \tag{A.1}$$

and

$$\left| \frac{\partial(t, \boldsymbol{p})}{\partial(t', \boldsymbol{p'})} \right| = \left| \det \begin{pmatrix} 1 & 0 & 0 \\ * & \cos(\alpha(t')) & -\sin(\alpha(t')) \\ * & \sin(\alpha(t')) & \cos(\alpha(t')) \end{pmatrix} \right| = 1 \tag{A.2}$$

In discrete time, the integral over t' becomes a sum. Hence,

$$\mathcal{Q}_{\boldsymbol{\alpha}}^* : \mathbb{R}^{WT \times N^2} \mapsto \mathbb{R}^{W \times N^2} \equiv \text{discrete version of } G(\lambda, t, \boldsymbol{p}) \mapsto \sum_{t'=1}^{T} G(\lambda, t', \boldsymbol{Q}_{-\alpha(t')} \boldsymbol{p}). \tag{A.3}$$

A.1.2 Scaling

Let $F, G : \mathbb{R} \times \mathbb{R} \times \mathbb{R}^2 \mapsto \mathbb{R}$ with finite support corresponding to the ASDI datacube domain. Then,

$$\begin{aligned}
\langle \mathcal{D}_{\boldsymbol{s}} F, G \rangle &= \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda, t, s(\lambda)^{-1} \boldsymbol{p}) G(\lambda, t, \boldsymbol{p}) \ \mathrm{d}\lambda \, \mathrm{d}t \, \mathrm{d}\boldsymbol{p} \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda', t, \boldsymbol{p'}) G(\lambda', t, s(\lambda') \boldsymbol{p'}) \ \mathrm{d}t \left| \frac{\partial(\lambda, \boldsymbol{p})}{\partial(\lambda', \boldsymbol{p'})} \right| \mathrm{d}\lambda' \, \mathrm{d}\boldsymbol{p'} \\
&= \int_{\mathbb{R}} \int_{\mathbb{R}} \int_{\mathbb{R}^2} F(\lambda', t, \boldsymbol{p'}) \left(s(\lambda')^2 G(\lambda', t, s(\lambda') \boldsymbol{p'}) \right) \ \mathrm{d}\lambda' \, \mathrm{d}t \, \mathrm{d}\boldsymbol{p'} \\
&= \langle F, \mathcal{D}_{\boldsymbol{s}}^* G \rangle
\end{aligned}$$

where we made the change of variable :

$$\begin{cases} \lambda' = \lambda \\ \boldsymbol{p'} = s(\lambda)^{-1}\boldsymbol{p} \end{cases} \tag{A.4}$$

and

$$\left|\frac{\partial(\lambda,\boldsymbol{p})}{\partial(\lambda',\boldsymbol{p'})}\right| = \left|\det\begin{pmatrix} 1 & 0 & 0 \\ * & s(\lambda') & 0 \\ * & 0 & s(\lambda')\end{pmatrix}\right| = s(\lambda')^2. \tag{A.5}$$

Hence, one should have

$$\mathcal{D}_{\boldsymbol{s}}^* : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{WT\times N^2} \equiv \text{discrete version of } G(\lambda,t,\boldsymbol{p}) \mapsto s(\lambda)^2 G(\lambda,t,s(\lambda)\boldsymbol{p}). \tag{A.6}$$

While this indeed provides the true adjoint in the continuous case, this is not the case in practice when using `opencv`. Using this adjoint makes the proximal algorithms diverge. Moreover, it does not at all satisfies the definition of the adjoint when tested on a real data cube. This can be due to the use of a zero-padding in the scaling operator. Indeed, with this zero-padding, the operator exactly acts as if we had kept fixed the first frame and down-scaled the other frames. The first frame completely fills the $n\times n$ image while the last frame appears smaller. Therefore, the meaning of the coefficient $s(\lambda)^2$ is ambiguous. By replacing the multiplicative coefficient by $s(\lambda)^{-2}$ in (A.7), it turns out that the following adjoint works correctly (see table A.1) :

$$\mathcal{D}_{\boldsymbol{s}}^* : \mathbb{R}^{WT\times n^2} \mapsto \mathbb{R}^{WT\times N^2} \equiv \text{discrete version of } G(\lambda,t,\boldsymbol{p}) \mapsto s(\lambda)^{-2} G(\lambda,t,s(\lambda)\boldsymbol{p}). \tag{A.7}$$

A.1.3 Convolution

Let $F, G : \mathbb{R}\times\mathbb{R}^2 \mapsto \mathbb{R}$ with finite support corresponding to a SDI datacube domain. Then,

$$\begin{aligned}
\langle \mathcal{T}_{\boldsymbol{\varphi}} F, G\rangle &= \int_{\mathbb{R}}\int_{\mathbb{R}^2} \left(\int_{\mathbb{R}^2} F(\lambda,\boldsymbol{p'})\varphi(\boldsymbol{p}-\boldsymbol{p'})\ \mathrm{d}\boldsymbol{p'}\right) G(\lambda,\boldsymbol{p})\ \mathrm{d}\lambda\,\mathrm{d}\boldsymbol{p} \\
&= \int_{\mathbb{R}}\int_{\mathbb{R}^2} F(\lambda,\boldsymbol{p'}) \left(\int_{\mathbb{R}^2} G(\lambda,\boldsymbol{p})\varphi(\boldsymbol{p}-\boldsymbol{p'})\,\mathrm{d}\boldsymbol{p}\right)\ \mathrm{d}\lambda\,\mathrm{d}\boldsymbol{p'} \\
&= \langle F, \mathcal{T}_{\boldsymbol{\varphi}}^* G\rangle.
\end{aligned}$$

Hence,

$$\mathcal{T}_{\boldsymbol{\varphi}}^* : \mathbb{R}^{W\times N^2} \mapsto \mathbb{R}^{W\times N^2} \equiv \text{discrete version of } G(\lambda,\boldsymbol{p}) \mapsto \int_{\mathbb{R}^2} G(\lambda,\boldsymbol{p'})\varphi(\boldsymbol{p'}-\boldsymbol{p})\,\mathrm{d}\boldsymbol{p'}, \tag{A.8}$$

which corresponds to the discrete version of the correlation operator.

A.1.4 Trajectorlet

There are two definitions of the trajectorlet operator :

1. $\Psi(.) = \mathcal{D}_{\boldsymbol{s}}\mathcal{Q}_{\boldsymbol{\alpha}}\mathcal{T}_{\boldsymbol{\varphi}}(.)$
2. $\Psi_b(.) = \sum_{g\in\Lambda_b}\sum_{\lambda=1}^{W}\langle .,\boldsymbol{\delta}_{\lambda,g}\rangle \boldsymbol{P}_{\lambda,g}$

Their adjoints are respectively :

(1)

$$\Psi^*(.) = \mathcal{T}_{\boldsymbol{\varphi}}^*\mathcal{Q}_{\boldsymbol{\alpha}}^*\mathcal{D}_{\boldsymbol{s}}^*(.),$$

(2)

$$\begin{aligned}
\langle \Psi_b(\boldsymbol{S}), \boldsymbol{Z}\rangle &= \left\langle \sum_{g\in\Lambda_b}\sum_{\lambda=1}^{W}\langle \boldsymbol{S}, \boldsymbol{\delta}_{\lambda,g}\rangle \boldsymbol{P}_{\lambda,g}, \boldsymbol{Z}\right\rangle \\
&= \sum_{g\in\Lambda_b}\sum_{\lambda=1}^{W}\langle \boldsymbol{S}, \boldsymbol{\delta}_{\lambda,g}\rangle\langle \boldsymbol{P}_{\lambda,g}, \boldsymbol{Z}\rangle \\
&= \left\langle \boldsymbol{S}, \sum_{g\in\Lambda_b}\sum_{\lambda=1}^{W}\langle \boldsymbol{P}_{\lambda,g}, \boldsymbol{Z}\rangle \boldsymbol{\delta}_{\lambda,g}\right\rangle \\
&= \langle \boldsymbol{S}, \Psi_b^*(\boldsymbol{Z})\rangle,
\end{aligned}$$

which yields :

$$\Psi_b^*(.) = \sum_{g\in\Lambda_b}\sum_{\lambda=1}^{W}\langle \boldsymbol{P}_{\lambda,g}, .\rangle \boldsymbol{\delta}_{\lambda,g}.$$

Remark: While $\Psi^*(.)$ is the theoretical adjoint, it satisfies its definition up to some numerical errors due to the interpolation. Table (A.1) shows some numerical experiments to illustrate to claim.

$\langle \Psi(\boldsymbol{S}), \boldsymbol{Z}\rangle$	$\langle \boldsymbol{S}, \Psi^*(\boldsymbol{Z})\rangle$
-198.93	-201.36
305.04	301.86
271.13	271.29
28.233	25.292
84.206	82.244
-91.744	-92.086
44.311	43.6
-201.05	-200.95

Table A.1: $\boldsymbol{S}$ and $\boldsymbol{Z}$ randomly generated with Gaussian i.i.d. entries. The mean absolute error over 100 simulations amounts to 2.974 representing a relative error of 5%.

A.2 The proximal gradient methods for FBAT

In this appendix, we provide the proof of (4.12) and (4.14).

FBAT$_\Sigma$

We consider the optimization problem defined in (4.12) that is rewritten below.

$$(\text{FBAT}_\Sigma) \quad \min_{\boldsymbol{\Sigma}\in\mathbb{R}^{r\times r}, \boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{\Sigma}\boldsymbol{V}_r^\top - \Psi(\boldsymbol{S})\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}. \tag{A.9}$$

We decompose this objective function in two terms :

$$\begin{aligned} f(\boldsymbol{S}, \boldsymbol{\Sigma}) &= \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{\Sigma}\boldsymbol{V}_r^\top - \Psi(\boldsymbol{S})\|_2^2 \\ g(\boldsymbol{S}, \boldsymbol{\Sigma}) &= \alpha\|\boldsymbol{S}\|_{2,1}. \end{aligned}$$

Recall the proximal gradient method :

$$(\boldsymbol{S}^{(i+1)}, \boldsymbol{\Sigma}^{(i+1)}) = \text{prox}_{\mu' g}\left((\boldsymbol{S}^{(i)}, \boldsymbol{\Sigma}^{(i)}) - \mu\nabla f(\boldsymbol{S}^{(i)}, \boldsymbol{\Sigma}^{(i)})\right) \tag{A.10}$$

We thus need to compute the gradient of $f(.,.)$ and the proximal operator of $g(.,.)$. On the one hand, by the chain rule, one can compute the gradient as follows :

$$\begin{array}{rcl} \dfrac{\partial f}{\partial \boldsymbol{S}}(\boldsymbol{S}, \boldsymbol{\Sigma}) &=& \Psi^*(\boldsymbol{R}(\boldsymbol{S}, \boldsymbol{\Sigma})) \\ \dfrac{\partial f}{\partial \boldsymbol{\Sigma}}(\boldsymbol{S}, \boldsymbol{\Sigma}) &=& \boldsymbol{V}_r^\top \boldsymbol{R}(\boldsymbol{S}, \boldsymbol{\Sigma})\boldsymbol{U}_r \\ \text{with } \boldsymbol{R}(\boldsymbol{S}, \boldsymbol{\Sigma}) &=& \Psi(\boldsymbol{S}) + \boldsymbol{U}_r\boldsymbol{\Sigma}\boldsymbol{V}_r^\top - \boldsymbol{Z} \end{array} \tag{A.11}$$

On the other hand, regarding the proximal operator, we have by definition :

$$\begin{aligned} \text{prox}_{\mu' g}(\boldsymbol{S}, \boldsymbol{\Sigma}) &= \arg\min_{\boldsymbol{\Sigma}'\in\mathbb{R}^{r\times r}, \boldsymbol{S}'\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\left(\|\boldsymbol{S} - \boldsymbol{S}'\|_2^2 + \|\boldsymbol{\Sigma} - \boldsymbol{\Sigma}'\|_2^2\right) + \mu'\|\boldsymbol{S}\|_{2,1} \\ &= (\text{prox}_{\mu'\|.\|_{2,1}}(\boldsymbol{S}), \text{prox}_0(\boldsymbol{\Sigma})). \end{aligned}$$

The definition of $\text{prox}_{\mu'\|.\|_{2,1}}(.)$ is given in appendix A.3, while by definition

$$\text{prox}_0(.) = (\mathcal{I} + 0)^{-1}(.) = \mathcal{I}.$$

Combining those results with (A.11) proves (4.12).

FBAT$_X$

We consider now the second optimization problem defined in (4.14) :

$$(\text{FBAT}_X) \quad \min_{\boldsymbol{X}\in\mathbb{R}^{n^2\times r}, \boldsymbol{S}\in\mathbb{R}^{W\times N^2}} \frac{1}{2}\|\boldsymbol{Z} - \boldsymbol{U}_r\boldsymbol{X}^\top - \Psi(\boldsymbol{S})\|_2^2 + \alpha\|\boldsymbol{S}\|_{2,1}. \tag{A.12}$$

Following the same arguments as above, one can directly write :

$$\begin{array}{rcl} \dfrac{\partial f}{\partial \boldsymbol{S}}(\boldsymbol{S}, \boldsymbol{X}) &=& \Psi^*(\boldsymbol{R}(\boldsymbol{S}, \boldsymbol{X})) \\ \dfrac{\partial f}{\partial \boldsymbol{X}}(\boldsymbol{S}, \boldsymbol{X}) &=& \boldsymbol{R}(\boldsymbol{S}, \boldsymbol{X})^\top \boldsymbol{U}_r \\ \text{with } \boldsymbol{R}(\boldsymbol{S}, \boldsymbol{X}) &=& \Psi(\boldsymbol{S}) + \boldsymbol{U}_r\boldsymbol{X}^\top - \boldsymbol{Z} \end{array} \tag{A.13}$$

and

$$\text{prox}_{\mu' g}(\boldsymbol{S}, \boldsymbol{X}) = (\text{prox}_{\mu'\|.\|_{2,1}}(\boldsymbol{S}), \text{prox}_0(\boldsymbol{X})). \tag{A.14}$$

A.3 Derivation of the proximal operator of $\|.\|_{2,1}$

Our goal is to derive a closed form for $\text{prox}_{\alpha\|.\|_{2,1}}(.) = (\mathcal{I} + \alpha\partial\|.\|_{2,1})^{-1}$ for some $\alpha > 0$. By separability of $\|.\|_{2,1}$, one first needs to compute $\text{prox}_{\alpha\|.\|_2}(.)$ and then to apply this operator in a column-wise manner. To justify this claim, we use the second definition of the proximal operator. Let $\boldsymbol{Y} = [\boldsymbol{y}_j]_{j=1}^n \in \mathbb{R}^{m\times n}$, then

$$\begin{aligned}
\text{prox}_{\alpha\|.\|_{2,1}}(\boldsymbol{Y}) &:= \arg\min_{\boldsymbol{X}\in\mathbb{R}^{m\times n}} \frac{1}{2}\|\boldsymbol{Y} - \boldsymbol{X}\|_2^2 + \alpha\|\boldsymbol{X}\|_{2,1} \\
&= \arg\min_{\boldsymbol{X}\in\mathbb{R}^{m\times n}} \sum_{j=1}^n \frac{1}{2}\|\boldsymbol{y}_j - \boldsymbol{x}_j\|_2^2 + \alpha\|\boldsymbol{x}_j\|_2 \\
&= \left[\arg\min_{\boldsymbol{x}_j\in\mathbb{R}^m} \frac{1}{2}\|\boldsymbol{y}_j - \boldsymbol{x}_j\|_2^2 + \alpha\|\boldsymbol{x}_j\|_2\right]_{j=1}^n \\
&= \left[\text{prox}_{\alpha\|.\|_2}(\boldsymbol{y}_1) \quad \cdots \quad \text{prox}_{\alpha\|.\|_2}(\boldsymbol{y}_n)\right].
\end{aligned}$$

We will now show that, for some $\boldsymbol{y} \in \mathbb{R}^m$,

$$\text{prox}_{\alpha\|.\|_2}(\boldsymbol{y}) = \begin{cases} \boldsymbol{y}\left(1 - \frac{\alpha}{\|\boldsymbol{y}\|_2}\right) & \text{if } \|\boldsymbol{y}\|_2 > \alpha \\ \mathbf{0} & \text{if } \|\boldsymbol{y}\|_2 \leq \alpha. \end{cases} \tag{A.15}$$

To this end, we use the first definition of the proximal operator : $\text{prox}_{\alpha\|.\|_2}(.) := (\mathcal{I} + \alpha\|.\|_2)^{-1}$. The first step is to find an expression for $\alpha\partial\|.\|_2$. For $\boldsymbol{x} \in \mathbb{R}^m$, this is directly obtained as :

$$\alpha\partial\|\boldsymbol{x}\|_2 = \begin{cases} \alpha\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} & \text{if } \boldsymbol{a}_j \neq \mathbf{0} \\ B_\alpha(\mathbf{0}) & \text{otherwise} \end{cases}$$

where $B_\alpha(\mathbf{0}) := \{\boldsymbol{g} \in \mathbb{R}^m : \|\boldsymbol{g}\|_2 \leq \alpha\}$. Indeed, when $\boldsymbol{x} \neq \mathbf{0}$, the gradient of the ℓ_2-norm exists and is the unique subgradient of this function in $\boldsymbol{x}$. When $\boldsymbol{x} = \mathbf{0}$, we can check that for any $\boldsymbol{g} \in B_\alpha(\mathbf{0})$, this vector satisfies the definition of a subgradient in $\alpha\partial\|\mathbf{0}\|_2$:

$$\forall \boldsymbol{y} \in \mathbb{R}^m, \alpha\|\boldsymbol{y}\|_2 \geq \|\boldsymbol{g}\|_2\|\boldsymbol{y}\|_2 \geq \alpha\|\mathbf{0}\|_2 + \langle \boldsymbol{g}, \boldsymbol{y} - \mathbf{0}\rangle.$$

To find the proximal operator, we now have to invert $\mathcal{I} + \alpha\partial\|.\|_2$. Therefore, given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^m\backslash\{\mathbf{0}\}$, we compute :

$$\begin{aligned}
(\mathcal{I} + \alpha\partial\|.\|_2)\boldsymbol{x} &= \boldsymbol{y} \\
\boldsymbol{x} + \alpha\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|_2} &= \boldsymbol{y} \\
\|\boldsymbol{x}\|_2\left(1 + \frac{\alpha}{\|\boldsymbol{x}\|_2}\right) &= \|\boldsymbol{y}\|_2 \\
\|\boldsymbol{x}\|_2 &= \|\boldsymbol{y}\|_2 - \alpha
\end{aligned} \tag{A.16}$$

which yields

$$\begin{aligned}
\boldsymbol{x} &= \boldsymbol{y}\left(1 + \frac{\alpha}{\|\boldsymbol{y}\|_2 - \alpha}\right)^{-1} \\
&= \boldsymbol{y}\left(\frac{\|\boldsymbol{y}\|_2 - \alpha}{\|\boldsymbol{y}\|_2}\right) \\
&= \boldsymbol{y}\left(1 - \frac{\alpha}{\|\boldsymbol{y}\|_2}\right).
\end{aligned} \tag{A.17}$$

Then,

$$\boldsymbol{x} = \mathbf{0} \Rightarrow \boldsymbol{y} \in B_\alpha(\mathbf{0}). \tag{A.18}$$

Combining (A.17) and (A.18) proves what we wanted to show.

A.4 Update rule in the accelerated gradient method

In the original accelerated proximal gradient method, the update of the momentum vector $\boldsymbol{v}^{(i)}$ is performed by minimizing a certain potential function $\psi^{(i)}$ defined as follows :

$$\begin{aligned}
\psi^{(i+1)}(\boldsymbol{x}) &= \psi^{(i)}(\boldsymbol{x}) + \theta^{(i)}\left(f(\boldsymbol{u}^{(i)}) + \langle \nabla f(\boldsymbol{u}^{(i)}), \boldsymbol{x} - \boldsymbol{u}^{(i)}\rangle + g(\boldsymbol{x})\right)\\
\psi^{(0)}(\boldsymbol{x}) &= \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|_2^2\\
\boldsymbol{v}^{(i+1)} &= \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \psi^{(i+1)}(\boldsymbol{x}).
\end{aligned}$$

This definition allows to proof the accelerated rate of convergence of this method in $\mathcal{O}(\varepsilon^{-1/2})$. In this appendix, we show that the minimizer of the potential function $\psi^{(i+1)}$ has a close from that can be written as a prox. First, we compute $\psi^{(i+1)}(\boldsymbol{x})$ as

$$\begin{aligned}
\psi^{(i+1)}(\boldsymbol{x}) &= \psi^{(0)}(\boldsymbol{x}) + \sum_{k=0}^{i} \theta^{(k)}(f(\boldsymbol{u}^{(k)}) + \langle \nabla f(\boldsymbol{u}^{(k)}), \boldsymbol{x} - \boldsymbol{u}^{(k)}\rangle + g(\boldsymbol{x}))\\
&= \psi^{(0)}(\boldsymbol{x}) + \underbrace{\left(\sum_{k=0}^{i} \theta^{(k)}(f(\boldsymbol{u}^{(k)}) + \langle \nabla f(\boldsymbol{u}^{(k)}), \boldsymbol{x}^{(0)} - \boldsymbol{u}^{(k)}\rangle)\right)}_{:=\bar{\boldsymbol{z}}^{(i+1)}}\\
&\quad + \left\langle \underbrace{\sum_{k=0}^{i} \theta^{(k)} \nabla f(\boldsymbol{u}^{(k)})}_{:=\Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}}, \boldsymbol{x} - \boldsymbol{x}^{(0)}\right\rangle + \left(\sum_{k=0}^{i} \theta^{(k)}\right) g(\boldsymbol{x})\\
&= \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|_2^2 + \bar{\boldsymbol{z}}^{(i+1)} + \langle \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}, \boldsymbol{x} - \boldsymbol{x}^{(0)}\rangle + \Theta^{(i+1)} g(\boldsymbol{x}).
\end{aligned}$$

Notice that

$$\begin{aligned}
\sum_{k=0}^{i} \theta^{(k)} \nabla f(\boldsymbol{u}^{(k)}) &= \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}\\
\Leftrightarrow \theta^{(i)} \nabla f(\boldsymbol{u}^{(i)}) &= \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)} - \Theta^{(i)}\bar{\boldsymbol{g}}^{(i)}\\
\Leftrightarrow \bar{\boldsymbol{g}}^{(i+1)} &= \frac{\Theta^{(i)}}{\Theta^{(i+1)}}\bar{\boldsymbol{g}}^{(i)} + \frac{\theta^{(i)}}{\Theta^{(i+1)}}\nabla f(\boldsymbol{u}^{(i)})\\
&= (1 - \gamma^{(i)})\bar{\boldsymbol{g}}^{(i)} + \gamma^{(i)}\nabla f(\boldsymbol{u}^{(i)}).
\end{aligned}$$

Finally, it follows that

$$\begin{aligned}
\boldsymbol{v}^{(i+1)} &= \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|_2^2 + \bar{\boldsymbol{z}}^{(i+1)} + \langle \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}, \boldsymbol{x} - \boldsymbol{x}^{(0)}\rangle + \Theta^{(i+1)} g(\boldsymbol{x})\right\}\\
&= \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{x}^{(0)}\|_2^2 + \langle \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}, \boldsymbol{x} - \boldsymbol{x}^{(0)}\rangle + \Theta^{(i+1)} g(\boldsymbol{x})\right\}\\
&= \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\frac{1}{2}\|\boldsymbol{x} - (\boldsymbol{x}^{(0)} - \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)})\|_2^2 + \Theta^{(i+1)} g(\boldsymbol{x})\right\}\\
&= \operatorname{prox}_{\Theta^{(i+1)}g}\left(\boldsymbol{x}^{(0)} - \Theta^{(i+1)}\bar{\boldsymbol{g}}^{(i+1)}\right)
\end{aligned}$$

where the last equality comes from the second definition of the proximal operator :

$$\operatorname{prox}_g(\boldsymbol{y}) := \arg\min_{\boldsymbol{x}\in\mathbb{R}^n} \left\{\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{y}\|_2^2 + g(\boldsymbol{x})\right\}. \tag{A.19}$$

A.5 Hyperparameters tuning

A.5.1 Selecting the rank of PCA

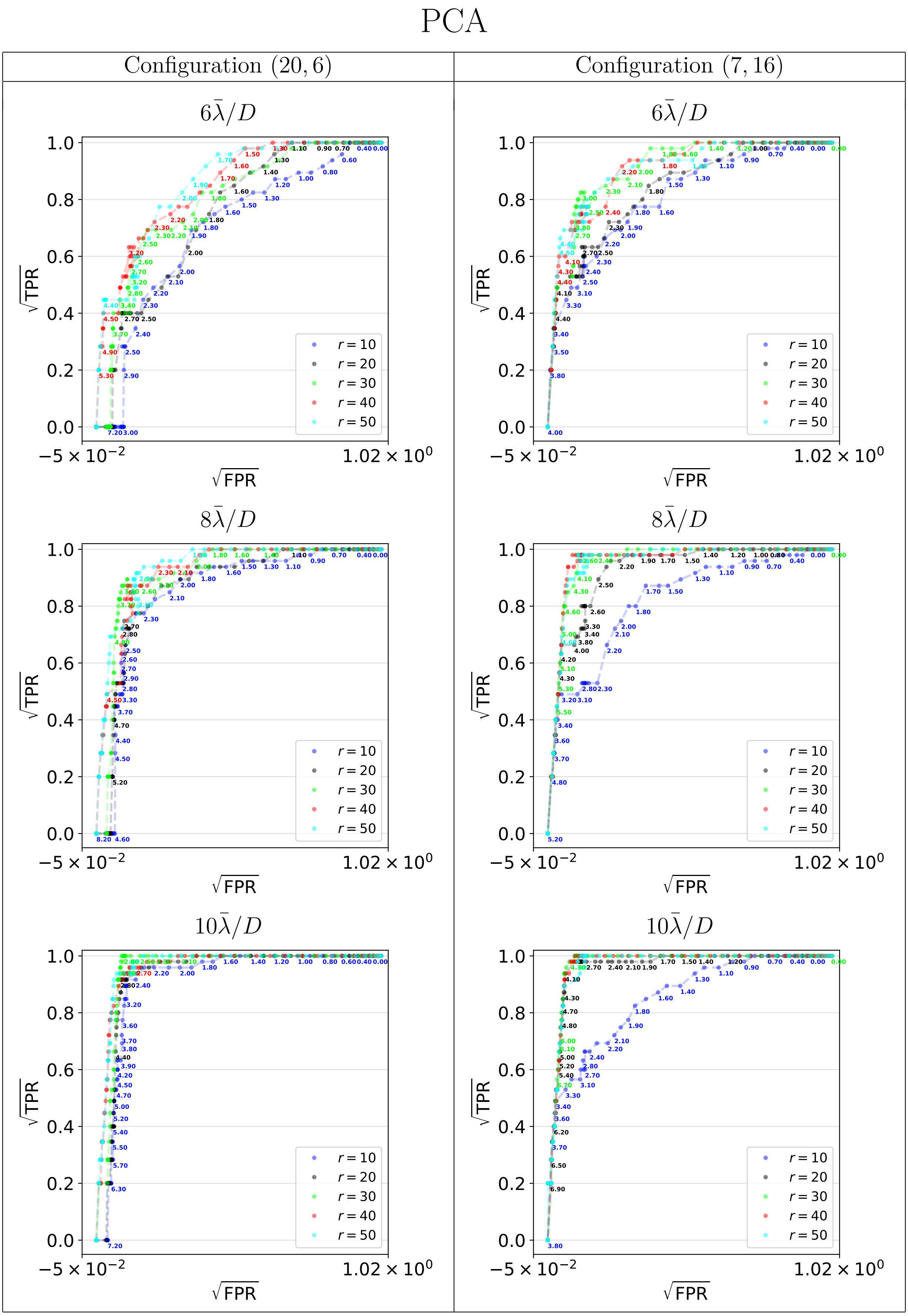

Figure A.1: PCA applied on both configurations with Flux = 1. Angular separation from top to bottom : $6\bar{\lambda}/D$, $8\bar{\lambda}/D$ and $10\bar{\lambda}/D$.

A.5.2 Selecting the α of FBAT$_X$-Flux

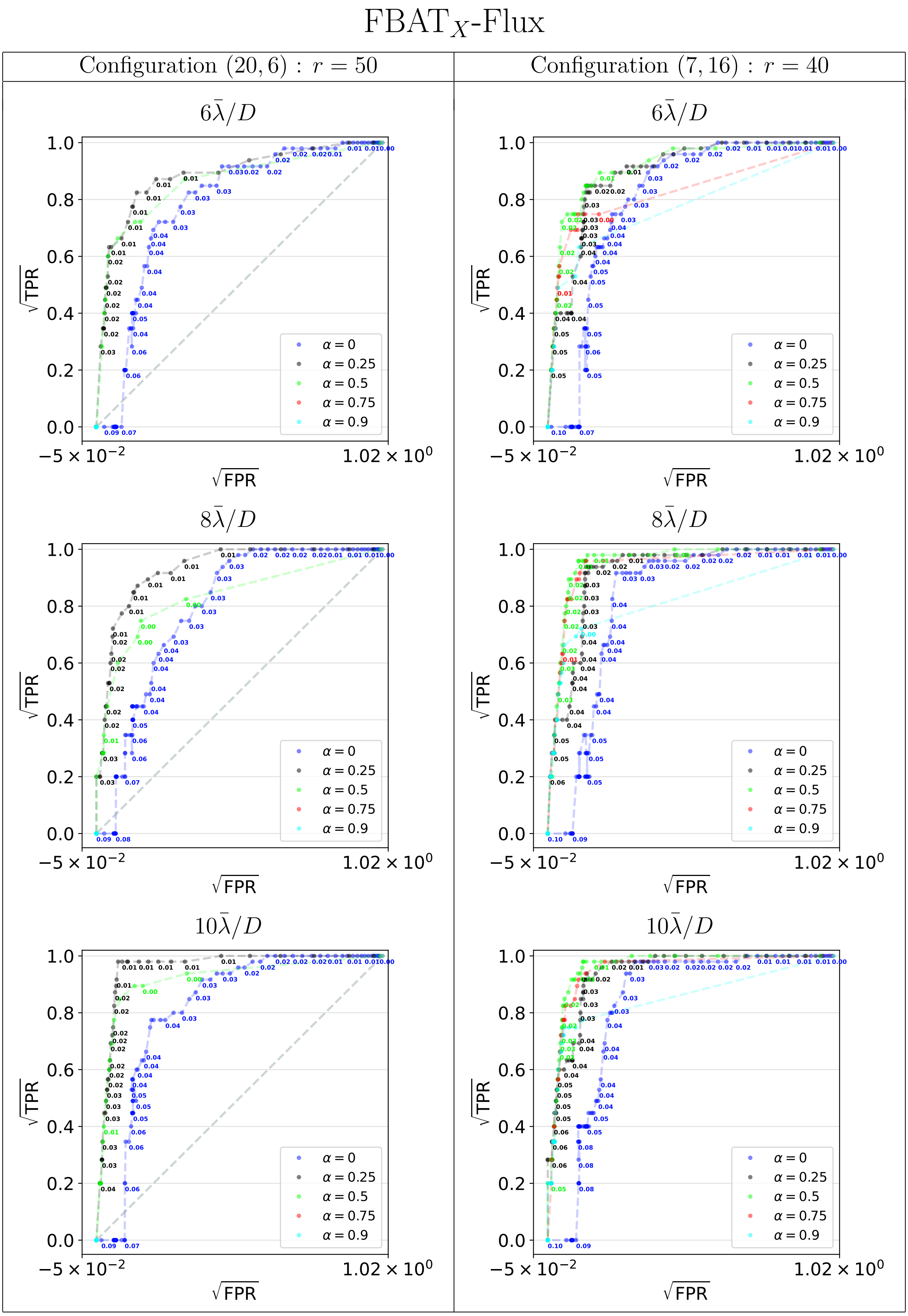

Figure A.2: FBAT$_X$-Flux applied on both configurations with Flux $= 1$. Angular separation from top to bottom : $6\bar{\lambda}/D$, $8\bar{\lambda}/D$ and $10\bar{\lambda}/D$.

A.5.3 Selecting the α of FBAT$_X$-SNR

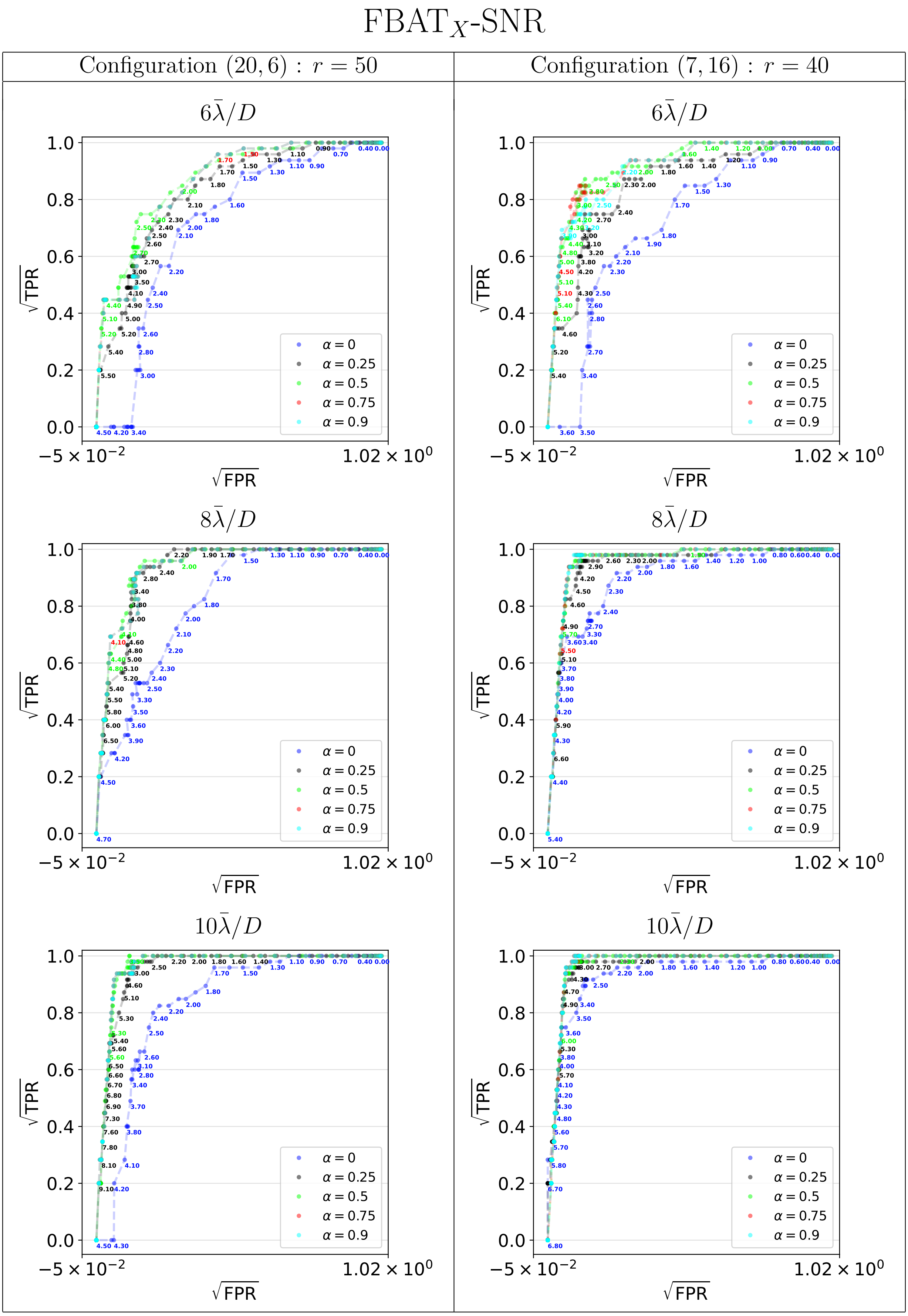

Figure A.3: FBAT$_X$-SNR applied on both configurations with Flux = 1. Angular separation from top to bottom : $6\bar{\lambda}/D$, $8\bar{\lambda}/D$ and $10\bar{\lambda}/D$.

A.5.4 Selecting the α of FBAT$_X$-Greedy

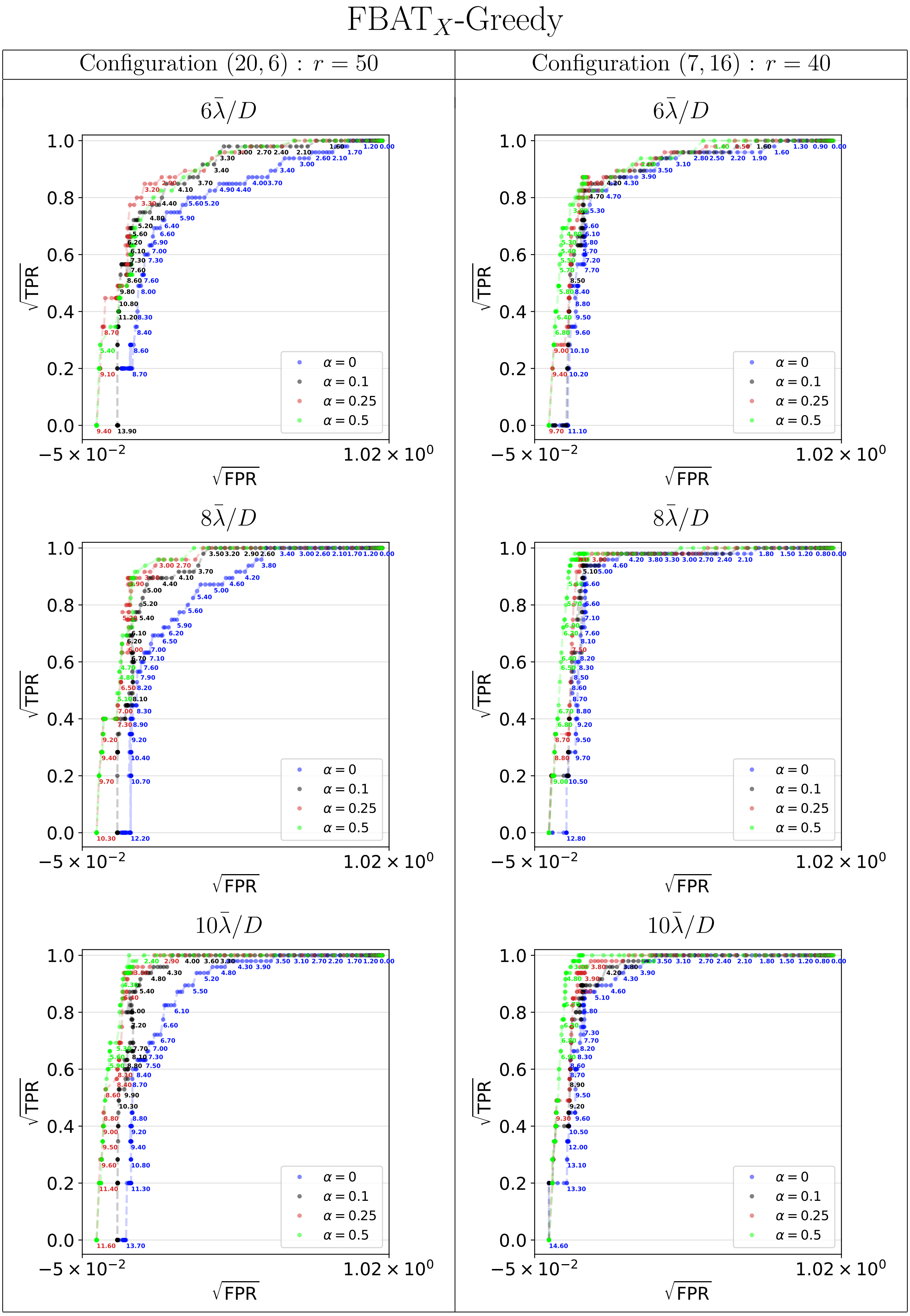

Figure A.4: FBAT$_X$-Greedy applied on both configurations with Flux $= 1$. Angular separation from top to bottom : $6\bar{\lambda}/D$, $8\bar{\lambda}/D$ and $10\bar{\lambda}/D$.

Bibliography

[1] Hans J. Deeg and Roi Alonso. Transit photometry as an exoplanet discovery method, 2024.

[2] C. Lovis and D. Fischer. Radial Velocity Techniques for Exoplanets. In S. Seager, editor, *Exoplanets*, pages 27–53. 2010.

[3] Thayne Currie, Beth Biller, Anne-Marie Lagrange, Christian Marois, Olivier Guyon, Eric Nielsen, Mickael Bonnefoy, and Robert De Rosa. Direct imaging and spectroscopy of extrasolar planets, 2023.

[4] Benoît Pairet. *Signal processing methods for high-contrast observations of planetary systems.* PhD thesis, Ecole polytechnique de Louvain, Université catholique de Louvain, 2012. Prom. : Jacques, Laurent ; Absil, Pierre-Antoine.

[5] Katherine B. Follette. An introduction to high contrast differential imaging of exoplanets and disks. *Publications of the Astronomical Society of the Pacific*, 135(1051):093001, sep 2023.

[6] Paolo Padovani and Michele Cirasuolo. The Extremely Large Telescope. *Contemporary Physics*, 64(1):47–64, January 2023.

[7] Aarynn L. Carter, Sasha Hinkley, Jens Kammerer, Andrew Skemer, Beth A. Biller, Jarron M. Leisenring, Maxwell A. Millar-Blanchaer, Simon Petrus, Jordan M. Stone, Kimberly Ward-Duong, Jason J. Wang, Julien H. Girard, Dean C. Hines, Marshall D. Perrin, Laurent Pueyo, William O. Balmer, Mariangela Bonavita, Mickael Bonnefoy, Gael Chauvin, Elodie Choquet, Valentin Christiaens, Camilla Danielski, Grant M. Kennedy, Elisabeth C. Matthews, Brittany E. Miles, Polychronis Patapis, Shrishmoy Ray, Emily Rickman, Steph Sallum, Karl R. Stapelfeldt, Niall Whiteford, Yifan Zhou, Olivier Absil, Anthony Boccaletti, Mark Booth, Brendan P. Bowler, Christine H. Chen, Thayne Currie, Jonathan J. Fortney, Carol A. Grady, Alexandra Z. Greebaum, Thomas Henning, Kielan K. W. Hoch, Markus Janson, Paul Kalas, Matthew A. Kenworthy, Pierre Kervella, Adam L. Kraus, Pierre-Olivier Lagage, Michael C. Liu, Bruce Macintosh, Sebastian Marino, Mark S. Marley, Christian Marois, Brenda C. Matthews, Dimitri Mawet, Michael W. McElwain, Stanimir Metchev, Michael R. Meyer, Paul Molliere, Sarah E. Moran, Caroline V. Morley, Sagnick Mukherjee, Eric Pantin, Andreas Quirrenbach, Isabel Rebollido, Bin B. Ren, Glenn Schneider, Malavika Vasist, Kadin Worthen, Mark C. Wyatt, Zackery W. Briesemeister, Marta L. Bryan, Per Calissendorff, Faustine Cantalloube, Gabriele Cugno, Matthew De Furio, Trent J. Dupuy, Samuel M. Factor, Jacqueline K. Faherty, Michael P. Fitzgerald, Kyle Franson, Eileen C. Gonzales, Callie E. Hood, Alex R. Howe, Masayuki Kuzuhara, Anne-Marie Lagrange, Kellen Lawson, Cecilia Lazzoni, Ben W. P. Lew, Pengyu Liu, Jorge Llop-Sayson, James P. Lloyd, Raquel A. Martinez, Johan Mazoyer, Paulina Palma-Bifani, Sascha P. Quanz, Jea Adams Redai, Matthias Samland, Joshua E. Schlieder, Motohide Tamura, Xianyu Tan, Taichi Uyama, Arthur Vigan, Jo-

hanna M. Vos, Kevin Wagner, Schuyler G. Wolff, Marie Ygouf, Xi Zhang, Keming Zhang, and Zhoujian Zhang. The jwst early release science program for direct observations of exoplanetary systems i: High-contrast imaging of the exoplanet hip 65426 b from 2 to 16 µm. *The Astrophysical Journal Letters*, 951(1):L20, July 2023.

[8] Simon Vary, Hazan Daglayan, Laurent Jacques, and Pierre-Antoine Absil. Low-rank plus sparse trajectory decomposition for direct exoplanet imaging, 2023.

[9] Hazan Daglayan, Simon Vary, and Pierre-Antoine Absil. An alternating minimization algorithm with trajectory for direct exoplanet detection. pages 417–422, 01 2023.

[10] Benoît Pairet, Faustine Cantalloube, and Laurent Jacques. MAYONNAISE: ADI data imaging processing pipeline. Astrophysics Source Code Library, record ascl:2204.009, April 2022.

[11] Raphaël Galicher and Johan Mazoyer. Imaging exoplanets with coronagraphic instruments. *Comptes Rendus. Physique*, 24(S2):1–45, March 2023.

[12] Laurent Pueyo. *Direct Imaging as a Detection Technique for Exoplanets*, pages 1–61. Springer International Publishing, Cham, 2018.

[13] Jean Surdej. Introduction to optical/ir interferometry: history and basic principles, June 2019.

[14] Nowak, M., Lacour, S., Lagrange, A.-M., Rubini, P., Wang, J., Stolker, T., Abuter, R., Amorim, A., Asensio-Torres, R., Bauböck, M., Benisty, M., Berger, J. P., Beust, H., Blunt, S., Boccaletti, A., Bonnefoy, M., Bonnet, H., Brandner, W., Cantalloube, F., Charnay, B., Choquet, E., Christiaens, V., Clénet, Y., Coudé du Foresto, V., Cridland, A., de Zeeuw, P. T., Dembet, R., Dexter, J., Drescher, A., Duvert, G., Eckart, A., Eisenhauer, F., Gao, F., Garcia, P., Garcia Lopez, R., Gardner, T., Gendron, E., Genzel, R., Gillessen, S., Girard, J., Grandjean, A., Haubois, X., Heißel, G., Henning, T., Hinkley, S., Hippler, S., Horrobin, M., Houllé, M., Hubert, Z., Jiménez-Rosales, A., Jocou, L., Kammerer, J., Kervella, P., Keppler, M., Kreidberg, L., Kulikauskas, M., Lapeyrère, V., Le Bouquin, J.-B., Léna, P., Mérand, A., Maire, A.-L., Mollière, P., Monnier, J. D., Mouillet, D., Müller, A., Nasedkin, E., Ott, T., Otten, G., Paumard, T., Paladini, C., Perraut, K., Perrin, G., Pueyo, L., Pfuhl, O., Rameau, J., Rodet, L., Rodríguez-Coira, G., Rousset, G., Scheithauer, S., Shangguan, J., Stadler, J., Straub, O., Straubmeier, C., Sturm, E., Tacconi, L. J., van Dishoeck, E. F., Vigan, A., Vincent, F., von Fellenberg, S. D., Ward-Duong, K., Widmann, F., Wieprecht, E., Wiezorrek, E., Woillez, J., and the GRAVITY Collaboration. Direct confirmation of the radial-velocity planet β pictoris c. *A& A*, 642:L2, 2020.

[15] Samuel W. Hasinoff. Photon, poisson noise. In *Computer Vision, A Reference Guide*, 2014.

[16] Kevin J. Ludwick. Signal-to-noise ratio for photon counting after photometric corrections. *Journal of Astronomical Telescopes, Instruments, and Systems*, 9(04), December 2023.

[17] Douglas Rabin. Solar coronagraphs. In *The WSPC Handbook of Astronomical Instrumentation.* World Scientific Publishing Co., Singapore, 2020.

[18] Julien Calbert. Learning integral operators from diagonal-circulant neural networks. Master's thesis, Ecole polytechnique de Louvain, Université catholique de Louvain, April 2020. Prom. : Jacques, Laurent ; Absil, Pierre-Antoine.

[19] G. Brussaard. *Handbook on Radiometeorology.* International Telecommunication Union, Place des Nations CH-1211 Geneva 20, 2013 edition.

[20] O. Absil, J. Milli, D. Mawet, A. M. Lagrange, J. Girard, G. Chauvin, A. Boccaletti, C. Delacroix, and J. Surdej. Searching for companions down to 2 AU from β Pictoris using the L'-band AGPM coronagraph on VLT/NACO. *Astronomy and Astrophysics*, 559:L12, November 2013.

[21] Christian Marois, David Lafrenière, René Doyon, Bruce Macintosh, and Daniel Nadeau. Angular differential imaging: A powerful high-contrast imaging technique*. *The Astrophysical Journal*, 641(1):556, apr 2006.

[22] S. Kiefer, A. J. Bohn, S. P. Quanz, M. Kenworthy, and T. Stolker. Spectral and angular differential imaging with sphere/ifs: Assessing the performance of various pca-based approaches to psf subtraction. *Astronomy & amp; Astrophysics*, 652:A33, August 2021.

[23] René Racine, Gordon A. H. Walker, Daniel Nadeau, René Doyon, and Christian Marois. Speckle Noise and the Detection of Faint Companions. 111(759):587–594, May 1999.

[24] V. Christiaens, S. Casassus, O. Absil, F. Cantalloube, C. Gomez Gonzalez, J. Girard, R. Ramírez, B. Pairet, V. Salinas, D. J. Price, C. Pinte, S. P. Quanz, A. Jordán, D. Mawet, and Z. Wahhaj. Separating extended disc features from the protoplanet in PDS 70 using VLT/SINFONI. 486(4):5819–5837, July 2019.

[25] Xiaowei Zhou, Can Yang, Hongyu Zhao, and Weichuan Yu. Low-rank modeling and its applications in image analysis. *ACM Comput. Surv.*, 47(2), dec 2014.

[26] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, 2013.

[27] Rémi Soummer, Laurent Pueyo, and James Larkin. Detection and characterization of exoplanets and disks using projections on karhunen–loÈve eigenimages. *The Astrophysical Journal Letters*, 755(2):L28, aug 2012.

[28] Adam Amara and Sascha P. Quanz. PYNPOINT: An image processing package for finding exoplanets. *Monthly Notices of the Royal Astronomical Society*, 427(2):948–955, 12 2012.

[29] C. A. Gomez Gonzalez, O. Absil, P.-A. Absil, M. Van Droogenbroeck, D. Mawet, and J. Surdej. Low-rank plus sparse decomposition for exoplanet detection in direct-imaging adi sequences: The llsg algorithm. *Astronomy & amp; Astrophysics*, 589:A54, April 2016.

[30] Jared Tanner and Simon Vary. Compressed sensing of low-rank plus sparse matrices, 2022.

[31] Benoît Pairet, Faustine Cantalloube, Carlos A Gomez Gonzalez, Olivier Absil, and Laurent Jacques. Stim map: detection map for exoplanets imaging beyond asymptotic gaussian residual speckle noise. *Monthly Notices of the Royal Astronomical Society*, 487(2):2262–2277, May 2019.

[32] Carlos Alberto Gomez Gonzalez, Olivier Wertz, Olivier Absil, Valentin Christiaens, Denis Defrère, Dimitri Mawet, Julien Milli, Pierre-Antoine Absil, Marc Van Droogenbroeck, Faustine Cantalloube, Philip M. Hinz, Andrew J. Skemer, Mikael Karlsson, and Jean Surdej. Vip: Vortex image processing package for high-contrast direct imaging. *The Astronomical Journal*, 154(1):7, June 2017.

[33] B. N. Madhukar and R. Narendra. Lanczos resampling for the digital processing of remotely sensed images, 2013.

[34] Faustine Cantalloube, Valentin Christiaens, Carles Cantero, Evert Nasedkin, A. Cioppa, Olivier Absil, J. M. Bonse, Philippe Delorme, Carlos A. Gomez-Gonzalez, S. Juillard, Johan Mazoyer, M. Samland, J.-B. Ruffio, and M. Van Droogenbroeck. Exoplanet imaging

data challenge, phase ii: characterization of exoplanet signals in high-contract images, August 2022.

[35] Juan Terven, Diana M. Cordova-Esparza, Alfonso Ramirez-Pedraza, and Edgar A. Chavez-Urbiola. Loss functions and metrics in deep learning, 2023.

[36] K. I. M. McKinnon. Convergence of the nelder–mead simplex method to a nonstationary point. *SIAM Journal on Optimization*, 9(1):148–158, 1998.

[37] M. Powell. The bobyqa algorithm for bound constrained optimization without derivatives. *Technical Report, Department of Applied Mathematics and Theoretical Physics*, 01 2009.

[38] Coralia Cartis, Jan Fiala, Benjamin Marteau, and Lindon Roberts. Improving the flexibility and robustness of model-based derivative-free optimization solvers, 2018.

[39] J. L. Beuzit, A. Vigan, D. Mouillet, K. Dohlen, R. Gratton, A. Boccaletti, J. F. Sauvage, H. M. Schmid, M. Langlois, C. Petit, A. Baruffolo, M. Feldt, J. Milli, Z. Wahhaj, L. Abe, U. Anselmi, J. Antichi, R. Barette, J. Baudrand, P. Baudoz, A. Bazzon, P. Bernardi, P. Blanchard, R. Brast, P. Bruno, T. Buey, M. Carbillet, M. Carle, E. Cascone, F. Chapron, J. Charton, G. Chauvin, R. Claudi, A. Costille, V. De Caprio, J. de Boer, A. Delboulbé, S. Desidera, C. Dominik, M. Downing, O. Dupuis, C. Fabron, D. Fantinel, G. Farisato, P. Feautrier, E. Fedrigo, T. Fusco, P. Gigan, C. Ginski, J. Girard, E. Giro, D. Gisler, L. Gluck, C. Gry, T. Henning, N. Hubin, E. Hugot, S. Incorvaia, M. Jaquet, M. Kasper, E. Lagadec, A. M. Lagrange, H. Le Coroller, D. Le Mignant, B. Le Ruyet, G. Lessio, J. L. Lizon, M. Llored, L. Lundin, F. Madec, Y. Magnard, M. Marteaud, P. Martinez, D. Maurel, F. Ménard, D. Mesa, O. Möller-Nilsson, T. Moulin, C. Moutou, A. Origné, J. Parisot, A. Pavlov, D. Perret, J. Pragt, P. Puget, P. Rabou, J. Ramos, J. M. Reess, F. Rigal, S. Rochat, R. Roelfsema, G. Rousset, A. Roux, M. Saisse, B. Salasnich, E. Santambrogio, S. Scuderi, D. Segransan, A. Sevin, R. Siebenmorgen, C. Soenke, E. Stadler, M. Suarez, D. Tiphène, M. Turatto, S. Udry, F. Vakili, L. B. F. M. Waters, L. Weber, F. Wildi, G. Zins, and A. Zurlo. SPHERE: the exoplanet imager for the Very Large Telescope. *Astronomy & amp; Astrophysics*, 631:A155, November 2019.

[40] R. U. Claudi, M. Turatto, R. G. Gratton, J. Antichi, M. Bonavita, P. Bruno, E. Cascone, V. De Caprio, S. Desidera, E. Giro, D. Mesa, S. Scuderi, K. Dohlen, J. L. Beuzit, and P. Puget. SPHERE IFS: the spectro differential imager of the VLT for exoplanets search, July 2008.

[41] M. Carbillet, P. Bendjoya, and L. et al. Abe. Apodized lyot coronagraph for sphere/vlt., 2011.

[42] F. Cantalloube, K. Dohlen, J. Milli, W. Brandner, and A. Vigan. Peering through SPHERE Images: A Glance at Contrast Limitations. *The Messenger*, 176:25–31, June 2019.

[43] Guillaume Bal. Introduction to inverse problems, lecture notes University of Chicago, 2019.

[44] Yury Korolev and Jonas Latz. Inverse problems, lecture notes Michaelmas term University of Cambridge, 2021.

[45] Neal Parikh and Stephen Boyd. Proximal algorithms. *Found. Trends Optim.*, 1(3):127–239, jan 2014.

[46] Michael Merz Mario V. Wüthrich. *Statistical Foundations of Actuarial Learning and its Applications*. Springer Nature Switzerland AG, Gewerbestrasse 11, 6330 Cham, Switzerland, 2022 edition.

[47] Jean-Pierre Corriou. *Models and Methods for Parametric Identification*, pages 387–419. Springer London, London, 2004.

[48] V. Isakov. *Inverse Problems for Partial Differential Equations.* Springer Nature, Gewerbestrasse 11, 6330 Cham, Switzerland, 2017 edition.

[49] Timothy C. Y. Chan, Rafid Mahmood, and Ian Yihang Zhu. Inverse optimization: Theory and applications, 2022.

[50] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[51] Yuri Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140:125–161, 2013.

[52] Alexandre d'Aspremont, Damien Scieur, and Adrien Taylor. Acceleration methods. *Foundations and Trends® in Optimization*, 5(1–2):1–245, 2021.

[53] G. Nunes Grapiglia. Optimization : Nonlinear programming, lecture 7, 2023.

[54] Lloyd N. Trefethen and David Bau, III. *Numerical Linear Algebra.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.

[55] D. Mawet, J. Milli, Z. Wahhaj, D. Pelat, O. Absil, C. Delacroix, A. Boccaletti, M. Kasper, M. Kenworthy, C. Marois, B. Mennesson, and L. Pueyo. Fundamental limitations of high contrast imaging set by small sample statistics. *The Astrophysical Journal*, 792(2):97, September 2014.

[56] Hazan Daglayan, Simon Vary, Valentin Leplat, Nicolas Gillis, and P. A. Absil. Direct exoplanet detection using l1 norm low-rank approximation, 2023.

[57] Hazan Daglayan, Simon Vary, Faustine Cantalloube, P. A. Absil, and Olivier Absil. Likelihood ratio map for direct exoplanet detection, 2022.

[58] Wikipedia. Receiver operating characteristic — Wikipedia, the free encyclopedia, 2003. [Online; accessed 29-May-2024].

[59] C. A. Gomez Gonzalez, O. Absil, and M. Van Droogenbroeck. Supervised detection of exoplanets in high-contrast imaging sequences. *Astronomy & amp; Astrophysics*, 613:A71, May 2018.

[60] Dahlqvist, C.-H., Cantalloube, F., and Absil, O. Auto-rsm: An automated parameter-selection algorithm for the rsm map exoplanet detection algorithm. *A& A*, 656:A54, 2021.

[61] S. Juillard, V. Christiaens, and O. Absil. Inverse-problem versus principal component analysis methods for angular differential imaging of circumstellar disks: The mustard algorithm. *Astronomy & amp; Astrophysics*, 679:A52, November 2023.

[62] Cantalloube, F., Mouillet, D., Mugnier, L. M., Milli, J., Absil, O., Gomez Gonzalez, C. A., Chauvin, G., Beuzit, J.-L., and Cornia, A. Direct exoplanet detection and characterization using the andromeda method: Performance on vlt/naco data⋆. *A&A*, 582:A89, 2015.

[63] Cornia, A., Mugnier, L. M., Carbillet, M., Sauvage, J.-F., Boccaletti, A., Vedrenne, N., Mouillet, D., Rousset, G., and Fusco, T. Optimal method for exoplanet detection by spectral and angular differential imaging. page 09005, 2010.

[64] Flasseur, Olivier, Denis, Loïc, Thiébaut, Éric, and Langlois, Maud. Paco asdi: an algorithm for exoplanet detection and characterization in direct imaging with integral field spectrographs. *A&A*, 637:A9, 2020.

[65] Flasseur, Olivier, Thé, Samuel, Denis, Loïc, Thiébaut, Éric, and Langlois, Maud. Rexpaco: An algorithm for high contrast reconstruction of the circumstellar environment by angular differential imaging. *A&A*, 651:A62, 2021.

[66] William B. Sparks and Holland C. Ford. Imaging spectroscopy for extrasolar planet detection. *The Astrophysical Journal*, 578(1):543–564, October 2002.

[67] Tom M. Ragonneau and Zaikun Zhang. Pdfo: A cross-platform package for powell's derivative-free optimization solvers, 2023.

[68] Jeffrey Larson, Matt Menickelly, and Stefan M. Wild. Derivative-free optimization methods. *Acta Numerica*, 28:287–404, May 2019.

[69] P. Baudoz, A. Boccaletti, J. Baudrand, and D. Rouan. The self-coherent camera: a new tool for planet detection. *Proceedings of the International Astronomical Union*, 1(C200):553–558, 2005.

[70] Chen Xie, Elodie Choquet, Arthur Vigan, Faustine Cantalloube, Myriam Benisty, Anthony Boccaletti, Mickael Bonnefoy, Celia Desgrange, Antonio Garufi, Julien Girard, Janis Hagelberg, Markus Janson, Matthew Kenworthy, Anne-Marie Lagrange, Maud Langlois, François Menard, and Alice Zurlo. Reference-star differential imaging on sphere/irdis. *Astronomy & amp; Astrophysics*, 666:A32, October 2022.

[71] de Boer, J., Langlois, M., van Holstein, R. G., Girard, J. H., Mouillet, D., Vigan, A., Dohlen, K., Snik, F., Keller, C. U., Ginski, C., Stam, D. M., Milli, J., Wahhaj, Z., Kasper, M., Schmid, H. M., Rabou, P., Gluck, L., Hugot, E., Perret, D., Martinez, P., Weber, L., Pragt, J., Sauvage, J.-F., Boccaletti, A., Le Coroller, H., Dominik, C., Henning, T., Lagadec, E., Ménard, F., Turatto, M., Udry, S., Chauvin, G., Feldt, M., and Beuzit, J.-L. Polarimetric imaging mode of vlt/sphere/irdis - i. description, data reduction, and observing strategy. *A& A*, 633:A63, 2020.

[72] de Regt, S., Ginski, C., Kenworthy, M. A., Caceres, C., Garufi, A., Gledhill, T. M., Hales, A. S., Huelamo, N., Kóspál, Á., Millar-Blanchaer, M. A., Pérez, S., and Schreiber, M. R. Polarimetric differential imaging with vlt/naco - a comprehensive pdi pipeline for naco data (pippin). *A& A*, 684:A73, 2024.

[73] Tristan van Leeuwen and Christoph Brune. Welcome to inverse problems and imaging, 2023.

[74] Alexandre Gramfort, Matthieu Kowalski, and Matti Hämäläinen. Mixed-norm estimates for the m/eeg inverse problem using accelerated gradient methods. *Physics in Medicine & Biology*, 57(7):1937, mar 2012.

[75] Gomez Gonzalez et al. Exoplanet imaging data challenge.

[76] G. Cugno, P. Patapis, T. Stolker, S. P. Quanz, A. Boehle, H. J. Hoeijmakers, G.-D. Marleau, P. Mollière, E. Nasedkin, and I. A. G. Snellen. Molecular mapping of the pds70 system: No molecular absorption signatures from the forming planet pds70 b. *Astronomy & amp; Astrophysics*, 653:A12, August 2021.

[77] André Uschmajew and Bart Vandereycken. On critical points of quadratic low-rank matrix optimization problems. *IMA Journal of Numerical Analysis*, 2020.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**