Master Thesis

Towards More Accurate and Efficient Beamformed Radio Interferometry Imaging

Lausanne, Spring 2015

Author:
Matthieu SIMEONI
matthieu.simeoni@epfl.ch

Supervisors:
Dr. Paul Hurley, IBM Research, Zurich
Pr. Victor Panaretos, SMAT, EPFL, Lausanne
Pr. Martin Vetterli, LCAV, EPFL, Lausanne

Swiss Federal Institute of Technology
Department of Mathematics

ABSTRACT

The Square Kilometre Array (SKA) will form the largest radio telescope ever built, generating on the order of one terabyte of data per second. To reduce the data flow sent to the central processor, hierarchical designs have been proposed: the data is primarily collected in groups of antennas, and summed coherently by beamforming. Historically, Fourier analysis has played a prominent role in radio astronomy interferometry, legitimated by the celebrated van Cittert-Zernike theorem. We show that, in the case of modern hierarchical designs, beamformed data has a less intimate, and thus more complicated relationship to the Fourier domain. Unsatisfactory attempts have been proposed to compensate, which implicitly retain the Fourier framework, and are limited to directive beamforming.

We show that when stepping away from Fourier, we can embed the data in a more natural domain originating from the telescope configuration and the specific beamforming technique. This leads to a new, more accurate, imaging pipeline. Standard techniques such as w-projection, and gridding are no longer needed, as the reconstruction is performed on the celestial sphere.

The proposed imager operates in two steps. First, a preconditioning based on the Gram-Schmidt orthogonalization procedure is performed, in order to facilitate the computation of the pseudoinverse sky estimate. Then, from this, the LASSO estimate is approximated very efficiently. The quality of this approximation is shown to be linked directly to the effective support of the instrument point spread function. Due to the greater flexibility of this framework, information-maximising beamforming techniques such as randomised beamforming can be readily incorporated. Moreover, we use the Bonferroni method to construct global confidence intervals onto the Gram-Schmidt least squares estimate, and use them to test the statistical significance of each pixel.

The complexity of the proposed technique is assessed and compared to the the state-of-the-art combined CLEAN and A-projection algorithm. In the case of LOFAR, we show that our algorithm can be from 2 to 34 times faster. The accuracy and sensitivity of the new technique is also shown, for simulated data, to be superior.

Table of Contents

1

Introduction

Over the last century, humanity has witnessed both the apogee and the decline of gigantic single-dish antennas, such as the Arecibo telescope in Puerto Rico, and its impressive diameter of 305 meters. These disproportionate instruments belong now to the history of radio astronomy, as they were progressively replaced by a new generation of telescopes, with higher sensitivity and lower cost. This technological breakthrough was made possible by efficiently leveraging large numbers of small antennas. The signals received by the individual antennas were correlated so that the resulting **phased array** would form a gigantic interferometer.

To achieve a higher resolution and sensitivity, the number of elements has increased exponentially. While the **LOw Frequency ARray (LOFAR)** [5], composed of about 20,000 dipole antennas spread across Europe, was freshly inaugurated, its successor, the **Square Kilometer Array (SKA)** was already in planning. This extraordinary instrument should consist, upon completion of its second stage, of up to a *million* antennas [7]. Among the many challenges, the main one is certainly the unprecedented amount of data generated, on the order of exabytes per day [14].

To reduce the data flow sent to the central processor, the SKA will follow a **hierarchical design**, a strategy currently deployed in LOFAR. In this scenario, antennas geographically close to one another are grouped together in **stations**. The data from individual antennas is then summed at the station level, by **beamforming** antenna signals. The beamformed data is then sent to the central processor for further processing [24] (see fig. 1.1 page 8).

There is currently an implicit assumption of an *equivalent* telescope in the data model for these hierarchical interferometers. This assumption basically states that the beamformed output of a station can be seen as coming from a virtual antenna with a specific **beamshape** [19]. This point of view, appealing due to the hierarchical design, allowed astronomers to formulate the problem in the well-known Measurement Equation formalism, in turn inherited from the celebrated van Cittert-Zernike theorem [20]. In the classical case, without beamforming, this theorem nicely relates the measurements of an interferometer to some layout-dependent samples of the 2D Fourier transform of the underlying sky image. This embedding of the data in the Fourier domain not only nicely encapsulates optical telescopes and radio telescopes in a common framework, but also enables the seductive use of the **Fast Fourier Transform (FFT)** and its low complexity in the imaging procedure.

But once beamforming comes into the picture, the relationship with the Fourier domain becomes significantly more complicated, and retaining it requires a lot of non-intuitive algorithm tweaking. Indeed, beamforming induces different beamshapes

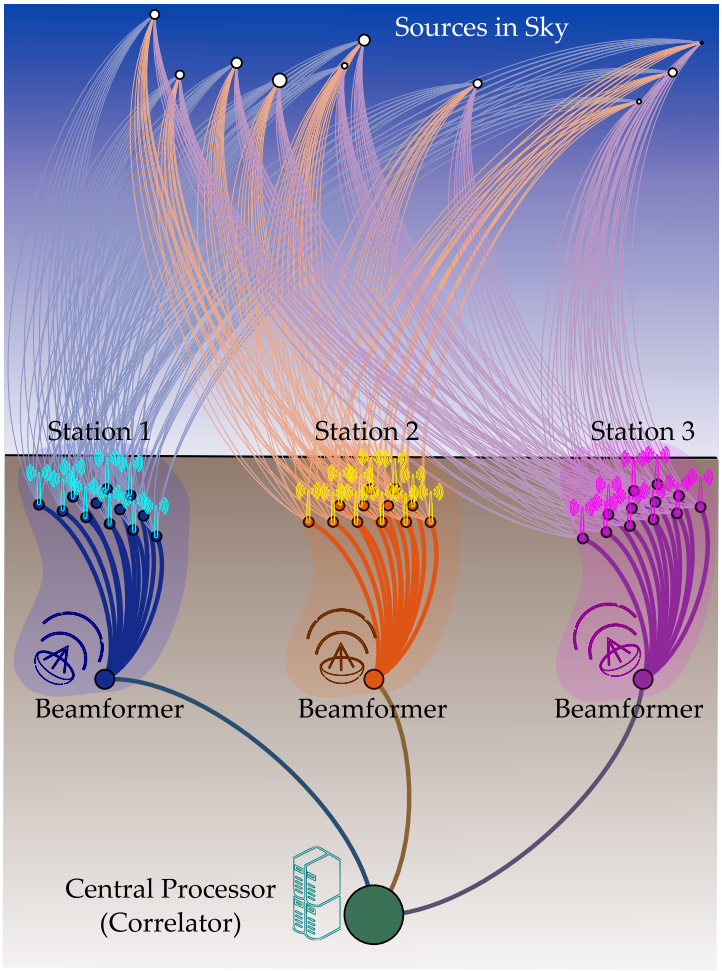


Figure 1.1: Schematic representation of a hierarchical design for modern radio telescopes. Signals received by individual antennas are beamformed at the station level, before being sent to the central processor for further processing and imaging.

at each stations, which therefore perceive the sky differently [25]. To account for these station-dependent beamshapes, the **A-projection algorithm** [3, 19] has been proposed. This algorithm essentially modifies the **gridding step**[1] by convolving the non-uniformly sampled Fourier domain with some beamshape-dependent kernels, different for each pair of station. In practice, this operation represents a serious bottleneck for the current imaging pipeline, as it is typically performed at each iteration of the **CLEAN algorithm** [10, 17], the most widely used imager in radio interferometry.

On top of this undesirable computational overhead, A-projection is not robust to variation in the beamforming strategy. Indeed, the A-projection algorithm is tailored to the use of **matched beamforming**, the de facto technique for modern radio interferometers such as LOFAR [5]. This beamformer can essentially be seen as a digital spatial filter mimicking the ability of dish antennas to focus at a certain location in the sky by aligning and summing coherently the signals received by each antenna from the point of focus [23]. But recent work [14, 15] has suggested the

[1] The interferometer samples non-uniformly the Fourier transform of the sky image. To enable the use of the FFT, these non-uniform samples need then to be placed onto a uniform grid, an operation usually referred to as **gridding**.

use of more versatile beamforming strategies, such as **randomized beamforming**, shown to maximize, in certain cases, the information transferred from station to correlator. Despite their advantages over matched beamforming, these beamforming techniques require alterations to the current imaging pipeline. Indeed, the lack of clear focus point in the beamshapes resulting from the use of randomized beamforming makes the positioning of the measurements in the Fourier plane ill-defined, and hence forbids the use of A-projection and other legacy imagers, which heavily rely on the Fourier domain.

In view of the above, this work strives to bring answers to the following questions: Is the failure of the classical imaging pipeline witnessed in the case of randomized beamforming symptomatic of a more fundamental flaw in the current problem formulation? Is the Fourier domain really the natural one for beamformed data? Could we propose a more general imaging pipeline readily usable with any beamforming strategy while remaining efficient enough for practical purposes?

We start, in the first two sections of chapter 2, by recalling the classical data model in its most general form and precisely state the assumptions upon which it is built. We then present how it is classically extended to the case of beamformed data by adopting a convenient (but, in our view, inappropriate) perspective on the data. In all that follows are the contributions of this thesis, summarized thus:

- **Chapter 2, section 3:** we reformulate the A-projection algorithm by a rigorous mathematical description, in terms more amenable to the mathematical and signal processing community.
- **Chapter 2, section 4:** we show that the CLEAN algorithm can be interpreted as an approximate gradient descent algorithm, which seeks to minimize the least squares problem by taking only canonical directions. We leverage this proximity in order to derive a loose lower bound on the number of iterations necessary for CLEAN to converge to a certain accuracy. This bound depends only on the data and the condition number of the problem.
- **Chapter 3, section 1:** we study the frequency response of the instrument to show that when beamforming is performed at the station level, the measurements should not be interpreted as samples of the 2D DFT of the underlying sky image. This result holds for *any* beamforming technique, and hence in particular for matched beamforming. Therefore, the inadequacy of the Fourier domain witnessed in the case of randomized beamforming was in reality not limited to this specific technique but symptomatic of a more fundamental flaw in the way beamformed data is currently processed.
- **Chapter 3, section 2:** we step away from the Fourier domain and propose a more natural measurement equation to relate beamformed data with the underlying sky image. This model depends only on the layout of the telescope, the characteristics of the antennas and the chosen beamforming technique. We compare this new model with the classical data model in terms of the accuracy of the reconstructed sky image. We perform this comparison for various imaging algorithms, and show the superiority of the new model.
- **Chapter 4, section 1:** we describe a direct solver for the least squares problem, based on the more general model derived in section 2 of chapter 3. The algorithm makes extensive use of the Gram-Schmidt orthogonalization procedure in order to precondition the problem and facilitate the reconstruction of the sky. In contrast to classical imagers, the reconstruction of the sky is made on the sphere, which is particularly interesting both for accuracy and efficiency. From the derived analytical description, we develop a discrete,

Figure 1.2: The various contributions of this thesis to the field of radio interferometry.

practical, version of the algorithm.

- **Chapter 4, section 1.3:** we derive meaningful statistics on the least squares estimate obtained with our new imager. In particular, we compute the covariance matrix of the sky estimate and leverage asymptotic properties of the Wishart distribution to construct global confidence intervals on the sky estimate using the Bonferroni method. These confidence intervals are used to perform a statistical test on the sky image, constructing what we call the significant image.
- **Chapter 4, section 2:** we present a very efficient scheme to approximate the LASSO estimate from the Gram-Schmidt least squares estimate, for the case where the number of pixels is greater than the number of measurements. We assess the quality of this approximation and show that it depends on the size of the effective support of the point spread function.
- **Chapter 4, section 3:** we discuss and demonstrate the robustness of our imaging pipeline to various experimental conditions, such as the noise level. Moreover, we show that, in contrast to classical imagers, the proposed methodology is readily applicable to any beamforming technique, such as randomized

beamforming.

- **Chapter 5, section 1:** we compute the complexity of both the proposed and the classical imaging pipeline, and compare them in the context of LOFAR and the SKA. We show the clear superior efficiency of our technique in the case of LOFAR across the board, and for certain scenarios for the SKA. We then discuss potential improvements that should make it also faster for all SKA use cases.
- **Chapter 5, section 2:** we assess the accuracy and sensitivity of both imaging pipelines on simulated data. The results of the experiments show the clear superiority of our algorithm with respect to the classical CLEAN + A-projection algorithm.

Finally, the thesis concludes and opens up potential avenues for future work.

2

The Current Imaging Pipeline

1 Basic Data Model

A radio interferometer is composed of antennas that measures a common incoming electromagnetic field at different locations on the ground [23]. The set of cross-correlations between the time series recorded by the different antennas provides samples of the **spatial coherency** of this electromagnetic field. These cross-correlations are usually referred to as **visibility measurements**. Provided certain assumptions on the sky signal, one can build a data model, linking these visibilities with the desired brightness distribution I, also referred to as **sky image**. Throughout the thesis, we will assume that the following holds:

Modeling Assumptions — ***Sky Signal.***

- **S1** *Celestial sources are in the far field, and lie on an hypothetical* ***celestial sphere*** $\mathbb{S}^2$*. Hence, the signals reaching the antennas are parallel [14].*
- **S2** *Signals emitted by the sources are narrow band, zero mean, circularly symmetric complex Gaussian processes. This yields the following* ***baseband representation*** *[23]:*

$$s(t, \boldsymbol{r}) = \hat{s}(t, \boldsymbol{r}) e^{j2\pi f_0 t}, \quad \hat{s}(t, \boldsymbol{r}) \sim \mathbb{C}\mathcal{N}(0, I(\boldsymbol{r})),$$

  where $s : \mathbb{R} \times \mathbb{S}^2 \to \mathbb{C}$ is a narrow-band signal coming from a direction $\boldsymbol{r} \in \mathbb{S}^2$ and with center frequency $f_0 \in \mathbb{R}$.
- **S3** *Signal coming from different directions in the sky are uncorrelated,*

$$\mathbb{E}[s(t, \boldsymbol{r}_1) s(t, \boldsymbol{r}_2)^*] = 0, \quad \forall \boldsymbol{r}_1, \boldsymbol{r}_2 \in \mathbb{S}^2.$$

From assumptions **S1** and **S2**, we can write [14] the total signal received by the ith antenna as

$$x_i(t) = \gamma_i \iint_{\mathbb{S}^2} \alpha(\boldsymbol{r}) \hat{s}(t, \boldsymbol{r}) e^{j2\pi f_0 \left(t - \langle \boldsymbol{r}, \frac{\boldsymbol{p}_i}{c} \rangle\right)} d\boldsymbol{r} \quad + \quad n_i(t), \tag{2.1}$$

with $\boldsymbol{p}_i \in \mathbb{R}^3$ the antenna's position on the ground, c the velocity of light, $\gamma_i \in \mathbb{C}$ a **complex gain**, $\alpha : \mathbb{S}^2 \to \mathbb{C}$ the **primary beamshape** of the antenna and $n_i(t) \sim \mathcal{N}(0, \sigma_n^2)$ additive **thermal noise**. Aside from the terms γ_i, $\alpha(\boldsymbol{r})$ and $n_i(t)$, this equation basically states that the total signal received from the sky by the ith antenna is the summation over the celestial sphere of the signals $s(t, \boldsymbol{r})$, with a time delay (which from assumption **S2** is a geometric delay [14]) accounting for the time

necessary for the signal to reach the antenna. The quantities γ_i, $\alpha(\boldsymbol{r})$ and $n_i(t)$ are introduced to model the imperfections of the instruments:

- The **complex gains** $\gamma_i \in \mathbb{C}$ account for the systematic errors introduced by the antennas. These gains usually vary slowly with time, and can therefore be assumed constant over the acquisition time. In practice, a **calibration** is performed, in order to estimate and (partially) correct for these gains.
- The **primary beamshape** $\alpha : \mathbb{S}^2 \to \mathbb{C}$ is a known feature of the instruments, usually uniform across the antennas [25]. It describes the sensitivity of the instrument towards the different directions on the celestial sphere (see fig. 2.1). When $\alpha(\boldsymbol{r}) = 1$, $\forall \boldsymbol{r} \in \mathbb{S}^2$, we say that the antennas have an **omnidirectional** field of view.
- The **thermal noise** $n_i(t) \sim \mathcal{N}(0, \sigma_n^2)$, is a stochastic corruption due to the electronics of the instruments [23]. The corruptions occurring at two different stations are assumed independent from one another. Finally, these corruptions are assumed uncorrelated with the signals coming from the sources.

Remark 1.1 *As described in [23], it is possible to add further refinements to 2.1 to account for ionospheric effects and other types of direction dependent effects. However, for the sake of simplicity, we voluntary exclude these effects from consideration. Finally, we neglect the polarization of the signal.*

From eq. (2.1) and assumption **S3**, we can obtain the relation between the visibility measurements $\{\mathbb{E}[x_i(t)x_k(t)^*]\}_{i,k}$ and the desired brightness distribution $I(\boldsymbol{r}) := Var(\hat{s}(t, \boldsymbol{r}))$. This relation is known as the **measurement equation**:

Theorem 2.1 — ***The Measurement Equation without Beamforming.*** Let $x_i(t) \in \mathbb{C}^{\mathbb{R}_+}$ and $x_k(t) \in \mathbb{C}^{\mathbb{R}_+}$ be the signals recorded by two antennas with respective positions $\boldsymbol{p}_i, \boldsymbol{p}_k \in \mathbb{R}^3$, gains $\gamma_i, \gamma_k \in \mathbb{C}$ and primary beamshapes $\alpha(\boldsymbol{r}) \in \mathbb{C}^{\mathbb{S}^2}$. Then, under assumptions **S1**, **S2** and **S3**, the cross-correlation $V_{i,k} \in \mathbb{C}$ between $x_i(t)$ and $x_k(t)$ is given by

$$\begin{aligned} V_{i,k} &:= \mathbb{E}[x_i(t)x_k(t)^*], \\ &= \gamma_i \gamma_k^* \iint_{\mathbb{S}^2} I(\boldsymbol{r}) |\alpha(\boldsymbol{r})|^2 e^{-j2\pi \langle \boldsymbol{r}, \frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0} \rangle} d\boldsymbol{r} + \sigma_n^2 \delta_{ik}, \end{aligned} \tag{2.2}$$

where $I(\boldsymbol{r})$ is the **sky image**, $\lambda_0 = f_0/c$ the wavelength of observation, σ_n^2 the variance of the thermal noise at the antennas and δ_{ik} the Kronecker delta function.

Vocabulary 1.1 — ***Measurement Equation.*** *In radio astronomy, a* ***measurement equation*** *is a model that describes the assumed relationship between the measurements of the telescope, and the underlying sky image. In this thesis, we will present many variations of measurements equations, with different assumptions used to derive them (some being more general than others). As always, the choice of working with one measurement equation rather than the other accounts to making a trade-off between preciseness of the model and computational convenience of its use. We will see in section 3 that some approximations made for efficiency reasons can have significant consequences on the resulting sky estimate.*

Vocabulary 1.2 — ***Baseline.*** *The normalized relative difference* $\frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0}$ *is commonly called a* ***baseline*** *between two antennas with respective positions* $\boldsymbol{p}_i \in \mathbb{R}^3$ *and* $\boldsymbol{p}_k \in \mathbb{R}^3$.

Remark 1.2 — ***Noise and Autocorrelations.*** *We observe in eq. (2.2) that the autocorrelations* $V_{i,i} = \mathbb{E}[x_i(t)x_i(t)^*]$ *are corrupted by additive noise while the cross-correlations are*

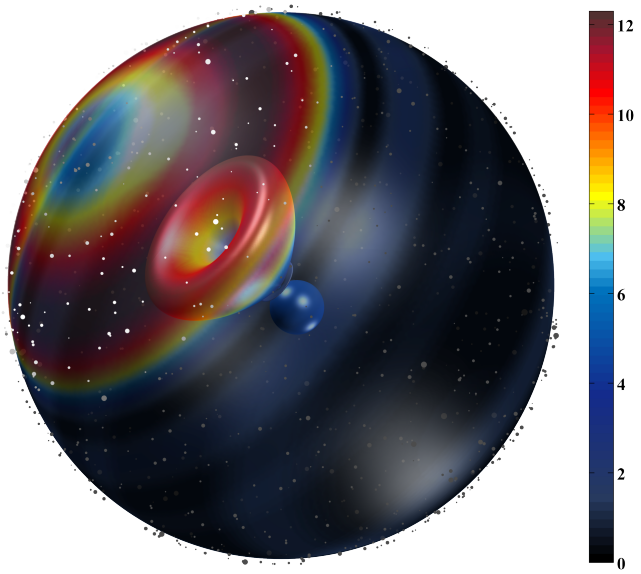


Figure 2.1: Primary beamshape of an HBA element of a LOFAR core station, for $f_0 = 175$ MHz. An HBA antenna is called a **tile** and is composed of 16 dually polarized dipoles, which are beamformed together towards a certain direction of the sky (here the zenith of LOFAR). The primary beamshape can be seen as the sensitivity of the antenna towards regions of the sky. The antenna perceives a modified version of the sky, with sources' intensities magnified according to the beamshape.

not[1]*. The autocorrelations are thus not used in the imaging process, and we will exclude them in all that follows.*

Assuming a proper calibration and an omnidirectional field of view for the antennas, eq. (2.2) simplifies to

$$V_{i,k} = \iint_{\mathbb{S}^2} I(\boldsymbol{r}) e^{-j2\pi\langle \boldsymbol{r}, \frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0}\rangle} d\boldsymbol{r}, \quad \forall i \neq k.$$

The resemblance between the above expression and the 3D Fourier transform is striking, with integration over the unit sphere instead of $\mathbb{R}^3$. When, as it is often the case, only a portion of the celestial sphere is of interest, one can project eq. (2.2) onto the plane tangent to the sphere at the center of the region of interest. This yields the **tangent plane measurement equation** without beamforming

$$V_{i,k} = \gamma_i \gamma_k^* \iint_{K \subset \mathbb{R}^2} \frac{I(l,m)}{\sqrt{1-l^2-m^2}} |\alpha(l,m)|^2 \mathcal{W}(l,m;w_{i,k}) e^{-j2\pi(u_{i,k}l + v_{i,k}m)} dl dm, \quad (2.3)$$

with $K \subset \mathbb{R}^2$ the compact support of I in $\mathbb{R}^2$,

$$\mathcal{W}(l,m;w_{i,k}) := e^{-j2\pi w_{i,k}(\sqrt{1-l^2-m^2}-1)}, \quad \forall l,m \in \mathbb{R}, \quad (2.4)$$

and

$$\boldsymbol{r} = l\boldsymbol{e}_1 + m\boldsymbol{e}_2 + (\sqrt{1-l^2-m^2} - 1)\boldsymbol{e}_3,$$

[1]This is because the noise corruptions at two different antennas are supposed independent from one another. In practice though, the estimated cross-correlations will also be affected by the noise as we can show that the variance of the estimates is increased in the presence of noise.

$$\frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0} = u_{i,k}\boldsymbol{e}_1 + v_{i,k}\boldsymbol{e}_2 + w_{i,k}\boldsymbol{e}_3.$$

The vectors $\boldsymbol{e}_1, \boldsymbol{e}_2 \in \mathbb{R}^3$ span the tangent plane and $\boldsymbol{e}_3 \in \mathbb{R}^3$ is orthogonal to this tangent plane, oriented towards the outside of the celestial sphere. When the field of view is small enough then we have $\mathcal{W}(l, m; w_{i,k}) \simeq 1$ and $\sqrt{1 - l^2 - m^2}$ is almost constant [20]. Therefore, assuming unit gains and an array of antennas with an omnidirectional field of view, we can re-write eq. (2.3) as

$$V_{i,k} \simeq \mathcal{V}(u_{i,k}, v_{i,k}) = \iint\limits_{K \subset \mathbb{R}^2} I(l, m) e^{-j2\pi(u_{i,k}l + v_{i,k}m)} dl dm, \tag{2.5}$$

with $\mathcal{V} : \mathbb{R}^2 \to \mathbb{C},\ (u, v) \mapsto \mathcal{F}\{I\}(u, v)$, the **visibility function**, Fourier transform of the sky image I. This special form of eq. (2.3) is known as the **van Cittert-Zernike theorem** [20] (or van Cittert-Zernike measurement equation).

In this specific case, we can interpret the visibilities $V_{i,k} \in \mathbb{C}$ as samples of the Fourier transform of $I(l, m)$. Then, recovering the sky image can be done by taking the inverse Fourier transform of the non-uniformly sampled visibility function:

$$I_D(l, m) = \mathcal{F}^{-1}\left\{ \sum_{i,k} \mathcal{V}(u, v)\delta(u - u_{i,k}, v - v_{i,k}) \right\}. \tag{2.6}$$

The image obtained this way is called the **dirty image**. It is usually polluted by artifacts (called **sidelobes**) due to our very crude knowledge of the visibility function. It is then necessary to apply a deconvolution algorithm to the dirty image, to correct for these artifacts. For point sources, the **CLEAN algorithm** [10, 17] is one of the most widely used.

For large fields of view, the approximation in eq. (2.5) breaks down, and a correction for the $\mathcal{W}$ term in eq. (2.3) needs to be performed. The **W-projection algorithm** [4] is commonly used for this purpose.

2 Beamforming at Stations

For modern radio telescopes such as LOFAR (or in the future the SKA), computing the correlations between each elements of the telescope is computationally prohibitive. To reduce the amount of data sent to the central correlator, one strategy is to group the various antennas in stations and beamform the signals from individual antennas together at the station level [24]. Then, the measurement equation previously derived needs updating to take into account the effect of beamforming on the data.

Let M be the number of stations, each composed of L antennas, with respective positions on the ground $\boldsymbol{p}_l^{(i)} \in \mathbb{R}^3$, $l = 1, \ldots, L$. We can concatenate the L signals collected by the antennas within station $i \leq M$ in a vector time series

$$\boldsymbol{x}_i : \begin{cases} \mathbb{R} \to \mathbb{C}^L, \\ t \mapsto \boldsymbol{x}_i(t) := \left(x_1^{(i)}(t), \ldots, x_L^{(i)}(t)\right)^T. \end{cases}$$

This yields, $\forall i \leq M$,

$$\boldsymbol{x}_i(t) = \iint_{\mathbb{S}^2} \hat{s}(t,\boldsymbol{r}) \times \alpha(\boldsymbol{r}) \left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r})\right) e^{j2\pi f_0 t} d\boldsymbol{r} \quad + \quad \boldsymbol{n}_i(t),$$

with $\odot$ the element-wise product, $\boldsymbol{\Gamma}_i = \left(\gamma_1^{(i)}, \ldots, \gamma_L^{(i)}\right) \in \mathbb{C}^L$ the concatenated elements' gains, $\boldsymbol{n}_i(t) = \left(n_1^{(i)}(t), \ldots, n_L^{(i)}(t)\right) \sim \mathbb{C}\mathcal{N}_L\left(\boldsymbol{0}, \sigma_n^2 I_L\right)$ some additive noise and $\boldsymbol{a}_i : \mathbb{S}^2 \to \mathbb{C}^L$ the **antenna steering vector** defined by

$$\boldsymbol{a}_i(\boldsymbol{r}) := \begin{pmatrix} e^{-j2\pi\langle \boldsymbol{r}, \frac{\boldsymbol{p}_1^{(i)}}{\lambda_0}\rangle} \\ \vdots \\ e^{-j2\pi\langle \boldsymbol{r}, \frac{\boldsymbol{p}_L^{(i)}}{\lambda_0}\rangle} \end{pmatrix}, \quad \forall \boldsymbol{r} \in \mathbb{S}^2. \tag{2.7}$$

The beamformed output $y_i : \mathbb{R} \to \mathbb{C}$ at each station is then given by

$$y_i(t) = \boldsymbol{w}_i^H \boldsymbol{x}_i(t), \quad i = 1, \ldots, M,$$

with $\boldsymbol{w}_i \in \mathbb{C}^L$ the **beamforming vector** used at station i. We finally have the following expression for $y_i(t)$:

$$y_i(t) = \iint_{\mathbb{S}^2} \hat{s}(t,\boldsymbol{r}) \times \alpha(\boldsymbol{r}) \boldsymbol{w}_i^H \left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r})\right) e^{j2\pi f_0 t} d\boldsymbol{r} \; + \; \boldsymbol{w}_i^H \boldsymbol{n}_i(t). \tag{2.8}$$

The beamforming vector $\boldsymbol{w}_i$ in eq. (2.8) can be chosen in many ways [14, 15], depending on the desired properties of $y_i(t)$. The most popular strategies in practice are directive beamforming techniques, such as **matched beamforming** and the **Minimum Variance Directionless Response (MVDR) beamformer** [23]. Both techniques can be seen as digital spatial filters, that provide the ability to focus at specific locations on the celestial sphere, by exploiting the telescope layout to appropriately combine the signals from the different antennas so that to maximize the contribution of a potential source at the selected location (see fig. 2.2 page 18).

Rather than deriving a new measurement equation by computing explicitly the correlation existing between two beamformed outputs y_i and y_k, astronomers prefer to adopt a different perspective on the data, that permits to bring hierarchically designed interferometers into the classical setting. This viewpoint is based on the following assumption:

Modeling Assumptions — ***Equivalent Telescope Model.***

T1 *Consider a hierarchically designed interferometer, organized in M stations with centroids $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_M \in \mathbb{R}^3$. Then, the stations' beamformed outputs $y_i(t)$ can be seen as coming from an equivalent telescope, composed of M* virtual *antennas positioned at the centroids of the stations, respectively $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_M \in \mathbb{R}^3$. To account for the effect of beamforming, those virtual antennas are allowed to have each different beamshapes.*

The rationale behind this apparently ad hoc statement comes from the study of an ideal case, where the sky would only be composed of a single source, with intensity σ_q^2 and position $\boldsymbol{r}_q \in \mathbb{S}^2$. In this very simple scenario, the variance of a station's

output is given by

$$\begin{aligned}\mathbb{E}[y_i(t)y_i^*(t)] =& \boldsymbol{w}_i^H \mathbb{E}[\boldsymbol{x}_i(t)\boldsymbol{x}_i^*(t)]\boldsymbol{w}_i, \\ =& |\underbrace{\alpha(\boldsymbol{r}_q)\boldsymbol{w}_i^H\left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r}_q)\right)}_{:=b_i(\boldsymbol{r}_q)}|^2\sigma_q^2 + \|\boldsymbol{w}_i\|^2\sigma_n^2. \end{aligned} \tag{2.9}$$

We observe in eq. (2.9) that the source's intensity is rescaled by the quantity $|b_i(\boldsymbol{r}_q)|^2$, and the thermal noise variance by the quantity $\|\boldsymbol{w}_i\|^2$. In comparison, the variance of the signal recorded by a single station's element is given by

$$\mathbb{E}[x_k^{(i)}(t)x_k^{(i)*}(t)] = |\alpha(\boldsymbol{r}_q)|^2\sigma_q^2 + \sigma_n^2.$$

It is then very tempting (but wrong, as we show in section 1 of chapter 3) to call $|b_i(\boldsymbol{r}_q)|^2$ the **beamshape** of the ith station, so that we could interpret the station as a virtual antenna [19], with a virtual thermal noise of variance $\|\boldsymbol{w}_i\|^2\sigma_n^2$.

Definition 2.1 — *Station Beamshape.* *Consider a hierarchical interferometer with M stations and L antennas per station, with primary beamshape $\alpha : \mathbb{S}^2 \to \mathbb{C}$. Let $\boldsymbol{w}_i \in \mathbb{C}^L$, $i = 1, \dots, M$, be the beamforming vectors at each station. Then, the **beamshape** $b_i : \mathbb{S}^2 \to \mathbb{C}$ of station i is defined as*

$$b_i : \begin{cases} \mathbb{S}^2 \to \mathbb{C}, \\ \boldsymbol{r} \mapsto b_i(\boldsymbol{r}) = \alpha(\boldsymbol{r})\boldsymbol{w}_i^H\left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r})\right), \end{cases} \tag{2.10}$$

with $\boldsymbol{\Gamma}_i \in \mathbb{C}^L$ the gains of the antennas within station i and $\boldsymbol{a}_i : \mathbb{S}^2 \to \mathbb{C}$ the antenna steering vector for station i, as defined in eq. (2.7).

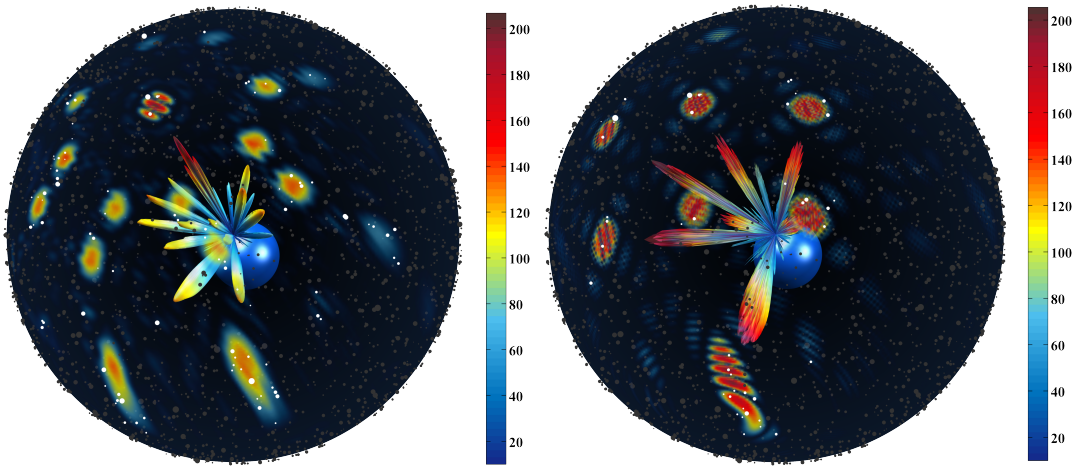
In the case of unitary gains and omnidirectional primary beamshape we get

$$\tilde{b}_i(\boldsymbol{r}) = \boldsymbol{w}_i^H\boldsymbol{a}_i(\boldsymbol{r}), \quad \forall \boldsymbol{r} \in \mathbb{S}^2,\ i = 1, \dots, M. \tag{2.11}$$

*The above quantity will be referred to as the **ideal beamshape** of station i in all that follows.*

Beamshapes from different stations of LOFAR are plotted on fig. 2.2. We observe that in contrast to the primary beamshape that was identical for every antenna, the station beamshape varies a lot from one station to the other. Roughly speaking, this means that each station has a different perception of the sky. We will see that accounting for this fact in the classical imaging pipeline significantly complicates sky reconstruction. At this point, we have reformulated the problem in terms of

an equivalent telescope composed of M virtual elements with beamshapes given in definition 2.1. This change of perspective nicely bring us into the range of application of the classical measurement equation, and we can straightforwardly apply theorem 2.1 to obtain



(a) Beamshape of the second core station of LOFAR, using matched beamforming towards the zenith of LOFAR, and a frequency of observation $f_0 = 120$ MHz.

(b) Beamshape of the eleventh core station of LOFAR, using matched beamforming towards the zenith of LOFAR, and a frequency of observation $f_0 = 120$ MHz.

Figure 2.2: Examples of station beamshapes for LOFAR. It appears than matched beamforming is effectively reducing the field of view of the station towards the region of interest (main lobe of the beamshape). However, some undesired portions of the sky also contribute to the measurements, because of the secondary sidelobes of the beamshape. These sidelobes can be attenuated by the use of the MVDR beamformer. Finally, we observe that conversely to the primary beamshape, the station beamshape varies a lot from one station to the other: each station is perceiving the sky differently.

Corollary 2.1 — *The Measurement Equation with Beamforming.* *Let $y_i(t) \in \mathbb{C}^{\mathbb{R}+}$ and $y_k(t) \in \mathbb{C}^{\mathbb{R}+}$ be the beamformed signals of two stations with centroids $\boldsymbol{p}_i, \boldsymbol{p}_k \in \mathbb{R}^3$, virtual complex gains $\gamma_i, \gamma_k \in \mathbb{C}$ and beamshapes $b_i(\boldsymbol{r})$, $b_k(\boldsymbol{r}) \in \mathbb{C}^{\mathbb{S}^2}$. Then, under assumptions **S1**, **S2**, **S3** and **T1**, the cross-correlation $V_{i,k} \in \mathbb{C}$ between $y_i(t)$ and $y_k(t)$ is given by*

$$V_{i,k} = \gamma_i \gamma_k^* \iint_{\mathbb{S}^2} I(\boldsymbol{r}) b_i(\boldsymbol{r}) b_k^*(\boldsymbol{r}) e^{-j2\pi \langle \boldsymbol{r}, \frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0} \rangle} d\boldsymbol{r} + \|\boldsymbol{w}_i\|^2 \sigma_n^2 \delta_{ik}, \tag{2.12}$$

*where $I(\boldsymbol{r})$ is the **sky image**, $\lambda_0 = f_0/c$ the wavelength of observation, σ_n^2 the variance of the thermal noise at the antennas, δ_{ik} the Kronecker delta function and $\boldsymbol{w}_i \in \mathbb{C}^L$ the beamforming vector used to beamform the signals at station i.*

Remark 2.1 — *Virtual Complex Gains.* *It can seem surprising to introduce virtual complex gains in eq. (2.12). Indeed, it seems reasonable to make the virtual elements inherit the imperfections of the actual antennas forming the station. But when looking back at the definition 2.1, we observe that the complex gains of the individual elements as well as the primary beamshape of those elements are already accounted for in the definition of the virtual elements beamshapes. So why would* virtual *elements introduce systematic errors in the measurements ? We can legitimate this by thinking at these complex gains as compensating for the approximation introduced by assumption **T1**, and the potential systematic mismatch that could exist between the true visibilities $V_{i,k} = \mathbb{E}[y_i(t) y_k^*(t)]$ and the visibilities that actual elements with the same specifications as the virtual elements would actually measure.*

Remark 2.2 — *Noise & Beamforming.* *We observe in eq. (2.12) that the thermal noise variance (that in theory corrupts only the auto-correlations but in practice corrupts also*

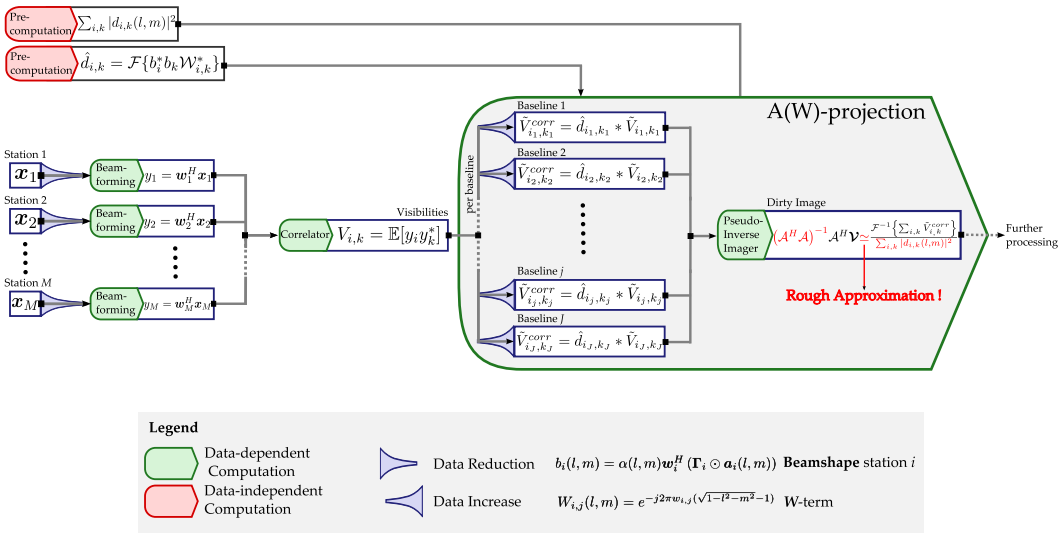


Figure 2.3: **Block diagram of the A(W)-projection algorithm.**

the cross-correlations estimates) is rescaled by the beamforming vector norm. Hence, beamforming can be leveraged to increase the signal to noise ratio, by for example magnifying the sources in the sky while maintaining the noise level unchanged[2]*. However it also means that if not chosen wisely, the beamforming vector can actually magnify the noise level, resulting in a more difficult sky reconstruction afterwards.*

Often, for convenience, eq. (2.12) is projected onto the tangent plane

$$V_{i,k} = \gamma_i \gamma_k^* \iint\limits_{K \subset \mathbb{R}^2} \frac{I(l,m)}{\sqrt{1-l^2-m^2}} b_i(l,m) b_k^*(l,m) \mathcal{W}(l,m;w_{i,k}) e^{-j2\pi(u_{i,k}l + v_{i,k}m)} dl dm, \tag{2.13}$$

with $K \subset \mathbb{R}^2$ the compact support of I, $\mathcal{W}(l,m;w_{i,k})$ and (l,m) as before, and (u_{ik}, v_{ik}, w_{ik}) being the uvw-coordinates of the baseline $(\boldsymbol{p}_i - \boldsymbol{p}_k)/\lambda_0$ between station i and k. Equation (2.13) will be referred to as the **tangent plane measurement equation** with beamforming.

3 Imaging with the A(W)-projection Algorithm

A naive Fourier inversion of the visibility function is no longer satisfactory in the case of beamformed data. Indeed, we have seen in the previous section that each station perceives the sky differently. To make sense of these measurements relatively to one another in the classical formulation, one needs first to apply a correction to the visibilities. The **A(W)-projection algorithm** [3, 19] is currently the most widely used technique for this purpose. It attempts to correct the visibilities for both the station-dependent beamshapes and the $\mathcal{W}$-term in eq. (2.13), by convolving each visibility with a compactly supported kernel, specific to the associated baseline (see fig. 2.3 for an overview of the algorithm). This operation is of course computationally expensive. Finally, we will see that the accuracy achievable is fundamentally limited, as the algorithm only partially corrects the visibilities.

[2] This is in effect what matched beamforming is doing.

3.1 Reshaping the Measurement Equation

The A-projection algorithm works with a $N \times N$ discrete 2D uniform grid of the sky portion of interest. Mathematically speaking, this is equivalent as assuming the following sky model:

$$I(l,m) = \sum_{i=1}^{N}\sum_{j=1}^{N} \sigma_{ij}^2 \delta(l-l_i, m-m_j), \quad \forall (l,m) \in K \subset \mathbb{R}^2,$$

with $\{(l_i, m_j)\}_{ij} \subset \mathbb{Z}^2$ forming a $N \times N$ uniform grid on the field of view and $\sigma_{ij}^2 \geq 0$ being the intensity (potentially null) of the pixel (i,j). Then, the sky can be equivalently represented as a matrix $\tilde{I} \in \mathbb{R}^{N\times N}$, defined as

$$\left(\tilde{I}\right)_{i,j} := I(l_j, m_i), \quad i,j = 1,\dots,N.$$

As it is often convenient to think of the sky image as a vector rather than a matrix, we need the vec$(\cdot)$ operator to $\tilde{I}$.

Definition 3.1 — *The vec$(\cdot)$ operator.* *Let $A \in \mathbb{K}^{N\times M}$ be a matrix, with $\mathbb{K}$ a field. Then, we define the vec$(\cdot)$ operator as*

$$vec : \mathbb{K}^{N\times M} \to \mathbb{K}^{NM}, \quad A \mapsto \left(A_{:,1}^T, \dots, A_{:,M}^T\right)^T,$$

with $A_{:,m} \in \mathbb{K}^N$ denoting the mth column of the matrix A.

This yields $\boldsymbol{I} := \text{vec}(\tilde{I}) \in \mathbb{R}^{N^2}$, where the columns of $\tilde{I}$ are stacked vertically. Then, we can re-write [19] the data model from eq. (2.13) as a product of linear operators successively acting on the sky image $\boldsymbol{I}$:

$$\boldsymbol{V}_{i,k} = G_{i,k} S_{i,k} F D_{i,k} \boldsymbol{I}, \quad i,k = 1,\dots,M,\ i \neq k, \tag{2.14}$$

where $\boldsymbol{V}_{i,k} \in \mathbb{C}^{N^2}$ and

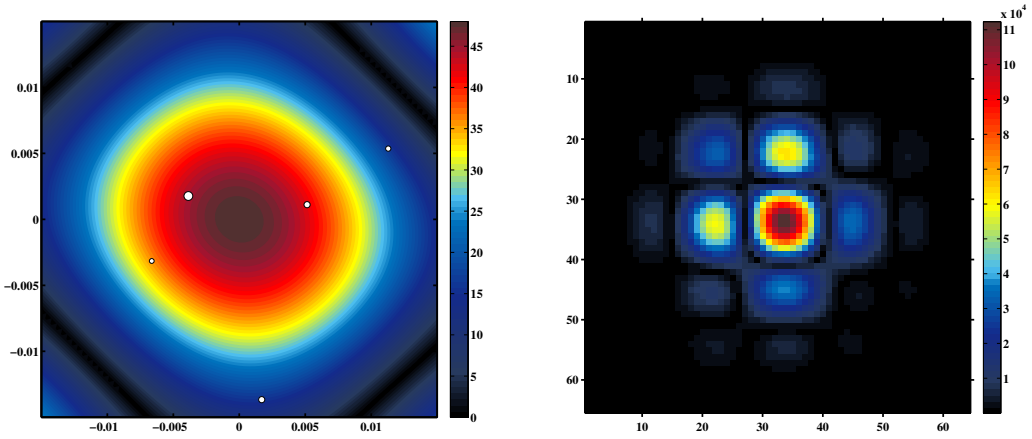
- $D_{i,k} := \text{diag}\left(\text{vec}\left(\tilde{D}_{i,k}\right)\right) \in \mathbb{C}^{N^2\times N^2}$ is a diagonal matrix. This term effectively rescales the brightness distribution $\boldsymbol{I}$ according to the beamshapes of the two stations, as well as the $\mathcal{W}$-term (see eq. (2.4)). The matrix $\tilde{D}_{i,k} \in \mathbb{C}^{N\times N}$ contains the rescaling factor for each pixel in the sky

  $$\left(\tilde{D}_{i,k}\right)_{c,r} := d_{ik}(l_r, m_c), \quad c,r = 1,\dots,N,$$

  with $d_{ik}(l,m) := b_i(l,m) \times b_k^*(l,m) \times \mathcal{W}(l,m;w_{i,k})$ (see fig. 2.4).
- $F \in \mathbb{C}^{N^2\times N^2}$ is the **2D DFT matrix**, reshaped by the vec$(\cdot)$ operator. Each column of F is given by

  $$F_{:,c} = \text{vec}(\text{DFT}_2\{E_{i_c,j_c}\}) \in \mathbb{C}^{N^2}, \quad c = 1,\dots,N^2,$$

  with DFT_2 the classical 2D discrete Fourier transform. The matrices $E_{i_c,j_c} \in \mathbb{R}^{N\times N}$ are given by $E_{i_c,j_c} = e_{i_c} e_{j_c}^T$, with $\{e_i\}_i \subset \mathbb{R}^N$ the canonical basis for $\mathbb{R}^N$. Finally, the bijection $(i_c, j_c) \leftrightarrow c$ which associates a pixel of the 2D image to its corresponding position in the vectorized image is described by the deterministic reordering performed by the vec$(\cdot)$ operator.



(a) Modulus of $d_{9,7}(l,m)$ for LOFAR, for a frequency of observation 75 MHz and a field of view of $\sim 2^\circ$. Stations are steered towards the zenith of LOFAR using matched beamforming.

(b) Zoom on the small support of the Fourier transform of $d_{9,7}(l,m)$.

Figure 2.4: Example of a baseline dependent rescaling function $d_{i,k}(l,m)$ for LOFAR. We observe that this function is rather smooth across the field of view, resulting in a small support in the Fourier domain.

- $S_{i,k} \in \{0,1\}^{N^2 \times N^2}$ is the **sampling matrix**. It samples the visibility function at the (gridded) frequencies $(u^{\Delta}_{i,k}, v^{\Delta}_{i,k}) \in \Delta$, nearest neighbors of the actual frequencies $(u_{i,k}, v_{i,k}) \in \mathbb{R}^2$ on the grid $\Delta \subset \mathbb{R}^2$ of the uv-plane. If $e_{i,k} \in \{1, \dots, N^2\}$ corresponds to the position of the sampled uv-point in the vectorize version of the visibility function $\boldsymbol{V}_{i,k} = FD_{i,k}\boldsymbol{I} \in \mathbb{C}^{N^2}$, then the terms of the matrix $S_{i,k}$ are given by

$$(S_{i,k})_{r,c} := \begin{cases} 1 & \text{if } r = c = e_{i,k}, \\ 0 & \text{otherwise.} \end{cases}$$

- $G_{i,k} := \gamma_i \gamma_k^* \in \mathbb{C}$ models the systematic and direction independent effects for the given baseline. For simplicity, we will assume that these gains have been estimated and the visibilities accordingly corrected, so that we have $G_{i,k} \simeq 1$.

Assuming an arbitrary numbering of the baselines, we can then stack eq. (2.14) vertically, which yields

$$\begin{aligned} \boldsymbol{\mathcal{V}} &= \underbrace{\begin{pmatrix} S_{i_1,k_1} & & 0 \\ & \ddots & \\ 0 & & S_{i_J,k_J} \end{pmatrix}}_{:=\mathcal{S} \in \{0,1\}^{JN^2 \times JN^2}} \underbrace{\begin{pmatrix} F & & 0 \\ & \ddots & \\ 0 & & F \end{pmatrix}}_{:=\mathcal{F} \in \mathbb{C}^{JN^2 \times JN^2}} \underbrace{\begin{pmatrix} D_{i_1,k_1} \\ \vdots \\ D_{i_J,k_J} \end{pmatrix}}_{:=\mathcal{D} \in \mathbb{C}^{JN^2 \times N^2}} \boldsymbol{I} \\ &= \mathcal{S}\mathcal{F}\mathcal{D}\boldsymbol{I} \\ &= \mathcal{A}\boldsymbol{I}, \end{aligned} \tag{2.15}$$

where J is the total number of baselines, $\boldsymbol{\mathcal{V}} := \left(\boldsymbol{V}^T_{i_1,k_1}, \dots, \boldsymbol{V}^T_{i_J,k_J}\right)^T \in \mathbb{C}^{JN^2}$ and $\mathcal{A} \in \mathbb{C}^{JN^2 \times N^2}$.

Equation (2.15) conveniently formulates our data model as a linear system linking

the visibilities to the sky image. As such, it is yet another measurement equation (it is actually the discrete analog of eq. (2.13)). The matrix $\mathcal{A}$ is not only rectangular but also generally *ill-conditioned*, which translates in a huge sensitivity of the solutions to potential data corruptions. Various strategies are then available to solve this linear system, each yielding potentially different estimates of the sky image with specific properties depending on the chosen algorithm. One intuitive solution to this linear system least squares, provided by

$$\boldsymbol{I}_D = \text{argmin}_{\boldsymbol{I}\in\mathbb{R}^{N^2}} \|\boldsymbol{\mathcal{V}} - \mathcal{A}\boldsymbol{I}\|_2^2.$$

In the classical case where no beamforming is performed, the **dirty image** proposed in eq. (2.6) can be shown to be the least square solution of the problem[3]. As already mentioned, this solution is generally corrupted by artifacts, which can mislead us to wrongly identify a structure within the sky estimate as a source. To minimize these corruptions, one can regularize the problem by biasing (usually with a L^1 penalization) the solution towards the set of sparse sky images[4]. Compressed sensing methods can then be used to recover the sky image. But such methods are usually very computationally demanding, so that astronomers prefer to work with the dirtier least square solution, cleaned by a deconvolution algorithm such as the CLEAN algorithm. The least square solution presents indeed a major advantage over competing solutions as it can be written in closed form using the **pseudoinverse** of $\mathcal{A}$,

$$\boldsymbol{I}_D = \left(\mathcal{A}^H\mathcal{A}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}}, \tag{2.16}$$

in the specific case where $\mathcal{A}$ is full column rank. In practice, because of the size of $\mathcal{A}$, it is prohibitive [19] to compute exactly the term $\left(\mathcal{A}^H\mathcal{A}\right)^{-1}$, that corrects for the geometric aberrations in $\mathcal{A}^H\boldsymbol{\mathcal{V}}$ (due to the columns of $\mathcal{A}$ not being orthogonal). A-projection considers then a very rough approximation of eq. (2.16)

$$\left(\mathcal{A}^H\mathcal{A}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}} \simeq \left(\mathcal{D}^H\mathcal{D}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}}, \tag{2.17}$$

where $\mathcal{A}^H\mathcal{A}$ is approximated by $\mathcal{D}^H\mathcal{D} = \sum_{j=1}^{J} D_{i_j,k_j}^H D_{i_j,k_j}$, average of the direction dependent effects across all baselines (see fig. 2.5 **(c)**). Each of the matrices D_{i_j,k_j} being by definition diagonal, the matrix $\mathcal{D}^H\mathcal{D}$ is itself diagonal, and therefore much easier to invert than the full matrix $\mathcal{A}^H\mathcal{A}$. The price to pay however for this computational convenience is high, as the resulting sky estimate can be severely affected by the remaining geometric artifacts, not fully corrected for (see fig. 2.5 **(e)** and **(f)**).

3.2 Algorithm Description

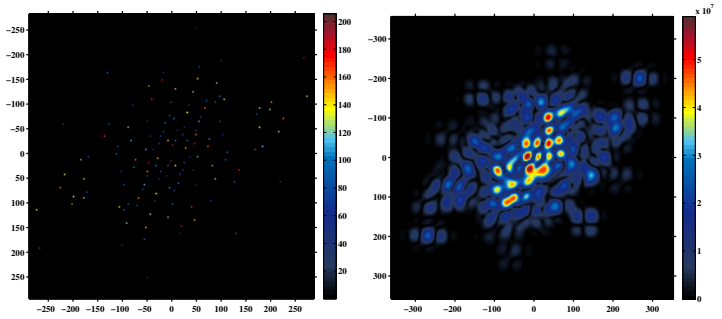
The **A-projection** algorithm is essentially an algorithm optimized for computing the approximate least square solution

$$\boldsymbol{I}_D = \left(\mathcal{D}^H\mathcal{D}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}},$$

focusing on the computation of the computationally most expensive term, $\mathcal{A}^H\boldsymbol{\mathcal{V}}$. Expanding this term we get

[3] Indeed, the dirty image in eq. (2.6) is by construction the closest approximation of the sky image within the subspace generated by the family of exponentials $\{e^{j2\pi(u_{i,k}l+v_{i,k}m)}\}_{i,k=1,\dots,J}$.

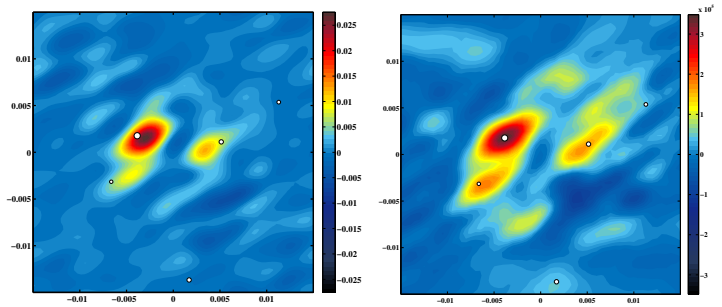
[4] With of course the a priori belief that the observed sky is mostly empty.



(a) Modulus of the visibility function without any correction for the station dependent beamshapes and the $\mathcal{W}$-term.

(b) Modulus of the visibility function obtained after A-projection (sum across all baselines of independently corrected components).

(c) First order correction term: average across all baselines of the functions $d_{i,k}(l,m)$.



(d) Dirty image without any correction for the the station dependent beamshapes and the $\mathcal{W}$-term (inverse Fourier transform of **(a)**).

(e) Dirty image after A-projection but without first order correction (inverse Fourier transform of **(b)**).

(f) Dirty image after A-projection and divided by **(c)** (first order approximation for the term $(\mathcal{A}^H\mathcal{A})^{-1}$)).

Figure 2.5: Some relevant quantities involved in the A-projection algorithm. We observe that accounting for the station dependent beamshapes and the $\mathcal{W}$-term permits to identify new sources in the dirty image, previously undetectable. However, the rough approximation of the pseudoinverse made in the A-projection algorithm for practical reasons leads to stronger artifacts in the dirty image. Superimposed white dots correspond to actual sources in the sky.

$$
\begin{aligned}
\mathcal{A}^H\boldsymbol{\mathcal{V}} &= \mathcal{D}^H\mathcal{F}^H\mathcal{S}^H\boldsymbol{\mathcal{V}} \\
&= \mathcal{D}^H\mathcal{F}^H\boldsymbol{\mathcal{V}} \\
&= \begin{pmatrix} D^H_{i_1,k_1} & \cdots & D^H_{i_J,k_J} \end{pmatrix} \begin{pmatrix} N^4F^{-1} & & 0 \\ & \ddots & \\ 0 & & N^4F^{-1} \end{pmatrix} \boldsymbol{\mathcal{V}} \\
&= N^4 \sum_{j=1}^{J} D^H_{i_j,k_j} F^{-1} \boldsymbol{V}_{i_j,k_j}.
\end{aligned}
$$

Therefore, the dirty image is given by:

$$
\boldsymbol{I}_D = N^4 \left(\mathcal{D}^H\mathcal{D}\right)^{-1} \sum_{j=1}^{J} D^H_{i_j,k_j} F^{-1} \boldsymbol{V}_{i_j,k_j}.
$$

To better understand the above equation, it is worth reinterpreting it as a continuous system. We get, for every $(l, m) \in \mathbb{R}^2$,

$$I_D(l,m) = \frac{\sum_{j=1}^{J} d^*_{i_j,k_j}(l,m) \times \mathcal{F}^{-1}\{\tilde{V}_{i_j,k_j}\}(l,m)}{\sum_{j=1}^{J} |d_{i_j,k_j}(l,m)|^2}, \tag{2.18}$$

with $\mathcal{F}$ the 2D Fourier transform, $\tilde{V}_{i_j,k_j}(u,v) = V_{i_j,k_j}\delta(u - u_{i_j,k_j}, v - v_{i_j,k_j})$ and $d_{i,k}(l,m) = b_i(l,m)b^*_k(l,m)\mathcal{W}(l,m;w_{i,k})$.

Equation (2.18) provides us with a recipe to approximately correct for the baseline dependent effects $d_{i,k} = b_i b^*_k \mathcal{W}(\cdot; w_{i,k})$.

1. For each baseline, consider the one-sample visibility function $\tilde{V}_{i_j,k_j}(u,v) = V_{i_j,k_j}\delta(u - u_{i_j,k_j}, v - v_{i_j,k_j})$,
   - Compute the inverse Fourier transform of this function.
   - Correct for the baseline dependent effects *in the image plane* by multiplying this function by $d^*_{i,k}$.
2. Sum these functions across all baselines,
3. Divide by $1/\sum_{j=1}^{J} |d_{i_j,k_j}(l,m)|^2$ as a first order approximation for $\left(\mathcal{A}^H\mathcal{A}\right)^{-1}$ in eq. (2.16).

This recipe is naive in terms of computational cost, as the correction for the beamshapes and the $\mathcal{W}$-term are performed in the image plane. Indeed, such a correction requires the multiplication of two $N \times N$ complex images per baseline, infeasible for next generation radio telescope, where the number of baselines will be extremely big.

One can significantly reduce the computational cost of this algorithm by leveraging the convolution theorem [19]. Indeed, the functions $d_{i,k}$ are usually rather smooth over the field of view [19] and their 2D Fourier transform have consequently small support in the uv-plane (see fig. 2.4 page 21). We can then compute equivalently I_D with the formula

$$I_D(l,m) = \frac{\sum_{j=1}^{J} \mathcal{F}^{-1}\{\mathcal{F}\{d^*_{i_j,k_j}\} * \tilde{V}_{i_j,k_j}\}(l,m)}{\sum_{j=1}^{J} |d_{i_j,k_j}(l,m)|^2}. \tag{2.19}$$

Because of the small support in the uv-plane of the functions $\mathcal{F}\{d^*_{i_j,k_j}\}$, the convolution $\mathcal{F}\{d^*_{i_j,k_j}\} * \tilde{V}_{i_j,k_j}$ is significantly cheaper to compute than the multiplication $d^*_{i_j,k_j}(l,m)\mathcal{F}^{-1}\{\tilde{V}_{i_j,k_j}\}(l,m)$ in eq. (2.18), performed in the image plane.

This finally yields the **A-projection** algorithm, given in algorithm 1.

Remark 3.1 — *w-stacking.* *Though significantly cheaper than the naive algorithm inspired from eq. (2.18), the implementation of A-projection presented in algorithm 1 can be further improved by factorising out the $\mathcal{W}$-term from the $d_{i,k}$ functions. Indeed this term is only dependent on $w_{i,k}$, the vertical coordinate of the considered baseline in the frame attached to the tangent plane. Because of potential telescope layout redundancy, some baselines can have identical w-coordinates (or at least close to one another). It can then be interesting to perform the $\mathcal{W}$ correction on groups of baselines rather than on each individual baselines. This technique is called **w-stacking** [19]. Depending on the layout of the telescope, this can significantly reduce the computational cost of the A-projection algorithm, as the $\mathcal{W}$-term is usually the most expensive to correct for (as it has a larger support in the uv-plane than the station beamshapes).*

Various quantities involved in the A-projection algorithm can be seen in fig. 2.5 page 23. Observe that accounting for the station dependent beamshapes and the $\mathcal{W}$-

Algorithm 1: The A-projection algorithm (Naive Implementation)

procedure A-PROJECTION ▷ INPUTS: visibilities, associated frequencies, station beamshapes and $\mathcal{W}$-term.

  $\mathcal{V}(u,v) \leftarrow 0$;

  for $j = 1 : J$ **do** ▷ *Perform correction for each baseline.*

    $\tilde{V}_{i_j,k_j}(u,v) \leftarrow V_{i_j,k_j}\delta(u - u_{i_j,k_j}, v - v_{i_j,k_j})$;

    $\tilde{V}^{corr}_{i_j,k_j}(u,v) \leftarrow \left(\mathcal{F}\{d^*_{i_j,k_j}\} * \tilde{V}_{i_j,k_j}\right)(u,v)$; ▷ *Cheaper in the uv-plane !*

    $\mathcal{V}(u,v) \leftarrow \mathcal{V}(u,v) + \tilde{V}^{corr}_{i_j,k_j}(u,v)$; ▷ *Add the corrected contributions from each baselines together.*

  end

  $I_D(l,m) \leftarrow \mathcal{F}^{-1}\{\mathcal{V}\}(l,m)$; ▷ *2D Fourier inversion.*

  $I_D(l,m) \leftarrow I_D(l,m) / \left(\sum_{j=1}^{J} |d_{i_j,k_j}(l,m)|^2\right)$; ▷ *First order correction for geometric artifacts.*

  return $I_D(l,m)$; ▷ OUTPUT: The (approximate) Dirty Image (L_2 solution)

term does improve the sky estimate. However, because of the approximation 2.17, the dirty image is polluted by artifacts, not fully corrected for. For this reason, the A-projection algorithm is often used in conjunction with the CLEAN algorithm [19]. The expensive correction needs then to be performed at each iteration of the CLEAN algorithm, making the imaging procedure even more costly.

4 Iterative approach: CLEAN as Gradient Descent

We have seen that dirty images obtained with algorithm 1 can be seriously affected by artifacts, resulting from the rough approximation of the deconvolution term $\left(\mathcal{A}^H\mathcal{A}\right)^{-1}$. To overcome this issue, a solution is to leverage the A-projection algorithm in order to design efficient iterative schemes aiming at obtaining an approximation to the least squares problem

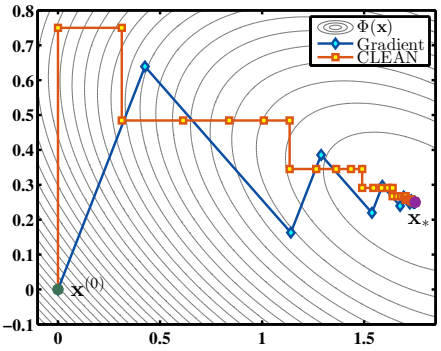
When the sky image under consideration is assumed sparse, it can be advantageous to penalize the above optimization problem, in order to guarantee a reasonable (depending on the strength of the penalty) sparsity of the solution

$$\boldsymbol{I}_D = \text{argmin}_{\boldsymbol{I}\in\mathbb{R}^{N^2}} \|\boldsymbol{\mathcal{V}} - \mathcal{A}\boldsymbol{I}\|_2^2 + \lambda\|\boldsymbol{I}\|_1,$$

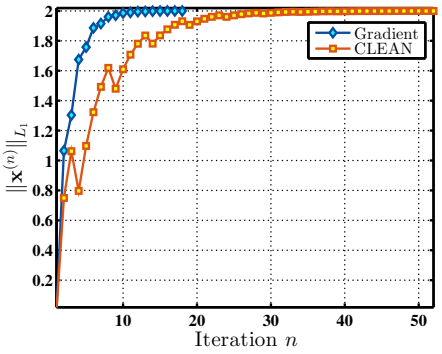
with $\lambda \in \mathbb{R}$ some penalty parameter, and $\|\cdot\|_1$ the L^1 norm.

The CLEAN algorithm is related to both optimization problems. It can be seen as an approximate gradient descent algorithm, following a constrained path going as close as possible from the gradient descent algorithm path. Starting from the null image $\hat{\boldsymbol{I}}^{(0)} = 0$, it iteratively updates the sky estimate according to the following equation

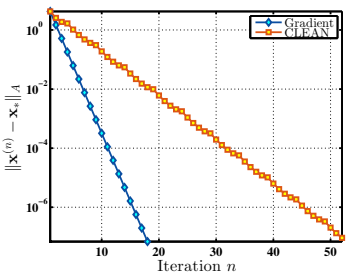
$$\hat{\boldsymbol{I}}^{(n+1)} := \hat{\boldsymbol{I}}^{(n)} + \tau\Psi\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)}), \quad n \geq 1, \tag{2.20}$$



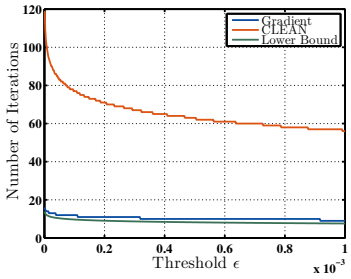
(a) Respective paths of the Gradient Descent algorithm and the CLEAN algorithm across iterations. The CLEAN path is constrained to take only canonical directions.



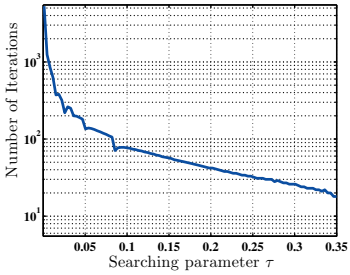
(b) L^1 norm of the estimates across iterations. The CLEAN estimate has a smaller L^1 norm (solution sparser) up to iteration 20.



(c) Distance (in terms of the A-norm) between the estimate $\boldsymbol{x}^{(n)}$ and the minimum $\boldsymbol{x}_*$, in logarithmic scale. The slopes of each line gives the convergence rate of the corresponding method.



(d) Number of iterations required to achieve an accuracy $\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\|_A < \epsilon$ for both algorithms. The green line gives the lower bound derived from theorem 2.2.



(e) Number of iterations before convergence for various searching parameters τ in the CLEAN algorithm (logarithmic scale). For $\tau > 0.35$ the algorithm does not converge anymore.

Figure 2.6: Comparison between the CLEAN algorithm and the gradient Descent Algorithm for a simple example. In this example, we apply the CLEAN and Gradient Descent algorithm to estimate the pseudoinverse solution to the linear system $A\boldsymbol{x} = \boldsymbol{b}$. A is full column-rank. If not specified differently, we take $\epsilon = 10^{-4}$ and $\tau = 0.1$.

with Ψ a nonlinear operator, that filters out every components from the residual image $\mathcal{A}^H(V - \mathcal{A}\hat{\boldsymbol{I}}^{(n)})$ except from the strongest, and $\tau \in \mathbb{R}$ the **searching parameter**, that determines how far we must travel along the direction $\Psi\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)})$. The multiplications by $\mathcal{A}$ and $\mathcal{A}^H$ in eq. (2.20) can be performed more efficiently using the A-projection algorithm. Keeping only the strongest component from the residual image guarantees that only one component of the sky estimate $\hat{\boldsymbol{I}}^{(n)}$ is updated at a time, with the hope that such an approach will help keeping the L^1 norm of the sky estimate low. The direction $\Psi\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)}) \in \mathbb{R}^{N^2}$ can also be seen as the closest approximation of the steepest descent direction in the canonical basis. Indeed, if we define

$$\Phi : \begin{cases} \mathbb{R}^{N^2} \to \mathbb{R}, \\ \boldsymbol{I} \mapsto \|\boldsymbol{\mathcal{V}} - \mathcal{A}\boldsymbol{I}\|_2^2, \end{cases}$$

then,

$$\begin{aligned}\nabla\Phi(\boldsymbol{I}) &= \frac{\partial}{\partial \boldsymbol{I}}\|\boldsymbol{\mathcal{V}} - \mathcal{A}\boldsymbol{I}\|_2^2 \\ &= \frac{\partial}{\partial \boldsymbol{I}}\left[\boldsymbol{I}^H\mathcal{A}^H\mathcal{A}\boldsymbol{I} - 2\boldsymbol{I}^H\mathcal{A}^H\boldsymbol{\mathcal{V}} + \boldsymbol{\mathcal{V}}^H\boldsymbol{\mathcal{V}}\right] \\ &= 2\mathcal{A}^H(\mathcal{A}\boldsymbol{I} - \boldsymbol{\mathcal{V}}).\end{aligned}$$

The steepest descent at $\hat{\boldsymbol{I}}^{(n)} \in \mathbb{R}^{N^2}$ is then given by

$$-\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}) = 2\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)}),$$

so that $\Psi\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)}) \propto \langle\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}), \boldsymbol{e}_{i_{max}}\rangle\boldsymbol{e}_{i_{max}}$, with

$$i_{max} = \text{argmax}\left\{\langle\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}), \boldsymbol{e}_i\rangle \;\middle|\; i = 1, \dots, N^2\right\},$$

and $\{\boldsymbol{e}_i,\, i = 1, \dots, N^2\} \subset \mathbb{R}^{N^2}$ the canonical basis. This confirms our previous claim: the direction followed by the CLEAN algorithm at each iteration is the steepest descent direction approximated by its main component (see fig. 2.6(a)). By taking only canonical directions, the CLEAN algorithm attempts to maintain a certain sparsity in the sky estimate, as only one component (i.e. pixel) is updated at a time (see fig. 2.6(b)).

Finally, the algorithm is stopped when the residual image goes below a certain threshold $\epsilon \in \mathbb{R}_+$, under which the sensitivity of the instrument is not good enough to discriminate potentially remaining sources from noise artifacts. That translates into the following stopping criterion:

$$\|\mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)})\|_\infty = \|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_\infty \leq \epsilon.$$

This is a sensible stopping criterion, as when $\nabla\Phi(\hat{\boldsymbol{I}}) = 0$ we have

$$\begin{aligned}\mathcal{A}^H\mathcal{A}\hat{\boldsymbol{I}} &= \mathcal{A}^H\boldsymbol{\mathcal{V}}, \\ \hat{\boldsymbol{I}} &= \left(\mathcal{A}^H\mathcal{A}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}},\end{aligned}$$

provided that $\mathcal{A}$ is full column rank. Hence, when $\epsilon = 0$, the CLEAN estimate and the pseudoinverse solution coincide. It also means that the relative sparsity of the CLEAN estimate can only be maintained for large enough ϵ, and will eventually breaks down as ϵ gets smaller (and the CLEAN estimate converges to the least squares solution). When $\mathcal{A}$ is not full-column rank, then the CLEAN estimate and the pseudoinverse solution are both a solution to the least squares problem, but do not coincide anymore (see fig. 2.7).

The CLEAN algorithm is summarized in algorithm 2. The lines 8 to 9 are usually performed when the CLEAN algorithm is stopped prematurely (large ϵ). Indeed, in such scenarios, the estimate $\hat{\boldsymbol{I}}^{(n)}$ consists of only a few components. To reduce this artificially high resolution, the image is convolved with a Gaussian kernel. The residuals are added to the image [23], as the astronomers might still be interested in the potential information left in it. The parameter τ in the update equation line 6 is chosen uniformly across all the iterations, which is suboptimal (in terms of number of operations needed to converge to the stopping criterion). The choice of this

Algorithm 2: The CLEAN Algorithm (helped by A-projection)

procedure CLEAN ▷ INPUTS: visibilities, $\epsilon > 0$, $\tau > 0$
  $\hat{\boldsymbol{I}}^{(0)} \leftarrow 0$;
  while $\|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_\infty > \epsilon$ **do** ▷ *Stopping criterion*
    $\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}) \leftarrow 2\mathcal{A}^H(\mathcal{A}\hat{\boldsymbol{I}}^{(n)} - \boldsymbol{\mathcal{V}})$; ▷ *Compute residuals using A-projection*
    $\hat{\boldsymbol{I}}^{(n)} \leftarrow \hat{\boldsymbol{I}}^{(n)} - \frac{\tau}{2}\Psi\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})$; ▷ *Update equation*
  end
  $\boldsymbol{I}_{res} \leftarrow \mathcal{A}^H(\boldsymbol{\mathcal{V}} - \mathcal{A}\hat{\boldsymbol{I}}^{(n)})$; ▷ *Residuals at the end*
  $\hat{\boldsymbol{I}}^{(n)} \leftarrow \hat{\boldsymbol{I}}^{(n)} \star \boldsymbol{\mathcal{G}}$; ▷ *Convolve with Gaussian kernel*
  $\hat{\boldsymbol{I}}^{(n)} \leftarrow \hat{\boldsymbol{I}}^{(n)} + \boldsymbol{I}_{res}$; ▷ *Add residuals*
  return $\hat{\boldsymbol{I}}^{(n)}$; ▷ OUTPUT: The CLEAN image.

parameter heavily influences the convergence of the CLEAN algorithm [10, 17, 23] (see fig. 2.6(e)). In comparison, the gradient descent algorithm can optimally select a searching parameter $\tau^{(n)}$ for each iteration. This is done by minimizing with respect to τ the cost function along the steepest descent direction

$$\tau^{(n)} = \text{argmin}_{\tau\in\mathbb{R}}\ \Phi\left(\hat{\boldsymbol{I}}^{(n)} - \frac{\tau}{2}\nabla\Phi\left(\hat{\boldsymbol{I}}^{(n)}\right)\right).$$

Differentiating w.r.t τ yields

$$\tau^{(n)} = \frac{\|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_2^2}{\|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_{\mathcal{A}}^2}, \tag{2.21}$$

with $\|\boldsymbol{x}\|_{\mathcal{A}} := \sqrt{\boldsymbol{x}^H\mathcal{A}^H\mathcal{A}\boldsymbol{x}}, \quad \forall\boldsymbol{x}\in\mathbb{R}^{N^2}$.

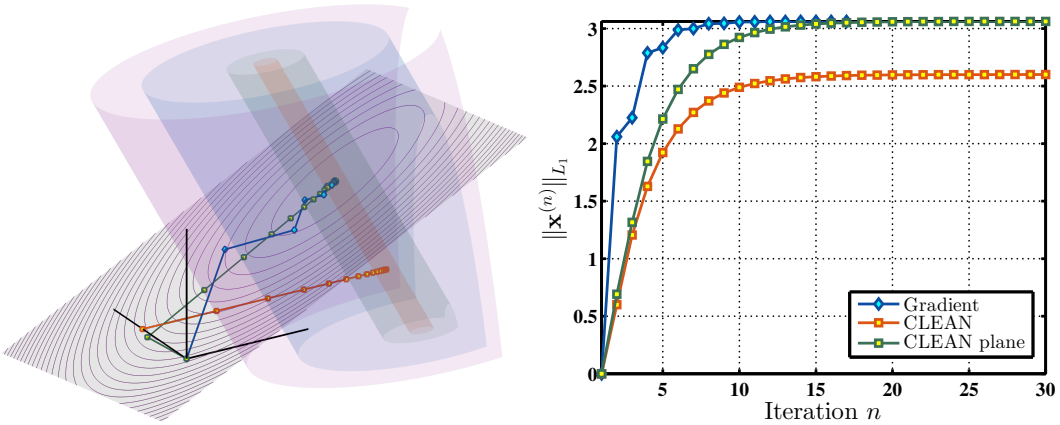
Hence, the CLEAN algorithm is suboptimal with respect to the gradient descent algorithm, both in terms of chosen directions and in terms of the choice of the searching parameter τ. Consequently, the number of iterations required to meet the stopping criterion $\|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_\infty > \epsilon$ will be necessarily larger for CLEAN algorithm than for gradient descent, which has the following convergence property [26]:

Theorem 2.2 — ***Convergence Rate of the Gradient Descent Algorithm.*** Let $f : E \to \mathbb{R}$ be a strictly convex quadratic form on the finite dimensional Euclidean space E. Let $H : E \to E$ be the Hessian of f, and $\boldsymbol{x}_* \in E$ the unique minimum of f. Then, if we use the gradient descent algorithm to minimize f starting from 0, we have, $\forall n \geq 1$

$$\|\boldsymbol{x}^{(n)} - \boldsymbol{x}_*\|_H \leq \left(\frac{\kappa_2(H)-1}{\kappa_2(H)+1}\right)^n \|\boldsymbol{x}_*\|_H,$$

with $\boldsymbol{x}^{(n)} \in E$ the estimate obtained with the gradient descent algorithm at iteration n, $\kappa_2(H)$ the **condition number** of H, and $\|\boldsymbol{x}\|_H := \sqrt{\langle H\boldsymbol{x},\boldsymbol{x}\rangle_E}, \quad \forall\boldsymbol{x}\in E$.

In our case, Φ is a strictly convex quadratic form, with Hessian given by $H := \mathcal{A}^H\mathcal{A}$. Thus, because CLEAN is suboptimal w.r.t the gradient descent algorithm, we can bound from below the number of iterations needed to achieve a certain



(a) Respective paths of the gradient descent (blue) and CLEAN (orange) across iterations. The CLEAN estimate is here different from the pseudoinverse solution. However, the projection of the CLEAN path onto the subspace generated by the rows of A converges to the pseudoinverse solution. The transparent cylinders correspond to isocontours of the residual sum of squares.

(b) L^1 norm of the estimates across iterations. The CLEAN estimate has always a smaller L^1 norm (solution sparser) then the gradient descent estimate.The projection of the CLEAN estimate onto the subspace generated by the rows of A has a smaller norm than the Gradient Descent estimate up to iteration 10.

Figure 2.7: Comparison between the CLEAN algorithm and the gradient Descent Algorithm for solving the linear system $A\boldsymbol{x} = \boldsymbol{b}$, with A is not full-column rank. When A is not full-column rank, then the CLEAN estimate and the pseudoinverse solution do not coincide anymore even after an infinite number of iterations. However, they both minimize the residuals, and hence the projection of the CLEAN estimate onto the subspace generated by the rows of A coincides with the pseudoinverse solution.

accuracy between the sky estimate $\boldsymbol{I}^{(n)}$ produced by the CLEAN algorithm and the pseudoinverse solution $\hat{\boldsymbol{I}} = \left(\mathcal{A}^H\mathcal{A}\right)^{-1}\mathcal{A}^H\boldsymbol{\mathcal{V}}$. For example, if we require

$$\|\boldsymbol{I}^{(n)} - \hat{\boldsymbol{I}}\|_H \leq \epsilon,$$

then we know than the number of iterations n_{CLEAN} required to achieve this accuracy with CLEAN will be bounded from below by the number of iterations required to achieve this accuracy with the gradient descent algorithm (see fig. 2.6(c) and (d)):

$$n_{CLEAN} > \frac{\log(\epsilon) - \log(\|\hat{\boldsymbol{I}}\|_H)}{\log\left(\frac{\kappa_2(H)-1}{\kappa_2(H)+1}\right)}, \tag{2.22}$$

provided that $\kappa_2(H) > 1$. We can see in (e) that (2.22) is not a very tight bound. We further observe that the convergence rate depends on the condition number $\kappa_2(H)$ of the matrix $\mathcal{A}^H\mathcal{A}$: the larger the condition number, the slower the convergence will be. In the special case where $\kappa_2(H) = 1$ then the gradient descent algorithm converges in a single iteration, and the above lower bound becomes less interesting, as it simply tells us than $n_{CLEAN} > 1$.

3

Understanding Visibilities

1 Visibilities and Fourier Samples

Interestingly enough, the A-projection algorithm described previously seems to disagree with the data model it is built upon. Indeed, the algorithm is based on the *equivalent* virtual telescope model **T1**, that permits (rather artificially) bringing beamformed astronomical data into the scope of the classical measurement equation formalism. Essentially, this model translates in mathematical terms the belief that, provided that an adequate correction for the station dependent beamshapes and the $\mathcal{W}$-term is performed, we can still see the visibilities as samples of the visibility function at locations given by the stations' uv-coordinates. But this viewpoint, that conveniently keeps alive the intimate link that classically binds radio astronomical data with the Fourier domain, is undermined by the nature of the correction to be applied to the visibility function. Looking back at figures 2.5 (a) and 2.5 (b), we can indeed observe that the correction, that consists in convolving individual components of the visibility function with baseline-dependent kernels, effectively *increases* the visibility function's support. But with this increased support, the number of visibilities and the cardinality of the sky image spectrum do not match anymore! This is rather problematic, as it seems to forbid any identification between visibilities and samples of the visibility function, a fact that directly contradicts our initial belief.

To try and provide an answer to this observation, we can analyze the frequency answer of the interferometer. For the sake of simplicity, assume in this section that the field of view is small enough to neglect the effect of the $\mathcal{W}$-term and assume an omnidirectionnal field of view for the antennas. We further assume that the complex gains have been already estimated and accounted for.

For insight, let's first look at the classical case where no beamforming is performed. In this scenario, and with the assumptions **S1**–**S3**, the van Cittert-Zernike theorem holds (see eq. (2.5) page 15)

$$V_{i,k} = \iint_{\mathbb{R}^2} I(l,m) e^{-j2\pi(u_{i,k}l + v_{i,k}m)} dl dm, \quad i \neq k.$$

The simplicity of the above equation helps in attaching a geometrical interpretation to a visibility. Indeed, the space $\mathbb{C}^{\mathbb{R}^2}$ of functions from $\mathbb{R}^2$ into $\mathbb{C}$ is a Hilbert space, that can be equipped with the standard inner product

$$\langle f, g \rangle := \iint_{\mathbb{R}^2} f(l,m) g^*(l,m) dl dm, \quad \forall f, g \in \mathbb{C}^{\mathbb{R}^2}.$$

With this definition, visibilities can thus be seen as the inner product between the sky image and elements from the Fourier basis:

$$V_{i,k} = \langle I, e^{j2\pi(u_{i,k}l + v_{i,k}m)} \rangle, \quad i \neq k.$$

This essentially means that the visibilities contain information about the (linear) resemblance between the sky image and elements of the Fourier basis, with frequencies . Hence, the set of visibilities $\{V_{i,k}\}_{i,k} \subset \mathbb{C}$, provides us with a complete description of the signal within the finite linear subspace spanned by the orthogonal family $\{e^{j2\pi(u_{i,k}l + v_{i,k}m)}\}_{i,k} \subset \mathbb{C}^{\mathbb{R}^2}$. The L^2 sky estimate, or dirty image I_D, is then simply the projection of I onto this subset, given by

$$\begin{aligned} I_D &= \sum_{i,k} \langle I, e^{j2\pi(u_{i,k}l + v_{i,k}m)} \rangle e^{j2\pi(u_{i,k}l + v_{i,k}m)}, \\ &= \sum_{i,k} V_{i,k} e^{j2\pi(u_{i,k}l + v_{i,k}m)}, \quad \forall (l,m) \in \mathbb{R}^2. \end{aligned} \tag{3.1}$$

We observe in eq. (3.1) that the dirty image is a linear combination of sinusoids, with frequencies determined by the telescope layout. Consequently, the support $\mathcal{S}_f \subset \mathbb{R}^2$ of the instrument's frequency response is given by

$$\begin{aligned} \mathcal{S}_f &= \text{supp} \left\{ \mathcal{F} \left(\sum_{i,k} e^{j2\pi(u_{i,k}l + v_{i,k}m)} \right) \right\}, \\ &= \text{supp} \left\{ \sum_{i,k} \delta(u - u_{i,k}, v - v_{i,k}) \right\}, \\ &= \bigcup_{i,k} \{(u_{i,k}, v_{i,k})\}. \end{aligned} \tag{3.2}$$

From eq. (3.2), we observe that $|\mathcal{S}_f| = |\{V_{i,k} | i \neq k\}| = L(L-1)$, with L the total number of antennas[1]. This equality in cardinality between the set of evidence and the set of Fourier elements makes an identification between visibilities and uv-samples possible.

Analogously, we can compute the support of the frequency response for a hierarchically designed radio telescope. If we do not use the equivalent telescope formulation **T1** but rather directly compute from eq. (2.8) the correlation between two beamformed outputs $y_i(t)$ and $y_k(t)$ we obtain an alternative data model

$$\mathbb{E}[y_i(t) y_k^*(t)] = \iint_{\mathbb{R}^2} I(l,m) \tilde{b}_i(l,m) \tilde{b}_k^*(l,m), \forall i \neq k, \tag{3.3}$$

with $\tilde{b}_i(l,m)$ the **ideal beamshape** of station i (defined in eq. (2.11) page 17). Once again, eq. (3.3) can be written as an inner product between the sky image and some

[1] Provided that we exclude the autocorrelations from the reconstruction.

periodic function:

$$V_{i,k} = \langle I, \tilde{b}_i^* \tilde{b}_k \rangle = \langle I, \tilde{\beta}_{i,k} \rangle,$$

with $\tilde{\beta}_{i,k} := \tilde{b}_i^* \tilde{b}_k \in \mathbb{C}^{\mathbb{R}^2}$.

To compute the frequency response of the instrument, we need to further develop the computation of these periodic functions. The ideal beamshape for station i is given by:

$$\begin{aligned}\tilde{b}_i(l,m) &= \boldsymbol{w}_i^H \boldsymbol{a}_i(l,m), \\ &= \sum_{h=1}^{L} \omega_h^{(i)*} e^{-j2\pi\left(u_h^{(i)} l + v_h^{(i)} m\right)}, \quad \forall (l,m) \in \mathbb{R}^2,\end{aligned}$$

with L the total number of antennas within station i, $(u_h^{(i)}, v_h^{(i)}) \in \mathbb{R}^2$ the uv-coordinates of antenna h within station i, and $\boldsymbol{w}_i = \left(\omega_1^{(i)}, \ldots, \omega_L^{(i)}\right)^T \in \mathbb{C}^L$ the beamforming vector used at station i. Then, the periodic function $\tilde{\beta}_{i,k}$ are given by

$$\begin{aligned}\tilde{\beta}_{i,k}(l,m) &= \sum_{h=1}^{L}\sum_{g=1}^{L} \omega_h^{(i)} \omega_g^{(k)*} e^{j2\pi\left[\left(u_h^{(i)} - u_g^{(k)}\right) l + \left(v_h^{(i)} - v_g^{(k)}\right) m\right]}, \\ &= \sum_{h=1}^{L}\sum_{g=1}^{L} \omega_h^{(i)} \omega_g^{(k)*} e^{j2\pi\left[u_{h,g}^{(i,k)} l + v_{h,g}^{(i,k)} m\right]}, \quad \forall (l,m) \in \mathbb{R}^2,\end{aligned}$$

with $(u_{h,g}^{(i,k)}, v_{h,g}^{(i,k)}) \in \mathbb{R}^2$ are the uv-coordinates of the baseline from antenna h in station i to antenna g in station k.

Again, the only information we have access to concerns the (linear) resemblance between the sky signal and the family of periodic functions $\{\tilde{\beta}_{i,k} | i \neq k\} \subset \mathbb{C}^{\mathbb{R}^2}$, so that the dirty image will necessarily be a linear combination of these functions. We can then compute the support of the frequency response as

$$\begin{aligned}\mathcal{S}_f &= \operatorname{supp}\left\{\mathcal{F}\left(\sum_{i,k}\sum_{h=1}^{L}\sum_{g=1}^{L} \omega_h^{(i)} \omega_g^{(k)*} e^{j2\pi\left[u_{h,g}^{(i,k)} l + v_{h,g}^{(i,k)} m\right]}\right)\right\}, \\ &= \operatorname{supp}\left\{\sum_{i,k}\sum_{h=1}^{L}\sum_{g=1}^{L} \omega_h^{(i)} \omega_g^{(k)*} \delta\left(u - u_{h,g}^{(i,k)}, v - v_{h,g}^{(i,k)}\right)\right\}, \\ &= \bigcup_{i,k}\bigcup_{h,g=1}^{L} \left\{\left(u_{h,g}^{(i,k)}, v_{h,g}^{(i,k)}\right)\right\}.\end{aligned}$$

This time, if M is the total number of stations, we observe that

$$|\mathcal{S}_f| = M(M-1)L^2 > M(M-1) = |\{V_{i,k} | i \neq k\}|.$$

In conclusion, when beamforming is performed at the station level, the support of the instrument frequency response has a greater cardinality than the set of visibilities. This evidently forbids any identification between the two sets, and as a

result undermines the legitimacy of the equivalent telescope formulation.

2 A New Measurement Equation

The classical measurement equation with beamforming (see eq. (2.12) page 18) was built on assumption **T1**, as an attempt to interpret beamformed data into the classical data model. To compensate for the inadequacy of this model, computationally expensive corrections need to be performed on the data during the imaging step. We have seen that performing those corrections in the Fourier domain with A-projection could, to some extent, reduce this computational overhead.

But maybe we could gain even more if we accepted to step away from the Fourier domain, and stopped performing expensive corrections on the data to make it fit in an obsolete framework. Indeed, the analysis carried out in the previous section, shows (in a very simple scenario) that the link between beamformed data and the Fourier domain is not as clear and intimate as it used to be. Therefore, rather than trying to optimize the current imaging pipeline, maybe we should reconsider the data model on which the imaging algorithms are built and see if we can propose a more natural framework to work with beamformed data.

In section 2 page 15, we have established a generic expression for the beamformed signal $y_i : \mathbb{R} \to \mathbb{C}$ of station i:

$$\begin{aligned} y_i(t) &= \iint_{\mathbb{S}^2} \hat{s}(t,\boldsymbol{r}) \times \alpha(\boldsymbol{r}) \boldsymbol{w}_i^H \left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r})\right) e^{j2\pi f_0 t} d\boldsymbol{r} \;+\; \boldsymbol{w}_i^H \boldsymbol{n}_i(t), \\ &= \iint_{\mathbb{S}^2} \hat{s}(t,\boldsymbol{r}) \times b_i(\boldsymbol{r}) e^{j2\pi f_0 t} d\boldsymbol{r} \;+\; \boldsymbol{w}_i^H \boldsymbol{n}_i(t), \end{aligned} \tag{3.4}$$

with the same notations as in section 2. At this stage, rather than wrongfully changing our viewpoint on the data to make it fall into the scope of the classical measurement equation, we can decide to let ourself be guided by the equations and establish our data model from the direct computation of the correlation exisitng between two beamformed signals $y_i(t)$ and $y_k(t)$. Using eq. (3.4) and the usual assumptions **S1** to **S3** yields

Theorem 3.1 — *The Natural Measurement Equation.* Let $y_i(t) \in \mathbb{C}^{\mathbb{R}+}$ and $y_k(t) \in \mathbb{C}^{\mathbb{R}+}$ be the beamformed signals of two stations with beamshapes $b_i(\boldsymbol{r})$, $b_k(\boldsymbol{r}) \in \mathbb{C}^{\mathbb{S}^2}$. Then, under assumptions **S1**, **S2** and **S3**, the cross-correlation $V_{i,k} \in \mathbb{C}$ between $y_i(t)$ and $y_k(t)$ is given by

$$V_{i,k} = \iint_{\mathbb{S}^2} I(\boldsymbol{r}) b_i(\boldsymbol{r}) b_k^*(\boldsymbol{r}) \, d\boldsymbol{r} \quad + \quad \|\boldsymbol{w}_i\|^2 \sigma_n^2 \delta_{ik}, \tag{3.5}$$

where $I(\boldsymbol{r})$ is the **sky image**, δ_{ik} the Kronecker delta function and $\boldsymbol{w}_i \in \mathbb{C}^L$ the beamforming vector used to beamform the antennas' signals from station i.

Remark 2.1 — *Comparison with the Classical Measurement Equation.* *Observe that, in contrast to the classical measurement equation eq. (2.2) derived under assumption **T1**, neither the virtual gains, nor the Fourier kernels $e^{-j2\pi\langle \boldsymbol{r}, \frac{\boldsymbol{p}_i - \boldsymbol{p}_k}{\lambda_0} \rangle}$ appear in eq. (3.5).*

Remark 2.2 — *Geometric Interpretation.* *Again, it is possible to interpret eq. (3.5) geometrically. Indeed, if we provide the space $\mathbb{C}^{\mathbb{S}^2}$ with the appropriate inner product,*

$$\langle f, g\rangle = \iint_{\mathbb{S}^2} f(\boldsymbol{r}) g^*(\boldsymbol{r}) d\boldsymbol{r}, \quad \forall f, g \in \mathbb{C}^{\mathbb{S}^2},$$

then we can see a visibility $V_{i,k}$ as the inner product between the sky image $I \in \mathbb{C}^{\mathbb{S}^2}$ and some function $\beta_{i,k} \in \mathbb{C}^{\mathbb{S}^2}$

$$V_{i,k} = \langle I, \beta_{i,k}\rangle, \quad i \neq k,$$

*with $\beta_{i,k}(\boldsymbol{r}) := b_i^*(\boldsymbol{r}) b_k(\boldsymbol{r})$ the **cross-station beamshapes**. Hence, after beamforming, the interferometer the sky image is no longer sampled in the Fourier basis, but in a family of functions $\{\beta_{i,k}\}_{i,k} \subset \mathbb{C}^{\mathbb{S}^2}$ (which might not necessarily form a basis).*

Theorem 3.1 can be more compactly written by regrouping the equations (3.5) for each visibility in a single equation, by stacking vertically the signals coming from each station in a vector time series $\boldsymbol{y}(t) : \mathbb{R} \to \mathbb{C}^M$,

$$\begin{aligned} \boldsymbol{y}(t) &= \iint_{\mathbb{S}^2} I(\boldsymbol{r}) \underbrace{\alpha(\boldsymbol{r}) W^H \left(\boldsymbol{\Gamma} \odot \boldsymbol{a}(\boldsymbol{r})\right)}_{:=\boldsymbol{b}(\boldsymbol{r})} d\boldsymbol{r} \; + \; W^H \boldsymbol{n}(t), \\ &= \iint_{\mathbb{S}^2} I(\boldsymbol{r}) \boldsymbol{b}(\boldsymbol{r}) d\boldsymbol{r} \; + \; W^H \boldsymbol{n}(t), \end{aligned} \tag{3.6}$$

with $\boldsymbol{b} : \mathbb{S}^2 \to \mathbb{C}^M$ containing the stations beamshapes stacked vertically, $\alpha : \mathbb{S}^2 \to \mathbb{C}$ the primary beamshape of the antennas, $\boldsymbol{a}(\boldsymbol{r}) : \mathbb{S}^2 \to \mathbb{C}^{ML}$, $\boldsymbol{\Gamma} \in \mathbb{C}^{ML}$ and $\boldsymbol{n}(t) : \mathbb{R} \to \mathbb{C}^{ML}$ respectively the antenna steering vector, the elements complex gains and the noise corruptions for the group of antennas formed by the entire telescope. More precisely, we have

$$\boldsymbol{a}(\boldsymbol{r}) = \left(\boldsymbol{a}_1(\boldsymbol{r}), \ldots, \boldsymbol{a}_M(\boldsymbol{r})\right)^T, \quad \forall \boldsymbol{r} \in \mathbb{S}^2,$$

$$\boldsymbol{\Gamma} = \left(\boldsymbol{\Gamma}_1, \ldots, \boldsymbol{\Gamma}_M\right)^T,$$

and

$$\boldsymbol{n}(t) = \left(\boldsymbol{n}_1(t), \ldots, \boldsymbol{n}_M(t)\right)^T, \quad \forall t \in \mathbb{R},$$

where $\boldsymbol{a}_1(\boldsymbol{r})$, $\boldsymbol{\Gamma}_i$ and $\boldsymbol{n}_i(t)$ are the antenna steering vector, complex gains and noise corruptions for station $i \leq M$. The matrix $W \in \mathbb{C}^{ML \times M}$ is called the **beamforming matrix** and is defined by

$$W := \begin{pmatrix} \boldsymbol{w}_1 & 0_{L\times 1} & \cdots & \cdots & 0_{L\times 1} \\ 0_{L\times 1} & \boldsymbol{w_2} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \boldsymbol{w}_{M-1} & 0_{L\times 1} \\ 0_{L\times 1} & \cdots & \cdots & 0_{L\times 1} & \boldsymbol{w}_M \end{pmatrix},$$

where $\boldsymbol{w}_i \in \mathbb{C}^L$ is the beamforming vector for station i.

Computing the covariance matrix $\Sigma := \mathbb{E}[\boldsymbol{y}(t)\boldsymbol{y}^H(t)] \in \mathbb{C}^{M\times M}$ yields

$$\Sigma = \iint_{\mathbb{S}^2} I(\boldsymbol{r})\boldsymbol{b}(\boldsymbol{r})\boldsymbol{b}^H(\boldsymbol{r})d\boldsymbol{r} \; + \; \sigma_n^2 W^H W. \tag{3.7}$$

Observe that the elements of Σ are such that $(\Sigma)_{i,k} = V_{i,k}, \quad \forall i,k = 1,\dots,M.$

Applying the vec$(\cdot)$ operator to eq. (3.7) allows to reformulate this matrix equation as a linear system

$$\boldsymbol{V} = \mathcal{B}I \; + \; \boldsymbol{\epsilon}, \tag{3.8}$$

where $\boldsymbol{V} = \text{vec}(\Sigma) \in \mathbb{C}^{M^2}$, $\boldsymbol{\epsilon} = \sigma_n^2 \text{vec}(W^H W) \in \mathbb{C}^{M^2}$. $\mathcal{B} : \mathbb{C}^{\mathbb{S}^2} \to \mathbb{C}^{M^2}$ is a linear operator given by

$$\mathcal{B} : \begin{cases} \mathbb{C}^{\mathbb{S}^2} \to \mathbb{C}^{M^2}, \\ I \mapsto \iint_{\mathbb{S}^2} I(\boldsymbol{r})\boldsymbol{b}^*(\boldsymbol{r}) \otimes \boldsymbol{b}(\boldsymbol{r})d\boldsymbol{r}, \end{cases}$$

using the known property [23] vec$(\boldsymbol{x}\boldsymbol{x}^H) = \boldsymbol{x}^* \otimes \boldsymbol{x}$, with $\otimes$ designating the Kronecker product.

3 Comparison with the Classical Measurement Equation

Before investigating potential new imaging algorithms that would make computationally feasible a sky reconstruction based on eq. (3.5), we propose to assess the performances of this more general model relatively to the classical model eq. (2.12) page 18. This performance analysis will focus only on the accuracy of the reconstructed sky image, without addressing (yet) the potential computational overhead caused by the use of one model rather than the other.

In all that follows, we will call **model** a linear operator $\mathcal{H} \in \mathbb{C}^{N_V \times N^2}$, that is said to describe the link between the true underlying sky image $\boldsymbol{I}_0 \in \mathbb{R}^{N^2}$ and the data $\boldsymbol{V} \in \mathbb{C}^{N_V}$ available to us

$$\boldsymbol{V} = \mathcal{H}\boldsymbol{I}_0. \tag{3.9}$$

Hence, a given model $\mathcal{H} \in \mathbb{C}^{N_V \times N^2}$ yields an associated measurement equation eq. (3.9). For the classical measurement equation described in section 2, the model is given by the matrix $\mathcal{A} \in \mathbb{C}^{JN^2 \times N^2}$, derived in eq. (2.15) page 21. In this case, we have $N_V = JN^2$, with J the total number of baselines used for the recovery of the sky.

Alternatively, one can use the more general measurement equation introduced in section 2. Then, the model is given by the matrix $B \in \mathbb{C}^{M^2 \times N^2}$, discrete analog of the linear operator $\mathcal{B}$ in eq. (3.8). For a grid $\Delta_{N^2} = \{\boldsymbol{r}_1, \dots, \boldsymbol{r}_{N^2}\} \subset \mathbb{S}^2$ on the field of view, this matrix is defined as

$$B := \left(\boldsymbol{b}^*(\boldsymbol{r}_1) \otimes \boldsymbol{b}(\boldsymbol{r}_1), \dots, \boldsymbol{b}^*(\boldsymbol{r}_{N^2}) \otimes \boldsymbol{b}(\boldsymbol{r}_{N^2})\right),$$

with $\boldsymbol{b}(\boldsymbol{r}) \in \mathbb{C}^M$ as in eq. (3.6) page 34. In practice, we often work with a reduced version of B, where we have removed the rows corresponding to auto-correlations, too severely corrupted by the thermal noise. In all that follows, we will assume that such a reduction on B has been performed.

To summarize, we propose to compare and assess the adequacy of the two concurrent models:

$$\textbf{M1: } \boldsymbol{V}_1 = \mathcal{A}\boldsymbol{I}_0, \quad \text{vs.} \quad \textbf{M2: } \boldsymbol{V}_2 = B\boldsymbol{I}_0,$$

where $\boldsymbol{V}_1$ and $\boldsymbol{V}_2$ are two alternative representations of the set of visibilities, reshaped according to the operators $\mathcal{A}$ and B previously described.

To this end, we investigate the accuracy of various sky estimates, obtained under the assumption of each model. More specifically, we focus on the least squares, CLEAN and LASSO estimates.

The two models will be stress-tested for the scenario described in table 3.1. This scenario was kept simple to insure a more controlled environment that would allow an easier interpretation of the simulation output, while remaining realistic enough to illustrate the conceptual differences existing between the two models.

3.1 Simulation Details

We used a custom simulation tool, as classical radio-astronomy packages such as CASA or OSKAR did not offer the flexibility to (easily) change the model used for the sky reconstruction. This prototypical tool, built as a MATLAB GUI, performs end-to-end processing of the data resulting from modern hierarchical radio telescopes: from sky generation and data acquisition to image reconstruction and cleaning.

In practice, the visibilities are estimated from a finite number of samples N_s of the stations beaformed outputs $y_i(t)$. Hence, we simulated the visibilities according to the following methodology, inspired from the actual processing chain of a modern hierarchical radio-telescope such as LOFAR:

1 **Generate and sample the time varying sources' complex amplitudes.** This can be achieved by drawing N independent observations

$$S := (\hat{\boldsymbol{s}}(t_1), \dots, \hat{\boldsymbol{s}}(t_{N_s})) \in \mathbb{C}^{Q \times N_s}$$

from the random vector $\hat{\boldsymbol{s}}(t) \sim \mathbb{C}\mathcal{N}_Q(0, \Sigma_s)$, with $\hat{\boldsymbol{s}}(t) \in \mathbb{C}^Q$ a Q-dimensional random vector containing the complex amplitudes of each sources within the sky, and $\Sigma_s = \text{diag}(\sigma_1^2, \dots, \sigma_Q^2) \in \mathbb{C}^{Q \times Q}$.

2 **Compute samples of the signals measured by each antenna.** This can be achieved by multiplying the previous samples by the **antenna steering array** $A \in \mathbb{C}^{ML \times Q}$ and corrupt this by some stochastic noise,

$$\boldsymbol{x}(t_i) = e^{j2\pi f_0 t_i} A\hat{\boldsymbol{s}}(t_i) + \boldsymbol{n}(t_i), \quad \forall i = 1, \dots, N_s$$

with $\boldsymbol{n}(t_1), \dots, \boldsymbol{n}(t_{N_s})$ independent observations from the random vector $\boldsymbol{n}(t) \sim \mathbb{C}\mathcal{N}_{ML}(0, \sigma_n^2 I_{ML})$. The matrix $A \in \mathbb{C}^{ML \times Q}$ is given by

$$A := (\boldsymbol{a}(\boldsymbol{r}_1), \dots, \boldsymbol{a}(\boldsymbol{r}_Q)),$$

with $\boldsymbol{a}(\boldsymbol{r}_q)$ the antenna steering vector towards the qth source. The samples can be stored in the empirical matrix defined by

$$X := (\boldsymbol{x}(t_1), \dots, \boldsymbol{x}(t_{N_s})) \in \mathbb{C}^{ML \times N_s}.$$

We have $X = AS + N$, with $N = (\boldsymbol{n}(t_1), \dots, \boldsymbol{n}(t_{N_s})) \in \mathbb{C}^{ML \times N_s}$.

3 **Apply beamforming to the antenna time-series.** Let $W \in \mathbb{C}^{M \times ML}$ be the

Sky Model

Name	Notation	Value
Number of sources	Q	5
Positions of sources	$(m_q, l_q)^T \in \mathbb{R}^2$	Uniform distribution
Sources intensities	σ_q^2	Log-normal distribution $(\mu, \sigma) = (\log(2), 0.7)$
Frequency/wavelength of sources	f_0, λ_0	75 MHz
Field of view	Θ	0.015 rad $\sim 1°$
Resolution	θ_{min}	100 arcsec
Center field of view	$\boldsymbol{r}_0$	RA:0H27M28S, DEC:52D54M55S

Antennas

Name	Notation	Value
Number of antennas per station	L	48
Position of antennas	$\boldsymbol{p}_j^{(k)} \in \mathbb{R}^3$	LOFAR HBA
Thermal Noise Variance	σ_n^2	500

Stations

Name	Notation	Value
Beamformer	–	Matched Beamforming
Number of stations	M	12
Position of stations	$\boldsymbol{p}_k \in \mathbb{R}^3$	LOFAR HBA Core Stations

Table 3.1: Description of the scenario chosen to assess the accuracy of the two models. The number of sources in the sky and the size of the field of view can in practice be much larger. This rather simplified scenario was preferred to more realistic and complex scenarios, as we believe its simplicity makes it more illustrative for the purpose of understanding the advantages and limitations of the investigated models.

beamforming matrix for the chosen beamforming technique. Then the corresponding samples from the beaformed signals at each station are given by

$$\boldsymbol{y}(t_i) = W^H \boldsymbol{x}(t_i), \quad \forall i = 1, \dots, N_s.$$

The empirical matrix Y is given by

$$Y = W^H X \in \mathbb{C}^{M \times N_s}.$$

4 **Construct an estimate of Σ.** We form the classical maximum-likelihood estimate of the covariance matrix:

$$\hat{\Sigma} = \frac{1}{N_s} Y Y^H = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{y}(t_i) \boldsymbol{y}(t_i)^H \in \mathbb{C}^{M \times M}.$$

In this very special case, as the mean of $\boldsymbol{y}(t)$ is known, the above estimator is unbiased [13]. The off-diagonal terms of $\hat{\Sigma}$ correspond then to the simulated visibilities.

For the subsequent experiments, we will assume that this simulation model is valid.

3.2 Experiment

Recall that the operator $\mathcal{H}$ designates either one of the models $\mathcal{A}$ or B. The set of visibilities $\boldsymbol{V}$ is assumed to have been correctly reshaped to agree with the corresponding model.

Least Squares Imaging

Given a set of visibilities $\boldsymbol{V} \in \mathbb{C}^{N_V}$ and a model $\mathcal{H} \in \mathbb{C}N_V \times N^2$, we want to find the sky estimate $\hat{\boldsymbol{I}}_{LS} \in \mathbb{R}^{N^2}$ that explains the best the data in the least squares sense:

$$\hat{\boldsymbol{I}}_{LS} = \text{argmin}_{\boldsymbol{I} \in \mathbb{R}^{N^2}} \|\boldsymbol{V} - \mathcal{H}\boldsymbol{I}\|_2^2. \tag{3.10}$$

When the operator $\mathcal{H}$ is full-column rank, the unique solution to the least squares optimization problem eq. (3.10) is given by solving the normal equations

$$\left(\mathcal{H}^H \mathcal{H}\right) \hat{\boldsymbol{I}}_{LS} = \mathcal{H}^H \boldsymbol{V} \;\Leftrightarrow\; \hat{\boldsymbol{I}}_{LS} = \mathcal{H}^\dagger \boldsymbol{V}, \tag{3.11}$$

where $\mathcal{H}^\dagger$ is the **Moore-Penrose pseudoinverse**, given in this case by

$$\mathcal{H}^\dagger = \left(\mathcal{H}^H \mathcal{H}\right)^{-1} \mathcal{H}^H.$$

As elegant and convenient as eq. (3.11) might be, direct computation is often computationally prohibitive in practice. One can then either approximate eq. (3.11) to make its computation affordable, or try and use an iterative solver such as the gradient descent algorithm to solve for the least squares problem eq. (3.10).

The first approach leads to the so-called dirty image

$$\hat{\boldsymbol{I}}_D = \left[\text{diag}\left(\mathcal{H}^H \mathcal{H}\right)\right]^{-1} \mathcal{H}^H \boldsymbol{V},$$

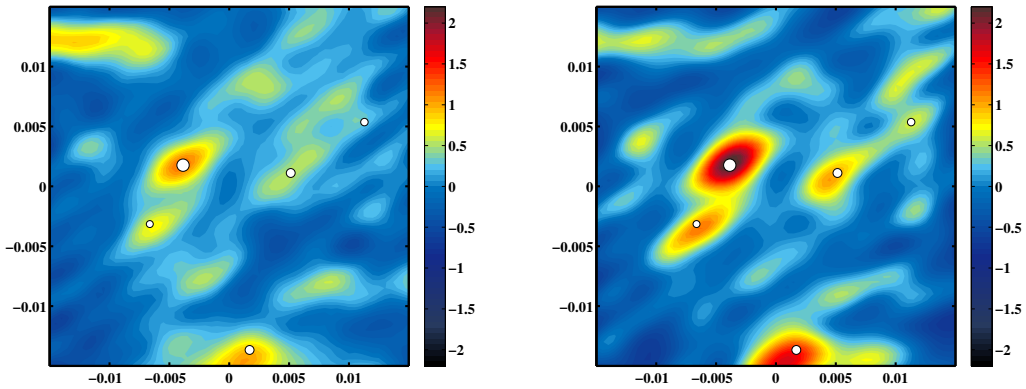
where the matrix $\mathcal{H}^H\mathcal{H}$ has been approximated by its diagonal part. The dirty images for models **M1** and **M2** are shown in fig. 3.1. For both, we observe that the resulting estimate is severely corrupted by artifacts, making difficult the recovery of actual sources within the image. This is maybe even more problematic for the dirty image obtained under the assumption of the classical model, whose dynamical range does not permit the proper discrimination between actual sources and artifacts (sidelobes have similar intensities as actual sources). In any case, the accuracy of both estimates is rather unsatisfying for practical purposes, which suggests the use of iterative solvers to estimate a solution to the least squares problem.

Here, we investigate the use of the gradient descent algorithm as a mean to approximate the least squares solution to eq. (3.10). Starting from the null image $\hat{\boldsymbol{I}}_{LS}^{(0)} = \boldsymbol{0}$, the estimate at iteration $n \geq 0$ is updated according to the following equation

$$\begin{aligned} \hat{\boldsymbol{I}}_{LS}^{(n+1)} &= \hat{\boldsymbol{I}}_{LS}^{(n)} - \tau^{(n)} \nabla\Phi\left(\hat{\boldsymbol{I}}_{LS}^{(n)}\right), \\ &= \hat{\boldsymbol{I}}_{LS}^{(n)} + \tau^{(n)} \mathcal{H}^H \left(\boldsymbol{V} - \mathcal{H}\hat{\boldsymbol{I}}_{LS}^{(n)}\right), \end{aligned} \tag{3.12}$$

with $\Phi\left(\boldsymbol{I}\right) = \|\boldsymbol{V} - \mathcal{H}\boldsymbol{I}\|_2^2$ the convex and differentiable functional to minimize. The parameter $\tau^{(n)} \geq 0$ is chosen optimally at each iteration according to eq. (2.21).

We run the gradient descent algorithm for both models with the same input



(a) Dirty Image for the model **M1** (Old Model)

(b) Dirty Image for the model **M2** (New Model)

Figure 3.1: Comparison of the dirty images obtained with the two concurrent models M1 and M2. For both models, we observe that the dirty image is severely corrupted by artifacts, which complicates the identification of actual sources within it. This is even more obvious for the classical model **M1** in (a), where we can observe that the dynamical range of the dirty image does not allow to properly discriminate between artifacts and actual sources.

visibilities estimated from 800 samples from the stations outputs $y_i(t)$. The least squares estimates for selected iterations of the gradient descent algorithm have been plotted on fig. 3.3 page 42 for the two concurrent models. We observe that the least squares estimate obtained under the assumption of the classical model exhibits rather unexpected behavior: the estimate first improves up to iteration 10 and then gradually starts to deteriorate as the number of iterations grows. In comparison, the least squares estimate obtained under the assumption of the new model does not seem to suffer from the same defect (see fig. 3.2). Before jumping to conclusions on whether or not one model is more adequately describing the problem, one has first to look at the condition number of the operators $\mathcal{A}$ and B. Indeed, it is well known that a bad conditioning of the model $\mathcal{H}$ can seriously affect the least squares estimate in the case of noisy data (as small perturbations in the data $\boldsymbol{V}$ can potentially be magnified importantly by the term $\left(\mathcal{H}^H\mathcal{H}\right)^{-1}$ in the pseudoinverse). Hence, it could be that the disturbing behavior witnessed on fig. 3.3 for the classical model is only the consequence of a bad conditioning of the matrix $\mathcal{A}$ (an issue that, even though problematic, can still be addressed by regularizing the least squares problem). When computing the condition numbers of both models for this specific example we get

$$\kappa_2\left(\mathcal{A}\right) = 166, \quad \kappa_2\left(B\right) = 974.72.$$

The classical model has a smaller condition number than the new model, and still the resulting estimate exhibits a much more dramatic deterioration as the number of iterations increases in the gradient descent algorithm. This suggests that the peculiar behavior witnessed on fig. 3.3 cannot be reduced to numerical considerations. The only sensible explanation left to us is then to interpret this deterioration of the estimate as a fundamental disagreement between the model and the actual sky image: the sky that would have generated the input set of visibilities according to the model $\mathcal{A}$ is different from the actual sky image. This observation seriously undermines the ability of the classical model to correctly relate beamformed data with the underlying sky image. In comparison, under the assumption that the

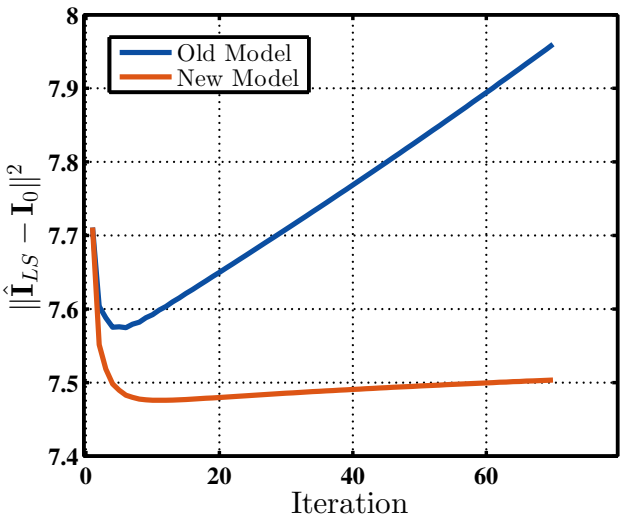


Figure 3.2: Accuracy of the least squares estimate with respect to the true sky for both models and various iterations of the gradient descent algorithm. We observe that the least squares estimate obtained under the assumption of the classical model exhibits a rather peculiar behavior: it first improves up to iteration 10 and then gradually starts to deteriorate as the number of iterations grows. In comparison, the least squares estimate obtained under the assumption of the new model seems not to suffer from the same defect.

simulation procedure described in section 3.1 is valid, our model seems much more adequate in this framework, which suggests its use in obtaining higher accuracy in the reconstructed sky.

The CLEAN Algorithm

In practice, the least squares estimate previously computed is rarely used. Indeed, this estimate has a tendency to spread the energy across the image plane, producing a dense sky estimate with extended structures and patterns, when the underlying sky image is most of the time assumed sparse.

For this reason, the CLEAN algorithm is often preferred in practice, as it usually produces sparser sky estimates. We have seen that it can be seen as a nonlinear version of the gradient descent algorithm, aiming at minimizing the optimization problem eq. (3.10), while (hopefully) producing a sparse[2] sky by taking only canonical directions (see section 4 for further details on the algorithm).

The number of CLEAN iterations as well as the gain parameter to use for each iteration are heavily application-dependent, and can seriously affect the performances of the algorithm if poorly chosen. In the lack of knowledge from the considered case, the rule of thumb is to use a very small value for the gain parameter: this will significantly reduce the speed of the algorithm, but will prevent it from making too big steps that could lead to divergence. Here, we chose the default input parameters for the CLEAN procedure implemented in the CASA package: $\tau = 0.1$ and $N_{iter} = 1000$. We then ran the algorithm as described in section 4 for both models **M1** and **M2**, with the same input data. Outputs of various iterations of the algorithm are available on fig. 3.4 page 43. We can observe that the CLEAN estimates for both models are sparser than the least squares estimates, with an increased dynamical range: the residual background floor presents a much smoother structure with reduced sidelobes, and the sources are much better resolved. The

[2]Understand here an estimate with a low L^1 norm.

CLEAN estimate for the classical model **M1** still exhibits a lot of false positives: many components rise above the background floor with intensities similar to actual sources. This is very problematic, as it complicates a lot the detection of actual sources among those wrongly detected sources[3]. With the model **M2**, the risk of confusion in the detection of actual sources is much smaller, as the actual sources clearly rise above the background floor.

Hence, once again, the model **M2** seems to yield better CLEAN estimates than the classical model, suggesting a better adequacy of our model with beamformed data.

LASSO Imaging

Even though the CLEAN algorithm usually produces sky images with relatively small L^1 norms, its performances heavily depend on the choice of the gain parameter as well as the stopping criterion. Hence, a more systematic approach to obtain sparse sky images could consist in regularizing the least squares optimization problem eq. (3.10). One possibility is for example to add a L^1 penalty term to the objective functional [9]. This yields the famous LASSO problem [21].

Given a set of visibilities $\boldsymbol{V} \in \mathbb{C}^{N_V}$ and a model $\mathcal{H} \in \mathbb{C}N_V \times N^2$, find the sky estimate $\hat{\boldsymbol{I}}_{LS} \in \mathbb{R}^{N^2}$ that explains the best the data in the following sense:

$$\hat{\boldsymbol{I}}_{LASSO} = \text{argmin}_{\boldsymbol{I}\in\mathbb{R}^{N^2}} \|\boldsymbol{V} - \mathcal{H}\boldsymbol{I}\|_2^2 + \lambda\|\boldsymbol{I}\|_1, \tag{3.13}$$

with $\lambda \in \mathbb{R}$ some penalty parameter. The penalized least squares optimization problem eq. (3.13) corresponds to the Lagrangian formulation of the constrained problem:

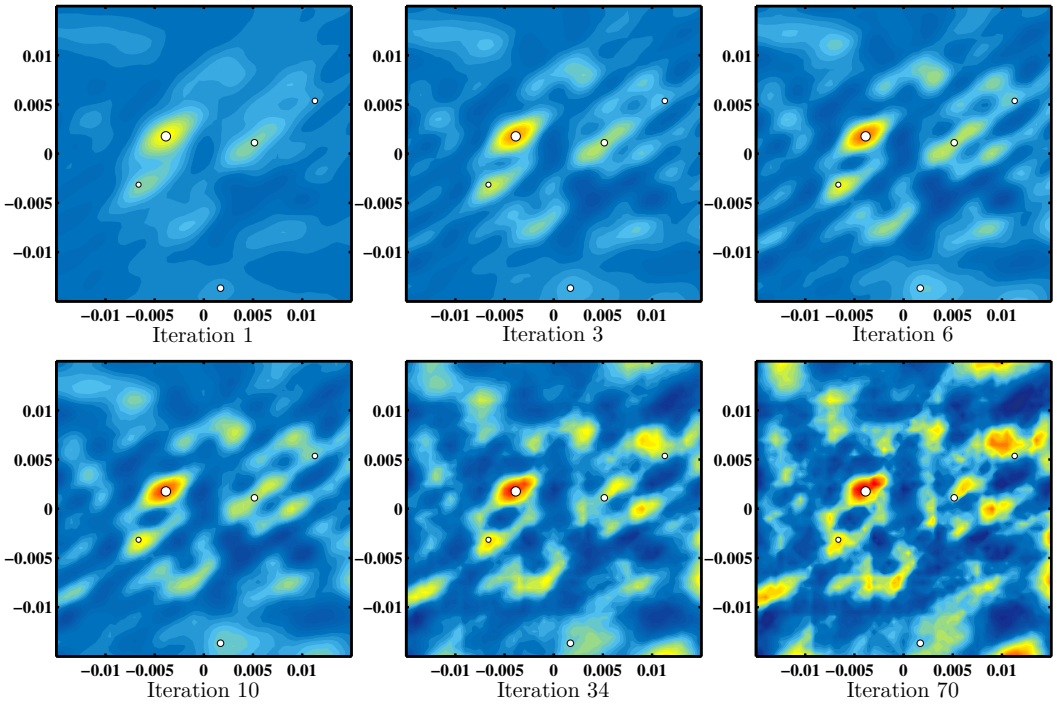
$$\hat{\boldsymbol{I}}_{LASSO} = \text{argmin}_{\boldsymbol{I}\in\mathbb{R}^{N^2}} \|\boldsymbol{V} - \mathcal{H}\boldsymbol{I}\|_2^2, \quad \text{such that } \|\boldsymbol{I}\|_1 \leq \epsilon,$$

where the penalty parameter λ is inversely related to the constraint parameter ϵ. The strength of the penalty parameter $\lambda \geq 0$ in eq. (3.13) controls the sparsity of the resulting sky estimate.

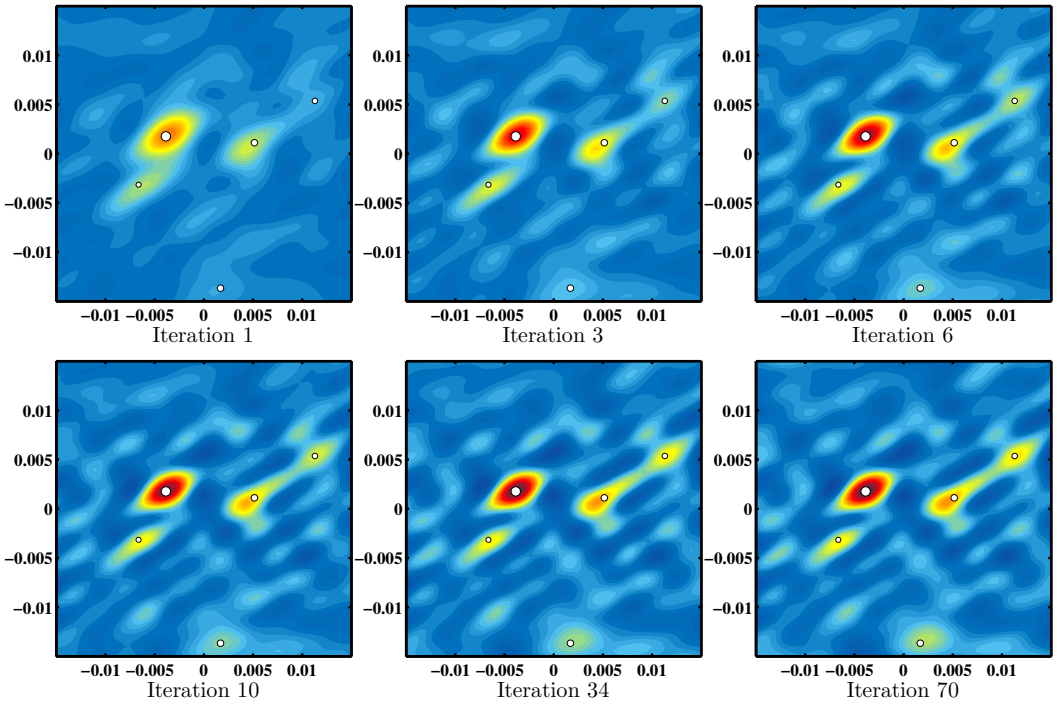
We used the **FISTA algorithm (Fast Iterative Shrinkage-Thresholding Algorithm)** [2] to solve for the LASSO problem for both models with the same input data. Outputs of the algorithm for various values of the penalty parameter λ are available on fig. 3.5. We observe that for any choice of λ, the sky estimate obtained with the new model **M2** is both sparser and more accurate than the sky estimate obtained with the classical model **M1**. Moreover, for small values of λ, skies estimated with the classical model present many false positives, that will one more time complicate the recovery of actual sources within the images. Finally, sky estimates obtained with the new model are much better aligned with the actual sources than for the classical model.

One more time, the new model **M2** seems to produce much more accurate estimates of the sky image, which confirms its superiority over the classical model.

[3] For example we can count on fig. 3.4 (a) approximately 13 potential sources with comparable intensities, while the sky is only composed of 5 actual sources. Choosing which of these sources are actual sources is a delicate task!

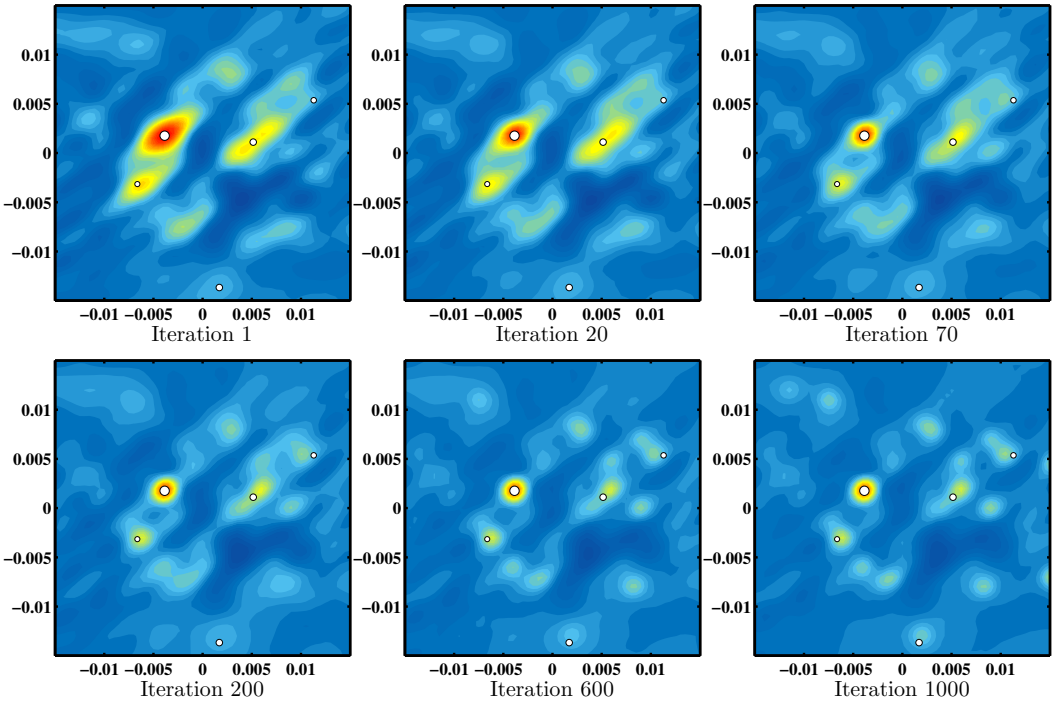


(a) Least squares estimate $\hat{\boldsymbol{I}}_{LS}^{(n)}$ for various iterations $n \geq 0$ of the gradient descent algorithm applied to the old model.

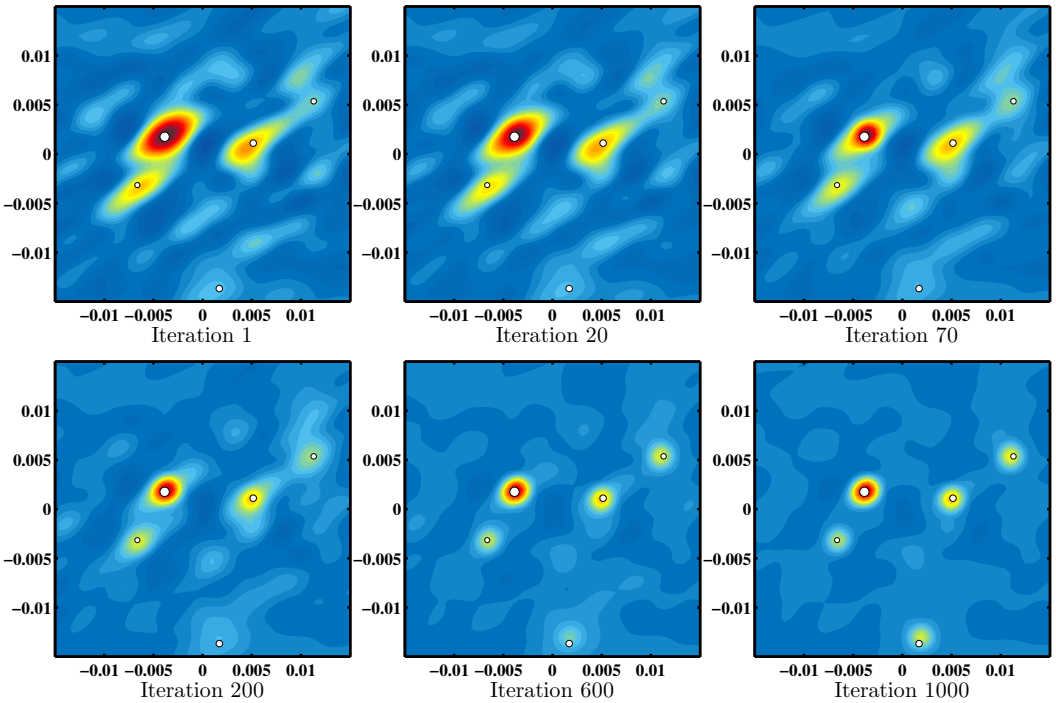


(b) Least squares estimate $\hat{\boldsymbol{I}}_{LS}^{(n)}$ for various iterations $n \geq 0$ of the gradient descent algorithm applied to the new model.

Figure 3.3: Gradient descent algorithm applied to both models as a mean to approximate the least squares solution to eq. (3.10). We observe that the least squares estimate for the classical model seems to deteriorate as the number of iterations grows. This peculiar behavior does not occur with the more general model we propose: this seems to suggest a better agreement of the latter with the beamformed data.

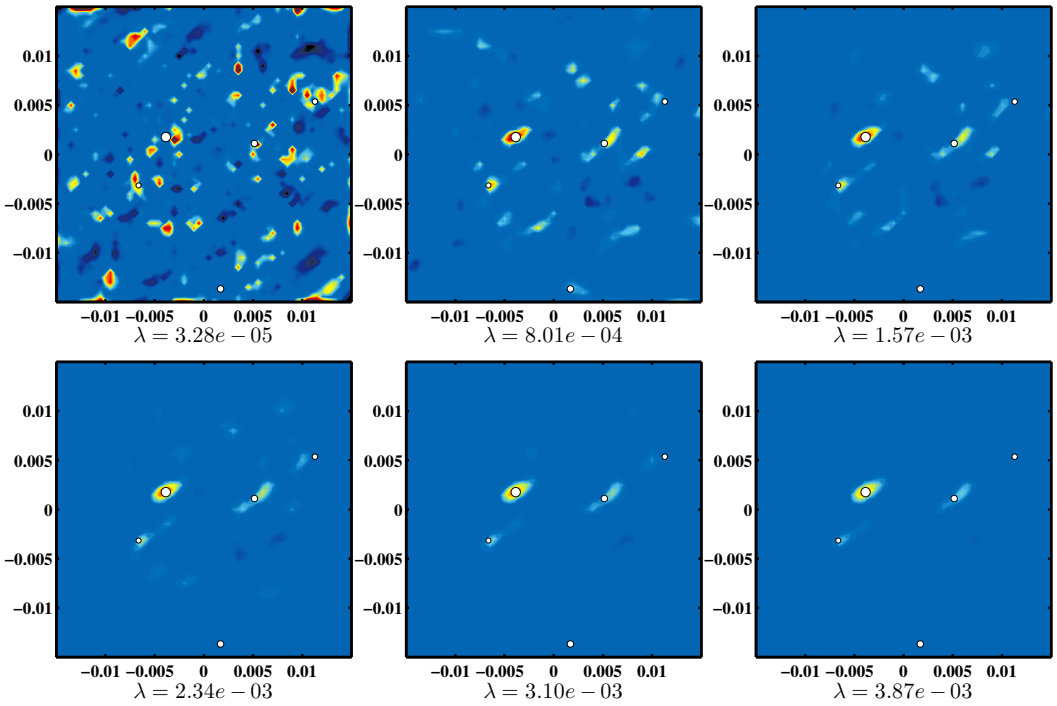


(a) CLEANed sky estimate $\hat{\boldsymbol{I}}^{(n)}_{CLEAN}$ for various iterations $n \geq 0$ of the CLEAN algorithm applied to the old model.

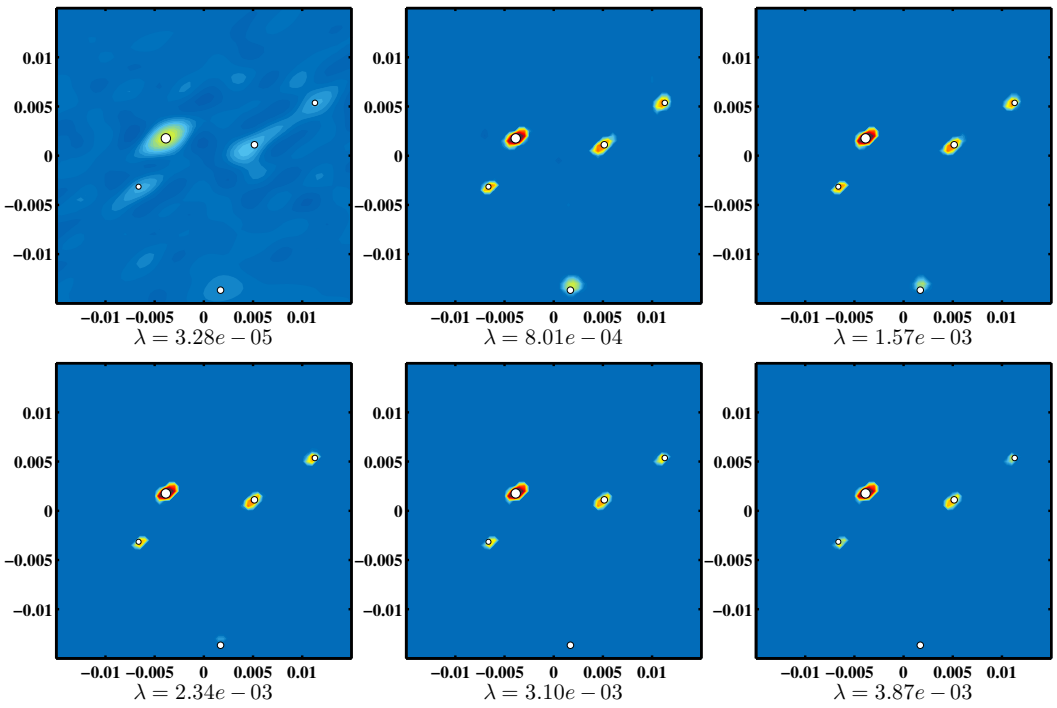


(b) CLEANed sky estimate $\hat{\boldsymbol{I}}^{(n)}_{CLEAN}$ for various iterations $n \geq 0$ of the CLEAN algorithm applied to the old model.

Figure 3.4: CLEAN algorithm applied to both models with the same input data. We observe that the identification of actual sources within the image is much easier with the new model **M2**, as the sources are better resolved and clearly rise above the residual background. In comparison, the CLEAN estimate obtained with the classical model model **M1** presents many components with similar intensities as actual sources, increasing the risk of confusion among actual and falsely detected sources.



(a) LASSO sky estimate for various values of λ obtained with the old model.



(b) LASSO sky estimate for various values of λ obtained with the new model.

Figure 3.5: Collection of LASSO estimates for both models with the same input data. We observe that for any choice of penalty parameter λ, the sky estimate obtained with the new model **M2** is both sparser and more accurate than the sky estimate obtained with the classical model **M1**. Moreover, for small values of λ, the sky estimated with the classical model presents many false positives, that will very certainly complicate the recovery of actual sources within the image.

4

Towards a New Imaging Pipeline

The results of the various experiments carried out in section 3 seem to indicate a clear superiority of the new model introduced in section 2 over the classical model, at least in terms of the accuracy of the recovered sky image for the three algorithms investigated. However, this preliminary analysis is not complete enough to legitimate the use of the new model, as it completely eludes the computational aspect of the problem. Indeed, despite its apparent lack of accuracy, the classical model still has a serious asset over us: the A-projection algorithm. As we have seen in section 3.2 page 22, this algorithm is essentially a fast way to apply $\mathcal{A}$ or $\mathcal{A}^H$, by leveraging the convolution theorem as well as the small support of the stations beamshapes in the Fourier domain. This permits an efficient computation of the residual image $\mathcal{A}^H(\boldsymbol{V} - \mathcal{A}\boldsymbol{I})$ that appears at each iterations of both the gradient descent algorithm and the CLEAN algorithm (see section 4 page 25), and hence helps in reducing the computational cost of those imaging algorithms.

Unfortunately, the A-projection algorithm cannot be extended to the new model, as it heavily depends on the convolution theorem and hence on the Fourier domain, from which we precisely agreed to step away, as we noticed that beamformed data and the Fourier domain were not as intimately linked as usually claimed. Deprived from the A-projection algorithm, we could then either try and derive an analog algorithm tailored to the new model in order to accelerate the multiplication by B or B^H, or we could alternatively try to reduce *significantly* the number of iterations required for the classical imagers to converge, so that the computational overhead introduced by applying B or B^H becomes less significant.

In this chapter, we propose a preconditioning of the problem based on the Gram-Schmidt orthogonalization process, and show that it critically reduces the number of iterations required for the classical imagers to converge. In particular, we show that with this preconditioning, the gradient descent algorithm converges in one iteration and that, under certain conditions, the LASSO solution can be either directly computed or reasonably well approximated from the least squares estimate.

Finally, we assess the robustness of the method to various variables of the problem: noise, chosen beamformer and telescope layout.

1 Least squares imaging and Orthogonalization

In the remark remark 2.2 page 34, we proposed a geometric interpretation of the visibilities in the new model. More specifically, we noticed that, with an appropriate choice of inner product on the space $\mathbb{C}^{\mathbb{S}^2}$, the visibilities could be interpreted as

inner product between the sky image and some functions:

$$V_{i,k} = \langle I, \beta_{i,k} \rangle, \quad \forall i \neq k, \tag{4.1}$$

with $\beta_{i,k}(\boldsymbol{r}) = b_i^*(\boldsymbol{r}) b_k(\boldsymbol{r}),\ \forall \boldsymbol{r} \in \mathbb{S}^2$, and the inner product in eq. (4.1) given by

$$\langle f, g \rangle = \iint_{\mathbb{S}^2} f(\boldsymbol{r}) g^*(\boldsymbol{r}) d\boldsymbol{r}, \quad \forall f, g \in \mathbb{C}^{\mathbb{S}^2}. \tag{4.2}$$

This is a very appealing observation, as it provides a simple and elegant least squares imager in the specific case where the family of functions $\{\beta_{i,k}\}$ form an orthonormal family with respect to the inner product eq. (4.2). Indeed, in that case, we can straightforwardly interpret the visibilities as coefficients of the sky image projected onto the subspace spanned by the basis $\{\beta_{i,k}\} \subset \mathbb{C}^{\mathbb{S}^2}$, and compute this projection (and thus the least squares estimate) using elementary linear algebra:

$$\hat{I}_{LS}(\boldsymbol{r}) = \sum_{i,k} \langle I, \beta_{i,k} \rangle \beta_{i,k}(\boldsymbol{r}) = \sum_{i,k} V_{i,k} \beta_{i,k}(\boldsymbol{r}). \tag{4.3}$$

Note that the functions $\beta_{i,k}$ depend only on the stations beamshapes, which are entirely determined and independent of the data once the antennas' gains have been estimated during the calibration process. In practice, depending on the layout of the considered telescope, the functions $\beta_{i,k}$ can be linearly independent, but they are certainly not orthogonal in general, forbidding the use of eq. (4.3). However, it is always possible to orthogonalize the family of functions a posteriori, provided that we accordingly relate the set of visibilities with this new orthogonal family. This inspires the following strategy (summarized in fig. 4.1):

Starting from the family of functions $\{\beta_{i,k}\}$ and the set of visibilities $\{V_{i,k} = \langle I, \beta_{i,k} \rangle\}$,

1. Orthogonalize the family of functions $\{\beta_{i,k}\}$. This yields a new orthogonal family of functions $\{\beta_{i,k}^{\perp}\}$. We have $|\{\beta_{i,k}^{\perp}\}| \leq |\{\beta_{i,k}\}|$, with equality when the original family of functions $\{\beta_{i,k}\}$ is linearly independent.
2. Compute from the original set of visibilities $\{V_{i,k} = \langle I, \beta_{i,k} \rangle\}$ a new set of visibilities $\{V_{i,k}^{\perp}\}$ such that $V_{i,k}^{\perp} = \langle I, \beta_{i,k}^{\perp} \rangle,\ \forall i, k$. Again, we have $|\{\beta_{i,k}^{\perp}\}| \leq |\{\beta_{i,k}\}|$, with equality when the original family of functions $\{\beta_{i,k}\}$ is linearly independent
3. Compute the least squares sky estimate by using eq. (4.3) with the orthogonal family $\{\beta_{i,k}^{\perp}\}$ and the associated modified set of visibilities $\{V_{i,k}^{\perp}\}$.

An algorithm that would effectively achieve the above recipe can now be designed. For simplicity, we will drop the double indexes and assume an arbitrary ordering of the functions and the associated visibilities. The orthogonalization step **1** can easily be achieved by applying the renowned **Gram-Schmidt orthogonalization process**.

By analyzing the steps of this procedure, we can then understand how to modify the set of visibilities accordingly, so that the geometric interpretation in eq. (4.1) will still be valid for the orthogonalized family. Without loss of generality, assume for this analysis that the initial family of functions $\{\beta_{i,k}\}$ is linearly independent[1].

[1] For a linearly dependent family, drop redundant functions and associated visibilities.

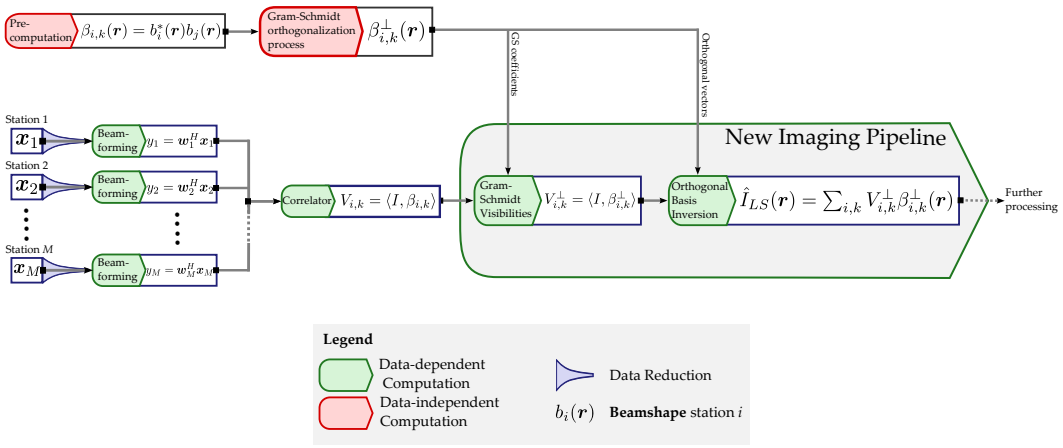


Figure 4.1: **Block diagram of the Gram-Schmidt-based least squares imager.**

Then, the Gram-Schmidt orthogonalization process can be described as follows:

- Start with the trivial family composed of only β_1 (choice of β_1 is arbitrary). Orthogonalizing a single-element family reduces to a normalization of β_1:

$$\beta_1^{\perp} = \frac{\beta_1}{\|\beta_1\|}, \tag{4.4}$$

  where $\|\beta_1\| := \sqrt{\langle \beta_1, \beta_1 \rangle}$ is the norm associated to the inner product defined in eq. (4.2).
- Increase iteratively the size of the orthogonal family by adding corrected elements from the input family. More precisely, for a given step i, remove the potential linear resemblances existing between β_i and the $i-1$ elements of the orthogonal family built so far, normalize the residual and add the resulting corrected element to the orthogonal family. This yields the following two-steps update equation

$$\tilde{\beta}_i = \beta_i - \sum_{k=1}^{i-1} \langle \beta_i, \beta_k^{\perp} \rangle \beta_k^{\perp}, \tag{4.5}$$

$$\beta_i^{\perp} = \frac{\tilde{\beta}_i}{\|\tilde{\beta}_i\|}, \tag{4.6}$$

  with $\{\beta_k^{\perp}\}_{k=1,\dots,i-1}$ the orthogonal family obtained up to iteration i.
- Stop when both families have the same cardinality.

From eqs. (4.4) to (4.6), an iterative scheme that produces a new set of visibilities such that $V_i^{\perp} = \langle I, \beta_i^{\perp} \rangle$ follows. For the first element $\beta_1^{\perp}$,

$$V_1^{\perp} = \langle I, \beta_1^{\perp} \rangle \overset{4.4}{=} \frac{1}{\|\beta_1\|} \langle I, \beta_1 \rangle = \frac{V_1}{\|\beta_1\|}, \tag{4.7}$$

where $V_1 = \langle I, \beta_1 \rangle$ the first element of the set of visibilities recorded by the telescope. After computing $V_1^\perp$, we can obtain $V_2^\perp$:

$$\begin{aligned} V_2^\perp &= \langle I, \beta_2^\perp \rangle, \\ &\overset{4.6}{=} \frac{1}{\|\tilde{\beta}_1\|} \langle I, \tilde{\beta}_1 \rangle, \\ &\overset{4.5}{=} \frac{1}{\|\tilde{\beta}_1\|} \left(\langle I, \beta_2 \rangle - \langle \beta_2, \beta_1^\perp \rangle \langle I, \beta_1^\perp \rangle \right), \\ &\overset{4.7}{=} \frac{1}{\|\tilde{\beta}_1\|} \left(V_2 - \langle \beta_2, \beta_1^\perp \rangle V_1^\perp \right), \end{aligned}$$

where we have used the bi-linearity of the inner product for the third step. We can then proceed iteratively to obtain the rest of the visibilities:

$$V_i^\perp = \frac{1}{\|\tilde{\beta}_i\|} \left(V_i - \sum_{k=1}^{i-1} \langle \beta_i, \beta_k^\perp \rangle V_k^\perp \right). \tag{4.8}$$

Equation (4.8) depends on the data V_i, the $i-1$ new visibilities computed up to iteration i and the coefficients $\|\tilde{\beta}_i\|$ and $\langle \beta_i, \beta_k^\perp \rangle$ computed during the Gram-Schmidt process. Hence, all those quantities are available, and the proposed scheme is viable.

The least squares sky estimate then follows from eq. (4.3):

$$\hat{I}_{LS}(\boldsymbol{r}) = \sum_{i=1}^{J} V_i^\perp \beta_i^\perp(\boldsymbol{r}).$$

The procedure is summarized and extended to the case where the input family is potentially linearly dependent on algorithm 3 page 49.

1.1 Continuous case

It is interesting to note that, contrarily to any of the classical imagers introduced in section 3, the previous approach works at the *continuous* level, at least theoretically. Indeed, we have the following identity [18, p. 154]

$$\iint_{\mathbb{S}^2} e^{j2\pi\langle \boldsymbol{r}, \boldsymbol{p} \rangle} d\boldsymbol{r} = 4\pi \times \operatorname{sinc}\left(2\pi \|\boldsymbol{p}\|_2\right), \quad \forall \boldsymbol{p} \in \mathbb{R}^3, \tag{4.9}$$

Moreover, using the results derived in section 1 of chapter 3, we see that, in the case of ideal gains and omnidirectional primary beamshape, we have

$$\beta_{i,k}(\boldsymbol{r}) = \sum_{h=1}^{L} \sum_{g=1}^{L} \omega_h^{(i)} \omega_g^{(k)*} e^{j2\pi\langle \boldsymbol{r}, \boldsymbol{p}_{h,g}^{(i,k)} \rangle}, \quad \forall \boldsymbol{r} \in \mathbb{S}^2,$$

where L is the number of antennas per station, $\omega_h^{(i)} \in \mathbb{C}$ the hth component of the beamforming vector $\boldsymbol{w}_i \in \mathbb{C}^L$ for station i, and $\boldsymbol{p}_{h,g}^{(i,k)} = \boldsymbol{p}_h^{(i)} - \boldsymbol{p}_g^{(k)}$.

Hence, the functions involved in the various iterations of the Gram-Schmidt algorithm essentially consist of (potentially very long) sums of complex exponentials of the form $e^{j2\pi\langle \boldsymbol{r}, \boldsymbol{p} \rangle}$. Therefore, we could in principle use eq. (4.9) directly in order to compute the various inner products and norms involved in the Gram-Schmidt

Algorithm 3: Gram-Schmidt orthogonalization process and modification of the visibilities

procedure ▷ INPUTS: Family of functions $\{\beta_i\}_{i=1,\dots,J}$ and associated visibilities $\{V_i\}_{i=1,\dots,J}$

$\beta_1^\perp \leftarrow \beta_1/\|\beta_1\|$;

$\mathcal{E} \leftarrow \{\beta_1^\perp\}$;

$V_1^\perp \leftarrow V_1/\|\beta_1\|$;

$j \leftarrow 1$;

for $i = 2$ **to** J **do**

$\tilde{\beta}_i \leftarrow \beta_i - \sum_{k=1}^{j} \langle \beta_i, \beta_k^\perp \rangle \beta_k^\perp$;

if $\|\tilde{\beta}_i\| \neq 0$ **then** ▷ *Drop* $\tilde{\beta}_i$ *and* V_i *if* $\|\tilde{\beta}_i\| = 0$

$j \leftarrow j + 1$;

$\beta_j^\perp \leftarrow \tilde{\beta}_i/\|\tilde{\beta}_i\|$;

$\mathcal{E} \leftarrow \{\beta_k^\perp | k = 1, \dots, j-1\} \cup \{\beta_j^\perp\}$;

$V_j^\perp \leftarrow \left(V_i - \sum_{k=1}^{j-1} \langle \beta_i, \beta_k^\perp \rangle V_k^\perp\right) / \|\tilde{\beta}_i\|$;

end

end

return $\mathcal{E} = \{\beta_i^\perp | i = 1, \dots, j\}$ and $\{V_i^\perp | i = 1, \dots, j\}$

algorithm.

Such an approach sounds particularly appealing, as it would provide us with a resolution independent least squares imager: all the computations would be performed analytically at the continuous level, and only at the very end we would form an image by sampling the continuous sky estimate. Unfortunately, this approach is only nice on paper, but completely infeasible in practice. Indeed, its effective implementation would involve very long and abominable expressions, that would very certainly be too costly to store and compute even with a software to perform symbolic computation: a very quick study of the trivial example of only two stations and 48 antennas per stations shows that above 5 millions of evaluations of the sinc function are required to perform the Gram-Schmidt algorithm analytically.

For practical purposes, we need then to sample the basis functions $\beta_{i,k}(\boldsymbol{r})$, to transform them into discrete quantities. This can be achieved in many different ways, and an efficient strategy still needs to be investigated. In what follows, we consider a uniform sampling of the basis functions $\beta_{i,k}(\boldsymbol{r})$ over the field of view.

1.2 Discrete Case

In keeping with previous methods, we choose to sample the different quantities involved in the problem on a uniform grid covering the field of view. For a grid $\Delta_{N^2} = \{\boldsymbol{r}_1, \dots, \boldsymbol{r}_{N^2}\} \subset \mathbb{S}^2$, we have seen in section 3 page 35 that the data model could be written as

$$\boldsymbol{V} = B\boldsymbol{I}_0, \tag{4.10}$$

where $\boldsymbol{V} = (V_1, \dots, V_J)^T \in \mathbb{C}^J$, $B \in \mathbb{C}^{J \times N^2}$ and $\boldsymbol{I}_0 = (I_0(\boldsymbol{r}_1), \dots, I_0(\boldsymbol{r}_{N^2})) \in \mathbb{C}^{N^2}$ is the the underlying sky image. Observe that the jth row of B is given by

$$(B)_{j,:} = (\beta_j^*(\boldsymbol{r}_1), \dots, \beta_j^*(\boldsymbol{r}_{N^2})),$$

where $\beta_j^*(\boldsymbol{r}) = (\boldsymbol{b}^*(\boldsymbol{r}) \otimes \boldsymbol{b}(\boldsymbol{r}))_j = b_{i_j}^*(\boldsymbol{r}) b_{k_j}(\boldsymbol{r})$ (see section 2 of chapter 2). In other words, the rows of B correspond to the conjugate transpose of sampled versions of the functions $\beta_{i,k}$, reordered according to the vec() operator. As is standard, we exclude from consideration the components of $\boldsymbol{V}$ (and hence the associated rows of B) that correspond to autocorrelations, thermal noise having rendered them untrustworthy.

Analogously to the continuous case, we can interpret eq. (4.10) as

$$\boldsymbol{V} = B\boldsymbol{I}_0 = \begin{pmatrix} \boldsymbol{\beta}_1^H \boldsymbol{I}_0 \\ \vdots \\ \boldsymbol{\beta}_J^H \boldsymbol{I}_0 \end{pmatrix} = \begin{pmatrix} \langle \boldsymbol{\beta}_1, \boldsymbol{I}_0 \rangle \\ \vdots \\ \langle \boldsymbol{\beta}_J, \boldsymbol{I}_0 \rangle \end{pmatrix},$$

with $\boldsymbol{\beta}_i = (\beta_i(\boldsymbol{r}_1), \dots, \beta_i(\boldsymbol{r}_{N^2}))^T \in \mathbb{C}^{N^2}$, and $\langle \cdot, \cdot \rangle$ the usual inner product on $\mathbb{C}^{N^2}$. Hence, if the family of vectors $\{\boldsymbol{\beta}_i\}_{i=1,\dots,J} \subset \mathbb{C}^{N^2}$ formed an orthonormal family, then we could recover the least squares sky estimate with

$$\hat{\boldsymbol{I}}_{LS} = B^H \boldsymbol{V} = \sum_{j=1}^{J} \langle \boldsymbol{\beta}_j, \boldsymbol{I}_0 \rangle \boldsymbol{\beta}_j. \tag{4.11}$$

Observe that eq. (4.11) corresponds also to the first iteration of the gradient descent algorithm used to minimize $\Phi(\boldsymbol{I}) = \|\boldsymbol{V} - B\boldsymbol{I}\|_2^2$ with initial estimate $\hat{\boldsymbol{I}}^{(0)} = \boldsymbol{0}$:

$$\begin{aligned} \hat{\boldsymbol{I}}^{(1)} &= \hat{\boldsymbol{I}}^{(0)} + \tau^{(1)} \nabla \Phi(\hat{\boldsymbol{I}}^{(0)}), \\ &= \boldsymbol{0} + \tau^{(1)} B^H (\boldsymbol{V} - B\boldsymbol{0}), \\ &= B^H \boldsymbol{V}, \end{aligned}$$

as $\tau^{(1)} = \|\nabla \Phi(\hat{\boldsymbol{I}}^{(0)})\|_2^2 / \|\nabla \Phi(\hat{\boldsymbol{I}}^{(0)})\|_B^2 = (\boldsymbol{V}^H \overbrace{BB^H}^{=Id} \boldsymbol{V}) / (\boldsymbol{V}^H \overbrace{BB^H}^{=Id} \overbrace{BB^H}^{=Id} \boldsymbol{V}) = 1$. Hence, in the specific case where the rows of B are orthogonal, the gradient descent converges exactly to the least squares estimate in *one single* step. This a remarkable result, as in the non-orthogonal case, an infinite number of iterations would theoretically be needed to approximate the least squares estimate with arbitrary precision.

It seems then than applying the same methodology as before in order to orthonormalize the family of vectors $\{\boldsymbol{\beta}_i\}_{i=1,\dots,J} \subset \mathbb{C}^{N^2}$ is a sensible thing to do, as it provides us with the best case scenario for the convergence of the gradient descent algorithm. The discrete version of algorithm 3 is given by algorithm 4. The least squares estimate of the sky image can be obtained from the outputs $B_\perp \in \mathbb{C}^{j \times N^2}$ and $\boldsymbol{V}_\perp \in \mathbb{C}^j$ of algorithm 4 as follows:

$$\hat{\boldsymbol{I}}_{LS} = B_\perp^H \boldsymbol{V}_\perp = \sum_{i=1}^{j} V_i^\perp \boldsymbol{\beta}_i^\perp = \sum_{i=1}^{j} \langle \boldsymbol{\beta}_j^\perp, \boldsymbol{I}_0 \rangle \boldsymbol{\beta}_i^\perp. \tag{4.12}$$

Algorithm 4: Gram-Schmidt Orthogonalization Process and Modification of the Visibilities (discrete version)

1: **procedure** ▷ INPUTS: Family of vectors $\{\boldsymbol{\beta}_i\}_{i=1,\dots,J}$
2: and associated visibilities $\{V_i\}_{i=1,\dots,J}$
3: $\boldsymbol{\beta}_1^{\perp} \leftarrow \boldsymbol{\beta}_1 / \|\boldsymbol{\beta}_1\|_2$;
4: $V_1^{\perp} \leftarrow V_1 / \|\boldsymbol{\beta}_1\|_2$;
5: $j \leftarrow 1$;
6: **for** $i = 2$ **to** J **do**
7: $\tilde{\boldsymbol{\beta}}_i \leftarrow \boldsymbol{\beta}_i - \sum_{k=1}^{j-1} \langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_k^{\perp} \rangle \boldsymbol{\beta}_k^{\perp}$;
8: **if** $\|\tilde{\boldsymbol{\beta}}_i\|_2 \neq 0$ **then** ▷ *Drop* $\tilde{\boldsymbol{\beta}}_i$ *and* V_i *if* $\|\tilde{\boldsymbol{\beta}}_i\|_2 = 0$
9: $j \leftarrow j + 1$;
10: $\boldsymbol{\beta}_j^{\perp} \leftarrow \tilde{\boldsymbol{\beta}}_i / \|\tilde{\boldsymbol{\beta}}_i\|_2$;
11: $V_j^{\perp} \leftarrow \left(V_i - \sum_{k=1}^{j-1} \langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_k^{\perp} \rangle V_k^{\perp} \right) / \|\tilde{\boldsymbol{\beta}}_i\|$;
12: **end**
13: **end**
14: $\boldsymbol{V}_{\perp} = \left(V_1^{\perp}, \dots, V_j^{\perp} \right)^T$;
15: $B_{\perp} = \begin{pmatrix} \boldsymbol{\beta}_1^{\perp H} \\ \vdots \\ \boldsymbol{\beta}_j^{\perp H} \end{pmatrix}$;
16: **return** $B_{\perp} \in \mathbb{C}^{j \times N^2}$ and $\boldsymbol{V}_{\perp} \in \mathbb{C}^j$

We will refer to this estimate as the **Gram-Schmidt least squares estimate**, as a reference to the orthogonalization procedure on which it relies.

In fig. 4.2 page 52, we compared the Gram-Schmidt estimate eq. (4.12) with the output after 500 iterations of the gradient descent algorithm used to minimize $\Phi(\boldsymbol{I}) = \|\boldsymbol{V} - B\boldsymbol{I}\|_2^2$. We observe that the two sky estimates appear visually and structurally very similar, even though the Gram-Schmidt estimate is still a 1000 times more accurate in terms of minimizing $\Phi(\boldsymbol{I})$.

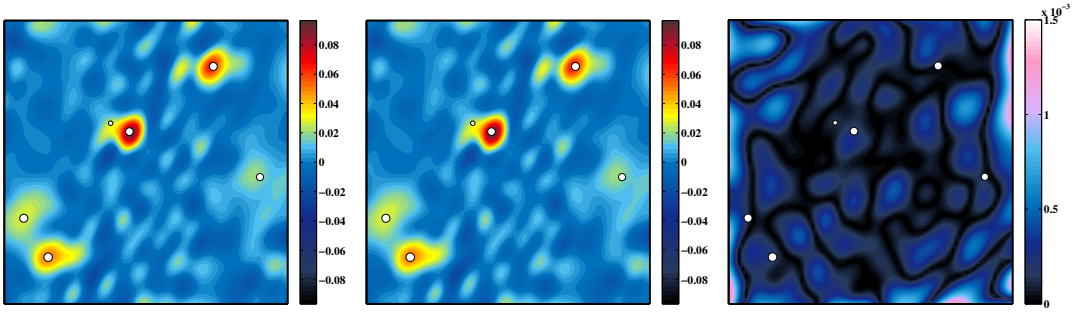
Equation (2.22) page 29, shows that, for this experiment, the minimum number of iterations needed to achieve an accuracy comparable to the Gram-Schmidt estimate with the gradient descent algorithm is 10,890. In contrast, we got the Gram-Schmidt estimate in one shot.

1.3 Statistical Properties of the Least Squares Estimate

The normalization step 10 in algorithm 4 is nonlinear for the input vectors $\{\boldsymbol{\beta}_i\}_{i=1,\dots,J}$. Hence, we cannot hope to represent $B_{\perp}$ in terms of a linear transformation of B. However, all the steps of algorithm 4 are linear for the visibilities, so that we can write

$$\boldsymbol{V}_{\perp} = \mathcal{G}\boldsymbol{V},$$

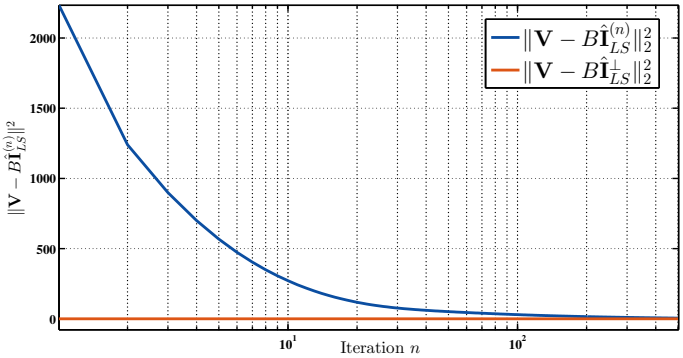
with $\mathcal{G} \in \mathbb{C}^{j \times J}$ an appropriate linear operator. In the specific case where the rows



(a) Gram-Schmidt least squares estimate obtained using algorithm 4 and eq. (4.12).

(b) Least squares estimate obtained after 500 iterations of the gradient descent algorithm.

(c) Absolute difference between the two sky estimates (different colormap).



(d) The accuracy of the sky estimate obtained with the gradient descent algorithm tends to the accuracy of the Gram-Schmidt least-squares estimate as the number of iterations grows to infinity.

Figure 4.2: Comparison of the least squares estimates obtained by: (a) performing the orthogonalization algorithm 4 and using eq. (4.12); **(b)** by using the gradient descent algorithm to minimize $\Phi(\boldsymbol{I}) = \|\boldsymbol{V} - B\boldsymbol{I}\|_2^2$. We observe that after letting run the gradient descent for a sufficient number of iterations, the two estimates appear visually very similar, even though the Gram-Schmidt least squares estimate is still more accurate in terms of minimizing the objective function $\Phi(\boldsymbol{I})$.

of B are linearly independent (and thus $j = J$), $\mathcal{G} \in \mathbb{C}^{J\times J}$ can easily be shown to be given by

$$\mathcal{G} = \prod_{i=1}^{J} G_i,$$

where $G_i \in \mathbb{C}^{J\times J}$ and

$$G_i = \begin{pmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & & & \vdots \\ \vdots & \ddots & 1 & \ddots & & & \vdots \\ -\frac{\langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_1^{\perp} \rangle}{\|\tilde{\boldsymbol{\beta}}_i\|_2} & \cdots & -\frac{\langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_{i-1}^{\perp} \rangle}{\|\tilde{\boldsymbol{\beta}}_i\|_2} & \frac{1}{\|\tilde{\boldsymbol{\beta}}_i\|_2} & \ddots & & \vdots \\ 0 & \cdots & \cdots & 0 & 1 & \ddots & \vdots \\ \vdots & & & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 \end{pmatrix} \leftarrow i\text{th row} . \tag{4.13}$$

This is particularly interesting for the purpose of assessing statistical properties of the least squares estimate. Indeed, in practice the visibilities are estimated from samples of the stations beamformed outputs (see section 3.1 page 36). Hence, instead of working with the ideal set of visibilities $\boldsymbol{V}$, we work with a realization of its estimate $\hat{\boldsymbol{V}}$, which is a random vector. As a linear transformation of a random vector (if computed with eq. (4.12)), the least squares estimate $\hat{\boldsymbol{I}}_{LS}$ will therefore be itself a random vector. Useful statistics on the sky estimate can then be derived from our knowledge of the random vector $\hat{\boldsymbol{V}}$.

Consider for example the case where $\hat{\boldsymbol{V}} = \text{vec}(\hat{\Sigma})$ with $\hat{\Sigma}$ given by the classical maximum-likelihood estimate of the covariance matrix (see section 3.1 page 36)

$$\hat{\Sigma} = \frac{1}{N_s} YY^H = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{y}(t_i)\boldsymbol{y}(t_i)^H \in \mathbb{C}^{M\times M}.$$

Then, as $\boldsymbol{y}(t)$ is assumed to be a Gaussian random vector, we know [13] the exact distribution of $\hat{\Sigma}$:

$$N_s\hat{\Sigma} \sim \mathcal{W}(N_s, \Sigma, 0),$$

where $\mathcal{W}$ denotes the **Wishart distribution** with parameters N_s (number of samples), Σ (covariance matrix containing the ideal visibilities) and 0 (mean of $\boldsymbol{y}(t)$). We can leverage this knowledge in order to built relevant statistics on the least squares sky estimate.

Variance of the Least Squares Estimate

We can show [16, 23] that the covariance matrix of $\hat{\boldsymbol{V}} = \text{vec}(\hat{\Sigma})$ is given by

$$\text{Var}(\hat{\boldsymbol{V}}) = \frac{1}{N_s}\Sigma^* \otimes \Sigma \in \mathbb{C}^{J\times J}.$$

Hence, the covariance matrix of the least squares estimate can be deduced by noting that eq. (4.12) can written as a linear transformation of $\hat{\boldsymbol{V}}$

$$\hat{\boldsymbol{I}}_{LS} = B_\perp^H \mathcal{G}\hat{\boldsymbol{V}}.$$

Thus, the variance of the least squares estimate is given by

$$\text{Var}(\hat{\boldsymbol{I}}_{LS}) = B_\perp^H \mathcal{G}\text{Var}(\hat{\boldsymbol{V}})\mathcal{G}^H B_\perp = \frac{1}{N_s} B_\perp^H \mathcal{G}\left(\Sigma^* \otimes \Sigma\right)\mathcal{G}^H B_\perp \in \mathbb{C}^{N^2\times N^2}. \tag{4.14}$$

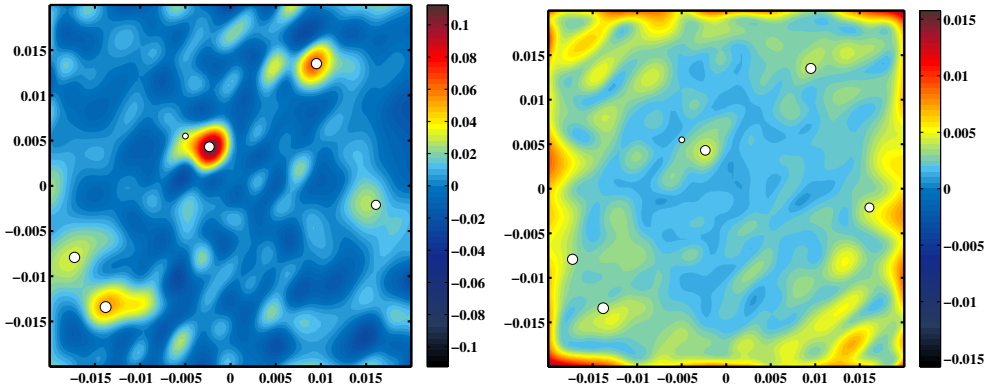
In practice, one can estimate the variance of the dirty image by replacing Σ by its estimate $\hat{\Sigma}$ in eq. (4.14). Selected second order moments of the Gram-Schmidt estimate are available in fig. 4.3.

Confidence Intervals and Significant Image

We can also invoke asymptotic arguments in order to construct confidence intervals on the sky image. Indeed, under general conditions, it is possible to show that the following asymptotic distribution holds [13]

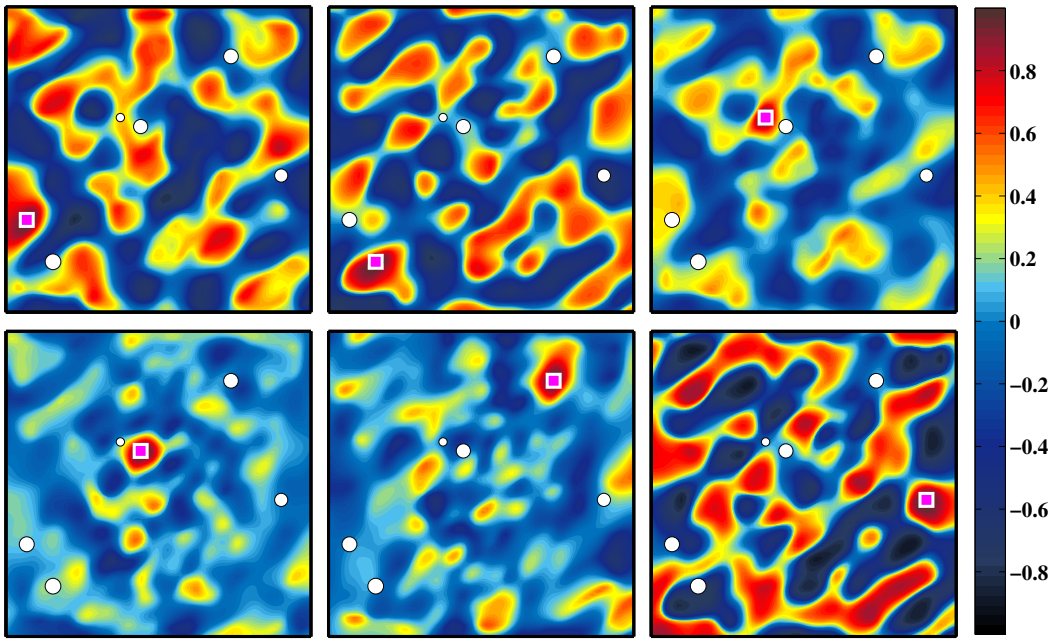
$$\sqrt{N_s}\left(\text{vec}(\hat{\Sigma}) - \text{vec}(\Sigma)\right) \overset{N_s\to\infty}{\sim} \mathcal{N}_{J^2}(0, \Sigma^* \otimes \Sigma).$$

Therefore we can deduce that each pixel of $\hat{\boldsymbol{I}}_{LS} = \left(\hat{I}_{LS}(\boldsymbol{r}_1), \ldots, \hat{I}_{LS}(\boldsymbol{r}_{N^2})\right)^T \in \mathbb{R}^{N^2}$



(a) Gram-Schmidt least squares estimate obtained using algorithm 4 and eq. (4.12).

(b) Variance of the individual pixels in the Gram-Schmidt least squares estimate. This image corresponds to the diagonal of the covariance matrix $\text{Var}(\hat{\boldsymbol{I}}_{LS})$.



(c) Correlation structure existing between selected pixels (marked with a pink square with white borders) and the rest of the sky estimate. Each image correspond to the ith row (or column) of the correlation matrix $\text{Corr}(\hat{\boldsymbol{I}}_{LS}) = \text{diag}(\text{Var}(\hat{\boldsymbol{I}}_{LS}))^{-1/2}\text{Var}(\hat{\boldsymbol{I}}_{LS})\text{diag}(\text{Var}(\hat{\boldsymbol{I}}_{LS}))^{-1/2}$. For example, if $\boldsymbol{r}_i$ denotes the position of the marked pixel in the top left picture, then the corresponding correlation image is given by $\hat{\boldsymbol{I}}^{(i)}_{corr}(\boldsymbol{r}_k) = \left(\text{Corr}(\hat{\boldsymbol{I}}_{LS})\right)_{i,:} = \text{corr}\left(\hat{I}_{LS}(\boldsymbol{r}_i), \hat{I}_{LS}(\boldsymbol{r}_k)\right)$, $k = 1, \ldots, N^2$.

Figure 4.3: **Second order moments of the Gram-Schmidt least squares estimate.**

has the following limiting distribution

$$\left(\hat{I}_{LS}(\boldsymbol{r}_i) - \mathbb{E}[\hat{I}_{LS}(\boldsymbol{r}_i)]\right) \overset{N_s \to \infty}{\sim} \mathcal{N}_1(0, (\text{Var}(\hat{\boldsymbol{I}}_{LS}))_{i,i}), \quad \forall i = 1, \ldots, N^2,$$

where $(\text{Var}(\hat{\boldsymbol{I}}_{LS}))_{i,i}$ denotes the ith diagonal term of the covariance matrix $\text{Var}(\hat{\boldsymbol{I}}_{LS}) \in \mathbb{C}^{N^2 \times N^2}$ given in eq. (4.14). Assuming that the number of samples N_s is sufficiently large so that this distribution approximately holds, we can then construct confidence intervals with level $0 \leq \alpha \leq 1$ on the true value of the intensity of the ith pixel of

the least squares estimate:

$$IC_{1-\alpha}^{(i)} = \left[\hat{I}_{LS}(\boldsymbol{r}_i) - \Phi_{1-\alpha}\sqrt{(\mathrm{Var}(\hat{\boldsymbol{I}}_{LS}))_{i,i}}\,;\, \hat{I}_{LS}(\boldsymbol{r}_i) + \Phi_{1-\alpha}\sqrt{(\mathrm{Var}(\hat{\boldsymbol{I}}_{LS}))_{i,i}}\right], \quad (4.15)$$

with $\Phi_{1-\alpha}$ the $1-\alpha$ quantile of the standard Gaussian distribution. Then,

$$\mathbb{P}\left[\hat{I}_{LS}(\boldsymbol{r}_i) \in IC_{1-\alpha}^{(i)}\right] = 1-\alpha, \quad \forall i = 1,\dots,N^2.$$

For all the pixels to be simultaneously contained in their respective marginal confidence interval with probability at least 0.95, we must choose $\alpha = 0.05/N^2$. Indeed,

$$\begin{aligned}
&\mathbb{P}\left(\bigcap_{i=1}^{N^2}\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \in IC_{1-\alpha}^{(i)}\right\}\right) = 0.95,\\
\Leftrightarrow&\mathbb{P}\left(\bigcup_{i=1}^{N^2}\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \notin IC_{1-\alpha}^{(i)}\right\}\right) = 0.05,
\end{aligned}$$

and

$$\mathbb{P}\left(\bigcup_{i=1}^{N^2}\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \notin IC_{1-\alpha}^{(i)}\right\}\right) \leq \sum_{i=1}^{N^2}\mathbb{P}\left(\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \notin IC_{1-\alpha}^{(i)}\right\}\right) = N^2\alpha.$$

Hence, if $\alpha = 0.05/N^2$, we have

$$\begin{aligned}
&\mathbb{P}\left(\bigcup_{i=1}^{N^2}\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \notin IC_{1-\alpha}^{(i)}\right\}\right) \leq 0.05,\\
\Leftrightarrow&\mathbb{P}\left(\bigcap_{i=1}^{N^2}\left\{\hat{I}_{LS}(\boldsymbol{r}_i) \in IC_{1-\alpha}^{(i)}\right\}\right) \geq 0.95.
\end{aligned}$$

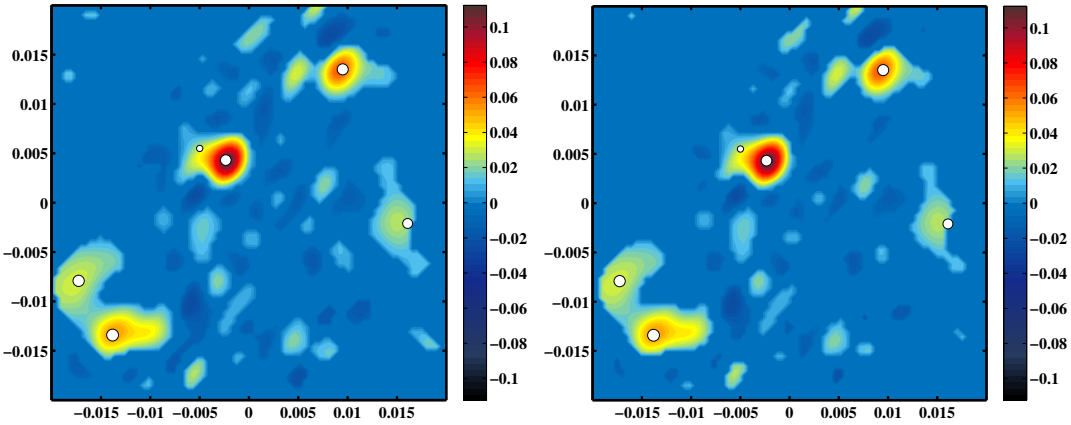
Confidence intervals obtained in this way are called the **Bonferroni intervals** [8] (see fig. 4.4). We can use these confidence intervals to perform a statistical test on each pixel $i = 1,\dots,N^2$,

$$H_0 : \hat{I}_{LS}(\boldsymbol{r}_i) = 0, \quad \text{v.s.} \quad H_1 : \hat{I}_{LS}(\boldsymbol{r}_i) \neq 0.$$

The hypothesis H_0 is accepted if $0 \in IC_{1-\alpha}^{(i)}$ and rejected otherwise. The **significant image** is then given by, $\forall i = 1,\dots,N^2$,

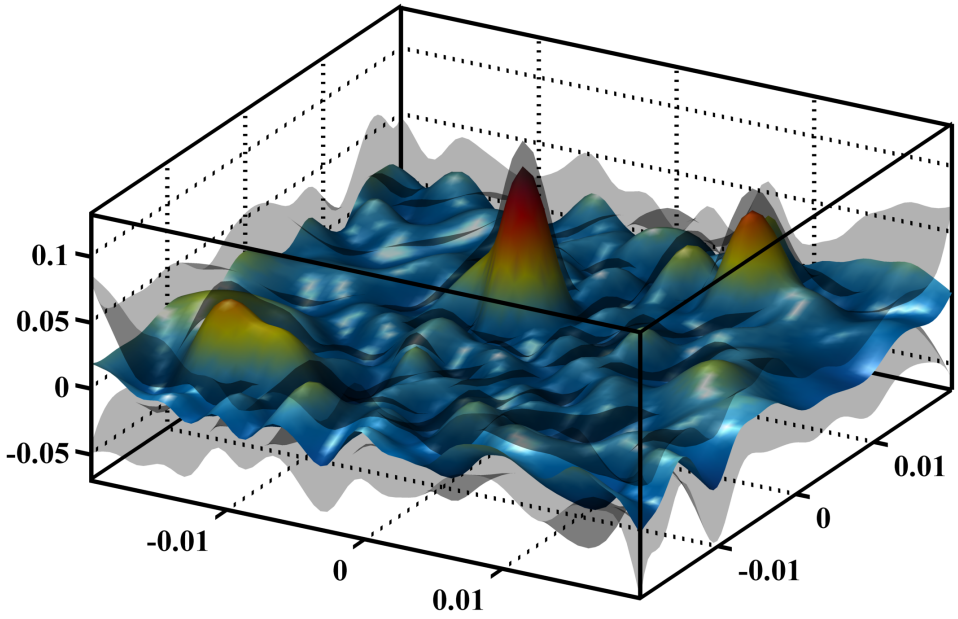
$$\hat{I}_{signif}^{\alpha}(\boldsymbol{r}_i) := \begin{cases} 0, & \text{if} \quad 0 \in IC_{1-\alpha}^{(i)},\\ \hat{I}_{LS}(\boldsymbol{r}_i), & \text{otherwise.}\end{cases}$$

Examples of significant images for the Gram-Schmidt estimate from fig. 4.3 (a) are provided on fig. 4.4. We observe that even with a global significance level of 0.01%, many false positives subsist in the least squares estimate (pixels that are significantly different from zero and yet do not correspond to an actual source).



(a) Significant Image for $\alpha = 0.05/N^2$ (pixels for which 0 does not belong to the 95% Bonferonni intervals).

(b) Significant Image for $\alpha = 0.001/N^2$ (pixels for which 0 does not belong to the 99.99% Bonferonni intervals).



(c) Gram-Schmidt least squares estimate and its 95% Bonferonni intervals.

Figure 4.4: **Examples of significant images and Bonferroni intervals.**

2 Solving for the LASSO using the Least Squares Estimate

In section 1, we demonstrated the merits of the orthogonal case in terms of accelerating the gradient descent algorithm and achieving higher accuracy in the least squares estimate. To benefit from those convenient properties, we have described a procedure based on the Gram-Schmidt algorithm, to orthogonalize the rows of B and accordingly modify the set of visibilities, so that the geometrical interpretability of the data was retained in the orthogonalized basis.

Because the proposed procedure was linear in the input data, we could successfully measure the uncertainty in the least squares estimate and build meaningful statistics on it. Among them, we proposed the significance image, that test the global significance of the image using the Bonferonni method. We could observe that many false positives remained in the least squares estimate, even with a significance level

as low as 0.01% (see fig. 4.4). Hence, it seems that the least squares estimate has some limitations for the purpose of producing sparse estimates of the sky.

To enforce sparsity of the sky image, we can alternatively try and solve the LASSO problem (see section 3 of chapter 3):

$$\hat{\boldsymbol{I}}_{LASSO} = \text{argmin}_{\boldsymbol{I}\in\mathbb{R}^{N^2}} \|\boldsymbol{V} - B\boldsymbol{I}\|_2^2 + \lambda\|\boldsymbol{I}\|_1, \tag{4.16}$$

for some $\lambda \geq 0$. The L^1 penalty on the sky estimate guarantees a relative sparsity of the LASSO estimate (depending on the strength of λ). Unfortunately, there is a price to pay in terms of complexity, as the resulting objective function becomes non-differentiable because of this penalty term, which forbids the use of classical iterative algorithms that rely on the gradient. Hence, the LASSO problem (4.16) is usually rather time-consuming to solve for large scale problems. However, in the blessed case where B has orthogonal columns, we can directly *extract* the LASSO estimate from the least squares estimate [21].

Theorem 4.1 — ***LASSO by Shrinkage-Thresholding of the Least Squares Estimate.*** Let $\boldsymbol{V} \in \mathbb{C}^J$, $\boldsymbol{I} \in \mathbb{C}^{N^2}$ and $B \in \mathbb{C}^{J\times N^2}$ a tall matrix ($N^2 \leq J$) with orthogonal columns. Consider the linear system $\boldsymbol{V} = B\boldsymbol{I}$, and let $\hat{\boldsymbol{I}}_{LS} = B^H\boldsymbol{V}$ be the least squares solution. Then, the LASSO estimate $\hat{\boldsymbol{I}}_{LASSO} \in \mathbb{C}^{N^2}$, solution to eq. (4.16) is given by

$$\hat{I}^i_{LASSO} = \text{sgn}\left(\hat{I}^i_{LS}\right) \times \left(\left|\hat{I}^i_{LS}\right| - \frac{\lambda}{2}\right)^+, \quad \forall i = 1, \dots, N^2, \tag{4.17}$$

where $\hat{I}^i_{LASSO}, \hat{I}^i_{LS} \in \mathbb{R}$ denote respectively the ith component of $\hat{\boldsymbol{I}}_{LASSO}$ and $\hat{\boldsymbol{I}}_{LS}$, $\lambda \geq 0$ is the penalty parameter in eq. (4.16), and

$$x^+ := \max(x, 0), \quad \forall x \in \mathbb{R}.$$

■ **Proof 2.1** The objective function to minimize is

$$\Phi_{LASSO}(\boldsymbol{I}) = \boldsymbol{V}^H\boldsymbol{V} + \boldsymbol{I}^T B^H B\boldsymbol{I} - 2\boldsymbol{I}^T B^H\boldsymbol{V} + \lambda\sum_{i=1}^{N^2}|I^i|, \tag{4.18}$$

where $I^i \in \mathbb{R}$ denotes the ith pixel of $\boldsymbol{I} \in \mathbb{R}^{N^2}$. First, note that the term $\boldsymbol{V}^H\boldsymbol{V}$ in eq. (4.18) can be dropped, as it does not depend on $\boldsymbol{I}$. Then, we observe that, when B has orthogonal columns, $B^HB = Id$ and $B^H\boldsymbol{V} = \hat{\boldsymbol{I}}_{LS}$. Hence, minimizing eq. (4.18) is equivalent to minimizing the quantity

$$\begin{aligned} &\|\boldsymbol{I}\|_2^2 - 2\boldsymbol{I}^H\hat{\boldsymbol{I}}_{LS} + \lambda\sum_{i=1}^{N^2}|I^i|, \\ \Leftrightarrow \quad &\sum_{i=1}^{N^2}\left(\left(I^i\right)^2 - 2\hat{I}^i_{LS}I^i + \lambda|I^i|\right). \end{aligned} \tag{4.19}$$

Now, eq. (4.19) is a sum of N^2 objective functions, with no variable in common. Hence, minimizing eq. (4.19) is equivalent to minimizing each of the terms of the summation:

$$\left(I^i\right)^2 - 2\hat{I}^i_{LS}I^i + \lambda|I^i|, \quad i \in \{1, \dots, N^2\}. \tag{4.20}$$

For a given pixel i, let $\hat{I}^i_{LASSO}$ be the minimizer of (4.20) (which exists as (4.20) is a convex

function). Then two possibilities result:

- **Case 1:** $\hat{I}^i_{LS} \geq 0$. Then, we have necessarily $\hat{I}^i_{LASSO} \geq 0$ otherwise we could always flip the sign of $\hat{I}^i_{LASSO}$ and get a smaller value for (4.20). Hence, there exists a neighborhood of $\hat{I}^i_{LASSO}$ in which the objective function for pixel i is given by

  $$\left(I^i\right)^2 - 2\hat{I}^i_{LS} I^i + \lambda I^i.$$

  Differentiating and equating to zero yields

  $$\hat{I}^i_{LASSO} = \hat{I}^i_{LS} - \frac{\lambda}{2} = \operatorname{sgn}\left(\hat{I}^i_{LS}\right)\left(\left|\hat{I}^i_{LS}\right| - \frac{\lambda}{2}\right)^+,$$

  as we have $\hat{I}^i_{LS} \geq 0$ and $\hat{I}^i_{LASSO} \geq 0$.
- **Case 2:** $\hat{I}^i_{LS} \leq 0$. Then, one more time, necessarily $\hat{I}^i_{LASSO} \geq 0$ otherwise we could always flip the sign of $\hat{I}^i_{LASSO}$ and get a smaller value for (4.20). Hence, there exists a neighborhood of $\hat{I}^i_{LASSO}$ in which the objective function for pixel i is given by

  $$\left(I^i\right)^2 - 2\hat{I}^i_{LS} I^i - \lambda I^i.$$

  Differentiating and equating to zero yields

  $$\hat{I}^i_{LASSO} = \hat{I}^i_{LS} + \frac{\lambda}{2} = \operatorname{sgn}\left(\hat{I}^i_{LS}\right)\left(\left|\hat{I}^i_{LS}\right| - \frac{\lambda}{2}\right)^+,$$

  as $\hat{I}^i_{LS} \leq 0$ and $\hat{I}^i_{LASSO} \leq 0$.

In conclusion, in both cases,

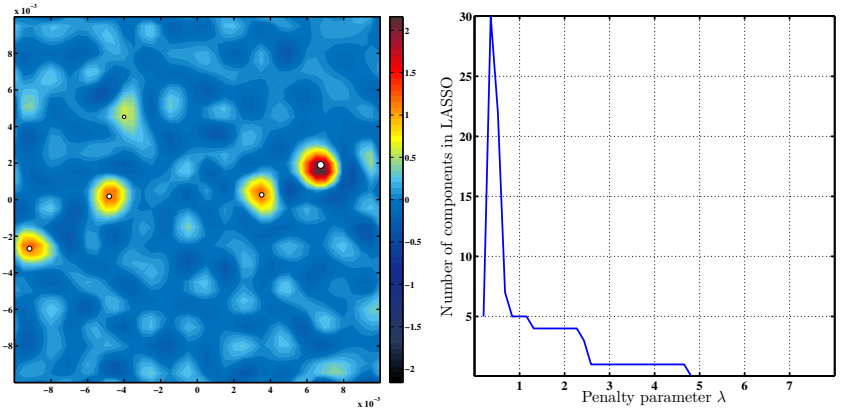
$$\hat{I}^i_{LASSO} = \operatorname{sgn}\left(\hat{I}^i_{LS}\right)\left(\left|\hat{I}^i_{LS}\right| - \frac{\lambda}{2}\right)^+, \quad \forall i = 1, \ldots, N^2,$$

which concludes the proof. ∎

Remark 2.1 *The parameter* λ *belongs to* $\left[0, 2 \cdot max_i\left\{\left|\hat{I}^i_{LS}\right|\right\}\right]$. *For* $\lambda = 0$, *the LASSO and least squares estimate coincide. For* $\lambda = 2 \cdot max_i\left\{\left|\hat{I}^i_{LS}\right|\right\}$, *the LASSO estimate is trivial:* $\hat{I}^i_{LASSO} = 0,\ \forall i = 1, \ldots, N^2$.

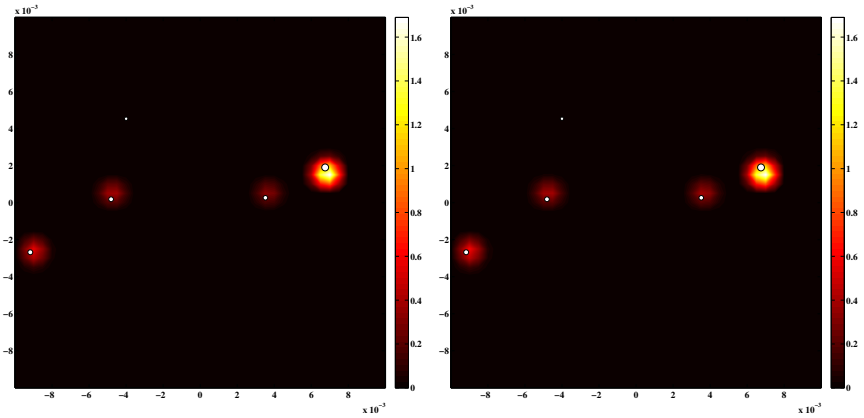
Hence, if B is a tall matrix and we use algorithm 4 to form $B_\perp$ and $\boldsymbol{V}_\perp$, then $B_\perp$ will consist of a square matrix with orthogonal rows (by construction) and hence orthogonal columns. Then, we can form the Gram-Schmidt least squares estimate using eq. (4.12) page 50, and extract the LASSO estimate from it by applying theorem 4.1 (see for example fig. 4.5). This is remarkable, as it provides us with a closed formula for the LASSO estimate, that can be computed at virtually no cost from the least squares estimate (for which we happen to have an efficient *direct* imager, based on algorithm 4). However, this computational convenience comes at a price in terms of the achievable resolution of the LASSO estimate, as B is required to be a tall matrix, and hence the number of pixels must be at most as big as the number of baselines (or more precisely the number of linearly independent vectors on which we are projecting the sky image).

The SKA is intended to comprise thousands of stations [7] (and hence potentially millions of baselines). Thus, the maximum achievable resolution would still be reasonable enough so that theorem 4.1 could be used to compute the LASSO estimate. However, for smaller phased arrays, this constraint on the resolution is more of an issue, that we would like to overcome without totally losing the advantages of the previously described strategy.



(a) Gram-Schmidt estimate obtained with 24 LOFAR core stations.

(b) Number of components in the LASSO estimate for various values of λ. Significant breaks in the slope can provide hints on how to choose λ.



(c) Approximation of the LASSO estimate using theorem 4.1, for $\lambda = 1.314$.

(d) LASSO estimate after 10 iterations of the FISTA algorithm, for $\lambda = 1.314$.

Figure 4.5: Exact computation of the LASSO estimate from the Gram-Schmidt estimate. Here the field of view is very small and the resolution is kept voluntary low, so that $N^2 = 400 < J = 553$ and we can invoke theorem 4.1 to compute the LASSO estimate. We observe that the LASSO estimate obtained this way is identical to the one computed with the FISTA algorithm (up to numerical inaccuracies).

Let see then what happens when B is not a tall matrix anymore (more columns than rows), but still has orthogonal rows. Then, minimizing eq. (4.16) is equivalent to minimizing the following quantity

$$\begin{aligned}\tilde{\Phi}(\boldsymbol{I}) &= \boldsymbol{I}^T B^H B \boldsymbol{I} - 2\boldsymbol{I}^T B^H \boldsymbol{V} + \lambda \sum_{i=1}^{N^2} |I^i|, \\ &= \boldsymbol{I}^T (B^H B)\boldsymbol{I} - \boldsymbol{I}^T \hat{\boldsymbol{I}}_{LS} + \lambda \sum_{i=1}^{N^2} |I^i|. \end{aligned} \tag{4.21}$$

As the columns of B are not orthogonal, the term $B^H B$ is not the identity, and hence we cannot decompose eq. (4.21) as a sum of independent objective functions as in proof 2.1. However, if $B^H B$ happened to be *almost* orthogonal, then we could

in some sense extend the approach of proof 2.1 to obtain, simply, an approximation of the LASSO estimate from the least squares estimate. Indeed, under the approximation $B^H B \simeq \text{diag}\left(B^H B\right)$, we can approximate eq. (4.21) by

$$\begin{aligned}\tilde{\Phi}(\boldsymbol{I}) &\simeq \boldsymbol{I}^T \text{diag}(B^H B)\boldsymbol{I} - \boldsymbol{I}^T \hat{\boldsymbol{I}}_{LS} + \lambda \sum_{i=1}^{N^2} |I^i|, \\ &= \sum_{i=1}^{N^2} \mu_i \left(I^i\right)^2 - 2\hat{I}^i_{LS} I^i + \lambda |I^i|, \end{aligned} \tag{4.22}$$

where $\mu_i \geq 0$ is the ith diagonal element of $B^H B$. Notice that eq. (4.22) is very similar to eq. (4.19), in the sense that the objective function can again be decomposed into a sum of N^2 objective functions, that do not share any variable in common. Hence, we can apply the same methodology as in proof 2.1, to get

Corollary 2.1 *Let $\boldsymbol{V} \in \mathbb{C}^J$, $\boldsymbol{I} \in \mathbb{C}^{N^2}$ and $B \in \mathbb{C}^{J \times N^2}$ a wide matrix ($N^2 \geq J$) with orthogonal rows. Further assume that B is such that $B^H B \simeq diag(B^H B)$, and $diag(B^H B)$ is positive definite.*

Consider then the linear system $\boldsymbol{V} = B\boldsymbol{I}$, and let $\hat{\boldsymbol{I}}_{LS} = B^H \boldsymbol{V}$ be the least squares solution to this system. Then, the LASSO estimate $\hat{\boldsymbol{I}}_{LASSO} \in \mathbb{C}^{N^2}$, solution to eq. (4.16) can be approximated by

$$\hat{I}^i_{LASSO} \simeq \frac{sgn\left(\hat{I}^i_{LS}\right)}{\mu_i} \left(\left|\hat{I}^i_{LS}\right| - \frac{\lambda}{2}\right)^+, \quad \forall i = 1, \ldots, N^2, \tag{4.23}$$

where $\hat{I}^i_{LASSO}, \hat{I}^i_{LS} \in \mathbb{R}$ denote respectively the ith component of $\hat{\boldsymbol{I}}_{LASSO}$ and $\hat{\boldsymbol{I}}_{LS}$, $\lambda \geq 0$ is the penalty parameter in eq. (4.16), and $\mu_i > 0$ is the ith diagonal element of the matrix $B^H B$.

Remark 2.2 *The quality of the approximation eq. (4.23) will depend on how close the matrix $B^H B$ is from being diagonal.*

Assume then that B is a wide matrix with associated visibilities $\boldsymbol{V}$, that we provide as inputs to algorithm 4 to form $B_\perp$ and $\boldsymbol{V}_\perp$. Then, $B_\perp$ has (by construction) orthogonal rows, and we would like to be able to use corollary 2.1 in order to approximate the LASSO estimate from the least squares estimate. To this end, we need to assess how close is $B_\perp^H B_\perp$ to being diagonal.

First, it is important to note that a priori there is no guarantee that $B_\perp^H B_\perp$ will be close to diagonal for any wide matrix $B_\perp$ with orthogonal rows. Hence, when it holds it is necessarily specific to the scenario investigated. More specifically, the validity of the approximation eq. (4.23) depends on the *layout* of the radio-telescope under consideration.

To make this assertion precise, note that, when using eq. (4.12) as our least squares imager, the ith column of $B_\perp^H B_\perp$ is nothing else but the **impulse response** of the instrument to a single source positioned at pixel i and with unit magnitude. Indeed, if $\boldsymbol{\delta}_i \in \mathbb{R}^{N^2}$ is the ith element of the canonical basis (or equivalently the **impulse image** with only one bright pixel of intensity one at position $\boldsymbol{r}_i$), then we have

$$B_\perp^H B_\perp \boldsymbol{\delta}_i = \left(B_\perp^H B_\perp\right)_{:,i}, = \sum_{j=1}^{J} \beta_j^*(\boldsymbol{r}_i)\boldsymbol{\beta}_j, = \sum_{j=1}^{J} \langle \boldsymbol{\beta}_j, \boldsymbol{\delta}_i \rangle \boldsymbol{\beta}_j = \boldsymbol{PSF}_i,$$

where $\boldsymbol{PSF}_i$ is usually called the **point spread function** of pixel i. We see that $\boldsymbol{PSF}_i$ is indeed the response of the instrument for an impulse at pixel i, and that it corresponds to the ith column of $B_\perp^H B_\perp$. Hence, understanding the structure of $B_\perp^H B_\perp$ is equivalent to understanding the properties of the individual point spread functions $\boldsymbol{PSF}_i$.

The nice thing is that, in practice, telescope layouts are designed so that these functions are well-behaved and with desirable properties [5, 23]. Usually, the point spread function $\boldsymbol{PSF}_i$ of pixel i consists in an extended bell-shaped function, maximal at pixel i, and decaying quickly as we move away from this pixel. Because of the tool's finite sensitivity, the support of this function is virtually finite (see fig. 4.6). Then, the spread of this function into the image plane will determine our ability (or inability) to recover point sources: the smaller the support, the better the recovery. If the layout of the telescope is optimized to maximize the frequency coverage in the Fourier domain, then we can expect the point spread functions to have most of its energy contained in a relatively small support.

Translated to the structure of $B_\perp^H B_\perp$, it means that the each column of $B_\perp^H B_\perp$ will be sparse, with a maximal positive element on the diagonal ($\boldsymbol{PSF}_i$ is maximal at pixel i) and (hopefully) only a few non-null off-diagonal elements (corresponding to the close neighbors of pixel i that belong to the support of $\boldsymbol{PSF}_i$). Hence, if the effective support of the point spread functions $\boldsymbol{PSF}_i$ are on average small enough with respect to the field of view, then we can reasonably neglect the few off-diagonal terms and approximate $B_\perp^H B_\perp$ by its diagonal elements.

In fig. 4.7 and fig. 4.8, we present the LASSO estimate approximated with corollary 2.1 for two different telescope layouts, and compare it to the LASSO estimate computed with the FISTA algorithm applied to eq. (4.16). In fig. 4.7, only 12 stations were used for the sky recovery. The corresponding average point spread function has a non-negligible effective support of about 5% of the field of view. Hence, in that scenario, the approximate LASSO estimate is not very accurate, as $B_\perp^H B_\perp$ is far from diagonal. In fig. 4.7, we increased the number of stations to 24. With this new layout, the point spread functions has a much smaller support (on average, around 1.8% of the total field of view). The accuracy of the LASSO estimate obtained with corollary 2.1 is consequently better, comparing much more with the output of the FISTA algorithm, which still provides a slightly more precise estimate (nearby sources are better isolated from one another).

Hence, when the support of the point spread functions is small enough, corollary 2.1 seems to provide, at a *very low* computational cost, a good enough approximation to the LASSO estimate. This computational convenience can be leveraged in the choice of λ. For example, we can compute the estimate for various values of λ, count the number of connected components in the resulting image and plot it as a function of λ. Breaks in the slope of this graph can then suggest interesting values of λ to investigate (see for example fig. 4.5 (b)). Of course, this is not the only possible criterion.

Remark 2.3 — *Gram-Schmidt and CLEAN.* *We have seen in section 4 page 25 that CLEAN was also used extensively in practice for the purpose of creating sparser estimates of the sky image. Thus, we tried to understand if CLEAN would benefit as well from the orthog-*

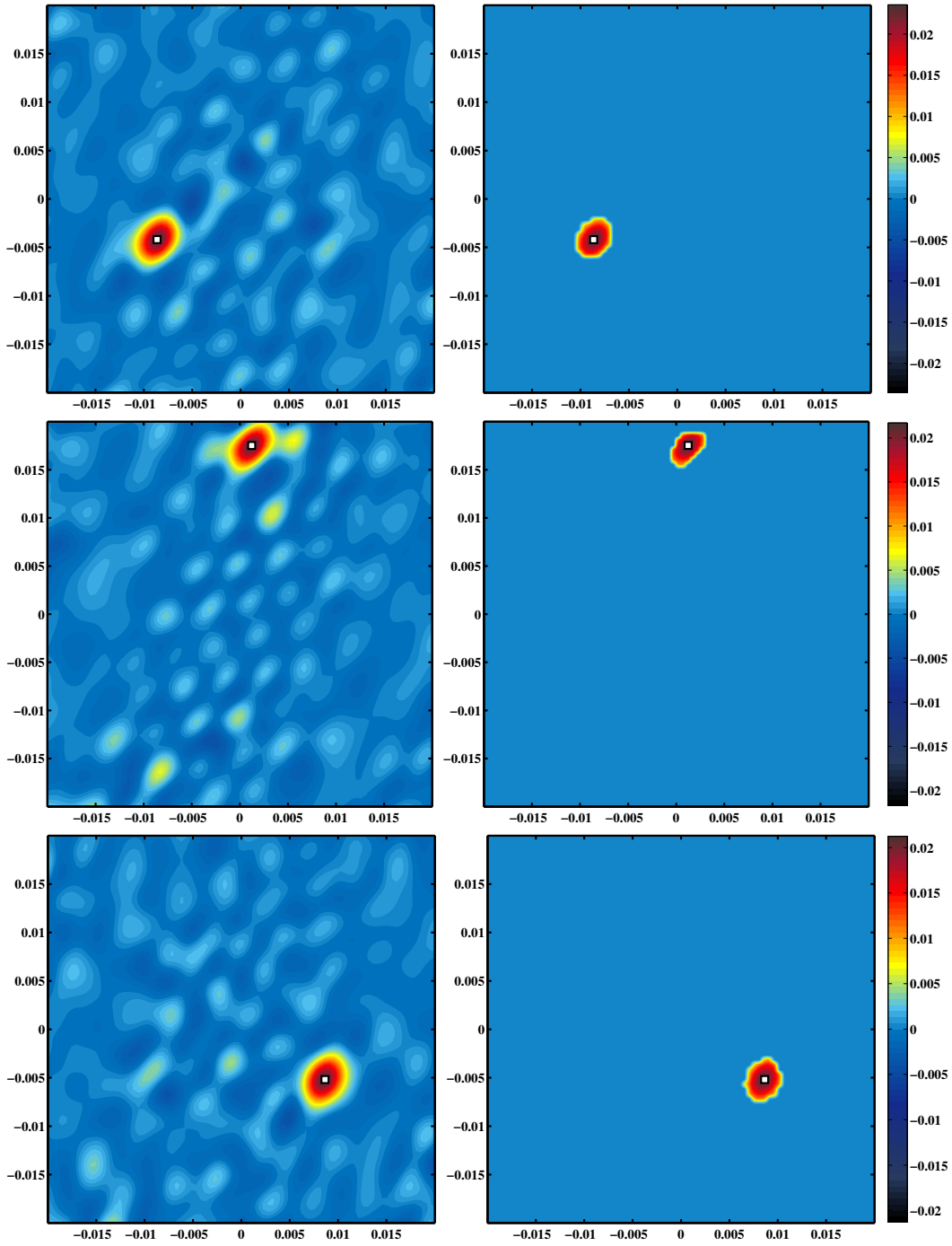
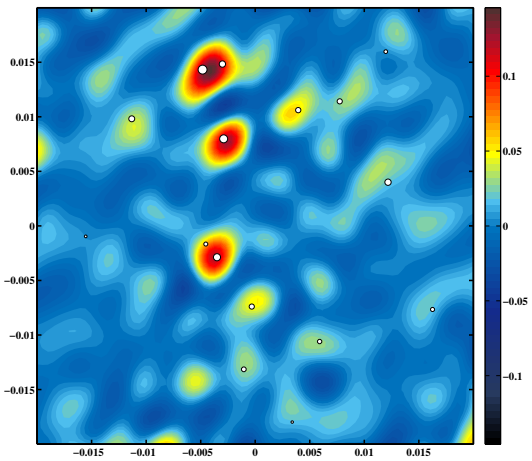
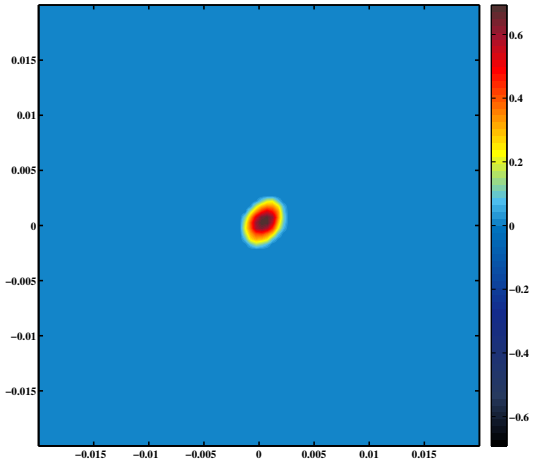


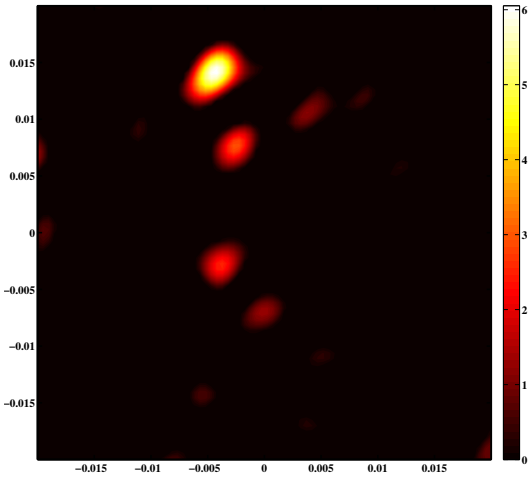
Figure 4.6: Examples of point spread functions for different positions in the sky (marked by a white square). Left: theoretical point spread function (ith column of $B_\perp^H B_\perp$), Right: significant point spread function (pixels of the point spread function that are significantly different from zero, for the given experimental conditions). We observe that, for given experimental conditions (noise level, number of samples), the point spread function has effectively a finite support. If this support is small enough with respect to the region of interest, we can safely neglect the off-diagonal terms of $B_\perp^H B_\perp$.



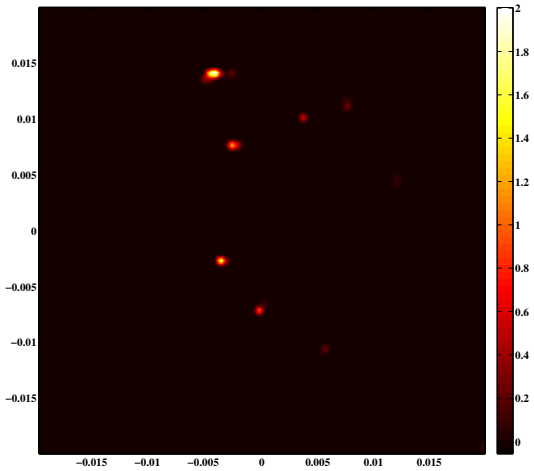
(a) Gram-Schmidt estimate obtained with 12 LOFAR core stations.



(b) Average point spread function. The support is $\sim 5\%$ of the field of view.

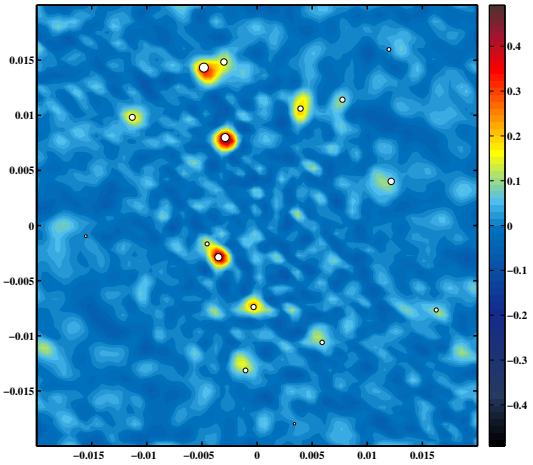


(c) Approximation of the LASSO estimate using corollary 2.1, for $\lambda = 0.08$.

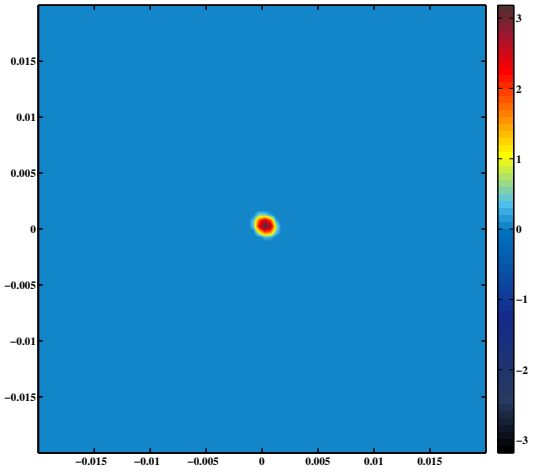


(d) LASSO estimate after 408 iterations of the FISTA algorithm, for $\lambda = 0.08$.

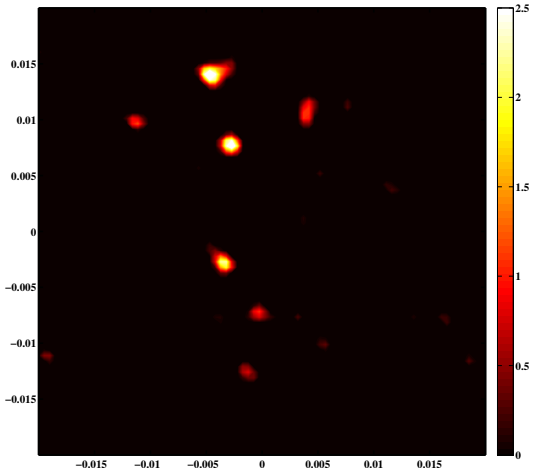
Figure 4.7: Extracting the LASSO estimate from the Gram-Schmidt estimate: comparison with the FISTA algorithm. In this example, we are using only 12 LOFAR core stations. We have $J = 132 < N^2 = 6724$, so we cannot use directly theorem 4.1. However, we can use corollary 2.1 to approximate the LASSO estimate. For this specific experiment, the approximated LASSO estimate has poor quality with respect to the LASSO estimate computed with the FISTA algorithm: the absolute intensities do not correspond, and the sources are poorly resolved (huge blobs). Moreover, some sources detected by the FISTA algorithm are not detected by the approximated LASSO estimate, and some false positives pollute the latter. This is because the point spread function has a too wide support for the given experimental conditions, and hence $B_\perp^H B_\perp$ cannot be approximated well enough by its diagonal. Indeed, for a noise level of $\sigma_n^2 = 500$, we observe that point spread function has an average support of size $\sim 5\%$ of the field of view (too wide).



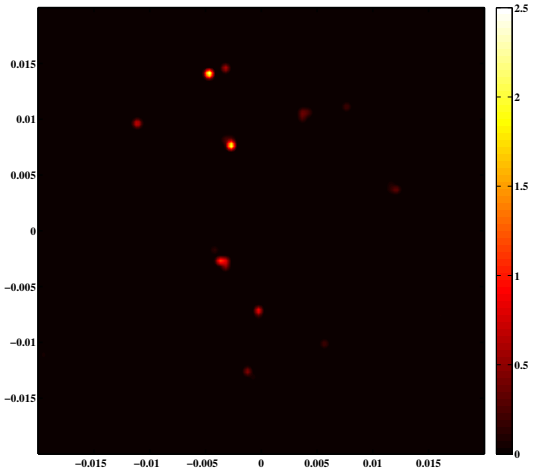
(a) Gram-Schmidt estimate obtained with 24 LOFAR core stations.



(b) Average point spread function. The support is $\sim 1.8\%$ of the field of view.



(c) Approximation of the LASSO estimate using corollary 2.1, for $\lambda = 0.15$.



(d) LASSO estimate after 166 iterations of the FISTA algorithm, for $\lambda = 0.15$.

Figure 4.8: Extracting the LASSO estimate from the Gram-Schmidt estimate: comparison with the FISTA algorithm. In this example, we are using only 24 LOFAR core stations. We have $J = 552 < N^2 = 6724$, so we cannot use directly theorem 4.1. However, we can use corollary 2.1 to approximate the LASSO estimate. This time, the quality of both estimates are comparable: the absolute intensities correspond, and the sources are well resolved. Of course, the LASSO estimate obtained with the FISTA algorithm is still more precise, but it is also much more costly to compute! The relatively good accuracy of the approximated LASSO estimate in this example can be explained by the increased frequency coverage of the telescope, that results in a smaller support for the point spread function ($\sim 1.8\%$ of the field of view). Hence, $B_\perp^H B_\perp$ can be better approximated by its diagonal, and corollary 2.1 provides a better approximation of the LASSO estimate.

onalization procedure. In section 4 page 25, we showed that CLEAN could essentially be interpreted as an approximate gradient descent algorithm constrained to take only canonical directions. Hence, because of the merits of the orthogonal case for the gradient descent algorithm, it would make sense, at least intuitively, that the perfectly conditioned model $B_\perp$ helps in the convergence of CLEAN. Unfortunately, a precise convergence analysis of the algorithm is made really difficult by the nonlinear nature of Ψ, as well as the extreme sensitivity of the method on the gain parameter τ. In the absence of theoretical arguments, we investigated the question experimentally, but our simulations did not show significant improvement in the convergence of CLEAN when used in conjunction with $B_\perp$.

3 Robustness

We have already identified the telescope layout as being a determinant factor for the quality of the LASSO estimate approximation provided by corollary 2.1. But this is of course not a sufficient condition, as the corollary relies on the availability of a reliable enough least squares estimate. Thus, in this section we investigate the robustness of the Gram-Schmidt least squares estimate to various parameters of the model. In particular, we propose a modification of algorithm 4, that improves the robustness of the least squares estimate to thermal noise. Then, we investigate the necessity of station calibration as for estimating antennas' gains and providing a more accurate computation of the station beamshapes. We conclude that station calibration does not improve significantly the sky estimate, and hence we can assume those gains to be unit gains when computing the beamshape. This is particularly convenient as it makes the computation of the beamshapes independent from the experimental conditions. Finally, we demonstrate the greater robustness of the Gram-Schmidt estimate to more exotic beamforming strategies proposed in the literature [14, 15] to maximize the information content within the beamformed outputs.

3.1 Improving the Stability of the Gram-Schmidt Estimate

The stability of algorithm 4 is heavily dependent on the conditioning of the input matrix B. Indeed, if the subspace spanned by the rows of B suffers from multicollinearity, it could be that one step of the Gram-Schmidt algorithm leads to numerical cancellation. This could occur when forming the quantity

$$\tilde{\boldsymbol{\beta}}_i \leftarrow \boldsymbol{\beta}_i - \sum_{k=1}^{i-1} \langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_k^\perp \rangle \boldsymbol{\beta}_k^\perp, \quad i = 1, \dots, J.$$

If the vector $\boldsymbol{\beta}_i$ is almost contained in the subspace spanned by the orthogonal family $\{\boldsymbol{\beta}_k^\perp \,|\, k = 1, \dots i-1\}$, then the $\tilde{\boldsymbol{\beta}}_i$ will have a very small norm, which might lead to numerical instabilities. In particular, a very small norm for $\tilde{\boldsymbol{\beta}}_i$ could potentially magnify a lot the noise in the measurements, resulting in a severely corrupted vector $\boldsymbol{V}_\perp$, and hence in a unreliable sky estimate. Indeed, the visibilities are modified by the algorithm according to the following equation

$$V_i^\perp \leftarrow \left(V_i - \sum_{k=1}^{i-1} \langle \boldsymbol{\beta}_i, \boldsymbol{\beta}_k^\perp \rangle V_k^\perp \right) / \|\tilde{\boldsymbol{\beta}}_i\|, \quad i = 1, \dots, J.$$

Therefore, the division by the term $\|\tilde{\boldsymbol{\beta}}_i\|$ in the above equation can severely magnify the noise in the original measurement set, and as a result affect the estimation of $V_i^\perp$

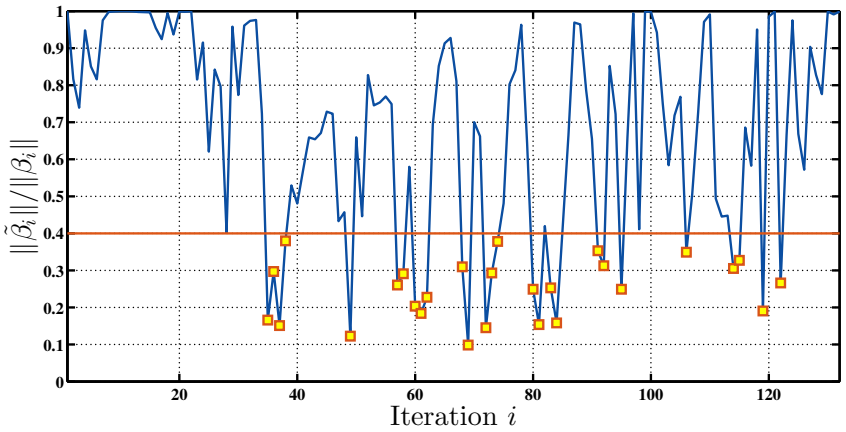


Figure 4.9: Results of the test eq. (4.24) for each iteration of algorithm 4. Discarding the vectors (marked by a yellow square) that fall below the threshold $\epsilon = 0.4$ (orange line) improves the stability of the algorithm, as it prevents the noise in the measurements to be magnified.

and all the subsequent visibilities $V_j^\perp$, $j \geq i$, whose estimation depends on $V_i^\perp$. This is particularly problematic in practice, as the visibilities have to be estimated, and hence exhibit inevitable fluctuations with respect to the true theoretical visibilities. Hence, to guarantee the stability of our algorithm with respect to those inaccuracies in the measurements, we added a check after line 7 of algorithm 4. This check compares $\|\tilde{\boldsymbol{\beta}}_i\|$ with $\|\boldsymbol{\beta}_i\|$ and if

$$\frac{\|\tilde{\boldsymbol{\beta}}_i\|}{\|\boldsymbol{\beta}_i\|} \leq \epsilon, \tag{4.24}$$

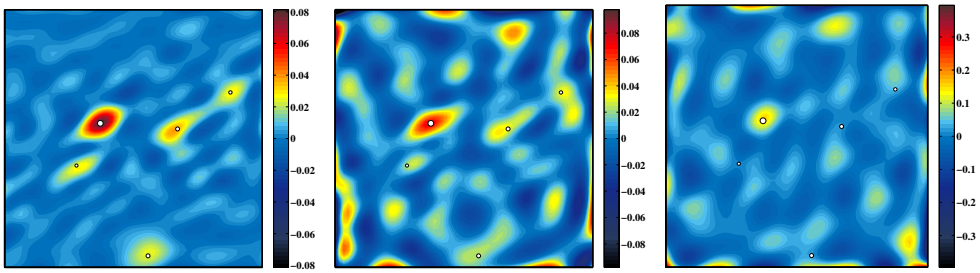
with $\epsilon \leq 1$ a given threshold, then the algorithm skips the following steps and jumps directly to the next iteration: $\boldsymbol{\beta}_i^\perp$ is not added to the orthogonal family $\{\boldsymbol{\beta}_k^\perp \mid k = 1, \dots i-1\}$, and the associated measurement $V_i^\perp$ is simply discarded (see fig. 4.9). Our investigations revealed that $\epsilon = 0.4$ was good enough for most of the cases. Figures 4.10 and 4.11 show the evolution of the Gram-Schmidt estimate as the noise grows, with and without the introduction of the test 4.24 in algorithm 4. We observe that the introduction of the test significantly improves the stability of the estimate, that remains reasonably accurate for noise levels as high as $\sigma_n^2 = 1000$. In comparison, in the absence of the test, the sky estimate quickly becomes totally unreliable. Hence, we recommend the use of (4.24) in the case of unreliability of the Gram-Schmidt estimate.

3.2 Sensitivity to Gains

We have so far assumed perfect antennas with unit gains for convenience in the computation of the station beamshapes. In practice however, the electronic gains and phases of the receivers are different [23], leading to a complication in computation of the beamshapes. Indeed, as defined in chapter 2, definition 2.1 page 17, the beamshape for the ith station of the telescope is given by

$$b_i(\boldsymbol{r}) = \alpha(\boldsymbol{r}) \boldsymbol{w}_i^H \left(\boldsymbol{\Gamma}_i \odot \boldsymbol{a}_i(\boldsymbol{r}) \right), \quad \forall \boldsymbol{r} \in \mathbb{S}^2, \tag{4.25}$$

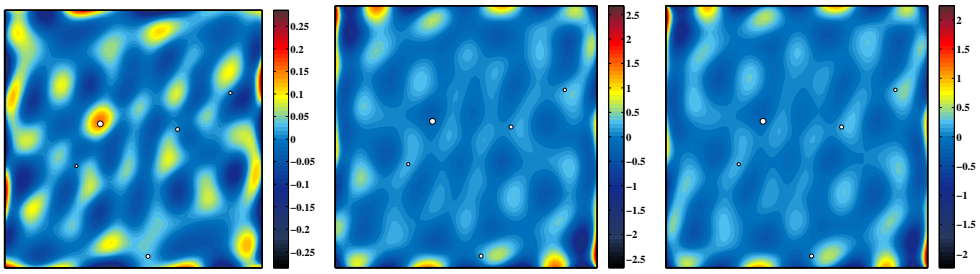
with $\boldsymbol{\Gamma}_i \in \mathbb{C}^L$ the gains of the antennas within station i and $\boldsymbol{a}_i : \mathbb{S}^2 \to \mathbb{C}$ the antenna steering vector for station i. Hence in theory, one would need to estimate the gains $\boldsymbol{\Gamma}_i$ for all stations in order to compute b_i, as those are unknown quantities that



(a) Ideal Visibilities (no estimation).

(b) Estimated visibilities ($\sigma_n^2 = 0$, $N_s = 1000$).

(c) Estimated visibilities ($\sigma_n^2 = 300$, $N_s = 1000$).

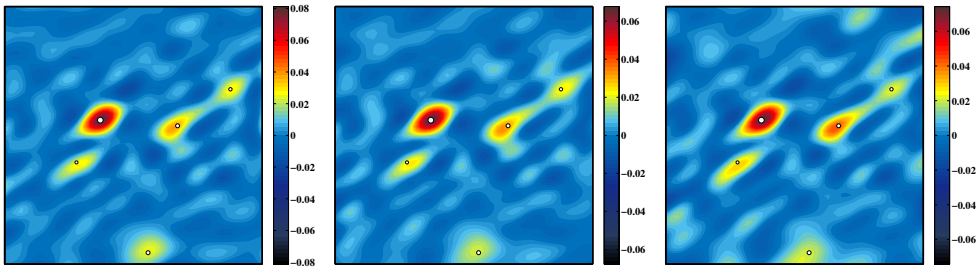


(d) Estimated visibilities ($\sigma_n^2 = 500$, $N_s = 1000$).

(e) Estimated visibilities ($\sigma_n^2 = 800$, $N_s = 1000$).

(f) Estimated visibilities ($\sigma_n^2 = 1000$, $N_s = 1000$).

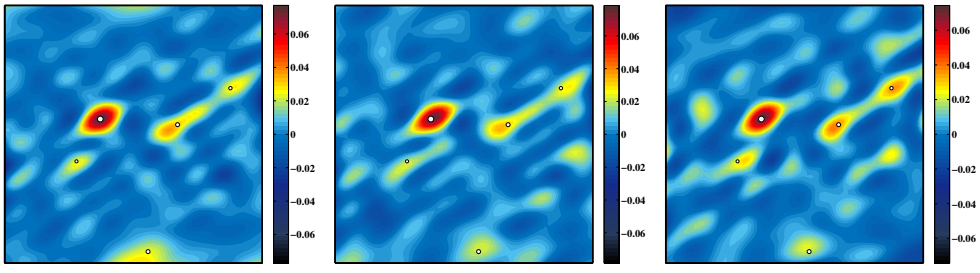
Figure 4.10: Evolution of the Gram-Schmidt Estimate with the noise level. Here, the test eq. (4.24) is not performed, which does not guarantee the stability of the sky estimate. We observe that the latter becomes totally unreliable for high noise levels.



(a) Ideal Visibilities (no estimation).

(b) Estimated visibilities ($\sigma_n^2 = 0$, $N_s = 1000$).

(c) Estimated visibilities ($\sigma_n^2 = 300$, $N_s = 1000$).

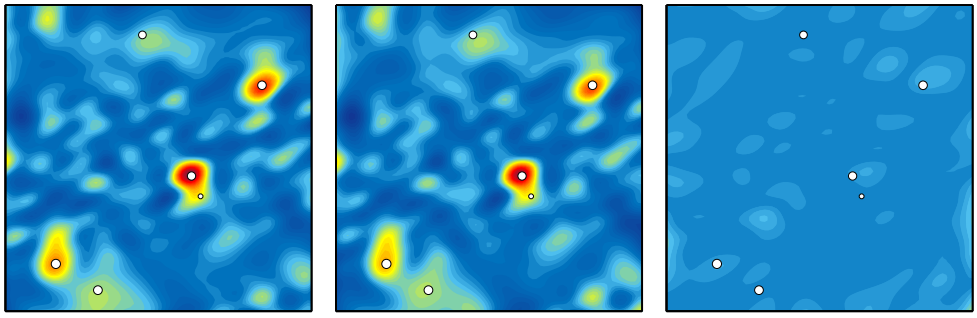


(d) Estimated visibilities ($\sigma_n^2 = 500$, $N_s = 1000$).

(e) Estimated visibilities ($\sigma_n^2 = 800$, $N_s = 1000$).

(f) Estimated visibilities ($\sigma_n^2 = 1000$, $N_s = 1000$).

Figure 4.11: Evolution of the Gram-Schmidt Estimate with the noise level. Here, the test eq. (4.24) is performed, and hence the sky estimate is much more stable. We observe that the latter remains reasonably accurate even for high noise levels.



(a) Unit gains for each receiver. **(b)** Different complex gains for each receiver. **(c)** Absolute difference between the two sky estimates.

Figure 4.12: Sensitivity of the Gram-Schmidt Estimate to non uniform gains across antennas. We observe that the Gram-Schmidt estimate is not very sensitive to the non uniform complex gains across antennas. Hence, the need for station calibration is questionable, as the improvement in the sky estimate quality seems not worth the expense of estimating the gains.

vary through time. This estimation procedure is called **station calibration**, and can represent a significant computational overhead. However, if the Gram-Schmidt estimate were robust enough to small fluctuations in the antennas' gains, then we could avoid this expensive step and approximate the beamshapes by

$$b_i(\boldsymbol{r}) \simeq \alpha(\boldsymbol{r}) \boldsymbol{w}_i^H \boldsymbol{a}_i(\boldsymbol{r}), \quad \forall \boldsymbol{r} \in \mathbb{S}^2. \tag{4.26}$$

To understand whether or not such an approximation is reasonable, we designed the following experiment:

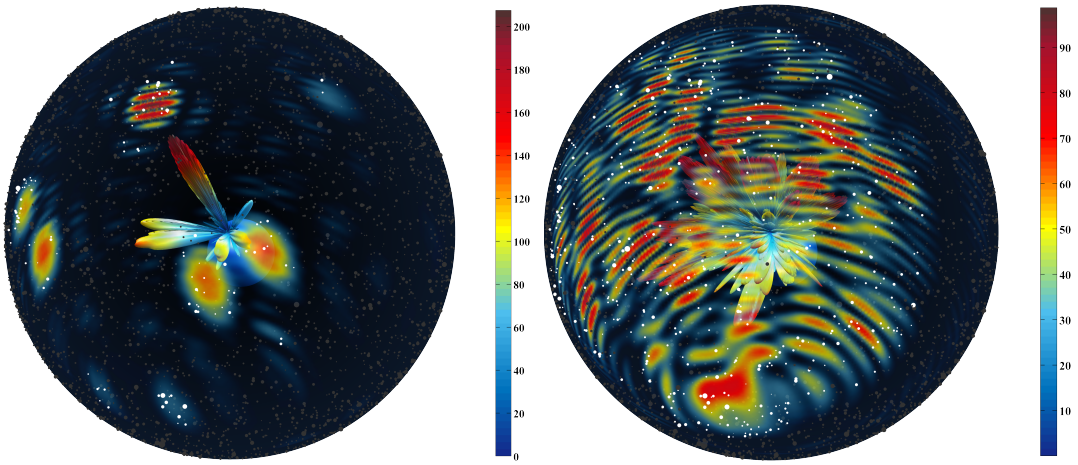
- For each antenna, simulate the gains $\gamma_i \in \mathbb{C}$ as follows

$$\gamma_i \overset{i.i.d}{\sim} Re^{j\theta}, \quad \forall i = 1, \dots, MJ,$$

  with $R \sim \mathcal{N}(1, 0.3)$ and $\theta \sim \mathcal{U}([0, 0.3])$. The parameters of both distributions have been chosen to produce realistic gains for the LOFAR telescope.
- For a given sky, generate samples that would be recorded by each of the antennas if they had unit gains. Duplicate the dataset and multiply the samples from each antenna in the second dataset by the associated complex gains γ_i previously simulated.
- Add Gaussian thermal noise (say $\sigma_n^2 = 500$ for example), beamform both datasets with matched beamforming and form two sets of visibilities.
- Choose $b_i(\boldsymbol{r}) = \alpha(\boldsymbol{r}) \boldsymbol{w}_i^H \boldsymbol{a}_i(\boldsymbol{r})$ for both measurement sets (neglecting that one has been corrupted by the non uniform gains), compute B and apply algorithm 4 to both measurement sets to obtain the associated Gram-Schmidt estimates. For simplicity, we choose a small enough field of view so that $\alpha(\boldsymbol{r}) \simeq 1$.

The results of the experiment are shown on fig. 4.12. Observe that approximating the beamshapes by (4.26) in the case of non unit gains does not affect much the Gram-Schmidt estimate. Hence, the imaging pipeline we propose seems not to be too sensitive to the individual antennas' gains, and the need for station calibration is then very questionable, as the improvement in the sky estimate quality it would bring does not seem worth the expense of estimating the gains.

The robustness of our algorithm to the use of (4.26) instead of (4.25) is particu-



(a) Beamshape of the third station of LOFAR, using matched beamforming towards the zenith of LOFAR and $f_0 = 75$ MHz.

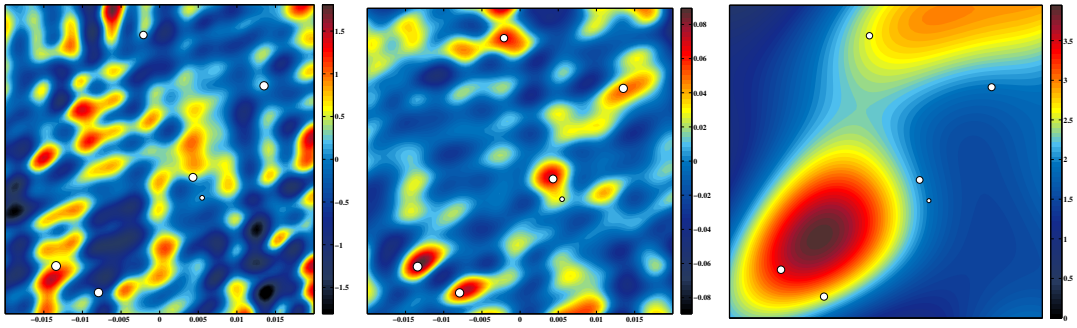
(b) Beamshape of the third station of LOFAR using randomized beamforming.

Figure 4.13: Matched beamforming vs. randomized beamforming. We observe that matched beamforming focuses on a small area of the sky and maximize the signal power coming from this area, while randomized beamforming tries to gather information from all the directions in the sky, to the price of a reduced sensitivity.

larly convenient as it provides us with an analytical description of the beamshapes depending only on the telescope layout, the chosen beamforming technique and the primary beamshapes. As all those quantities are independent from measurements and experimental conditions, it is possible to precompute the orthogonalized functions, and store them together with the coefficients computed at each iteration. The visibilities can then be corrected afterwards.

3.3 On the Use of Information-Maximizing Beamforming Techniques

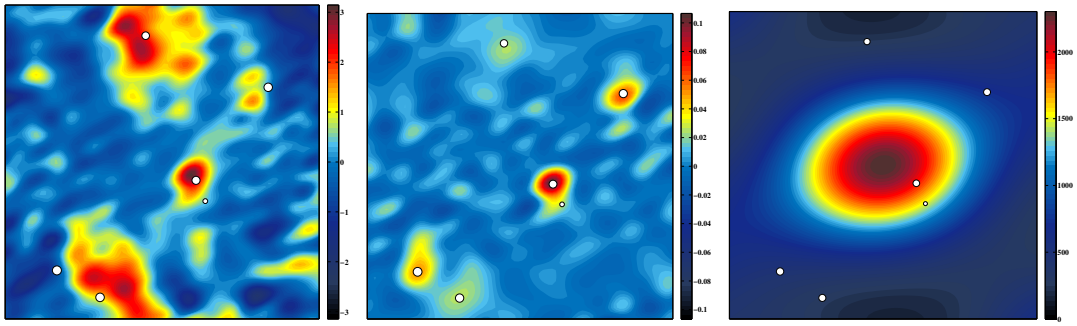
With the increasing number of antennas in modern radio interferometers, beamforming was introduced as a technique to reduce the amount of data to be sent to the central processor. As we have seen previously, the de facto method for what is effectively distributed lossy data compression is matched beamforming [5]. This beamforming technique can essentially be seen as a digital spatial filter mimicking the ability of dish antennas to focus at a certain location in the sky by aligning and summing coherently the signals received by each antenna from the point of focus [14, 23]. In terms of information content, this beamformer is maximizing the information coming from a certain direction in the sky, while keeping unchanged the noise level. Hence, it will perform optimally for very small field of view with high measurement noise, but will be less adapted for surveying large portions of the sky. In some sense, matched beamforming is trading off sky coverage for capturing more signal power. To assess more generally this trade-off from an information point of view, more versatile beamforming strategies have been proposed [14, 15], based on a randomization of the beamforming vectors at each station. The resulting beams capture information from all directions in the sky, and hence maximize the information content within the beamformed output (see fig. 4.13). As it stands, the current imaging pipeline is not ready for such beamforming techniques, as the computation of the stations uv-coordinates is usually done in the plane attached to the center of the focus region (in the case of matched beamforming). When



(a) Dirty image obtained with A-projection (Randomized Beamforming).

(b) Gram-Schmidt least squares estimate (Randomized Beamforming).

(c) Average of stations' beamshapes (Randomized Beamforming).



(d) Dirty image obtained with A-projection (Matched Beamforming).

(e) Gram-Schmidt least squares estimate (Matched Beamforming).

(f) Average of stations' beamshapes (Matched Beamforming).

Figure 4.14: Robustness of the classical and Gram-Schmidt imaging pipeline to the use of different beamforming techniques. We observe that, when using randomized beamforming at each station, the dirty image obtained with A-projection is completely unreliable (not a single source is aligned with the dirty map, and many artifacts pollute the estimate and forbid sources identification). With the Gram-Schmidt imaging pipeline however, the least squares estimate is much more accurate, and resolves most of the sources. For comparison purposes, we provide the sky estimates obtained with each technique when using matched beamforming. For both, the resulting sky image presents less dirtiness. However, it is interesting to note that the relative intensities of the sources located on the outer part of the image are better recovered by randomized beamforming (at least in the case of the Gram-Schmidt estimate). Because of the reduced sensitivity of random beams, the noise level was set to zero for this specific experiment.

using randomized beamforming, there is no clear focus region (as we try to capture as much as possible from the entire field of view) and hence it is less clear how the uv-coordinates should be computed. This thesis was partially motivated by the need for a new imaging pipeline that would be robust to any beamforming technique. Hence, we assessed the performances of the Gram-Schmidt imaging pipeline with randomized beamforming, and showed that it offers a more robust environment than the classical imaging pipeline.
More specifically, we investigated the following beamforming strategy [15]:

R1 For each station $i = 1, \dots, M$, each component $\omega_k^{(i)} \in \mathbb{C}$ of the beamforming vector $\boldsymbol{w}_i \in \mathbb{C}^L$ is chosen randomly according to

$$\omega_k^{(i)} \sim \mathbb{C}\mathcal{N}(0, 1).$$

Then, to avoid magnifying the noise variance, the beamforming vector is

normalized: $\boldsymbol{w}_i \leftarrow \boldsymbol{w}_i / \|\boldsymbol{w}_i\|_2$.

On fig. 4.14, we can see that beamforming strategy **R1** has terrible consequences for the accuracy of the dirty image produced with A-projection: the sky estimate becomes totally unreliable, and polluted by strong artefacts, which forbid any identification of the actual sources within the dirty map. The Gram-Schmidt estimate however, is far more robust to the use of random beams, and remains reasonably accurate in comparison to the sky estimate obtained with matched beamforming. It is important to note here that when using a single beamshape per station matched beamforming always performs better than randomized beamforming, as pointed out in [15]. The benefit of randomized beamforming starts when using two or more beamshapes per station (i.e., each station produces two beamformed outputs).

4 Gram-Schmidt and the QR-factorization

In all that follows, we will assume that $B \in \mathbb{C}^{J\times N^2}$ is a wide matrix ($N^2 \geq J$) with full-row rank. In section section 1.3, we showed that the output visibilities $\boldsymbol{V}_\perp \in \mathbb{C}^j$ of algorithm 4 could be written in terms of a linear transformation of the original data $\boldsymbol{V} \in \mathbb{C}^J$:

$$\boldsymbol{V}_\perp = \mathcal{G}\boldsymbol{V},$$

where $\mathcal{G} \in \mathbb{C}^{j\times J}$ an appropriate linear operator, determined by the various coefficients appearing in the orthogonalization procedure of the rows of B. When B is full-row rank, we have $j = J$ and we showed that in that case $\mathcal{G}$ could be written as a product of lower triangular matrices:

$$\mathcal{G} = \prod_{i=1}^{J} G_i,$$

where each of the $G_i \in \mathbb{C}^{J\times J}$ are defined in eq. (4.13) page 52.

We have then,

$$\begin{aligned} & \boldsymbol{V}_\perp = B_\perp \boldsymbol{I}, \\ \Leftrightarrow \quad & \mathcal{G}\boldsymbol{V} = B_\perp \boldsymbol{I}, \\ \Leftrightarrow \quad & \boldsymbol{V} = \mathcal{G}^{-1} B_\perp \boldsymbol{I}, \end{aligned}$$

where the first equality hold by construction of $\boldsymbol{V}_\perp$ and $B_\perp$ and the last equality is valid as $\mathcal{G}$ is the product of invertible matrices (obvious when looking at the definition of the G_i's).

But we also have $\boldsymbol{V} = B\boldsymbol{I}$, and hence,

$$B = \mathcal{G}^{-1} B_\perp.$$

If we transpose this equation we get

$$B^T = (B_\perp)^T \left(\mathcal{G}^{-1}\right)^T. \tag{4.27}$$

To simplify, we define $Q_1 := (B_\perp)^T \in \mathbb{C}^{N^2\times J}$ and $R_1 := \left(\mathcal{G}^{-1}\right)^T \in \mathbb{C}^{J\times J}$. By construction, the columns of Q_1 form an orthonormal family, and hence we can complement this family in an orthogonal basis that spans the entire space $\mathbb{C}^{N^2}$. We

can then expand Q_1 in a square unitary matrix $Q \in \mathbb{C}^{N^2 \times N^2}$ by filling in the $N^2 - J$ additional columns by the complementary orthonormal family $Q_2 \in \mathbb{C}^{N^2 \times (N^2 - J)}$:

$$Q = (Q_1 \; Q_2) \, .$$

Moreover, as $\mathcal{G}$ is a lower triangular matrix, then $\mathcal{G}^{-1}$ is a lower triangular matrix as well and R_1 is consequently an upper triangular matrix. Let call $R \in \mathbb{C}^{N^2 \times J}$ the matrix defined by

$$R := \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix},$$

with $\mathbf{0} \in \mathbb{C}^{(N^2 - J) \times J}$ the null matrix. Then, we can rewrite eq. (4.27) as

$$B^T = QR = (Q_1 \; Q_2) \begin{pmatrix} R_1 \\ \mathbf{0} \end{pmatrix}. \tag{4.28}$$

We recognize here a QR-factorization of the tall matrix B^T (Q is unitary and R_1 upper triangular). Equation (4.27) can then be interpreted as the **thin or reduced QR-factorization** [22]. Hence, the Gram-Schmidt least squares estimate could equivalently be produced proceeding as follows:

- Compute the reduced QR-factorization of B^T. That yields $B^T = Q_1 R_1$, and hence
$$\boldsymbol{V} = R_1^T Q_1^T \boldsymbol{I}.$$
- Compute $\boldsymbol{V}_\perp$ as
$$\boldsymbol{V}_\perp = \left(R_1^T\right)^{-1} \boldsymbol{V}.$$
The matrix R_1^T being lower triangular,$\boldsymbol{V}_\perp$ can be efficiently computed by **forward substitution** (very low complexity).
- Compute the Gram-Schmidt least squares estimate as
$$\hat{\boldsymbol{I}}_{LS} = Q_1^* \boldsymbol{V}_\perp.$$

The above procedure, summarized in algorithm 5, is, in linear algebra terms, equivalent toalgorithm 4 (see fig. 4.15). However, the QR approach is particularly promising, as there are means to compute the QR factorization of a matrix more efficiently than with the naive Gram-Schmidt procedure. For example, the Householder algorithm computes the QR-factorization of a matrix $B^T \in \mathbb{C}^{N^2 \times J}$ in $\propto 2N^2J^2 - \frac{2}{3}J^3$ flops while Gram-Schmidt takes $\propto 2N^2J^2$ flops [22]. Moreover, the Householder algorithm involves only multiplications by unitary matrices, and hence the algorithm is numerically stable [22]. For those reasons, we recommend the use of the above procedure in practice rather than the direct, classical Gram-Schmidt process.

Algorithm 5: Gram-Schmidt Imaging using the QR-factorization of B^T.

procedure ▷ INPUTS: A tall, full-row rank matrix $B \in \mathbb{C}^{J \times N^2}$ and a set of visibilities $\{V_i\}_{i=1,\dots,J}$

  $[Q_1, R_1] \leftarrow \texttt{QR\_economic}(B^T)$; ▷ *Compute the reduced QR-factorization of* B^T.

  $R \leftarrow R_1^T$;

  for $i = 1$ **to** J **do** ▷ *Forward substitution*

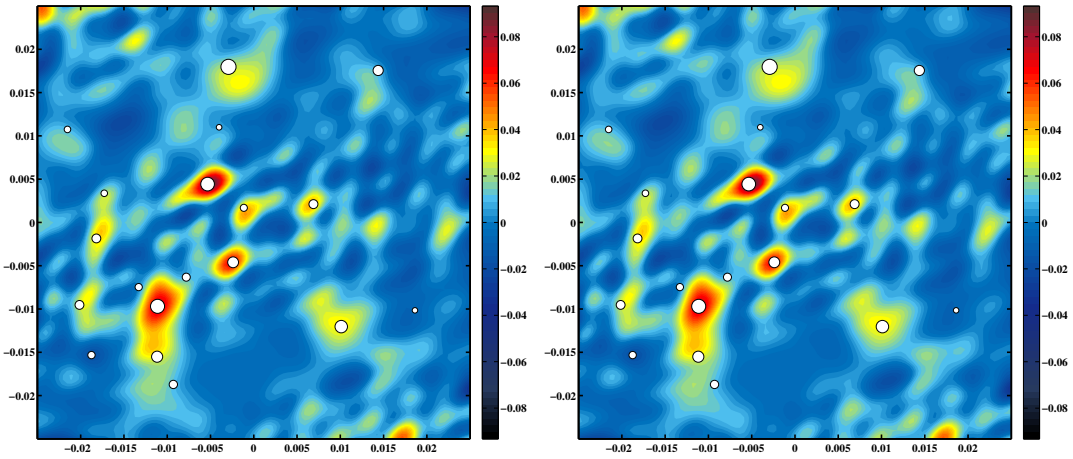
    $V_i^{\perp} \leftarrow \left(V_i - \sum_{k=1}^{i-1} R_{i,k} V_k^{\perp}\right) / R_{i,i}$;

  end

  $\boldsymbol{V}_{\perp} = \left(V_1^{\perp}, \dots, V_J^{\perp}\right)^T$;

  $\hat{\boldsymbol{I}}_{LS} = Q_1^* \boldsymbol{V}_{\perp}$;

  return $\hat{\boldsymbol{I}}_{LS}$.



(a) Least squares estimate obtained with Gram-Schmidt.

(b) Least squares estimate obtained with the QR-factorization.

Figure 4.15: Comparison between the least squares estimates obtained with Gram-Schmidt and the QR-factorization. We observe that up to numerical accuracies, the two estimate are the same. However, the use of the QR-factorization is computationally more efficient.

5

Comparison to State of the Art

In this chapter, we compare the imaging pipeline introduced in chapter 4 to the classical imaging pipeline, both in terms of number of operations, accuracy and sensitivity of the resulting sky estimates. More precisely, we investigate the cost of computing the least squares estimate and the approximate LASSO estimate by using algorithm 5 and corollary 2.1. We compare it to the cost of the CLEAN algorithm used in conjunction with the A-projection algorithm (see section 4 page 25), and identify realistic scenarios for which our imaging pipeline uses fewer operations than the classical imaging pipeline. Finally, we investigate the quality of the sky estimates obtained with both techniques on simulated data. We show that, when applicable, corollary 2.1 provides an approximate LASSO estimate with higher quality than the classical CLEAN estimate.

1 Complexity Analysis

The subsequent derivations are based on the following assumptions:

- Adding, multiplying, subtracting or dividing two real numbers takes a constant time 1.
- Evaluating trigonometric functions (sinus, cosinus) or square roots for a given $x \in \mathbb{R}$ takes a constant time 1.
- Logical tests between two real numbers take constant time 1.
- The Fast Fourier transform of a vector of size N^2 has cost $5N^2 \log_2\left(N^2\right) - 10N^2 + 16$ (radix-2 FFT algorithm [11, 12]).

Finally, we will use extensively the following results:

- Adding, or subtracting two complex numbers $\alpha, \beta \in \mathbb{C}$ has cost 2.
- Multiplying two complex numbers $\alpha, \beta \in \mathbb{C}$ has cost 6.

1.1 Complexity of the New Imaging Pipeline

For this complexity analysis, we use the following notation:

- L antennas per stations.
- M stations.
- J baselines.
- N^2 pixels.

To simplify the computation, we decompose the operations count into the following independent steps:

1 Sample the M stations beamshapes $\{b_i(\boldsymbol{r}),\, i = 1, \ldots, J\}$ over a uniform grid of size N^2 covering the field of view.
2 Form the J rows of $B \in \mathbb{C}^{J\times N^2}$ by pairwise multiplying the sample beamshapes with one another. The jth row of B is given by $\boldsymbol{\beta}_j = \left(\beta_j^*(\boldsymbol{r}_1), \ldots, \beta_j^*(\boldsymbol{r}_{N^2})\right) \in \mathbb{C}^{1\times N^2}$, with $\beta_j^*(\boldsymbol{r}) = b_{i_j}^*(\boldsymbol{r}_n) b_{k_j}^*(\boldsymbol{r})$.
3 Compute the reduced QR-factorization of the tall matrix $B^T \in \mathbb{C}^{N^2\times J}$.
4 Compute $\boldsymbol{V}_\perp \in \mathbb{C}^J$ by forward substitution.
5 Compute the least squares estimate by evaluating $Q_1^* \boldsymbol{V}_\perp$.
6 Extract the approximate LASSO estimate by using corollary 2.1 page 60.

The computational cost of each of is detailed below.

Step 1: Sampling the Stations Beamshapes

The beamshape of station $i \leq M$ is given by

$$b_i(\boldsymbol{r}) = \sum_{h=1}^{L} \omega_h^{(i)^*} e^{-j2\pi\langle \boldsymbol{r}, \boldsymbol{p}_h^{(i)}\rangle}, \quad \forall \boldsymbol{r} \in \mathbb{S}^2.$$

Hence, to evaluate the beamshape for a given point $\boldsymbol{r}_1$ on the sphere we need 8+6 operations for each summation term, so in total we have to perform $14L + 2(L-1)$ operations to compute each of the summation terms and sum them together. As we have to repeat this operation for each pixel and station, the total cost of sampling the J beamshapes is given by

$$c_b = (14L + 2(L-1))MN^2 = 16LMN^2 - 2MN^2.$$

Step 2: Computing the rows of B

The jth row of B is given by $\boldsymbol{\beta}_j = \left(\beta_j^*(\boldsymbol{r}_1), \ldots, \beta_j^*(\boldsymbol{r}_{N^2})\right) \in \mathbb{C}^{1\times N^2}$, with $\beta_j^*(\boldsymbol{r}) = b_{i_j}^*(\boldsymbol{r}_n) b_{k_j}^*(\boldsymbol{r})$. Hence, computing one row of B involves the pairwise multiplication of two N^2 complex vectors, or N^2 multiplications of two complex numbers. The total number of operations required to form the J rows of B is then given by

$$c_B = 6JN^2.$$

Step 3: Computing the reduced QR-factorization of B^T

The complexity of the Householder algorithm to compute the reduced QR-factorization of a $N^2 \times J$ matrix, with $N^2 \geq J$, is given by [1,22]

$$c_{QR} = 8N^2J^2 - \frac{8}{3}J^3 + 22N^2J - 8J^2 + \frac{32}{3}J.$$

Step 4: Computing $\boldsymbol{V}_\perp$ by forward substitution

The ith component of $\boldsymbol{V}_\perp \in \mathbb{C}^J$ is given by

$$V_i^\perp \leftarrow \left(V_i - \sum_{k=1}^{i-1} R_{i,k} V_k^\perp\right) / R_{i,i}.$$

Hence, the computation of $V_i^\perp$ requires $6(i-1)+2(i-2)+2+2$ operations. Summing this for i going from 1 to J we get the total cost of computing $\boldsymbol{V}_\perp \in \mathbb{C}^J$:

$$
\begin{aligned}
c_V &= \left(\sum_{i=2}^{J} 6(i-1) + 2(i-2) + 4 \right) + 2, \\
&= 6 \sum_{i=1}^{J-1} i + 2 \sum_{i=1}^{J-2} i + 4(J-1) + 2, \\
&= 10(J-1) + 8 \sum_{i=1}^{J-2} i + 2, \\
&= 10(J-1) + 8 \frac{(J-2)(J-1)}{2} + 2, \\
&= 4J^2 - 2J + 4.
\end{aligned}
$$

Step 5: Computing the Least Squares Estimate

From algorithm 5, we have that

$$\hat{\boldsymbol{I}}_{LS} = Q_1^* \boldsymbol{V}_\perp.$$

For efficiency, the reduced QR-factorization does not explicitly compute the matrix Q_1. It is possible to compute the above product implicitly (see [22]). The number of operations required to compute $\hat{\boldsymbol{I}}_{LS}$ is then given by

$$c_{LS} = 18JN^2 - 9J^2 - 11J.$$

Step 6: Extract the Approximate LASSO Estimate

Here, the worst case is for $N^2 \geq J$. From corollary 2.1 we see that the approximate LASSO estimate is then given by

$$\hat{I}^i_{LASSO} \simeq \frac{\operatorname{sgn}\left(\hat{I}^i_{LS}\right)}{\mu_i} \left(\left| \hat{I}^i_{LS} \right| - \frac{\lambda}{2} \right)^+, \quad \forall i = 1, \ldots, N^2.$$

The term μ_i is given by $\mu_i = \sum_{j=1}^{J} \beta_j(\boldsymbol{r}_i)\beta_j^*(\boldsymbol{r}_i)$. The computation of this term requires $4J - 1$ operations. Hence, the cost of computing one pixel is $4J + 6$, and the total cost to form the image is

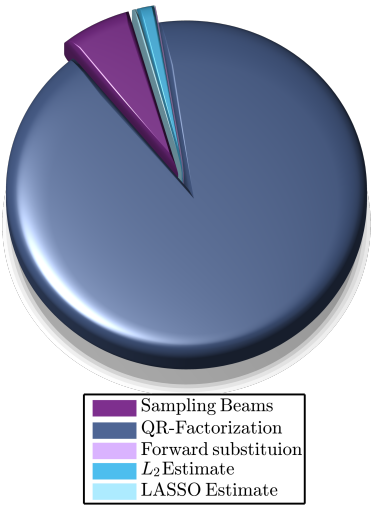
$$c_{LASSO} = 4JN^2 + 6N^2.$$

Total Cost

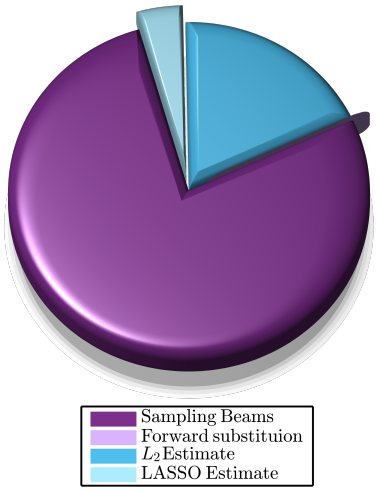
Summing the number of operations necessary for each step, we get:

$$c_{GS} = 8N^2J^2 - \frac{8}{3}J^3 + 50N^2J + (16L - 2)MN^2 + 6N^2 - 13J^2 - \frac{7}{3}J + 4. \quad (5.1)$$

The leading term is $8N^2J^2 - \frac{8}{3}J^3$. Observe that the total number of operations behaves linearly with the number of pixels and quadratically with the number of visibilities. In fig. 5.1, we can see that the QR-factorization of B^T is clearly the most expensive step of the algorithm ($\sim 92\%$ of the total number of operations for this specific example). Then come the computation of the basis elements ($\sim 6.4\%$), the least squares estimate ($\sim 1.5\%$), the LASSO estimate ($\sim 0.3\%$) and finally the



(a) Relative cost of the individual steps of the algorithm with respect to the total cost.



(b) Relative cost of the individual steps of the algorithm with respect to the total cost minus the cost of the QR-factorization (zoom on the left upper-part of **(a)**).

Figure 5.1: Contribution of the different steps involved in the new imaging pipeline to the total cost eq. (5.1). For this example, we chose a small number of stations $M = 12$, so that the QR-factorization cost does not completely dominate the other steps and make the chart unreadable. The size of the image is $N^2 = 1000 \times 1000$ pixels. We observe that the QR-factorization is clearly the most expensive step of the algorithm ($\sim 92\%$ of the total number of operations for this specific example). Then, comes the computation of the basis elements ($\sim 6.4\%$), the least squares estimate ($\sim 1.5\%$), the LASSO estimate ($\sim 0.3\%$) and the massaging of the visibilities by forward substitution (very cheap $\sim 0.0004\%$).

computation of the new set of visibilities $\boldsymbol{V}_\perp$ by forward substitution (by far the cheapest step, only $\sim 0.0004\%$ of the total cost).

1.2 Complexity of the Classical Imaging Pipeline

For comparison, we provide here a simplified complexity analysis of the CLEAN+A-projection algorithm (see algorithm 2). We exclude from consideration w-stacking (see remark 3.1), as it is very dependent on the telescope layout and the specific experimental conditions (pointing direction). A more complete and general derivation of the complexity of the A-projection algorithm can be found in [19]. For this analysis, we use the following notation:

- L antennas per stations.
- M stations.
- J baselines.
- N^2 pixels.
- Beamshape kernels of size N_s^2. For typical LOFAR observations, $N_s^2 = 11 \times 11$ pixels (see [19]).
- Average size of the w-kernels: N_w^2. For typical LOFAR observations, $N_w^2 = 30 \times 30$ pixels (see [19]).
- Oversampling parameter O (for nearest neighbors interpolation). Usually, we take $O = 9$.
- Size of the synthesized beam (convolution after the CLEAN iterations): N_B^2. Typically, we set $N_B^2 = 5 \times 5$.

- N_{iter} CLEAN iterations, typically in the range 10^4 to 10^6 (see [9,19]).

Let's first address the cost of the CLEAN algorithm, independently from the cost of the A-projection algorithm.

Cost of the CLEAN algorithm

The CLEAN algorithm (see algorithm 2 page 28) performs at each iteration the following steps:

1. Compute the residual image,
$$\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}) \leftarrow \mathcal{A}^H(\mathcal{A}\hat{\boldsymbol{I}}^{(n)} - \boldsymbol{\mathcal{V}}),$$
2. Find the strongest component in the residual image and update the corresponding component in the CLEAN image
$$\hat{\boldsymbol{I}}^{(n)} \leftarrow \hat{\boldsymbol{I}}^{(n)} - \tau\Psi\nabla\Phi(\hat{\boldsymbol{I}}^{(n)}),$$
3. Compute the quantity $\|\nabla\Phi(\hat{\boldsymbol{I}}^{(n)})\|_\infty$ and compare it to the threshold $\epsilon > 0$ (stopping criterion).

The first step involves two multiplication by the matrix $\mathcal{A}$ and its conjugate transpose. This is performed efficiently by applying twice the A-projection algorithm. We denote for now by c_{Aproj} the computational cost of applying the A-projection algorithm, and will compute later this cost. Apart from these two operations, we have J complex subtractions to perform, to form the residual visibilities. Hence, the cost of forming the residual image for a given iteration is given by

$$c_{res} = 2c_{Aproj} + 2J.$$

Finding the strongest component in the residual image requires N^2 logical tests. The update of the CLEAN estimate requires one real multiplication and one real addition. Finally, the computation and testing of the stopping criterion requires $N^2 + 1$ logical tests. Hence, in total, each loop of the CLEAN algorithm involves the following number of operations

$$c_{loop} = 2c_{Aproj} + 2J + 2N^2 + 3.$$

Once the stopping criterion has been met, two steps are performed by the algorithm: the CLEAN estimate is convolved with the synthesized beam (typically a truncated Gaussian kernel) to correct for the artificial high resolution in the estimate, and the residual image for the last estimate is compute one last time and added to the CLEAN estimate in background. Here, because the synthesized beam is very small, the convolution step is more efficiently performed by direct computation rather than by the use of the convolution theorem and the FFT algorithm. The convolution step involves then $N^2(2N_B^2 - 1)$ arithmetic operations. Hence, the computational cost of the post-processing steps is given by

$$c_{post} = N^2(2N_B^2 - 1) + c_{res} + N^2 = N^2(2N_B^2 - 1) + 2c_{Aproj} + 2J + N^2.$$

Finally, if the CLEAN algorithm has performed N_{iter} loops, we have a total cost for the algorithm of

$$c_{CLEAN} = N_{iter}\left[2c_{Aproj} + 2J + 2N^2 + 3\right] + N^2(2N_B^2 - 1) + 2c_{Aproj} + 2J + N^2.$$

It remains now to compute the number of operations required to apply A-projection, and the complexity analysis will be complete.

Cost of the A-projection algorithm

The cost of multiplying by $\mathcal{A}$ or $\mathcal{A}^H$ is the same. Hence, here we will discuss only the number of operations required to apply $\mathcal{A}^H$ to a set of visibilities, as we have developed this case before (see algorithm 1). The algorithm operates as follows:

1. For each baseline (i, j) between station i and j,
   - Compute the beamshapes b_i, b_j and the w-term $\mathcal{W}_{i,j}$,
   - Form the Fourier kernel associated to the baseline. This can be done by multiplying the beamshapes and w-term together in the image plane, and then taking the Fourier transform of it: $\mathcal{F}\{b_i^* b_j \mathcal{W}_{i,j}^*\}$.
   - Convolve with the one-sample visibility function of the corresponding baseline $\tilde{V}_{i,j}(u, v) = V_{i,j}\delta(u - u_{i,j}, v - v_{i,j})$.
2. Sum the corrected contributions $\mathcal{F}\{b_i^* b_j \mathcal{W}_{i,j}^*\} * \tilde{V}_{i,j}$ from each baseline together.
3. Take the 2D inverse FFT of the result.

The computation of the beamshapes and the w-term requires the sampling of those continuous quantities on a regular grid. Because we are only interested in the spectral properties of these quantities, each of them are sampled at their corresponding Nyquist rate. For the beamshapes, it can be shown [19] that this rate is inversely proportional to the station diameter (normalized by the wavelength of observation). Because each LOFAR core station has the same diameter, we end up with M images with identical resolution N_s^2 (with typically, $N_s = 11$). Using the first step of the new imaging pipeline analysis, we see that the computation of those M images involves the following number of operations:

$$c_b = 16LMN_s^2 - 2MN_s^2.$$

The Nyquist rate for the w-term depends on the w-coordinate of the specific baseline under consideration. For simplicity, we assume here that each of the w-terms are sampled at the same rate, resulting in J complex images with resolution N_w^2 (with typically $N_w = 30$). The w-term for baseline (i, k) is given by:

$$\mathcal{W}(l, m; w_{i,k}) = e^{-j2\pi w_{i,k}(\sqrt{1-l^2-m^2}-1)}.$$

Hence, for each pixel (l, m) we have to perform 4 real multiplications, 3 additions, 1 real square root and two evaluations of trigonometric functions. This results in 10 arithmetic operations per pixel, and finally

$$c_w = 10JN_w^2.$$

To multiply in the image plane the beamshapes and w-term of a specific baseline, we need to have them at the same resolution. As $N_w > N_s$, we can simply use zero-padding and FFT to interpolate the beamshapes at the resolution N_w^2:

$$b_i^{N_w} = (S^{N_w})^{-1}\mathcal{F}^{-1}\left\{\mathcal{Z}_{N_s}^{N_w}\mathcal{F}\{S^{N_s} b_i\}\right\},$$

with $\mathcal{Z}_{N_s}^{N_w}$ the zero-padding operator and S^{N_s} the **prolate spheroidal** at resolution N_s^2, a window function used to avoid aliasing due to sharp edges [19]. This operation has a cost of

$$\begin{aligned} c_{interp} &= 2N_s^2 + c_{FFT}(N_s^2) + (N_w^2 - N_s^2) + c_{FFT}(N_w^2) + 2N_w^2, \\ &= N_s^2 + 3N_w^2 + 10\left(N_s^2 \log_2(N_s) - N_s^2 + N_w^2 \log_2(N_w) - N_w^2\right) + 32. \end{aligned}$$

Once this interpolation step has been performed, we can straightforwardly compute the Fourier kernel by multiplying the beamshapes and the w-term together and take the FFT of the result. For later use, this FFT is computed at the higher resolution $O^2 N_w^2$ (by zero-padding), which has cost

$$c_{kernel} = 12N_w^2 + (O^2 N_w^2 - N_w^2) + (10O^2 N_w^2 \log_2(ON_w) - 10O^2 N_w^2 + 16).$$

Because $\tilde{V}_{i,j}$ consists of a rescaled Dirac function, the convolution $\mathcal{F}\{b_i^* b_j \mathcal{W}_{i,j}^*\} * \tilde{V}_{i,j}$ is simply a rescaling and shifting of the Fourier kernel at the uv-coordinates of the corresponding baseline. In practice, as the uv-coordinates do not fall on the grid, they are first shifted to their nearest grid neighbors. To minimize the effect of this shift on the sky estimate, this nearest neighbor interpolation is performed on an oversampled grid of size $O^2 N^2$. The convolution is then performed on that oversampled grid which is finally downsampled to its original size N^2. Neglecting the cost of the nearest neighbor interpolation, we get a total cost of

$$c_{conv} = 6O^2 N_w^2 + N^2.$$

The above operations are repeated for each of the J baselines and the resulting visibilities functions are summed together and Fourier transformed. That yields the following cost for A-projection:

$$c_{Aproj} = J\left(2c_{interp} + c_{kernel} + c_{conv}\right) + 2(J-1)N^2 + (5N^2 \log_2\left(N^2\right) - 10N^2 + 16). \tag{5.2}$$

The terms c_b and c_w have not been included in eq. (5.2), as they do not actually contribute to each call of the A-projection algorithm. Indeed, in practice the beamshapes and w-terms are computed once and saved in memory, and hence the corresponding costs of those operations should only contribute once to the total cost of CLEAN + A-projection. We finally get the total cost for CLEAN + A-projection:

$$c_{CLEAN} = N_{iter}\left[2c_{Aproj} + 2J + 2N^2 + 3\right] + N^2(2N_B^2 - 1) + 2c_{Aproj} + 2J + N^2 + c_b + c_w, \tag{5.3}$$

with c_{Aproj} given in eq. (5.2). Replacing quantities N_s, N_w, N_B and N_{iter} by their typical values for LOFAR helps in understanding better the leading term of eq. (5.3). If we choose $N_s = 11$, $N_w = 30$, $N_B = 5$ and $N_{iter} = 10^4$, we obtain

$$\begin{aligned} &c_{Aproj} \simeq \\ &N^2\left[60 \cdot 10^3 J + 200 \cdot 10^3 \log_2(N) - 219 \cdot 10^3\right] + 115 \cdot 10^9 J + 1936LM - 242M + 3.5 \cdot 10^5. \end{aligned}$$

For that specific choice of parameters, we observe that the leading term of eq. (5.3) is approximately $N^2\left[60\cdot 10^3 J + 100\cdot 10^3 \log_2(N^2)\right]$. Hence, the CLEAN+A-projection algorithm grows as $N^2\log_2(N^2)$ with the number of pixels, and linearly with the number of baselines J.

1.3 Comparison of the Computational Efficiency of both Imaging Pipelines

In this section, we use eq. (5.1) and eq. (5.3) to compare the computational efficiency of both imaging pipelines. We first perform this comparison for the LOFAR telescope and then investigate the scalability of both imaging pipelines to very large radio telescope such as the SKA.

Comparison for LOFAR

Recall that LOFAR is composed of $M = 46$ stations each composed of $L = 48$ elements (for the HBA stations). Moreover, for a typical LOFAR observation we have the following [19]:

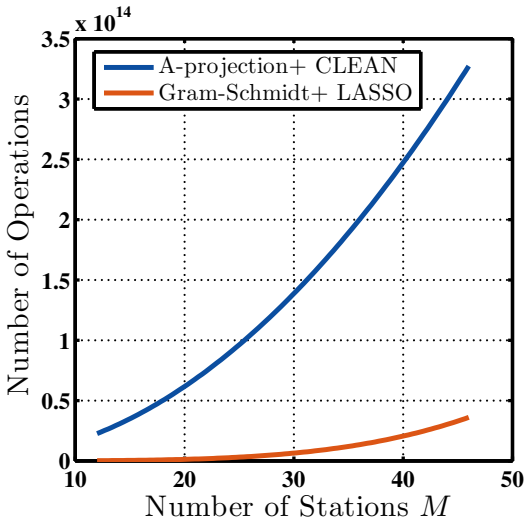
- $N_s^2 = 11\times 11$ pixels,
- $N_w^2 = 30\times 30$ pixels,
- $O = 9$,
- $N_B^2 = 5\times 5$,
- $N_{iter} = 10^4$.

As it is conceptually easier to think in terms of stations rather than in terms of baselines, we transformed the dependency of eq. (5.1) and eq. (5.3) on J in a dependency on M (the number of stations), by using the relation $J = M(M-1)$. Figure 5.2 shows the evolution of the number of operations for both algorithms, with respect to the resolution of the output image and the number of stations used for the recovery. We can see that for every practical case, the Gram-Schmidt imaging pipeline is computationally more efficient than the current one. In table (e) of fig. 5.2, we see that the Gram-Schmidt imaging pipeline can be 2 to 34 times faster than CLEAN + A-projection, depending on the scenario investigated.

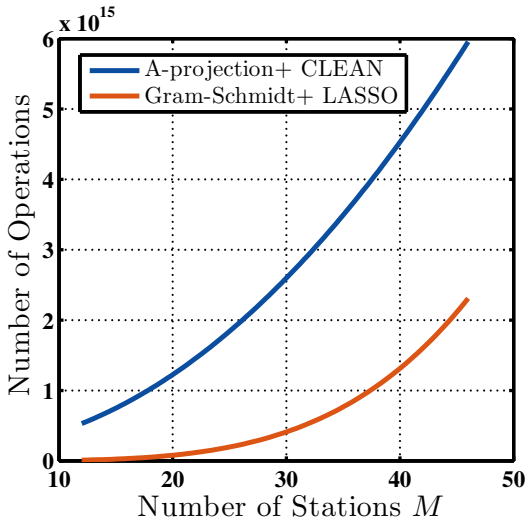
Comparison for the SKA

The SKA will have different characteristics from LOFAR, which we need to account for in our analysis. The construction of the telescope is organized in two phases (SKA1 and SKA2), with the first phase SKA1 starting in 2018. As the design for SKA2 has not been decided yet, we will take here the specifications for the SKA1 LOW as a reference [6]:

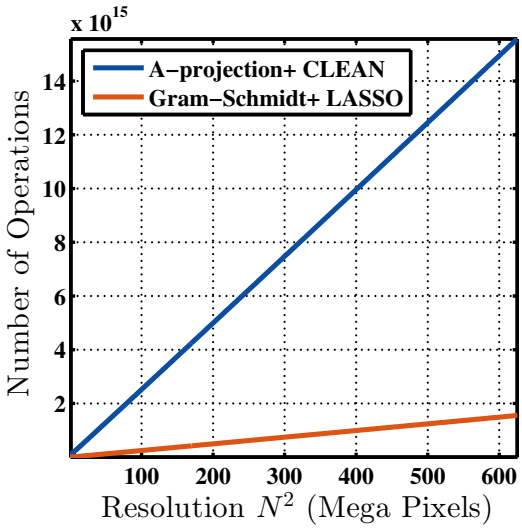
- **Dipole Antennas:** The SKA1 LOW will be composed of $\sim 130,000$ dipole antennas, spread between 500 stations. This gives us an average of $L = 260$ antennas per stations.
- **Stations:** The SKA1 LOW will be composed of $M = 500$ stations, with a diameter of approximately a 100 meters. This is twice the size of the largest LOFAR HBA stations (international stations have a radius of 56.5 meters). The sampling rate of the stations' beamshapes being inversely proportional to the stations' diameters, the SKA stations' beamshapes will have to be sampled at twice the rate of the LOFAR stations' beamshapes. For LOFAR, we had typically $N_s = 11$, hence for SKA1 LOW we choose $N_s = 22$.
- **Collecting area:** The SKA1 will have a total collecting area of $419,000$ square meters. This is approximately 8 times more than the $52,000$ square meters of LOFAR. To achieve a similar collecting area, the distance between the stations of LOFAR should be multiplied by ~ 2.8. Hence, because the average size N_w



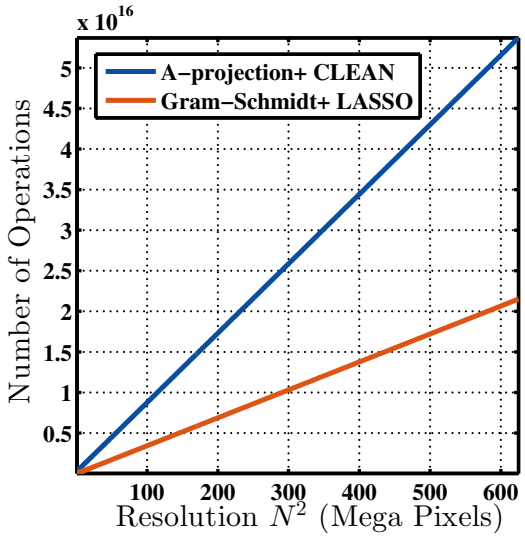
(a) Resolution $N^2 = 1024 \times 1024$ pixels.



(b) Resolution $N^2 = 8192 \times 8192$ pixels.



(c) $M = 24$ stations used for the recovery.



(d) $M = 46$ stations used for the recovery.

c_{Aproj}/c_{GS}		Number of Stations		
		$M = 24$	$M = 38$	$M = 46$
Resolution	$N^2 = 1024 \times 1024$	34.051	13.364	9.0748
	$N^2 = 2048 \times 2048$	15.804	8.6764	4.1174
	$N^2 = 4096 \times 4096$	11.304	4.2796	2.8832
	$N^2 = 8192 \times 8192$	10.239	3.8369	2.579

(e) Ratio c_{Aproj}/c_{GS}. One cell of the tabular reads: *For a resolution N^2 and a number of stations M, Gram-Schmidt + LASSO is x times faster than CLEAN + A-projection* (with x the content of the cell).

Figure 5.2: Number of operations for both imaging pipelines for various scenarios. We observe that for any practical scenario, the new imaging pipeline is at least twice as fast as the classical imaging pipeline. For some specific (but realistic) scenarios, the new imaging pipeline can be 34 times faster than the classical one.

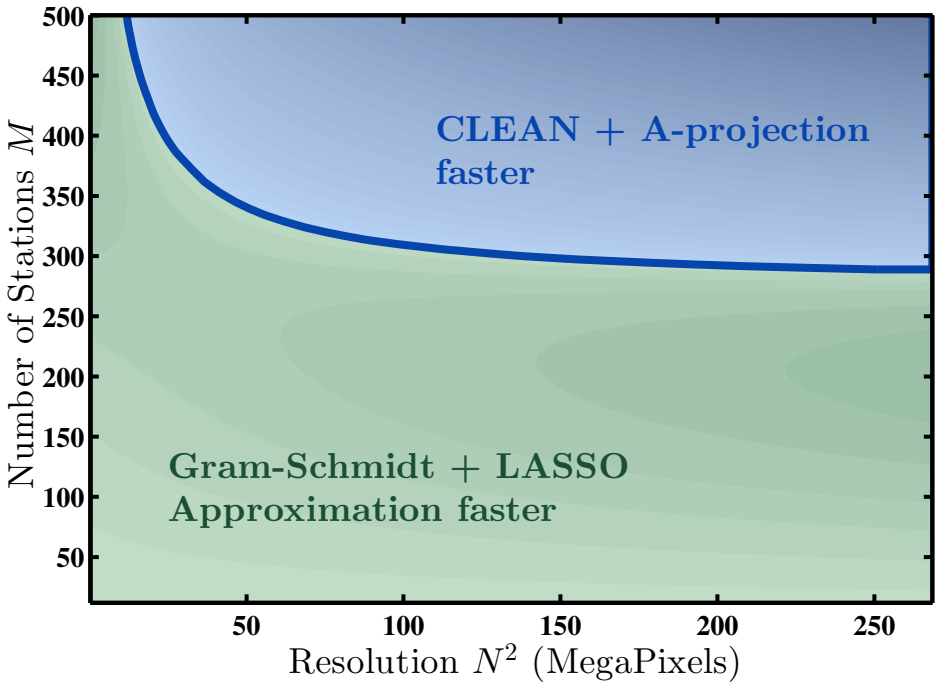


Figure 5.3: Operational Zones of both imaging pipelines for the SKA. To construct this plot we took the difference between c_{Aproj} and c_{GS}. In the blue zone, we have $c_{Aproj} - c_{GS} < 0$, which means that the Gram-Schmidt imaging pipeline is more expensive than the classical imaging pipeline. In the green zone, we have $c_{Aproj} - c_{GS} > 0$, which means that the Gram-Schmidt imaging pipeline is less expensive than the classical imaging pipeline. Finally the blue line correspond to points (N^2, M) for which both imaging pipelines are equally expensive.

of the w-kernel is proportional to the average of the w-term across baselines, we can expect N_w to be bigger for SKA1. Here, we estimate N_w to be 2.8 times larger than for LOFAR. That yields $N_w = 2.8 \times 30 = 84$. This is a rather speculative value, and one would need to investigate in more details the exact layout of SKA1 to come up with a more reliable estimate.

- **Sensitivity:** The sensitivity of SKA1 LOW will be approximatively 8 times higher than that of LOFAR. This means that the threshold for CLEAN will be potentially 8 times lower than for LOFAR (for equivalent experimental conditions). In order to meet such an accuracy, the number of CLEAN iterations are likely to increase significantly. To fix the ideas we choose $N_{iter} = 1.5 \times 10^5$, or 15 times more iterations than we previously assumed for LOFAR.
- **Resolution:** The SKA1 LOW should have a resolution 1.2 times superior to LOFAR. Hence the size N_B of the synthesized beam will be reduced (indeed, the size of the synthesized beam is usually chosen according to the minimal achievable resolution). Here, we take $N_B = 5/1.2 \simeq 4$.

With the above choice of parameters, we investigated the relative efficiency of both imaging pipelines at the scale of the SKA. Figure 5.3 presents the operational zones of both imaging algorithms. We can see that for a number of stations below 300, the Gram-Schmidt imaging pipeline is faster thant the classical imaging pipeline, for any resolution. From 300 to 500 stations, the Gram-Schmidt imaging pipeline remains faster for low resolutions (< 10 Mega Pixels), but becomes too expensive for high resolution images. In this area, the CLEAN + A-projection algorithm is faster. Of course, these results are strongly dependent on the various approximations made in order to extend the analysis to the SKA telescope. Still,

this preliminary analysis reveals that without any optimization of the new imaging pipeline, there might exist scenarios for which the CLEAN + A-projection imager remains cheaper to perform in the case of the SKA. This could become even more problematic with SKA2, where the number of stations should be further increased.

However, the Gram-Schmidt imaging pipeline is still at a very early stage, and we are very confident that it can be greatly optimized. In particular, the sampling and orthogonalization of the basis elements are currently done at the same resolution as the final sky estimate, which represents a huge computational overhead. It would then be interesting to try and perform those steps at a lower resolution, if possible. Finally, the previous analysis considers only the specific case where the sky image is reconstructed from a single set of visibilities, estimated during a certain time interval. In practice, to reduce noise, many set of visibilities estimated during different time intervals are combined to produce more reliable sky images [23]. Because of the rotation of the earth, the layout of the telescope relatively to the direction of observation is different from one time interval to the other. However, because of the deterministic nature of the earth rotation, we believe that there is a lot of redundancy in the basis corresponding to each time intervals, that could certainly be exploited in order to accelerate our estimation procedure. Similarly, one could also try to exploit the redundancies existing between different frequency channels, when the sky is surveyed at different frequencies simultaneously. Currently, none of those redundancies and structures are exploited by the classical imagers. We believe that our framework is better suited to exploit such redundancies, and hence could become significantly cheaper than the classical imaging pipeline, and this even for a very large number of stations such as in the SKA.

2 Comparative Accuracy and Sensitivity Analysis

In this section, we investigate the accuracy of the sky estimates obtained with both imaging pipelines, as well as their sensitivity to the various experimental conditions, such as the thermal noise variance σ_n^2 as well as the number of samples N_s used for the estimation of the visibilities. For those experiments, the 24 HBA stations from LOFAR were used, and the frequency channel of observation was chosen at $f_0 = 120$ MHz. Different skies were randomly generated, and the corresponding visibilities were simulated according to the methodology described in section 3.1 of chapter 3.

For each experiment, two sky estimates were produced with the same input data. The first was obtained by applying classical CLEAN+A-projection. The second was produced by approximating the LASSO estimate from the Gram-Schmidt least squares estimate, as described in section 2 of chapter 4. For CLEAN, we took $\tau = 0.1$ and chose a custom stopping criterion based on a statistical test, to free ourselves from the arbitrariness of the choice of a uniform threshold. At each iteration, the pixels within the residual image are tested (as described in section 1.3 of chapter 4). If no pixel can be declared significantly different from zero with a level $\alpha = 1\%$, then the algorithm is stopped (as the potential remaining sources within the residuals could anyway not be distinguished from noise for the given experimental conditions). Moreover, we did not add the residuals to the CLEAN estimate after the last iteration, to facilitate the comparison with the approximate LASSO estimate obtained with the new imaging pipeline. For the computation of the approximate LASSO estimate, we used strategies such as in fig. 4.5 page 59 to

help in the choice of the penalty parameter.

The first experiment was to evaluate the sensitivity of both estimates to the noise variance and the number of samples used for estimating the visibilities. As a measure of the estimate quality, we used the root mean squared error, given by

$$RMSE(\hat{\boldsymbol{I}}) = \mathbb{E}\left[\|\hat{\boldsymbol{I}} - \boldsymbol{I}_0\|_2\right], \tag{5.4}$$

with $\hat{\boldsymbol{I}} \in \mathbb{R}^{N^2}$ a given sky estimate, and $\boldsymbol{I}_0 \in \mathbb{R}^{N^2}$ the true underlying sky image (available to us as we are simulating). This metric is commonly used to assess the bias-variance trade-off. For each experimental conditions, we estimated eq. (5.4) by simulating three[1] different sets of visibilities, producing the associated estimates, and finally taking the average of the individual root squared errors.

Figure 5.4 shows the results of the sensitivity analysis. The underlying sky image used for simulating the data is given in (a). Examples of sky estimates obtained with Gram-Schmidt + LASSO and CLEAN + A-projection are respectively available in (b) and (c). In fig. 5.4 (d), we reported the evolution of the root mean squared error as a function of the number of samples, for both estimates and a fixed noise level of $\sigma_n^2 = 1000$. There are many interesting observations to make from this plot. To begin with, notice that the Gram-Schmidt + Lasso estimation procedure produces much more accurate sky estimates than CLEAN + A-projection for the chosen metric (on average 1.5 times more accurate). Moreover, observe that, while the Gram-Schmidt + LASSO estimate is clearly improving with the number of samples, this is less obvious for the CLEAN + A-projection estimate, whose RMSE hardly decays with the number of samples. This seems to suggest that the CLEAN + A-projection algorithm is not able to fully exploit this increased accuracy in the measurements in order to produce better sky estimates. However, this behavior could also be due to the choice of metric, which might not properly capture the evolution of the CLEAN + A-projection estimate with the number of samples. Indeed, two sky estimates can appear very different visually, but still have comparable quality in terms of the root mean squared metric[2]. Hence, to validate the above observation, one would need to repeat the experiment with different metrics (peak signal-to-noise ratio, structural similarity, entropy...), and see if similar conclusions can still be drawn.

In fig. 5.4 (e), we investigated the sensitivity of both estimation procedures to the noise variance σ_n^2 for a fixed number of samples $N_s = 800$. Once again, observe that the Gram-Schmidt + LASSO estimation procedure produces much more accurate sky estimates than the CLEAN + A-projection imager, for every noise level tested. Moreover, we observe that Gram-Schmidt + LASSO is more robust to the noise variance, as the RMSE of the estimates obtained for various noise levels are extremely comparable. In comparison, the CLEAN + A-projection algorithm is more affected by the noise level, as it can be seen that the RMSE of the estimates produced with this imager increases with the noise variance (although the increase is not dramatic). Once again, one would need to repeat the experiment for different metrics to validate those observations.

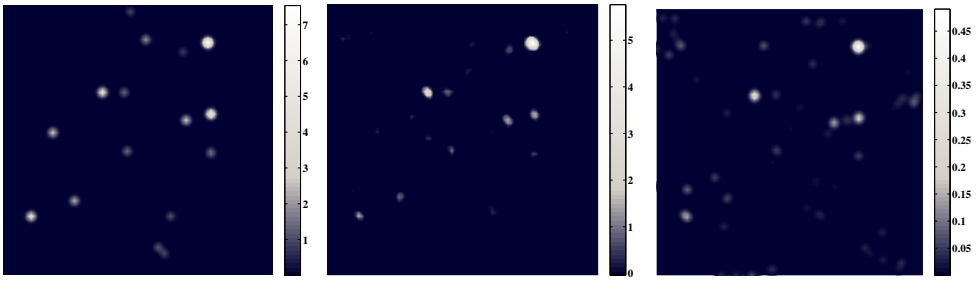
Finally, we also provide in fig. 5.5 the sky estimates obtained with both estimation procedures for five different skies. For each, observe that the range of the Gram-

[1] This is evidently too few a number of simulations for a good estimate of RMSE, but because of the time constraints we could not afford to run more simulations.

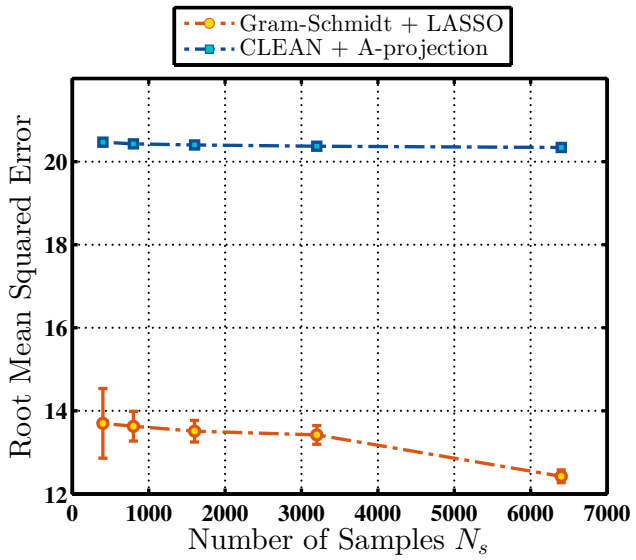
[2] For example, consider a sky estimate polluted by a false positive at a certain position. If we artificially change the position of the false positive in the estimate we still have the same root mean squared error, while the two skies might appear structurally very different for a human observer.

Schmidt + LASSO estimate is closer to the actual range of the underlying sky image. Moreover, a visual inspection of the images reveals that the CLEAN + LASSO estimates are more conservative than the CLEAN + A-projection estimates: they sometimes miss faint sources, but have far less false positives.

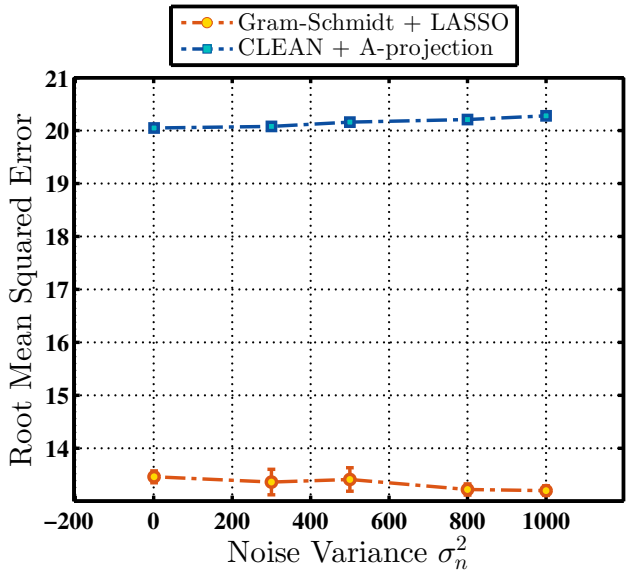
From those experimental conditions at least, we can safely claim that, when applicable, the new imaging pipeline presented in this document produces more accurate, reliable and robust sky estimates than the classical imaging pipeline.



(a) True sky image. **(b)** Gram-Schmidt + LASSO ($N_s = 800$, $\sigma_n^2 = 1000$). **(c)** CLEAN + A-projection ($N_s = 800$, $\sigma_n^2 = 1000$).

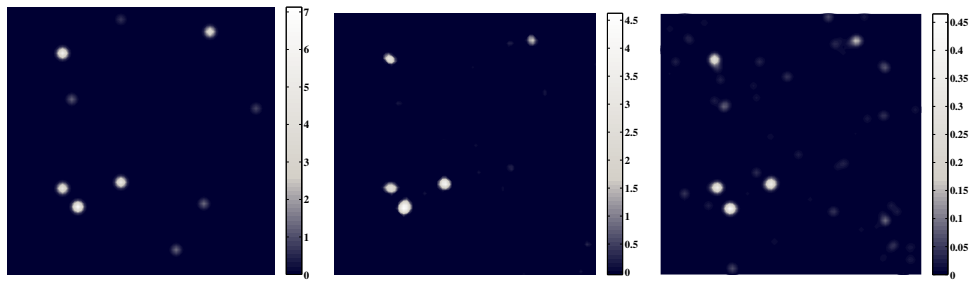


(d) Root mean squared error for both imaging pipelines as a function of the number of samples. The noise level is set at $\sigma_n^2 = 1000$.

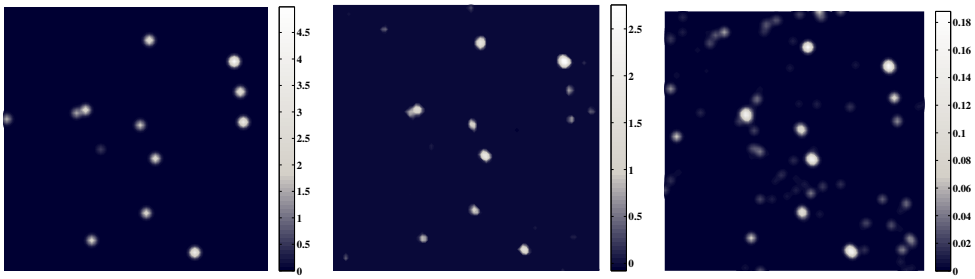


(e) Root mean squared error for both imaging pipelines as a function of the noise level. The number of samples is set at $N_s = 800$.

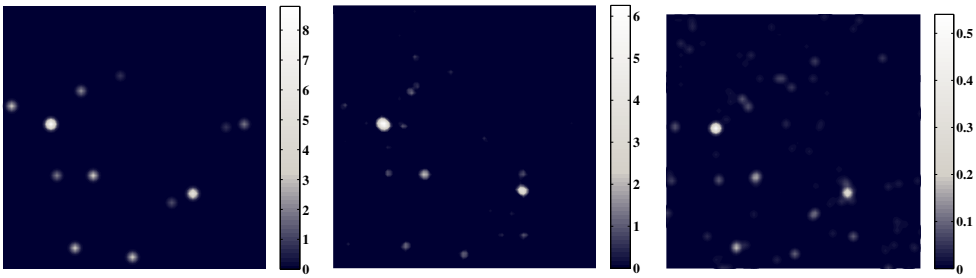
Figure 5.4: Accuracy of the sky estimates obtained with both imaging pipeline for various experimental conditions. We observe that the sky estimates obtained with the new imaging pipeline are more accurate and less sensitive to the noise (if we choose the root mean squared error as a measure of the quality of the sky estimate). Moreover they have less false positives then the sky estimates obtained with CLEAN and the A-projection algorithm.



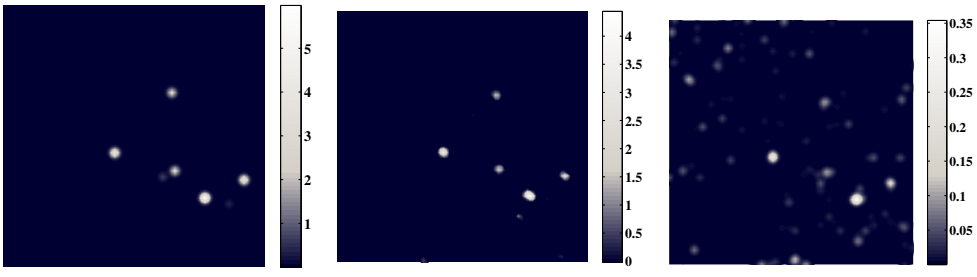
(a) True sky image. **(b)** Gram-Schmidt + LASSO. **(c)** CLEAN + A-projection.



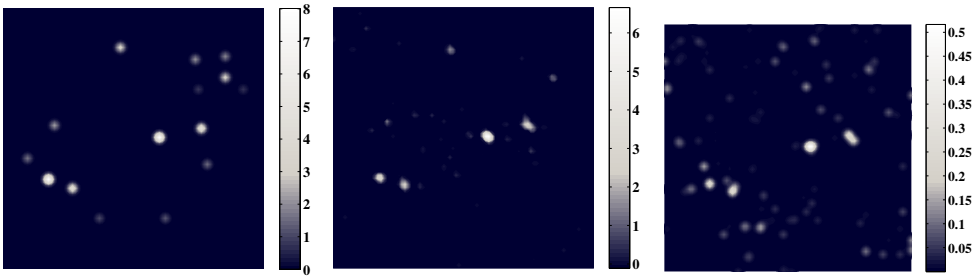
(d) True sky image. **(e)** Gram-Schmidt + LASSO. **(f)** CLEAN + A-projection.



(g) True sky image. **(h)** Gram-Schmidt + LASSO. **(i)** CLEAN + A-projection.



(j) True sky image. **(k)** Gram-Schmidt + LASSO. **(l)** CLEAN + A-projection.



(m) True sky image. **(n)** Gram-Schmidt + LASSO. **(o)** CLEAN + A-projection.

Figure 5.5: Accuracy of the sky estimates obtained with both imaging pipeline for various skies. We observe that the sky estimates obtained with the new imaging pipeline are more accurate (in terms of absolute intensities) and less polluted by false positives than the estimates obtained with the classical imaging pipeline. For all the experiments, we chose $N_s = 800$ and $\sigma_n^2 = 500$.

6

Conclusions

We started off by investigating how to integrate randomized beamforming into the existing imaging chain for radio interferometry. This was prompted by a criticism that the technique was fine but there was no way to place the correlation output onto the “uv-plane" (Fourier domain). By studying the frequency response of the instrument, we noticed that there was no way either for de-facto beamforming. This explains why striving to retain the appealing Fourier domain has resulted in cumbersome work-arounds, which also turn out to be less accurate and introduce computational overhead.

It became attractive to examine the problem from an orthogonal perspective, literally. This amounts to a QR-decomposition of the system to result in a more natural basis. The whole chain is then more intuitive, linear, and flexible. It allows, for example, in statistical confidence in the resultant image, and incorporation of general beamforming.

The QR-decomposition is currently over-sampled, and does not exploit redundancy in calculation across the basis elements and observation times. Yet it still manages to outperform the state-of-the-art for the LOFAR telescope, and many scenarios for SKA1. We are quite confident that with optimization, the technique will perform much faster than presently implemented.

Another possible direction for future work would be to modify the beamforming to help in the orthogonalization step. Finally, orthogonalization at an earlier stage, for complexity reasons, on the antenna time-series samples is also a promising avenue of research.

Acknowledgments

I would like to express my sincere gratitude to my academic advisors Profs. Victor Panaretos and Martin Vetterli, as well as to the entire DOME WP6 team in IBM Zurich, for having offered me the incredible opportunity to work on that passionating subject.

On the EPFL side, I would like to thank specifically Victor for his brilliant guidance and very insightful advice throughout the thesis.

On the IBM side, my warmest greetings go to Dr. Paul Hurley, for his day-to-day support and his incommensurable help, as much for technical inquiries than for the actual writing of this thesis. In addition to his precious advice, he has always been very available and enthusiastic about my work, which I sincerely appreciated. It was very pleasing for me to work among the DOME WP6 team, in such an stimulating

environment. A special thought goes to Dr. Sanaz Kazemi, that welcomed me and greatly helped me during my stay in ASTRON (the Netherlands Institute for Radio Astronomy).

Bibliography

[1] Abhishek Awasthi, Rohit Guttal, Naofal Al-Dhahir, and Poras T Balsara. Complex QR decomposition using fast plane rotations for MIMO applications. *Communications Letters, IEEE*, 18(10):1743–1746, 2014.

[2] 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.

[3] S Bhatnagar, U Rau, and K Golap. Wide-field wide-band interferometric imaging: The WB A-projection and hybrid algorithms. *The Astrophysical Journal*, 770(2):91, 2013.

[4] Tim J Cornwell, Kumar Golap, and Sanjay Bhatnagar. The noncoplanar baselines effect in radio interferometry: The W-projection algorithm. *Selected Topics in Signal Processing, IEEE Journal of*, 2(5):647–657, 2008.

[5] Marco De Vos, Andre W Gunst, and Ronald Nijboer. The LOFAR telescope: System architecture and signal processing. *Proceedings of the IEEE*, 97(8):1431–1437, 2009.

[6] P Dewdney, W Turner, R Millenaar, R McCool, J Lazio, and T Cornwell. Ska1 system baseline design. *Document number SKA-TEL-SKO-DD-001 Revision*, 1, 2013.

[7] Peter E Dewdney, Peter J Hall, Richard T Schilizzi, and T Joseph LW Lazio. The square kilometre array. *Proceedings of the IEEE*, 97(8):1482–1496, 2009.

[8] Olive Jean Dunn. Estimation of the medians for dependent variables. *The Annals of Mathematical Statistics*, pages 192–197, 1959.

[9] Hugh Garsden, JN Girard, Jean-Luc Starck, S Corbel, C Tasse, A Woiselle, JP McKean, AS van Amesfoort, J Anderson, IM Avruch, et al. LOFAR sparse image reconstruction. *Astronomy & Astrophysics*, 575:A90, 2015.

[10] JA Högbom. Aperture synthesis with a non-regular distribution of interferometer baselines. *Astronomy and Astrophysics Supplement Series*, 15:417, 1974.

[11] Steven G Johnson and Matteo Frigo. A modified split-radix fft with fewer arithmetic operations. *Signal Processing, IEEE Transactions on*, 55(1):111–119, 2007.

[12] Douglas L Jones. Decimation-in-time (dit) radix-2 fft. *The Connexions Project. http://cnx. org*, 2006.

[13] Robb J Muirhead. *Aspects of multivariate statistical theory*, volume 197. John Wiley & Sons, 2009.

[14] Orhan Ocal. Data Reduction Algorithms for Radio Astronomy Antenna Stations. Master's thesis, EPFL, Lausanne, Switzerland, 2014.

[15] Orhan Ocal, Paul Hurley, Giovanni Cherubini, and Sanaz Kazemi. Collaborative randomized beamforming for phased array radio interferometers. *arXiv preprint arXiv:1411.4002*, 2014.

[16] Björn Ottersten, Petre Stoica, and Richard Roy. Covariance matching estimation techniques for array signal processing applications. *Digital Signal Processing*, 8(3):185–210, 1998.

[17] UJ Schwarz. Mathematical-statistical description of the iterative beam removing technique (method CLEAN). *Astronomy and Astrophysics*, 65:345, 1978.

[18] Elias M Stein and Guido L Weiss. *Introduction to Fourier analysis on Euclidean spaces*, volume 1. Princeton university press, 1971.

[19] Cyril Tasse, S van der Tol, J van Zwieten, Ger van Diepen, and S Bhatnagar. Applying full polarization A-Projection to very wide field of view instruments: An imager for LOFAR. *Astronomy & Astrophysics*, 553:A105, 2013.

[20] Greg B Taylor, Chris Luke Carilli, and Richard A Perley. Synthesis imaging in radio astronomy II. In *Synthesis Imaging in Radio Astronomy II*, volume 180, 1999.

[21] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[22] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

[23] Alle-Jan van der Veen and Stefan J Wijnholds. Signal processing tools for radio astronomy. In *Handbook of Signal Processing Systems*, pages 421–463. Springer, 2013.

[24] MCH Wright. A model for the ska. Technical report, SKA memo 16, 2002.

[25] Sarod Yatawatta. LOFAR beamshapes and their use in calibration and imaging. Technical report, ASTRON, Tech. Report, 2007., 2009.

[26] Ya-xiang Yuan. A short note on the Q-linear convergence of the steepest descent method. *Mathematical programming*, 123(2):339–343, 2010.