P9 Distributed Image Reconstruction for the new Radio Interferometers

Jonas Schwammberger

June 28, 2019

Abstract

Contents

1 Introduction

1.1 Radio interferometry system

This project is focused on distributing image Reconstruction for radio interferometers, which is only one of three steps in the pipeline from measurements to the final image. We give a quick overview over the whole pipeline in figure 1 and how Radio Interferometers work in principle: The antennas observe the arriving electromagnetic wave, gets processed in three steps, correlation, calibration and image reconstruction.

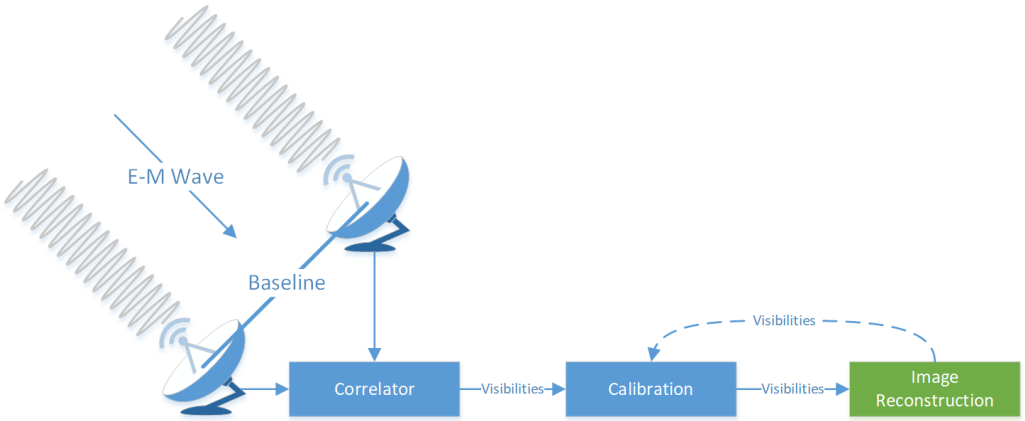


Figure 1: Radio interferometer system

First, the electromagnetic wave gets measured by the different antennas of the interferometer. The measurements of each antenna pair get correlated into a Fourier component (called Visibility in Radio Astronomy). Each antenna pair measures a complex-valued, noisy visibility of the sky. The distance and orientation of the antenna pair relative to the incoming signal, called the baseline, dictates which visibility gets measured. The longer the baseline the higher-order visibility gets measured, resulting in a higher angular resolution. The image 2a shows the sampled visibilities in the Fourier space of the MeerKAT interferometer. Every dot is a single measurement. After correlation, the visibility data is saved for later processing.

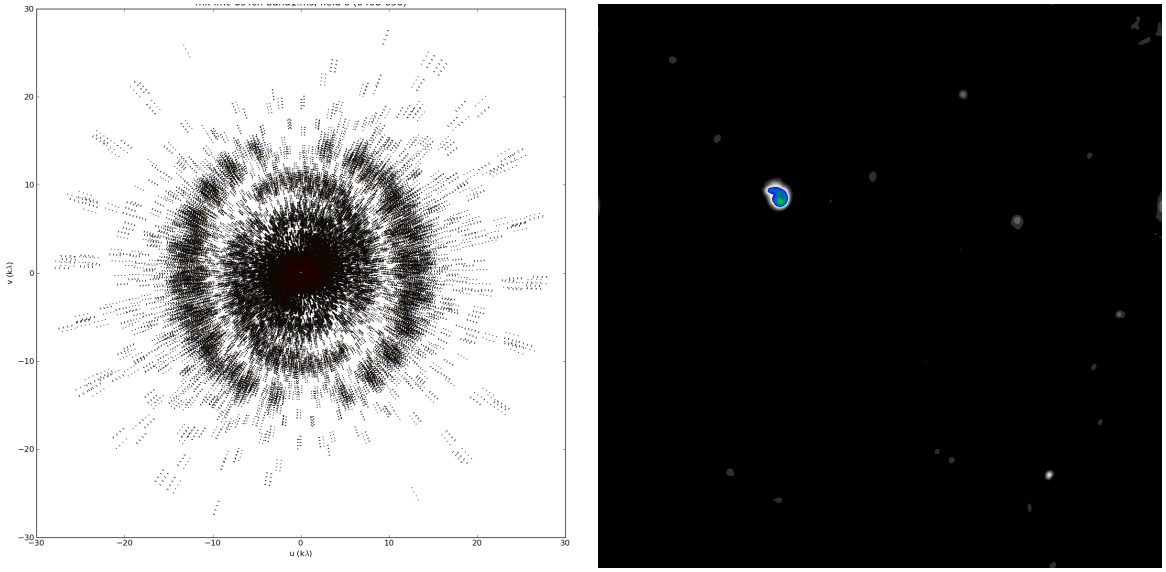
The calibration step is done after all visibility data has been recorded. This step corrects the amplitude and phase of the measurements for varying antenna sensitivities, pointing errors and other effects. Also, this step removes corrupted data from the measurements. After the calibration step, the visibilities still contain noise.

The last step is responsible for reconstructing an image from the calibrated, noisy visibilities. The figure 2 shows a real-world example of a reconstruction from the MeerKAT radio interferometer. It arrives at the reconstructed image by inverting the measurement equation, and deciding which part of the visibilities is noisy, and which part is the signal. These two problems, handling the noise and inverting the measurement equation, are central to image reconstruction for radio interferometers. They influence both the quality and the runtime costs of the reconstruction.

1.1.1 The measurement equation

The measurement equation models how the electromagnetic wave gets distorted on its path from the celestial source through the ionosphere and finally through the antenna of the interferometer[1]. It abstracts all effects we wish to correct in the image reconstruction in one equation. As such, there is no single unified measurement equation for all interferometers, and generally depends on the instrument.

In this project we use the measurement equation (1.1). It consists of three parts: The visibility measurements



(a) Measurements in the Fourier space. *(b) Reconstructed image.*

Figure 2: Example of an image reconstruction for visibility measurements of the MeerKAT radio interferometer

$V(u,v,w)$[1], the observed image with a normalization factor $\frac{I(x,y)}{c(x,y)}$ and the Fourier Transform $e^{2\pi i[...]}$. u v and w represent the axes in the Fourier Domain, while the x and y axes represent the angles away from the image center. A pixel in $I(x,y)$ represents how much radio emission was measured from a the direction x, y. The image 2a shows an example for $V()$, while 2b shows an example for $I()$.

$$V(u,v,w) = \int\int \frac{I(x,y)}{c(x,y)} e^{2\pi i[ux+vy+w(c(x,y)-1)]}\, dx\, dy\,, \quad c(x,y) = \sqrt{1-x^2-y^2} \tag{1.1}$$

The radio interferometer essentially observes the sky in the Fourier domain. If we want to retrieve the observed sky image $I()$, all we need to do in theory is calculate the inverse Fourier Transform. Note however that the visibilities $V(u,v,w)$ are three dimensional, while the image $I(x,y)$ only has two. Also note that the third component w only depends on the directions x and y. In a sense, the Visibilities $V()$ and the image $I()$ have a two dimensional Fourier relationship ($V(u,v,w) = \int\int I(x,y)e^{2\pi i[ux+vy]}\, dx\, dy$), but with a directionally dependent correction factor $e^{2\pi i[...+w(c(x,y)-1)]}$.

The third component w is an example of a Directionally Dependent Effect (DDE) which have a tendency to increase the runtime costs of the image reconstruction. The w-component keeps us from using the Fast Fourier Transform (FFT) for the measurement equation (1.1). Research in this area tries to use approximations which lets us use faster algorithms like the FFT, and correct for DDE's accurately enough [2, 3, 4]. In this project, the w-correction is the only DDE we handle.

1.2 System of linear equations

Even though the Fourier Transform in the measurement equation (1.1) contains a w-correction factor, it is still linear. Even with more complex measurement equations, the relationship between the visibilities $V()$ and the $I()$ remain linear[5]. If we ignore the noise for the sake of demonstration, this means we can represent

[1] V in the equation (1.2) is a vector. We use the lower-case v to denote the axis in the Fourier space uvw, and the upper-case letter to denote the visibility vector.

the measurement equation as a system of linear equations (1.2), where F is the Fourier Transform with w-correction, x is the image we are searching, V are the calibrated visibilities. To reconstruct the image x, we simply need to search for a solution to (1.2).

$$\underset{x}{find} \quad V - Fx = 0 \tag{1.2}$$

Even though we generally have more visibilities than pixels in the reconstruction, which makes (1.2) overdetermined, there is no unique solution to (1.2). There are potentially many candidate images that solve (1.2). The actually observed image is among the candidates, but from the measurements alone we cannot decide which it is. This makes the image reconstruction for radio interferometers an ill-posed inverse problem.

The reason why there is no unique solution to (1.2) lies in the visibility measurements V. When we look back at figure 2a, it is clear to see that the interferometer does not sample the visibilities in a uniform way. There are regions with a high sample density. The density decreases when we move further away from the center. In short, the radio interferometer measures an incomplete set of visibilities. It does not contain all the data we require to reconstruct the image, which makes finding the image an ill-posed inverse problem.

1.2.1 The theory of compressed sensing

$$\underset{x}{minimize} \ \|V - Fx\|_2^2 + \lambda P(x) \tag{1.3}$$

1.3 Image reconstruction in practice

We know how to solve the ill-posed image reconstruction problem in theory. We formulate a minimization problem (1.3), specify a prior function that capture our prior knowledge, and find the optimum image with an appropriate optimization algorithm. In practice however we have a hard time representing the dense Fourier Transform matrix F in the equation (1.3). It is the size of number of visibilities times pixels in the reconstruction. Even older radio interferometers easily produce several million visibilities, with a million pixels in the reconstructed image. We cannot represent F explicitly. However, we can reformulate the image reconstruction (1.3) as a deconvolution problem, which reduces the dimensions of the measurement V and removes the large Fourier transform matrix F from the minimization objective.

1.3.1 Reformulating as a deconvolution problem

The Fourier transform matrix F in (1.3) is a product of two operations: The Fourier transform F, and the masking operator M. The masking operator is a matrix with zero entries for all Fourier components invisible to the instrument. So far, M was implicitly contained in F of (1.3). To derive the deconvolution problem, we represent M explicitly. For the sake of demonstration, let us assume our visibility measurements $V(u, v, w)$ lie on a discrete grid. $V(u, v, w)$ is zero for all components that the interferometer could not measure. We then can represent the transformation from image to visibility space with the Fourier transform F followed by a masking operation M, and we arrive at the image reconstruction problem (1.4). This is identical to (1.3), except for the factorization of F.

$$\text{original:} \quad \underset{x}{minimize} \ \|V - MFx\|_2^2 + \lambda P(x) \tag{1.4}$$

$$\text{in-painting:} \quad \underset{V_2}{minimize} \ \|V - MV_2\|_2^2 + \lambda P(F^{-1}V_2) \tag{1.5}$$

$$\text{deconvolution:} \quad \underset{x}{minimize} \ \|I_{dirty} - x * PSF\|_2^2 + \lambda P(x) \tag{1.6}$$

Note that M represents the degradation, the corruption introduced by incomplete sampling in the visibility space. M is the important operator. The measurements V, or the reconstructed image x can be in any space we wish. For example, we do not actually need to reconstruct the image in image space. In theory, we can reformulate an equivalent problem (1.5), in which we in-paint the missing visibilities. Or, we can use the Fourier transform on the visibility measurements V and the masking operator M, which leads us to the deconvolution problem (1.6).

Since the deconvolution formulation is vital for the major/minor cycle architecture, we have a closer look at (1.6). The effect of incomplete sampling in Fourier space is equal to a convolution with a Point Spread Function PSF in image space. I.e. $PSF = F^{-1}M$. The measurements are now represented as the ”dirty” image, $I_{dirty} = F^{-1}V$. We try to find the deconvolved image x, while only knowing the convolution kernel PSF and the convolved, dirty image I_{dirty}. This is still an ill-posed inverse problem. We have potentially many different deconvolutions that fit the dirty image, and we search the most likely candidate according to some prior $P(x)$.

The deconvolution (1.6) and the original image reconstruction problem (1.3) are equivalent. Both arrive at the same result. But the deconvolution problem is easier to handle in practice: I_{dirty} and PSF are generally more compact representations of V and M. There is one last issue: Calculating the dirty image from the measurements ($I_{dirty} = F^{-1}V$) again needs the impractically large Fourier transform matrix F. This is solved in the major/minor cycle algorithm.

1.3.2 The major/minor cycle

The major/minor cycle architecture shown in figure 3 consists of two parts: The minor cycle, which iteratively deconvolves the dirty image with the PSF (it minimizes (1.6)). The major cycle is responsible for estimating the dirty image. It consists of two steps: the gridding and the Fast Fourier Transformation (FFT).

1.3.3 Approximations under the major cycle

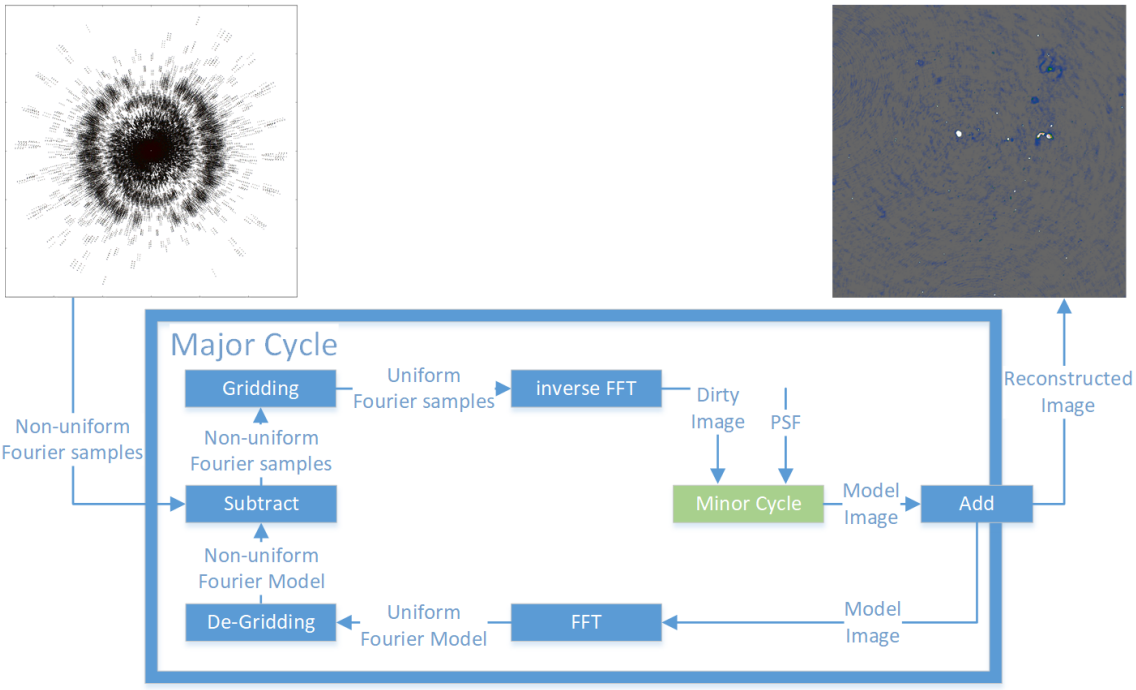


Figure 3: The Major/Minor Cycle Architecture

2 State of the art Image Reconstruction

2.1 Gridder

2.1.1 -stacking

2.1.2 IDG

2.2 Reconstruction

2.2.1 CLEAN

2.2.2 SARA

3 Simple distributed image reconstruction

Show architecture

Different architectures, why don't we do consensus algorithms.

Message Passing Interface (MPI)

Gridding and Devonvolution

We use the recently developed Image Domain Gridder (IDG)

3.1 The Image Domain Gridder

What the gridding actually does. In theory, just an interpolation. Interpolation kernel like Spheroidal, Kaiser-Bessel functions etc. Generally the most intensive step, because the number of grid cells is generally a lot smaller than the number calibrated Visibilities.

However, here we move from the three dimensional Visibilities to two dimensions. DDE's fall down in the area of the gridder.

So how we do an interpolation

Veeneboer et al[2] developed the Image Domain Gridder. It uses Subgrids and solves each subgrid seperately. It is in the image domain, because it can do Radio Interferometer specific corrections efficiently. Furthermore, it leads to a structure which is primed for GPU processing. We use this algorithm to distribute the gridding.

W-Projection, Spheroidal are convolutions in the Fourier space.

The figure 4 shows the different parts of the image domain algorithm.

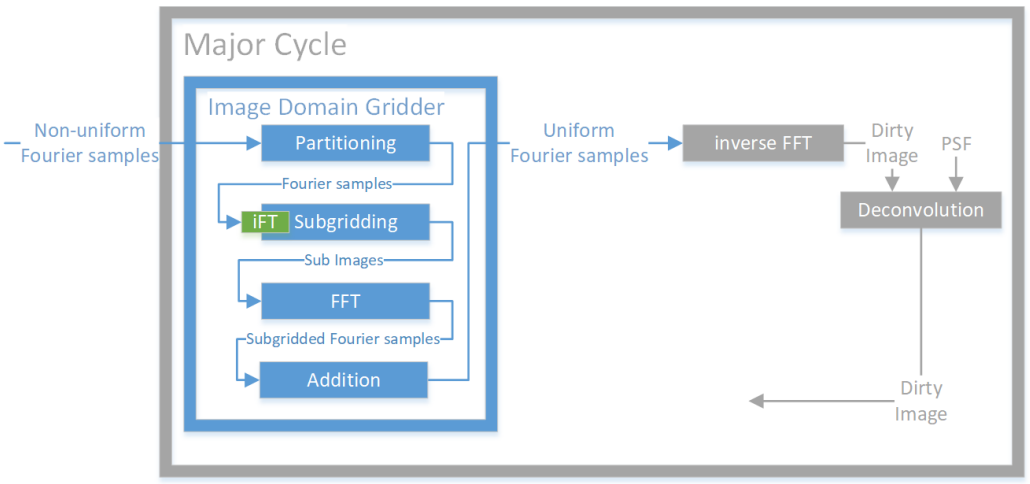


Figure 4: Image Domain Gridder in the Major Cycle Architecture

Algorithm

3.2 Coordinate Descent deconvolution

Coordinate Descent Algorithm why

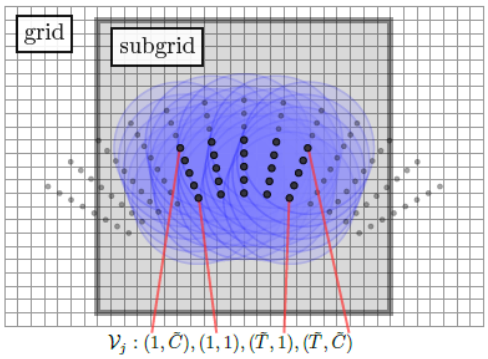


Figure 5: Subgrid

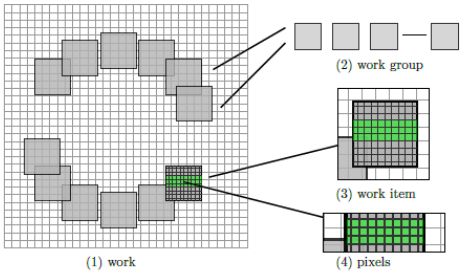


Figure 6: parallel

Variants

Implementation Correlate the dirty image Find max How to calculate a

3.2.1 ElasticNet Regularization

Why

Formula

Effect

Implementation

May even speed up convergence for correlated pixel values compared to L1 or L2[6]. But was not investigated in this project

3.3 Major Cycle convergence

Putting it all together

We have the Minor Cycle, which is easy to converge.

Coordinate Descent Path optimization [6] Danger that CD takes too many pixel into a Major Cycle. Lower bound per iteration, PSF sidelobe can still be too low, danger when many psf sidelobes overlap

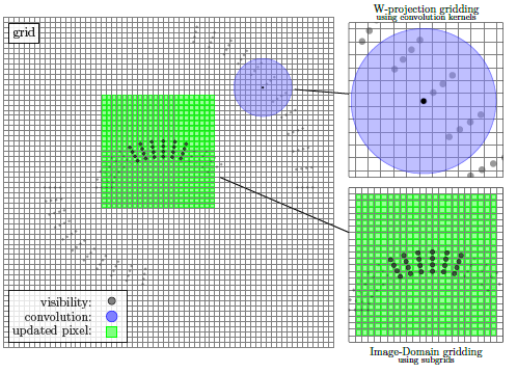
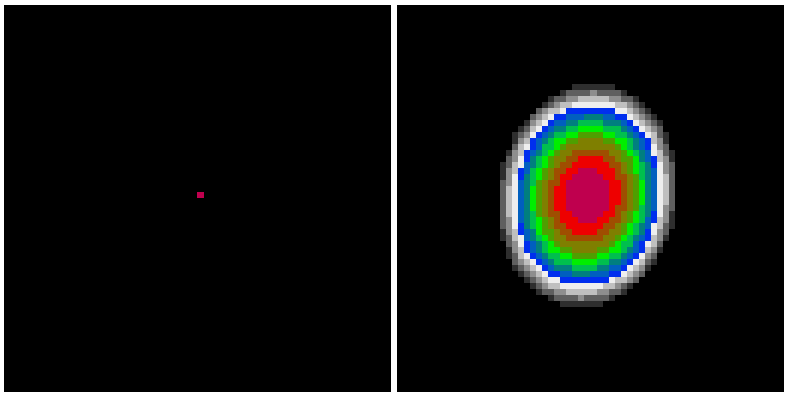


Figure 7: Image Domain Gridder in the Major Cycle Architecture



(a) Effect of the pure L1 norm (λ = 1.0) on a single point source.

(b) Effect of the pure L2 norm (λ = 1.0) on a single point source.

Figure 8: Effect of the L1 and L2 Norm separately.

3.4 Test on MeerKAT data

4 Conclusion

References

[1] Oleg M Smirnov. Revisiting the radio interferometer measurement equation-i. a full-sky jones formalism. Astronomy & Astrophysics, 527:A106, 2011.

[2] Bram Veenboer, Matthias Petschow, and John W Romein. Image-domain gridding on graphics processors. In 2017 IEEE International Parallel and Distributed Processing Symposium (IPDPS), pages 545–554. IEEE, 2017.

[3] AR Offringa, Benjamin McKinley, Natasha Hurley-Walker, FH Briggs, RB Wayth, DL Kaplan, ME Bell, Lu Feng, AR Neben, JD Hughes, et al. Wsclean: an implementation of a fast, generic wide-field imager for radio astronomy. Monthly Notices of the Royal Astronomical Society, 444(1):606–619, 2014.

[4] Luke Pratley, Melanie Johnston-Hollitt, and Jason D McEwen. A fast and exact w-stacking and w-projection hybrid algorithm for wide-field interferometric imaging. arXiv preprint arXiv:1807.09239, 2018.

[5] Oleg M Smirnov. Revisiting the radio interferometer measurement equation-i. a full-sky jones formalism. Astronomy & Astrophysics, 527:A106, 2011.

[6] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. Journal of statistical software, 33(1):1, 2010.

[7] Arwa Dabbech, Chiara Ferrari, David Mary, Eric Slezak, Oleg Smirnov, and Jonathan S Kenyon. More-sane: Model reconstruction by synthesis-analysis estimators-a sparse deconvolution algorithm for radio interferometric imaging. Astronomy & Astrophysics, 576:A7, 2015.

List of Figures

List of Tables

5 attachment

6 Larger runtime costs for Compressed Sensing Reconstructions

The MeerKAT instrument produces a new magnitude of data volume. An image with several million pixels gets reconstructed from billions of Visibility measurements. Although MeerKAT measures a large set of Visibilities, the measurements are still incomplete. We do not have all the information available to reconstruct an image. Essentially, this introduces ”fake” structures in the image, which a reconstruction algorithm has to remove. Additionally, the measurements are noisy.

We require an image reconstruction algorithm which removes the ”fake” structures from the image, and removes the noise from the measurements. The large data volume of MeerKAT requires the algorithm to be both scalable and distributable. Over the years, several reconstruction algorithms were developed, which can be separated into two classes: Algorithms based on CLEAN, which are cheaper to compute and algorithms based on Compressed Sensing, which create higher quality reconstructions.

CLEAN based algorithms represent the reconstruction problem as a deconvolution. First, they calculate the ”dirty” image, which is corrupted by noise and fake image structures. The incomplete measurements essentially convolve the image with a Point Spread Function (PSF). CLEAN estimates the PSF and searches for a deconvolved version of the dirty image. In each CLEAN iteration, it searches for the highest pixel in the dirty image, subtracts a fraction PSF at the location. It adds the fraction to the same pixel location of a the ”cleaned” image. After several iterations, the cleaned image contains the deconvolved version of the dirty image. CLEAN accounts for noise by stopping early. It stops when the highest pixel value is smaller than a certain threshold. This results in a light-weight and robust reconstruction algorithm. CLEAN is comparatively cheap to compute, but does not produce the best reconstructions and is difficult to distribute on a large scale.

Compressed Sensing based algorithms represent the reconstruction as an optimization problem. They search for the optimal image which is as close to the Visibility measurements as possible, but also has the smallest regularization penalty. The regularization encodes our prior knowledge about the image. Image structures which were likely measured by the instrument result in a low regularization penalty. Image structures which were likely introduced by noise or the measurement instrument itself result in high penalty. Compressed Sensing based algorithms explicitly handle noise and create higher quality reconstructions than CLEAN. State-of-the-art Compressed Sensing algorithms show potential for distributed computing. However, they currently do not scale on MeerKATs data volume. They require too many computing resources compared to CLEAN based algorithms.

This project searches for a way to reduce the runtime costs of Compressed Sensing based algorithms. One reason for the higher costs is due to the non-uniform FFT Cycle. State-of-the-art CLEAN and Compressed Sensing based algorithms both use the non-uniform FFT approximation in a cycle during reconstruction. The interferometer measures the Visibilities in a continuous space in a non-uniform pattern. The image is divided in a regularly spaced, discrete pixels. The non-uniform FFT creates an approximate, uniformly sampled image from the non-uniform measurements. Both, CLEAN and Compressed Sensing based algorithms use the non-uniform FFT to cycle between non-uniform Visibilities and uniform image. However, a Compressed Sensing algorithm requires more non-uniform FFT cycles for reconstruction.

CLEAN and Compressed Sensing based algorithms use the non-uniform FFT in a similar manner. However, there are slight differences in the architecture. This project hypothesises that The previous project searched for an alternative to the non-uniform FFT cycle. Although there are alternatives, there is currently no replacement which leads to lower runtime costs for Compressed Sensing. Current research is focused on reducing the number of non-uniform FFT cycles for Compressed Sensing algorithms.

CLEAN based algorithms use the Major Cycle Architecture for reconstruction. Compressed Sensing based algorithms use a similar architecture, but with slight modifications. Our hypothesis is that we may reduce the number of non-uniform FFT cycles for Compressed Sensing by using CLEAN’s Major Cycle Architecture.

6.1 CLEAN: The Major Cycle Architecture

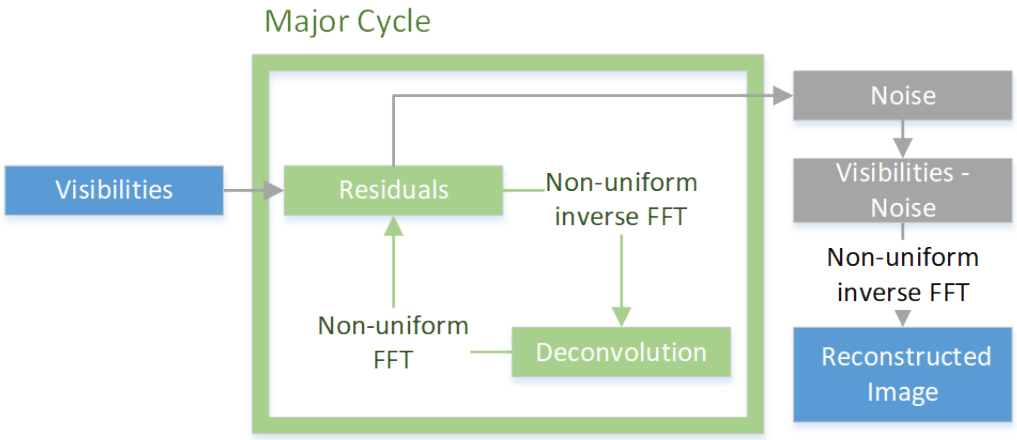


Figure 9: The Major Cycle Architecture

Figure 9 depicts the Major Cycle Architecture used by CLEAN algorithms. First, the Visibilities get transformed into an image with the non-uniform FFT. The resulting dirty image contains the corruptions of the measurement instrument and noise. A deconvolution algorithm, typically CLEAN, removes the corruption of the instrument with a deconvolution. When the deconvolution stops, it should have removed most of the observed structures from the dirty image. The rest, mostly noisy part of the dirty image gets transformed back into residual Visibilities and the cycle starts over.

In the Major Cycle Architecture, we need several deconvolution attempts before it has distinguished the noise from the measurements. Both the non-uniform FFT and the deconvolution are approximations. By using the non-uniform FFT in a cycle, it can reconstruct an image at a higher quality. For MeerKAT reconstruction with CLEAN, we need approximately 4-6 non-uniform FFT cycles for a reconstruction.

6.2 Compressed Sensing Architecture

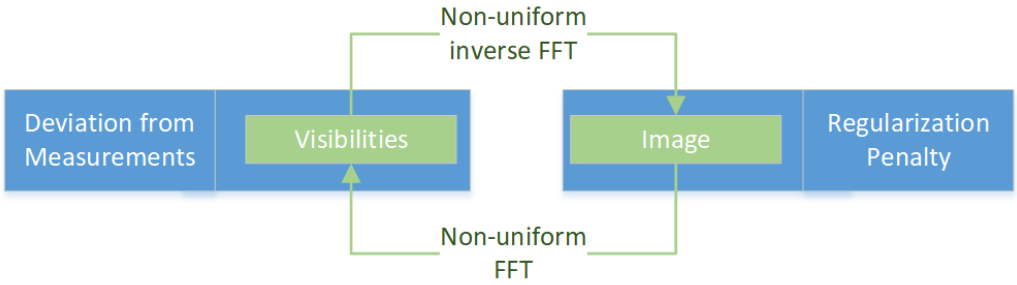


Figure 10: State-of-the-art Compressed Sensing Reconstruction Architecture

Figure 10 depicts the architecture used by Compressed Sensing reconstructions. The Visibilities get transformed into an image with the non-uniform FFT approximation. The algorithm then modifies the image so it reduces the regularization penalty. The modified image gets transformed back to Visibilities and the algorithm then minimizes the difference between measured and reconstructed Visibilities. This is repeated until the algorithm converges to an optimum.

In this architecture, state-of-the-art Compressed Sensing algorithms need approximately 10 or more non-uniform FFT cycles to converge. It is one source for the higher runtime costs. For MeerKAT reconstructions

the non-uniform FFT tends to dominate the runtime costs. A CLEAN reconstruction with the Major Cycle Architecture already spends a large part of its time in the non-uniform FFT. Compressed Sensing algorithms need even more non-uniform FFT cycle on top of the ”Image Regularization” step being generally more expensive than CLEAN deconvolution. There is one upside in this architecture: State-of-the-art algorithms managed to distribute the ”Image Regularization” operation.

6.3 Hypothesis for reducing costs of Compressed Sensing Algorithms

Compressed Sensing Algorithms are not bound to the Architecture presented in section 6.2. For example, we can design a Compressed Sensing based deconvolution algorithm and use the Major Cycle Architecture instead.

Our hypothesis is: We can create a Compressed Sensing based deconvolution algorithm which is both distributable and creates higher quality reconstructions than CLEAN. Because it also uses the Major Cycle architecture, we reckon that the Compressed Sensing deconvolution requires a comparable number of non-uniform FFT cycles to CLEAN. This would result in a Compressed Sensing based reconstruction algorithm with similar runtime costs to CLEAN, but higher reconstruction quality and higher potential for distributed computing.

6.4 State of the art: WSCLEAN Software Package

6.4.1 W-Stacking Major Cycle

6.4.2 Deconvolution Algorithms

CLEAN MORESANE

6.5 Distributing the Image Reconstruction

6.5.1 Distributing the Non-uniform FFT

6.5.2 Distributing the Deconvolution

7 Handling the Data Volume

The new data volume is a challenge to process for both algorithms and computing infrastructure. Push for parallel and distributed algorithms. For Radio Interferometer imaging, we require specialized algorithms. The two distinct operations, non-uniform FFT and Deconvolution, were difficult algorithms for parallel or distributed computing.

The non-uniform FFT was historically what dominated the runtime []. Performing an efficient non-uniform FFT for Radio Interferometers is an active field of research[3, 4], continually reducing the runtime costs of the operation. Recently, Veeneboer et al[2] developed a non-uniform FFT which can be fully executed on the GPU. It speeds up the most expensive operation.

In Radio Astronomy, CLEAN is the go-to deconvolution algorithm. It is light-weight and compared to the non-uniform FFT, a cheap algorithm. It is also highly iterative, which makes it difficult for effective parallel or distributed implementations. However, compressed sensing based deconvolution algorithms can be developed with distribution in mind.

7.1 Fully distributed imaging algorithm

Current imaging algorithms push towards parallel computing with GPU acceleration. But with Veeneboer et al's non-uniform FFT and a compressed sensing based deconvolution, we can go a step further and create a distributed imaging algorithm.

8 Image Reconstruction for Radio Interferometers

In Astronomy, instruments with higher angular resolution allows us to measure ever smaller structures in the sky. For Radio frequencies, the angular resolution is bound to the antenna dish diameter, which puts practical and financial limitations on the highest possible angular resolution. Radio Interferometers get around this limitation by using several smaller antennas instead. Together, they act as a single large antenna with higher angular resolution at lower financial costs compared to single dish instruments.

Each antenna pair of an Interferometer measures a single Fourier component of the observed image. We can retrieve the image by calculating the Fourier Transform of the measurements. However, since the Interferometer only measures an incomplete set of Fourier components, the resulting image is ”dirty”, convolved with a Point Spread Function (PSF). Calculating the Fourier Transform is not enough. To reconstruct the from an Interferometer image, an algorithm has to find the observed image with only the dirty image and the PSF as input. It has to perform a deconvolution. The difficulty lies in the fact that there are potentially many valid deconvolutions for a single measurement, and the algorithm has to decide for the most likely one. How similar the truly observed image and the reconstructed images are depends largely on the deconvolution algorithm.

State-of-the-art image reconstructions use the Major Cycle architecture (shown in Figure 11), which contains three operations: Gridding, FFT and Deconvolution.

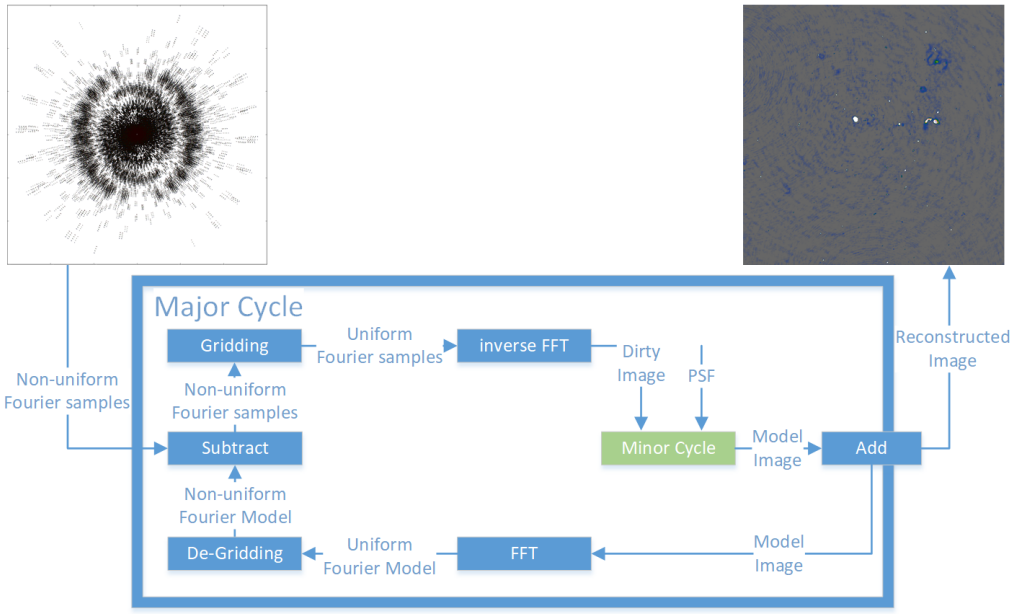


Figure 11: The Major Cycle Architecture of image reconstruction algorithms

The first operation in the Major Cycle, Gridding, takes the non-uniformly sampled Fourier measurements from the Interferometer and interpolates them on a uniformly spaced grid. The uniform grid lets us use FFT to calculate the inverse Fourier Transform and we arrive at the dirty image. A deconvolution algorithm takes the dirty image plus the PSF as input, producing the deconvolved ”model image”, and the residual image as output. At this point, the reverse operations get applied to the residual image. First the FFT and then De-gridding, arriving at the non-uniform Residuals. The next Major Cycle begins with the non-uniform Residuals as input. The cycles are necessary, because the Gridding and Deconvolution operations are only approximations. Over several cycles, we reduce the errors introduced by the approximate Gridding and Deconvolution. The final, reconstructed image is the addition of all the model images of each Major Cycle.

8.1 Distributed Image Reconstruction

New Interferometer produce an ever increasing number of measurements, creating ever larger reconstruction problems. A single image can contain several terabytes of Fourier measurements. Handling reconstruction problems of this size forces us to use distributed computing. However, state-of-the-art Gridding and Deconvolution algorithms only allow for limited distribution. How to scale the Gridding and Deconvolution algorithms to large problem sizes is still an open question.

Recent developments make a distributed Gridder and a distributed Deconvolution algorithm possible. Veeneboer et al[2] found an input partitioning scheme, which allowed them to perform the Gridding on the GPU. The same partitioning scheme can potentially be used to distribute the Gridding onto multiple machines. For Deconvolution, there exist parallel implementations for certain algorithms like MORESANE[7]. These can be used as a basis for a fully distributed image reconstruction.

In this project, we want to make the first steps towards an image reconstruction algorithm, which is distributed from end-to-end, from Gridding up to and including deconvolution. We create our own distributed Gridding and Deconvolution algorithms, and analyse the bottlenecks that arise.

8.2 First steps towards a distributed Algorithm

In this project, we make the first steps towards a distributed Major Cycle architecture (shown in figure 11) implemented C#. We port Veeneboer et al's Gridder, which is written in C++, to C# and modify it for distributed computing. We implement a simple deconvolution algorithm based on the previous project and create a first, non-optimal distributed version of it.

In the next step, we create a more sophisticated deconvolution algorithm based on the shortcomings of the first implementation. We use simulated and real-world observations of the MeerKAT Radio Interferometer and measure its speed up. We identify the bottlenecks of the current implementation and explore further steps.

From the first lessons, we continually modify the distributed algorithm and focus on decreasing the need for communication between the nodes, and increase the overall speed up compared to single-machine implementations. Possible Further steps:

- Distributed FFT
- Replacing the Major Cycle Architecture
- GPU-accelerated Deconvolution algorithm.

A state-of-the-art reconstruction algorithm has to correct large number of measurement effects arising from the Radio Interferometer. Accounting for all effects is out of the scope for this project. We make simplifying assumptions, resulting in a proof-of-concept algorithm.

9 Ehrlichkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende schriftliche Arbeit selbstständig und nur unter Zuhilfenahme der in den Verzeichnissen oder in den Anmerkungen genannten Quellen angefertigt habe. Ich versichere zudem, diese Arbeit nicht bereits anderweitig als Leistungsnachweis verwendet zu haben. Eine Überprüfung der Arbeit auf Plagiate unter Einsatz entsprechender Software darf vorgenommen werden.
Windisch, June 28, 2019

Jonas Schwammberger