P9 Distributed Image Reconstruction for the new Radio Interferometers

Jonas Schwammberger

April 24, 2019

Abstract

Contents

1 Introduction

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.

New Radio Interferometers are built Higher sensitivity Create images at a hIgher angular resolution Like MeerKAT Does not measure the sky image Difficulty creating an image

And interferometers do not measure the sky directly. Interferometers do not measure the sky in pixels. Each antenna pair measures a Fourier Component. But the image reconstruction forms an ill-posed inverse problem We have many possible images that fit the measurements. Image reconstruction has to find the most likely image.

But produce a huge amount of data. larger problem size require distributed computing so far, it was difficult to separate the image reconstruction Too much work was multiplied by the number of nodes. Mostly done on a limited number of shared-memory systems

Target to distribute the image reconstruction First tests

1.1 Radio Interferometry

The figure 1 shows the whole imaging pipeline for an interferometer, which consists of three steps: Correlator, Calibration and Image Reconstruction. The incoming electromagnetic wave gets measured by the different antennas of the interferometer. The measurements of each antenna pair get correlated into a complex-valued Fourier Component (called Visibility in Radio Astronomy). Each antenna pair therefore measures amplitude and phase of a single Visibility (Fourier Component) of the sky image. What Visibility is measured depends on the distance of the antenna pairs, called the baseline. The longer the baseline, the higher-order Visibility gets measured. Meaning more angular resolution.

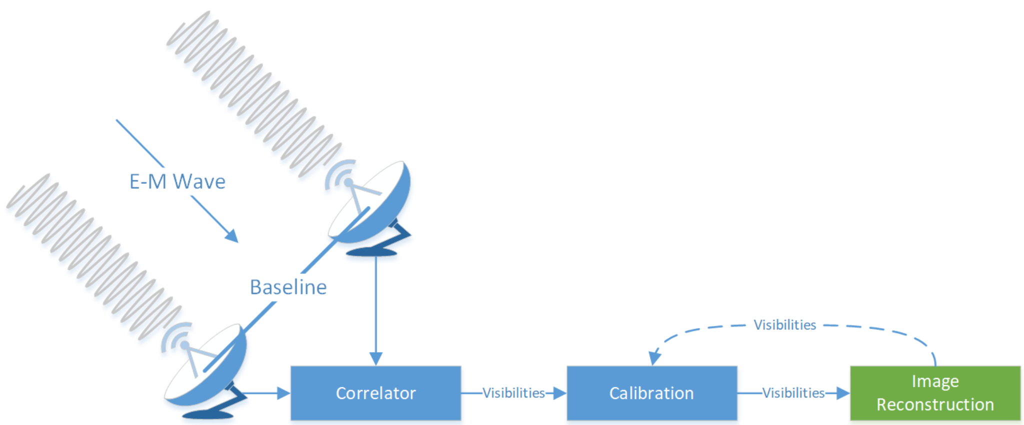


Figure 1: Interferometer System

What Visibilities get measured depends on the antenna layout. The earth rotation modifies the baselines. Due to the earth's rotation, we can sample different Visibilities. (filling the aperture). Noisy measurements Only a limited number of Visibilities. This is where the incompleteness comes in. We do not have enough data. After the Correlator, this is what is saved as the raw measurements.

In the Calibration step, we correct for (Interferometer stuff), Pointing errors, Calibrate the amplitude and phase of the visibilities. Correct for pointing errors. (The interferometer measures relative values). Flagging, remove hopelessly noisy data. Done before imaging, manual labour.

The last step, the Image Reconstruction is where this work focuses its attention. This is where the ill-posed inverse problem is. We want to find the image corresponding to the calibrated visibilities. [Improving the Calibration with partial results of the image reconstruction, done in practice but not part of this project.] The Image Reconstruction part is where the ill-posed inverse problem arises.

1.2 Image Reconstruction Problem

We want to find the image $I()$ which fits the calibrated Visibilities $V()$. The relationship between the Image and the Visibilities is shown in equation (1.1).

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

[Not Really 2d Relationship. if $c() << 1$ Fourier part $e^{2\pi i[ux+vy]}$]. This is the Fourier transform, but in our case we have an extra term $w(c(x,y)-1)$ that keeps us from using the 2d fourier transform, and by extend the 2d FFT. But still a linear relationship. Meaning we can express the relationship from Visibilities $V()$ and Image $I()$ with a matrix, which we call F. Create a set of linear equations and solve for the image. Different ways of representing the image reconstruction problem.

$$\text{analysis: } \underset{I}{find} \quad FI = V \tag{1.2}$$

$$\text{in-painting: } \underset{V_{Unif}}{find} \quad MV_{Unif} = V, \qquad I = iFFT(V_{Unif}) \tag{1.3}$$

$$\text{deconvolution: } \underset{I}{find} \quad I \star PSF = I_{Dirty}, \qquad I_{Dirty} = F^{-1}V \tag{1.4}$$

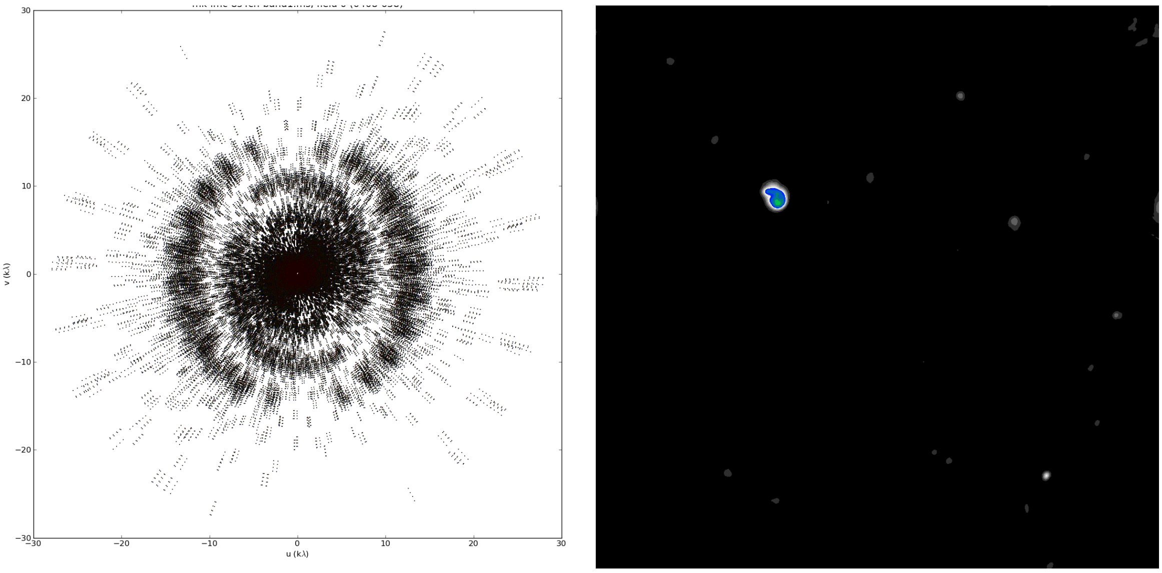
Explain analysis Explain in-painting explain deconvolution All are a way to represent the Image reconstruction problem. If the problem had a unique solution, all three representation would find it. Interestingly enough, in Radio Astronomy, we generally have more Visibilities than Pixels in the reconstruction. the first two (ref) are over-determined problems, meaning we have more linear equations than free variables in the system. At first glance, it seems like we can solve equation (**??**). Why equation (**??**) is ill-posed. Depends on the properties of $V()$ An example from the MeerKAT Radio Interferometer is shown in figure 2.

In figure 2a shows the Visibility Samples in the UV space. We have densely sampled areas, and holes. We have too much data in some areas, while too little in others. Furthermore each Visibility measurement is noisy. These two properties make that many possible images fit the measurements. Which makes the whole inverse problem ill-posed. From the measurements alone, we cannot find the observed image from all possible solutions. We need prior information

F is too big for any practical application. It has the size of pixels times Visibilities. 4k*4k pixels, times 4 billion Visibilities. Approximate F. Decide on a representation!

To solve the image reconstruction problem in Radio Astronomy, we need two things.

- Decide on a representation.
- Prior knowledge about the image.
- If needed, a fast approximation of the Matrix F



(a) *Measurements $V()$ in the UV plane.*

(b) *A reconstructed image $I()$ which fits the measurements.*

Figure 2: The Image Reconstruction Problem

1.3 Solving the Image Reconstruction Problem: The Major/Minor Cycle architecture

Current state of the art way of solving the image reconstruction problem. Major Minor Cycle architecture. Representation: Deconvolution. The Major Cycle is a fast approximation of the Transform Matrix F. The Minor Cycle is a deconvolution algorithm. In this architecture, the Deconvolution algorithm is responsible for including prior knowledge about the image. Representation.

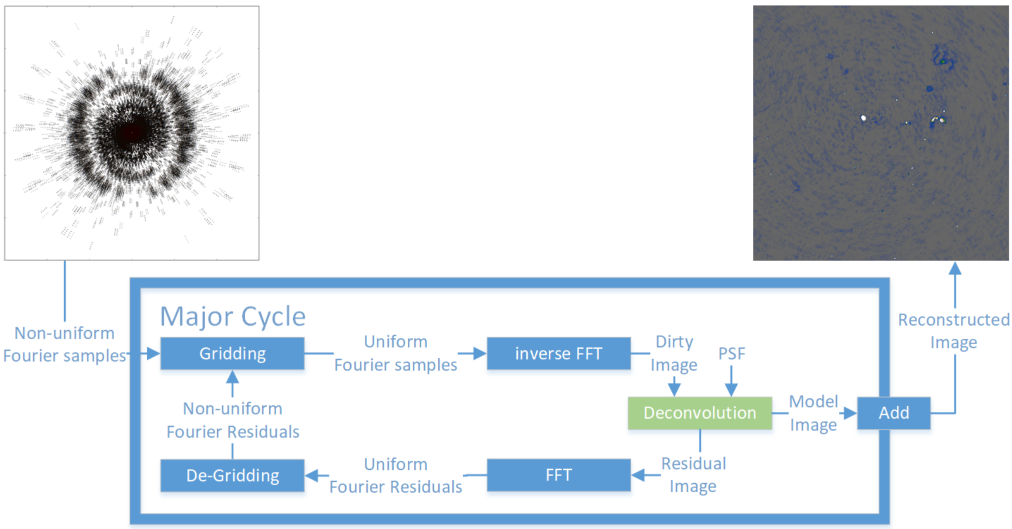
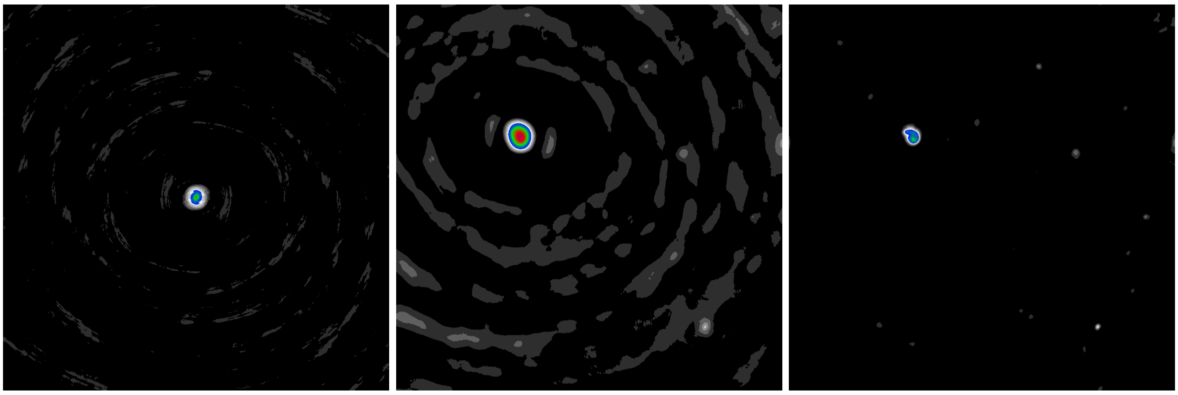


Figure 3: The Major Cycle Architecture



(a) Point Spread Function. *(b) Dirty image* *(c) Deconvolved image.*

Figure 4: Image reconstruction of two simulated point sources.

1.3.1 Deconvolution

1.3.2 Prior Information about the Image

Positivity constraint. Contains two classes of objects: Point sources, which are essentially stars, and extended emissions, which span over several pixels.

1.3.3 Fast Approximation

for $V()$ Incompleteness, holes in the UV space Continuous space Non-uniform sampling, we have more densely sampled uv space

For $I()$

Is uniformly sampled Generally fewer pixels than visibilities Has bright sources

To transform from $V()$ to $I()$, we have to overcome a few problems. invert the Almost-Fourier-Equation from (1.1). It is not really the 2d Fourier Transform. and as such we cannot use the FFT. How to do this efficiently. Transform from continuous into discrete space Into uniformly-sampled space from non-uniform Lossy Transform

What a high quality reconstruction is. Problem of bright sources in the image Noise and High dynamic range So a large part of $V()$ belongs to a few bright sources, but we want to find faint sources "hidden". image

1.3.4 Representation

Devonbolution vs in-painting. vs image transform

We do a deconvolution. Terminology of the dirty image.

Still ill-posed. How can we solve this? we need additional information about the image. We find the most likely image $I()$ given the measurements.

Focus on how to distribute the problem. How to handle this efficiently. How can we distribute this.

1.4 The Major Cycle Architecture

Major cycle how to reconstruct the image with deconvolution In an efficient manner

$V()$ problems of non-uniform sampling, and the 3 dimensions. keep us from using the Fast Fourier Transform. We first interpolate on a regularly spaced grid, in the "Gridder". Use the FFT. For large numbers of Visibilities, this is faster. than inverting equation directly (1.1). And now we do a deconvolution in image space

So we have three basic components, Gridder, FFT and Deconvolution algorithms. The Major Cycle Architecture makes this a cycle. Shown in figure 4.

Why the cycle is necessary Find the fainter sources in later iterations Because we can only estimate the psf.

1.4.1 Minor Cycle

2 Distributing the Image Reconstruction

Distributing the whole image reconstruction. So far, only parts were distributed. First time that end to end, everything gets distributed.

OpenMPI

Gridding and Devonvolution

2.1 Distributed Gridder: The IDG algorithm

Veeneboer et al[1] 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 5 shows the different parts of the image domain algorithm.

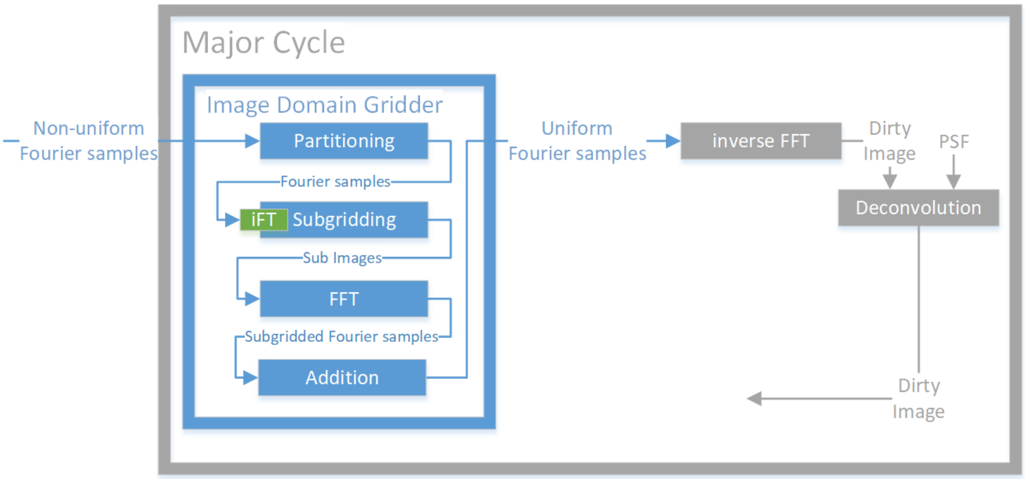


Figure 5: Image Domain Gridder in the Major Cycle Architecture

Algorithm

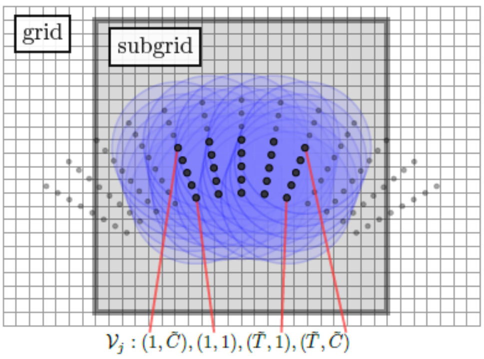


Figure 6: Subgrid

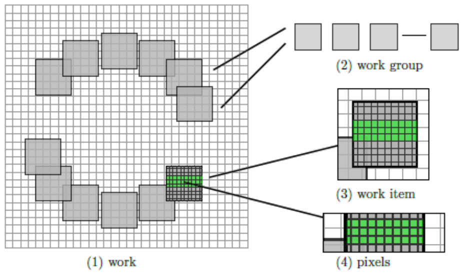


Figure 7: parallel

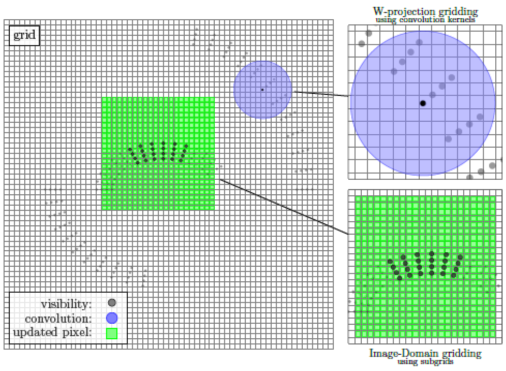


Figure 8: Image Domain Gridder in the Major Cycle Architecture

2.2 Distributed Deconvolution: Coordinate Descent

3 Conclusion

References

[1] 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.

[2] 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.

[3] 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.

[4] Arwa Dabbech, Chiara Ferrari, David Mary, Eric Slezak, Oleg Smirnov, and Jonathan S Kenyon. Moresane: 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

4 attachment

5 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.

5.1 CLEAN: The Major Cycle Architecture

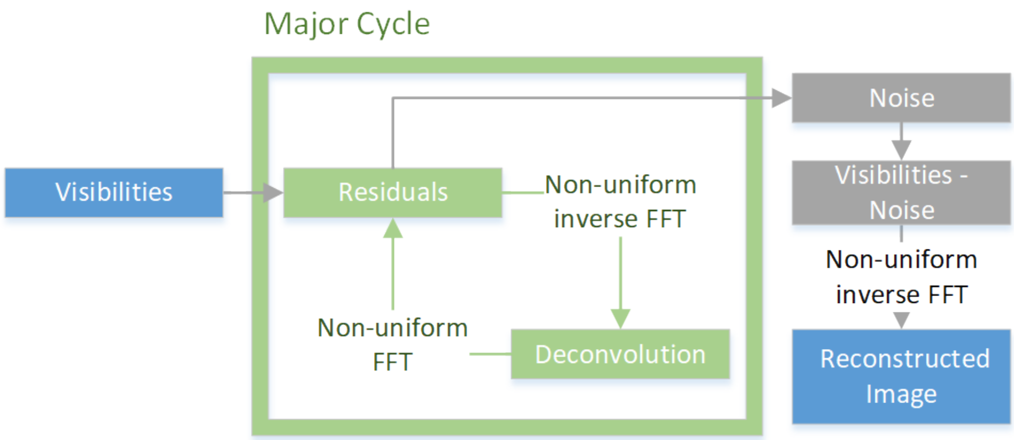


Figure 9: The Major Cycle Architecture

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.

5.2 Compressed Sensing Architecture

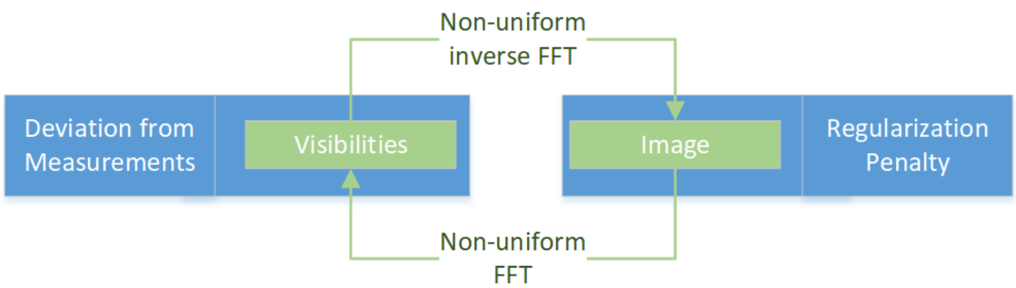


Figure 10: State-of-the-art Compressed Sensing Reconstruction Architecture

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.

5.3 Hypothesis for reducing costs of Compressed Sensing Algorithms

Compressed Sensing Algorithms are not bound to the Architecture presented in section 5.2. 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.

5.4 State of the art: WSCLEAN Software Package

5.4.1 W-Stacking Major Cycle

5.4.2 Deconvolution Algorithms

CLEAN MORESANE

5.5 Distributing the Image Reconstruction

5.5.1 Distributing the Non-uniform FFT

5.5.2 Distributing the Deconvolution

6 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[2, 3], continually reducing the runtime costs of the operation. Recently, Veeneboer et al[1] 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.

6.1 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.

7 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.

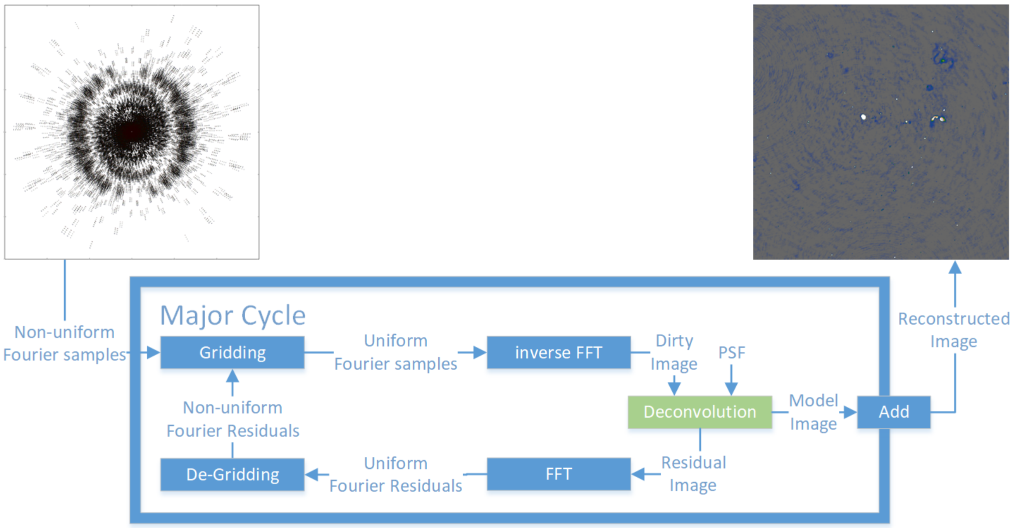


Figure 11: The Major Cycle Architecture of image reconstruction algorithms

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.

7.1 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[1] 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[4]. 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.

7.2 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.

8 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, April 24, 2019

Jonas Schwammberger