A fast and exact w-stacking and w-projection hybrid algorithm for wide-field interferometric imaging

Luke Pratley,[1] Melanie Johnston-Hollitt,[2] and Jason D. McEwen[1]

[1]Mullard Space Science Laboratory (MSSL),
University College London (UCL),
Holmbury St Mary, Surrey RH5 6NT, UK
[2]International Centre for Radio Astronomy Research (ICRAR),
Curtin University,
1 Turner Ave., Technology Park, Bentley, 6102, WA, Australia

ABSTRACT

The standard wide-field imaging technique, the w-projection, allows correction for wide-fields of view for non-coplanar radio interferometric arrays. However, calculating exact corrections for each measurement has not been possible due to the amount of computation required at high resolution and with the large number of visibilities from current interferometers. The required accuracy and computational cost of these corrections is one of the largest unsolved challenges facing next generation radio interferometers such as the Square Kilometre Array. We show that the same calculation can be performed with a radially symmetric w-projection kernel, where we use one dimensional adaptive quadrature to calculate the resulting Hankel transform, decreasing the computation required for kernel generation by several orders of magnitude, whilst preserving the accuracy. We confirm that the radial w-projection kernel is accurate to approximately 1% by imaging the zero-spacing with an added w-term. We demonstrate the potential of our radially symmetric w-projection kernel via sparse image reconstruction, using the software package PURIFY. We develop a distributed w-stacking and w-projection hybrid algorithm. We apply this algorithm to individually correct for non-coplanar effects in 17.5 million visibilities over a 25 by 25 degree field of view MWA observation for image reconstruction. Such a level of accuracy and scalability is not possible with standard w-projection kernel generation methods. This demonstrates that we can scale to a large number of measurements with large image sizes whilst still maintaining both speed and accuracy.

1. INTRODUCTION

Since the advent of radio interferometry in the 1940s (Pawsey et al. 1946; Ryle & Vonberg 1948) radio astronomers have built an impressive suite of interferometric imaging techniques to allow signals from collections of antennas to be used collectively to image astronomical sources. As successive generations of interferometric arrays were built and operated, techniques were developed to obtain an estimate of the true sky brightness distribution, and to correct for different instrumental affects inherent in the process. Among these methods are processes such as deconvolution of the antenna response, so-called 'CLEANing' (Högbom 1974; Schwarz 1978; Steer et al. 1984; Pratley & Johnston-Hollitt 2016), and methods to account for wide-field and other direction dependent effects (DDEs) such as w-projection (Cornwell et al. 2008) and a-projection (Bhatnagar et al. 2008).

In the past where the field of view of instruments was relatively small, it was common practice to assume curvature was negligible and proceed with a two dimensional Fourier transform over the uv-plane (using cartesian coordinates). With the arrival of next generation telescopes, such as the LOw Frequency ARray

Corresponding author: Luke Pratley
Luke.Pratley@gmail.com

(LOFAR; van Haarlem et al. 2013), Murchison Wide-field Array (MWA; Tingay et al. 2013), and Hydrogen Epoch of Reionization Array (HERA; DeBoer et al. 2017), telescopes became non-coplanar arrays with extremely large fields of view. Such instruments are precursors to the low frequency component of the Square Kilometre Array (SKA-LOW), and are already encountering 'big data' challenges. Imaging and correcting for DDEs (with wide-field of view DDEs being the most basic) are among the most computationally intensive and critical challenges that needs to be solved if the SKA is to meet its scientific goals, in areas such as the Epoch of Reionization (EoR) (Koopmans et al. 2015) and Cosmic Magnetism (Johnston-Hollitt et al. 2015). Until now, the approach to account for the third Fourier dimension, w, has been to use mathematical approximations to correct for this term and the associated wide-field effects in the measurement equation, reducing the problem back to a two dimensional Fourier transform via the so-called 'w-projection algorithm' (Cornwell et al. 2008; Tasse et al. 2013; Offringa et al. 2014).

However, the w-projection algorithm kernels, used to correct for non-coplanar array and sky curvature, to date have been computationally expensive to calculate, with kernel generation dominated by the Fast Fourier Transform (FFT) (Scaife 2015). In particular the gridding kernel (anti-aliasing kernel) and w-chirp are multiplied in image space, and then an FFT is applied to generate the w-projection kernel (Cornwell et al. 2011). This means it has not been possible to generate a kernel for each w-term individually, instead they are generated as w-planes, approximately correcting for a group of w-terms.

For extremely wide-fields of view, this becomes expensive in computation and memory, and requires both high resolution sampling to model the spherical curvature and extra zero padding to increase sub-pixel accuracy in the uv-domain. Such a cost in kernel construction has motivated alternative imaging strategies, such as image domain gridding (van der Tol et al. 2018). Even for small fields of view with high resolution, it is not possible to perform an FFT for each visibility on large data sets, limiting the kernel calculation to a small number of w-planes. However, Merry (2016) mathematically showed that for narrow fields of view the w-projection kernel can be approximated as separable into a product of two 1d kernels, reducing the resources required to generate w-planes.

In this work, we set out to improve the analytic understanding of wide-field interferometry, in the hopes that it would provide clues on how to improve the strategy of expensive kernel construction. We start by presenting the non-standard analytic expression for the 3d Fourier transform used to create the w-projection kernel. Then using the analytic expression for the Fourier transform of a spherical shell and enforcing the horizon window with a convolution kernel, we arrive at the 3d expression for the sky curvature and horizon in the uvw-domain. The real component of the kernel is a radial Sinc function in uvw. It is also clear that the horizon window produces the imaginary component, which is a Hilbert transform of the real component. With this understanding, we investigate construction through 3d convolution in the uvw-domain to generate gridding kernels. However, this proves computationally challenging due to rapid osculations and large function support[1].

We find it is less challenging to generate the w-projection kernel via a Fourier integral using 2d adaptive quadrature, due to the smoothness of the window function and the chirp. However, under the condition that the window function has radial symmetry, this 2d Fourier integral is equivalent to 1d Hankel transform. We show that such a 1d Hankel transform can be fast and accurately computed with adaptive quadrature compared to the 2d Fourier integral, and produces the same imaging results.

We discuss the computational impact of having a 1d radially symmetric w-projection kernel, such as reducing the dimension of w-planes from 2d to 1d radial planes, allowing new possibilities for reducing kernel construction costs.

Lastly, we provide a demonstration of exact correction of the w-component to simulated big data sparse image reconstruction using the software package PU-RIFY (Carrillo et al. 2014; Pratley et al. 2018), using the hybrid of w-stacking and w-projection with distributed computation on a high performance computing cluster. Correction of the w-component for each measurement is only possible with the developments in this work, a radially symmetric w-projection kernel and distributed computation with w-stacking.

The developments presented here provide an accurate route for reducing the computational overhead for next generation wide-field imaging, thus providing a step forward on the path to realizing the SKA.

This work starts with an introduction to the interferometric measurement equation and the w-projection algorithm in Section 2, Section 3 extends the w-projection derivation starting from a 3d setting. The calculation of a 1d radially symmetric w-projection kernel is derived in Section 4. The 1d radially symmetric kernel is

---

[1] By the support of a function we mean the region of the domain where the function has non-zero output.

then numerically validated and benchmarked in Section 5. Section 6 details and demonstrates the computationally distributed w-stacking and w-projection hybrid algorithm that is possible with a 1d w-projection kernel. This work is concluded in Section 7.

2. INTERFEROMETRIC MEASUREMENT EQUATION

The interferometric measurement equation for a radio telescope can be represented by the following integral

$$
y(u,v,w') = \int x(l,m)a(l,m)\frac{e^{-2\pi i w'(\sqrt{1-l^2-m^2}-1)}}{\sqrt{1-l^2-m^2}} \\
e^{-2\pi i(lu+mv)}\,\mathrm{d}l\mathrm{d}m\,, \quad (1)
$$

(u,v,w') are the baseline coordinates and (l,m,n) are directional cosines restricted to the unit sphere. In this work, we define $w' = w + \bar{w}$, where $\bar{w}$ is the average value of w-terms, and w is the effective w-component (with zero mean). x is the sky brightness, $n(\boldsymbol{l}) = \sqrt{1-l^2-m^2}$ is a parametrization of the upper hemisphere, and a includes direction dependent effects such as the primary beam and Field of View (FoV). The measurement equation is a mathematical model of the measurement operation that allows one to calculate model measurements y when provided with a sky model x. Having such a measurement equation allows one to find a best fit model of the sky brightness, for a given set of (incomplete) measurements. Many techniques are available for inverting a measurement equation in an attempt to find a best fit model. This includes traditional methods such as CLEAN (Högbom 1974) and Maximum Entropy (Ables 1974; Cornwell & Evans 1985), and state of the art deconvolution methods such as Sparse Regularization algorithms (Onose et al. 2016; Pratley et al. 2018; Dabbech et al. 2018). There are many other variations of the measurement equation, that can include general direction dependent effects and polarization (McEwen & Scaife 2008; Smirnov 2011; Price & Smirnov 2015). But, all interferometric measurement equations can be derived from the van Cittert-Zernike theorem (Zernike 1938).

This measurement equation is typically approximated by a non-uniform fast Fourier transform, since it reduces the computational complexity from $\mathcal{O}(MN)$ to $\mathcal{O}(MJ^2 + N\log N)$, where N is the number of pixels M is the number of visibilities, and J is the number of weights to interpolate off the fast Fourier transform (FFT) grid for each axis (Fessler & Sutton 2003; Thompson et al. 2008). This process is traditionally known as degridding. The version of the measurement equation relevant in this work is represented by the following linear operations

$$
\boldsymbol{y} = \mathsf{WGCFZS}\boldsymbol{x} \quad (2)
$$

S represents a gridding correction and correction of baseline independent effects such as $\bar{w}$, Z represents zero padding of the image, F is an FFT, G represents a sparse circular convolution matrix that interpolates measurements off the grid and the combined GC includes baseline dependent effects such as variations in the primary beam and w-component in the interpolation, and W are weights applied to the measurements. This linear operator represents the application of the measurement equation, so is typically called a measurement operator $\boldsymbol{\Phi} = \mathsf{WGCFZS}$ with $\boldsymbol{\Phi} \in \mathbb{C}^{M\times N}$.

In this case, $\boldsymbol{x}_i = x(\boldsymbol{l}_i)$ and $\boldsymbol{y}_i = y(\boldsymbol{u}_i)$ are discrete vectors in $\mathbb{C}^{N\times 1}$ and $\mathbb{C}^{M\times 1}$ of the sky brightness and visibilities, respectively.

Since the measurement operator is linear it has an adjoint operator $\boldsymbol{\Phi}^\dagger$, which essentially, consists of applying these operators in reverse. Additionally, it is possible to represent these operators in matrix form, however, this is not always efficient or practical.

The dirty map can be calculated by $\boldsymbol{\Phi}^\dagger\boldsymbol{y}$, and the residuals by $\boldsymbol{\Phi}^\dagger\boldsymbol{\Phi}\boldsymbol{x} - \boldsymbol{\Phi}^\dagger\boldsymbol{y}$.

2.1. *Gridding and degridding*

Degridding, also known as the NUFFT, is the process of applying the linear operators GFZS. There are many works in the literature describing this process (see Section 4 of Pratley et al. 2018 for a brief review). The zero padding, Z, (normally by a factor of 2) is to increase accuracy of degridding/gridding of visibilities, by up sampling in the Fourier domain. The choice of interpolation weights in G, known as the gridding kernel, affects the aliasing error, where ghost periodic structures can appear in the dirty map from outside the imaged region. An ideal gridding kernel would be a sinc interpolation kernel, which would prevent any ghosting from the imaged region with a box function, but this has a large support (highly non localized). Well known kernels, such as prolate spheroidal wave functions (PSWF) and Kaiser Bessel functions, are known to suppress the ghosting through apodisation while having minimal support on the Fourier grid (Fessler & Sutton 2003; Offringa et al. 2014; Pratley et al. 2018). This apoidisation is then corrected for with the gridding correction S.

Importantly, the size of the cell in a grid is inversely proportional to the field of view, and the number of cells in a grid determines the resolution of the image.

2.2. *The projection algorithm*

The projection algorithm has been developed to model baseline dependent effects. Typically, DDEs in the measurement equation such as the primary beam and w-term are multiplied with the sky intensity in the image domain. Since they are baseline dependent, a separate primary beam and w-term would need to be multiplied for each baseline – which is computationally inefficient as this involves applying a different gridding/degridding process for each baseline.

If we define our baseline dependent DDEs as

$$c(l, m; w) = a(l, m) \frac{e^{-2\pi i w(\sqrt{1-l^2-m^2}-1)}}{\sqrt{1-l^2-m^2}}\,, \qquad (3)$$

the measurement equation can be expressed as

$$y(u, v, \bar{w} + w) = \int x(l, m) e^{-2\pi i \bar{w}(\sqrt{1-l^2-m^2}-1)}$$
$$\times c(l, m; w) e^{-2\pi i (lu+mv)}\, \mathrm{d}l \mathrm{d}m\,. \qquad (4)$$

We can use the convolution theorem, which states that for functions f and g we have $\mathcal{F}^{-1}\{\mathcal{F}\{f\}\mathcal{F}\{g\}\} = f \star g$, where convolution in 3d is defined as

$$(f \star g)(x, y, z) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(t, r, q)$$
$$\times g(x - t, y - r, z - q)\, \mathrm{d}t \mathrm{d}r \mathrm{d}q\,. \qquad (5)$$

This produces the expression

$$y(u, v, w) = \tilde{y}(u, v, 0) \star C(u, v, w)\,, \qquad (6)$$

where $\tilde{y}(u, v, 0)$ is the Fourier transform of the sky brightness

$$\tilde{y}(u, v, 0) = \int x(l, m) e^{-2\pi i \bar{w}(\sqrt{1-l^2-m^2}-1)}$$
$$\times e^{-2\pi i (lu+mv)}\, \mathrm{d}l \mathrm{d}m\,. \qquad (7)$$

where the projection kernel C is the Fourier representation of c, and $\star$ is the convolution operation.

2.3. Projection with convolutional degridding

Since the convolution with gridding kernels is already baseline dependent, we can include the projection convolution in the gridding process. If we let $G(u, v)$ be a gridding kernel, and the Fourier transform of the window function $g(l, m)$, we find

$$y(u, v, w) = \int \left[\frac{x(l, m)}{g(l, m)} \right] e^{-2\pi i \bar{w}(\sqrt{1-l^2-m^2}-1)} \times$$
$$g(l, m) c(l, m; w) e^{-2\pi i (lu+mv)}\, \mathrm{d}l \mathrm{d}m\,, \qquad (8)$$

this suggests that we should define a new convolutional kernel

$$[GC](u, v, w) = G(u, v) \star C(u, v, w) \qquad (9)$$

$$y(u, v, w) = \tilde{y}(u, v, 0) \star [GC](u, v, w)\,, \qquad (10)$$

where $\tilde{y}(u, v, 0)$ is now the Fourier transform of the gridding corrected sky brightness

$$\tilde{y}(u, v, 0) = \int \frac{x(l, m) e^{-2\pi i \bar{w}(\sqrt{1-l^2-m^2}-1)}}{g(l, m)}$$
$$\times e^{-2\pi i (lu+mv)}\, \mathrm{d}l \mathrm{d}m\,. \qquad (11)$$

Traditionally, the kernel is window separable in l and m, i.e. $g(l, m) = g(l)g(m)$. But, as relevant for the later sections of this work, it can be a radial function, i.e. a function of $\sqrt{l^2 + m^2}$ only.

This shows that we can include the projection convolution in the gridding process through the kernel GC in Equation 10 and the operator **GC** seen in Equation 2. In the next section, we derive expressions for the chirp kernel C in uvw-space from a 3d setting.

3. PROJECTION ALGORITHM IN 3D SETTING

In this section, we derive the 3d w-projection kernel C_H formula including the horizon. We start using a measurement equation which can be expressed to include the horizon explicitly and any restrictions of our signal to the sphere. We restrict the signal above horizon in 3d through the Heaviside step function

$$\Theta(n) = \begin{cases} 1 & n > 0 \\ \frac{1}{2} & n = 0 \\ 0 & n < 0 \end{cases} \qquad (12)$$

and to the sphere through the Dirac delta function, yielding $\delta(1 - l^2 - m^2 - n^2)$,

$$c_\mathrm{H}(l, m, n; w') = \Theta(n)\, \delta(1 - l^2 - m^2 - n^2) e^{+2\pi i w'}\,. \qquad (13)$$

This leads to the measurement equation

$$y(u, v, w') = \int_{-\infty,-\infty,-\infty}^{\infty,\infty,\infty} x(l, m) a(l, m) c_\mathrm{H}(l, m, n; w')$$
$$e^{-2\pi i (lu+mv+nw')} \mathrm{d}l \mathrm{d}m \mathrm{d}n\,. \qquad (14)$$

where equivalent 3d equations can be found in Thompson (1999); Cornwell et al. (2008); Thompson et al. (2008). Unlike the previous section, the above equation has no $1/n$ term. This term is provided by the Dirac composition rule, which is shown in the next subsection.

3.1. w-projection including the horizon directly

In section, we show that the kernel in the work of Cornwell et al. (2008) is equivalent to including both the horizon and spherical effects in the projection algorithm in a full 3d setting. The Fourier transform of Equation 13 is

$$C_{\rm H}(u,v,w) = \int_{0,-\infty,-\infty}^{\infty,\infty,\infty} \delta(1 - l^2 - m^2 - n^2)$$
$$\mathrm{e}^{-2\pi i(lu+mv+nw)}\mathrm{e}^{+2\pi iw}\mathrm{d}l\mathrm{d}m\mathrm{d}n\,. \tag{15}$$

We find that the Dirac delta function is zero at two values of $n = n_{\pm}$, where $n_{\pm} = \pm\sqrt{1-l^2-m^2}$ are the two roots. In addition, we have $\delta(n^2 - n_+^2) = (\delta(n-n_+) - \delta(n-n_-))/(2n_+)$, however, the horizon eliminates the $n = n_-$ root from the integral. Using the composition rule for the Dirac delta function we have

$$C_{\rm H}(u,v,w) = \int_{0,-1,-1}^{1,1,1} \frac{\delta(n-n_+)}{2} \frac{\mathrm{e}^{-2\pi iw\sqrt{1-l^2-m^2}}}{\sqrt{1-l^2-m^2}}$$
$$\mathrm{e}^{-2\pi i(ul+mv)}\mathrm{e}^{+2\pi iw}\mathrm{d}l\mathrm{d}m\mathrm{d}n\,, \tag{16}$$

where the bounds of integration are now restricted to the sphere. and doing an integral over n we find

$$C_{\rm H}(u,v,w) = \int_{-1,-1}^{1,1} \frac{\mathrm{e}^{-2\pi iw(\sqrt{1-l^2-m^2}-1)}}{2\sqrt{1-l^2-m^2}} \tag{17}$$
$$\times \mathrm{e}^{-2\pi i(ul+mv)}\mathrm{d}l\mathrm{d}m\,.$$

This is the standard expression used for the w-projection kernel in Cornwell et al. (2008), with the inclusion of a factor of $1/2$ from there being two roots and normalization of the Dirac Delta function. To date, there is no analytical solution for this integral beyond approximations. One reason this integral may be difficult to solve analytically, is the breaking of spherical symmetry when including the horizon.

Having no analytic solution to this integral poses a problem in understanding the properties of $C_{\rm H}(u,v,w)$. This has lead to various approximations of $C_{\rm H}(u,v,w)$, where the solution can be used estimate its support and amplitude.

We can expand $w(\sqrt{1-l^2-m^2}-1)$ in a Taylor expansion to a given order. We can expand in $(\sqrt{1-l^2-m^2}-1)$ to first order, we find

$$w(\sqrt{1-l^2-m^2}-1) = -\frac{w(l^2+m^2)}{2} + \mathcal{O}(w(l^2+m^2)^2)\,. \tag{18}$$

This has the assumption $w(l^2+m^2)^2 \ll 1$. Also choosing a small field of view $(l^2 + m^2)^2 \ll 1$ leads to

$$\frac{\mathrm{e}^{-2\pi iw(\sqrt{1-l^2-m^2}-1)}}{2\sqrt{1-l^2-m^2}} \to \frac{\mathrm{e}^{\pi iw(l^2+m^2)}}{2}\,. \tag{19}$$

In Cornwell et al. (2008), they state the above small field of view approximation, which is a Gaussian. The Fourier transform of a Gaussian function is also Gaussian, and leads to

$$C_{\rm H}(u,v,w) \propto \frac{\mathrm{e}^{i\pi\frac{(u^2+v^2)}{w}}}{iw}\,, \tag{20}$$

however, they comment that this expression breaks down at large fields of view and diverges at $w = 0$. By choosing to fix the sky to a parabola, rather than the sphere, we arrive at the same approximation above. First we choose

$$c_{\rm H}(l,m,n;w') = \frac{1}{2}\delta\left(n + \frac{l^2+m^2}{2}\right)\,, \tag{21}$$

then by integrating over n in Equation 14 we arrive at same small field of view approximation.

3.2. w-projection with exact spherical correction

We choose to replace the horizon with a window function, where the expression for the full sphere is

$$c_{\rm H}(l,m,n;w') = h(n)\delta(1 - l^2 - m^2 - n^2)\,. \tag{22}$$

Any scaling from this window function can be corrected in the upper hemisphere of the measurement equation

$$y(u,v,w') = \int_{-\infty,-\infty,-\infty}^{\infty,\infty,\infty} \frac{x(l,m)a(l,m)}{h(\sqrt{1-l^2-m^2})}c_{\rm H}(l,m,n;w')$$
$$\mathrm{e}^{-2\pi i(ul+mv+nw')}\mathrm{e}^{+2i\pi w'}\mathrm{d}l\mathrm{d}m\mathrm{d}n\,. \tag{23}$$

3.2.1. No horizon

When $h(n) = 1$ there is no horizon and the w-projection kernel is calculated from

$$C(u,v,w) = \int_{-\infty,-\infty,-\infty}^{\infty,\infty,\infty} \delta(1 - l^2 - m^2 - n^2)$$
$$\mathrm{e}^{-2\pi i(ul+mv+nw)}\mathrm{e}^{+2\pi iw}\mathrm{d}l\mathrm{d}m\mathrm{d}n\,. \tag{24}$$

The Fourier transform of this equation has an analytic solution that can be simply expressed as a real valued function

$$C(u,v,w) = 2\pi\mathrm{sinc}(2\pi\sqrt{u^2+v^2+w^2})\mathrm{e}^{+2\pi iw}\,, \tag{25}$$

as shown in Vembu (1961), which is solved in spherical coordinates due to symmetry. This solution dates back as far as Poisson (1820), and similar problems have been solved in 2 dimensions in Parseval (1805). The units of (u,v,w) are implicitly chosen to depend on the directional cosines (l,m,n), meaning $\sqrt{u^2+v^2+w^2} = 1$ corresponds to the largest spatial scales.

The Sinc function above represents limits on the resolution in (u,v,w) due to the field of view being bounded

to the sphere. The uncertainty principle states that restricting the field of view is equivalent to enforcing a resolution limit on $C(u, v, w)$. At a small field of view, this kernel is effectively a delta function of small support. However, as the field of view increases, the kernel becomes a radial Sinc function with extended support and rapid oscillations. When mosaicking multiple fields of view, resolution in (u, v, w) is increased (as discussed in Ekers & Rots (1979) and Thompson (1999)), however, the total field of view will be limited to the sphere as represented by this radial Sinc function.

Since $x(l, m)$ is independent of n it will project both onto the sphere for n and $-n$. While $C(u, v, w)$ models the curvature of the sphere, it allows a reflection of $x(l, m)$ for $-1 \leq n < 0$. This is why a horizon window function needs to be included in the analysis.

3.2.2. *Projecting above the Horizon*

If we let $H(w)$ be the Fourier transform of $h(n)$, we find that the horizon effect can be understood through the convolution theorem

$$C_{\mathrm{H}}(u, v, w) = H(w) \star C(u, v, w).\qquad(26)$$

We can get an expression for the horizon limited w-projection kernel in the (u, v, w) domain in terms of the w-projection kernel for the full sphere. Choosing $h(n) = \Theta(n)$ with $H(w) = \frac{1}{2}\left[\delta(w) - \frac{i}{\pi w}\right]$, we find an expression equivalent to Equation 17 in the (u, v, w) domain

$$C_{\mathrm{H}}(u, v, w) = \frac{1}{2}C(u, v, w) - \frac{i}{2\pi}\int_{-\infty}^{\infty}\frac{C(u, v, t)}{w - t}\mathrm{d}t\,,\ (27)$$

where the second term is a Hilbert transform of the sphere along the w-axis. Another equivalent expression can be found by choosing a box function $h(n) = \Pi(n + \frac{1}{2})$ for the horizon window, by setting $H(w) = \mathrm{e}^{i\pi w}\frac{\sin(\pi w)}{\pi w}$,

$$C_{\mathrm{H}}(u, v, w) = \int_{-\infty}^{\infty}\mathrm{d}t\,\mathrm{e}^{i\pi t}\mathrm{sinc}(\pi t)C(u, v, w - t)\,.\ (28)$$

We are not aware of an analytic solution to this convolution, which could improve understanding of the behavior of wide field effects.

3.3. *Convolution with a gridding kernel*

To calculate the w-projection kernel, we could convolve the chirp with the gridding kernel in the (u, v, w) domain

$$[GC](u, v, w) = \int_{-\infty,-\infty,-\infty}^{\infty,\infty,\infty} G(p)G(q)H(r)$$
$$C(u - p, v - q, w - r)\mathrm{d}p\mathrm{d}q\mathrm{d}r\,.\qquad(29)$$
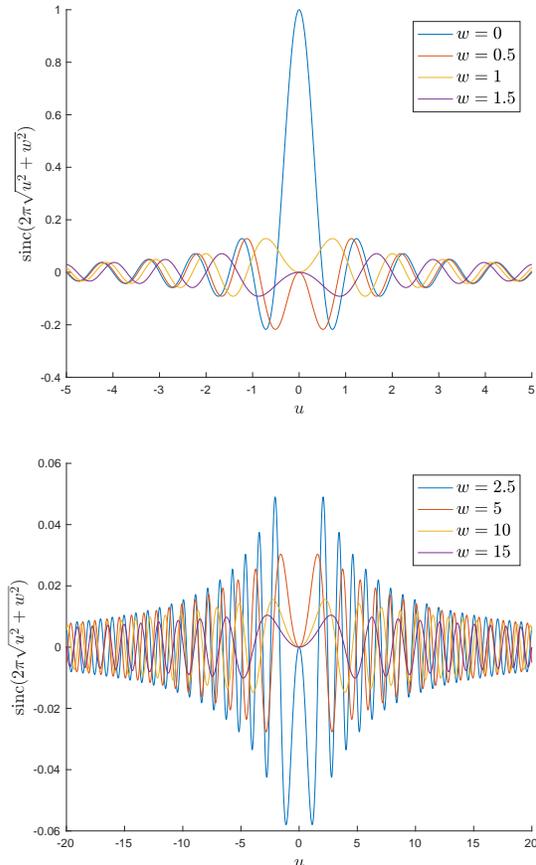
Figure 1. The oscillations of C, without the complex phase, as a function of u for given w. Equation 30, which is used to calculate the pixel size of a uv-grid, shows that many of these oscillations can occur over the convolution window, making numerical integration difficult for convolution with the gridding kernels G and the horizon H. Hence, we find that convolution by numerical integration is difficult. Additionally, we see that C has a large support that increases with w. The top figure shows the standard Sinc function at $w = 0$, and the bottom figure shows the spread of C over a wider range of u as w increases.

However, the challenge with computing this three dimensional integral is the extended support of H and C in w. Additionally, $C(u, v, w)$ will have rapid oscillation in (u, v) for small values of w, making accurate numerical integration and convolution expensive, see Figure 1. Therefore, we avoid this approach in kernel calculation, and present an alternative approach in the next section.

4. KERNEL CALCULATION METHODS

In the previous section, we discussed the properties of the w-projection kernel in the (l, m, n) and (u, v, w) domains. We expected that the properties for numerical convolution with the chirp and the gridding kernel are more favorable by multiplying the window and the

chirp in the image domain, then performing a Fourier transform to generate the kernel in the Fourier domain. This should increase accuracy and reduce the total computation.

In this section, we describe two methods for calculating the w-projection kernel using the Fourier transform. The first is numerical integration using adaptive quadrature in 2d, the second is to restrict the imaged region to a radial field of view, allowing for a radially symmetric kernel that can be integrated with adaptive quadrature in 1d. In the following section we compare the numerical accuracy and speed of the two kernel construction methods. The scaling $\Theta(1 - l^2 - m^2)/\sqrt{1 - l^2 - m^2}$ is included in the gridding and primary beam correction, because it is baseline independent. We do not include this term in the gridding kernel, and we apply this in the image domain with all other baseline independent effects.

4.1. Cartesian integration

To calculate the Fourier coefficients of the w-projection corrected gridding kernel, we need to perform a Fourier series with boundary conditions determined by the size of the window. We let Δu and Δv determine the conversion between pixel and baseline coordinates, $u = u_{\mathrm{pix}}\Delta u$ and $v = v_{\mathrm{pix}}\Delta v$ where u_{pix} and v_{pix} are integer pixel values. This factor is given by

$$\Delta u = \left[2\alpha \sin\left(\frac{N_x \pi \mathrm{cell}}{2 \times 60 \times 60 \times 180.}\right)\right]^{-1}. \tag{30}$$

where cell is the size of a pixel in arc-seconds, α is the oversampling ratio, and N_x is the image width of the x-axis. A similar formula is given for Δv, with respect to the y-axis. We use this field of view to integrate over the imaged region, and including the bounds of the sphere

$$[GC](u_{\mathrm{pix}}, v_{\mathrm{pix}}, w, \Delta u, \Delta v) = \int_{-\alpha/(2\Delta u), -\alpha/(2\Delta v)}^{\alpha/(2\Delta u), \alpha/(2\Delta v)} \mathrm{e}^{-2\pi i w(\sqrt{1-l^2-m^2}-1)} g(\Delta u l) g(\Delta v m)$$
$$\times \mathrm{e}^{-2\pi i(\Delta u u_{\mathrm{pix}} l + \Delta v v_{\mathrm{pix}} m)} \mathrm{d}l \mathrm{d}m. \tag{31}$$

We then change coordinates $l = x/\Delta u$ and $m = y/\Delta v$ to be relative to the imaged region

$$[GC](u_{\mathrm{pix}}, v_{\mathrm{pix}}, w, \Delta u, \Delta v) = \frac{1}{\Delta u \Delta v} \int_{-\alpha/2, -\alpha/2}^{\alpha/2, \alpha/2} \mathrm{e}^{-2\pi i w(\sqrt{1-x^2/\Delta u^2 - y^2/\Delta v^2}-1)} g(x) g(y)$$
$$\times \mathrm{e}^{-2\pi i(u_{\mathrm{pix}} x + v_{\mathrm{pix}} y)} \mathrm{d}x \mathrm{d}y. \tag{32}$$

Here $g(l)$ is the window function that determines the gridding kernel and $[GC]$ is the w-projection corrected gridding kernel. It is worth noticing that when $w = 0$, there is no dependence on Δu or Δv, unless the condition $l^2 + m^2 \leq 1$ is to be enforced.

Depending on the convention of the FFT operation $\mathbf{F}$ in the measurement operator, there could be a phase offset of $\mathrm{e}^{\pm 2\pi i u_{\mathrm{pix}}/2}$ and $\mathrm{e}^{\pm 2\pi i v_{\mathrm{pix}}/2}$ required to centre the image[2]. The region of integration is determined by the zero padded field of view (we have used zero padding by a factor of $\alpha = 2$).

4.2. Polar integration

By performing a change of coordinates, this integral can also be evaluated in polar coordinates

$$[GC](u_{\mathrm{pix}}, v_{\mathrm{pix}}, w, \Delta u, \Delta v) = \frac{1}{\Delta u \Delta v} \int_{0,0}^{\alpha/2, 2\pi} g(r \cos(\theta)) g(r \sin(\theta)) \mathrm{e}^{-2\pi i w(\sqrt{1 - r^2 \cos^2(\theta)/\Delta u^2 - r^2 \sin^2(\theta)/\Delta v^2}-1)}$$
$$\times \mathrm{e}^{-2\pi i(u_{\mathrm{pix}} r \cos(\theta) + v_{\mathrm{pix}} r \cos(\theta))} r \mathrm{d}r \mathrm{d}\theta, \tag{33}$$

The region is circular rather than rectangular, which is a fundamental difference with the Cartesian expression in Equation 32 (the boundary conditions for the Fourier series lie on a circle, rather than a square).

---

[2] This is due the difference of centering the coordinates in the middle or at the corner of the image, which can require an FFT shift.

The enforces a Sinc convolution with the w-projection for the rectangular boundary condition, and a Airy Pattern convolution (first order Bessel Function) for the circular boundary condition. This translates to a slightly different interpolation when up-sampling the w-projection kernel, Sinc interpolation in the rectangular case, and $J_1(4\pi\sqrt{u^2+v^2}/\alpha)/(2\sqrt{u^2+v^2}/\alpha)$ interpolation in the circular case, both enforcing a band-limit.

It is important to state, this boundary is at the edge of the zero-padded region, which suggests that there would be little difference in practice because it is far outside of the gridding corrected region, and will not change suppression of aliasing error (which is the purpose of the window function/gridding convolution function). This means that while the kernels are fundamentally different due to the boundary condition, they will perform the same role, and the entire measurement operators will be equivalent after gridding correction and zero-padding.

4.3. Radial symmetry

We now make our window function radially symmetric $g(l)g(m) \to g(\sqrt{l^2+m^2})$, and choose $\Delta u = \Delta v$ so that the chirp is also radially symmetric. This allows us to take the Fourier transform of a radially symmetric function, which is calculated using a 1d integral rather than the 2d polar integral in Equation 33, and is known as a Hankel transform[3]. This is given by

$$[GC](\sqrt{u_{\mathrm{pix}}^2+v_{\mathrm{pix}}^2},w,\Delta u) = \frac{2\pi}{\Delta u^2}\int_0^{\alpha/2} g(r)\mathrm{e}^{-2\pi i w(\sqrt{1-r^2/\Delta u^2}-1)}J_0\left(2\pi r\sqrt{u_{\mathrm{pix}}^2+v_{\mathrm{pix}}^2}\right)r\mathrm{d}r\,, \qquad (34)$$

where J_0 is a zeroth order Bessel function. The restriction of $r/\Delta u < 1$ is built into the bounds of the integration. This has the large computational advantage of only sampling along the radius, reducing how the computation scales with field of view and w. There is also an increase in accuracy, since there is no sampling in θ. Furthermore, the condition that we require $\Delta u = \Delta v$ is not difficult to accommodate in many cases.

4.4. Adaptive quadrature

To compute Equation 32, we use adaptive multidimensional integration. In a multi-variate setting, quadrature is also known as cubature.

We use the software package Cubature[4] which has implementations of these algorithms. We use the h-adaptive cubature method to evaluate the integrals in this work, which uses the work of Genz & Malik (1980) and Berntsen et al. (1991) to perform integration using an adaptive mesh to approximate the integral, until convergence is reached (h is in reference to a length parameter of the mesh). Cubature also has a p-adaptive method (Ernst 1989), which uses polynomial based quadrature, increasing the polynomial order of the integrand until the integration has converged, and is expected to converge faster than h-adaptive methods for smooth integrands.

The p-adaptive would converge faster than the h-adaptive method for the 1d-integration, while providing results as accurate within numerical error. However, the accuracy of the p-adaptive method was not as accurate for 2d-integration, especially in the presence of discontinuities. For this reason, we use the p-adaptive method for 1d-integration but the h-adaptive method for 2d-integration.

4.5. Kaiser-Bessel gridding kernel

In this work, we use a Kaiser-Bessel gridding kernel. Kaiser-Bessel functions have been used as convolutional gridding kernels for decades (Greisen 1979; Jackson et al. 1991; Fessler & Sutton 2003), and have a simpler form than the prolate spheroidal wave functions, while providing similar performance (Greisen 1979). The zeroth order Kaiser-Bessel function can be expressed as

$$G(u_{\mathrm{pix}}) = \frac{I_0\left(\beta\sqrt{1-\left(\frac{2u_{\mathrm{pix}}}{J}\right)^2}\right)}{I_0(\beta)}\,, \qquad (35)$$

where u_{pix} has units of pixels, J is the support in units of pixels, I_0 is the zeroth order modified Bessel function of the first kind, and β determines the spread of the Kaiser-Bessel function (Jackson et al. 1991; Fessler & Sutton 2003). The Fourier Transform of $G(u_{\mathrm{pix}})$ is

$$g(x) = \mathrm{sinc}\left(\sqrt{\pi^2x^2J^2-\beta^2}\right)\,. \qquad (36)$$

To correct for the convolution, the image is divided by $g(l)$ (Jackson et al. 1991; Fessler & Sutton 2003)

$$s(x) = [g(x)]^{-1}\,. \qquad (37)$$

---

[3] Birkinshaw (1994) suggested that convolutions between radially symmetric functions can be efficiently computed using a Hankel Transform but in different astronomical contexts.

[4] https://github.com/stevengj/cubature

The work of Fessler & Sutton (2003) shows that for $\beta = 2.34J$ the Kaiser-Bessel kernel performs similarly to the optimal min-max kernel considered.

In this work, we use the Kaiser-Bessel gridding kernel to calculate w-projection kernels, by using $g(x)$ in Equations 32 and 34. For other possible window functions and anti-aliasing kernels, see Thompson et al. (2008) and Pratley et al. (2018).

5. VALIDATION OF RADIALLY SYMMETRIC KERNEL

In this section we numerically evaluate Equation 32, and present a cross section of the kernel, showing its variation with sub-pixel accuracy. We then numerically evaluate Equation 34, showing that it provides the same accurate sub-pixel accuracy, with orders of magnitude less function evaluations during the quadrature computation.

5.1. *Quadrature convergence conditions*

The kernel function is normalized to one when $(u, v, w) = (0, 0, 0)$, and an estimate error tolerance η on the quadrature calculated kernel $[GC]^\eta(u_{\mathrm{pix}}, v_{\mathrm{pix}}, w)$ is used for quadrature convergence of the kernel, such that the absolute difference is less than η

$$|[GC](u_{\mathrm{pix}}, v_{\mathrm{pix}}, w) - [GC]^\eta(u_{\mathrm{pix}}, v_{\mathrm{pix}}, w)| \leq \eta. \quad (38)$$

It is also possible to use the relative difference

$$\frac{|[GC](u_{\mathrm{pix}}, v_{\mathrm{pix}}, w) - [GC]^\eta(u_{\mathrm{pix}}, v_{\mathrm{pix}}, w)|}{|[GC]^\eta(u_{\mathrm{pix}}, v_{\mathrm{pix}}, w)|} \leq \eta, \quad (39)$$

which would constrain smaller values of $[GC]^\eta(u_{\mathrm{pix}}, v_{\mathrm{pix}}, w)$ to be calculated more accurately, at the cost of more computation.

There is a downside of using absolute difference, for example, if you are calculating kernels to an absolute accuracy of 10^{-2} and the kernels have values below 10^{-2} then these values may not be accurate. The relative difference is an ideal alternative, but it can cause an inconsistent level of accuracy across the measurement operator, and more computation can go into small values that may not contribute much in practice. If the support size is known accurately before computation, this may help.

We assume that the support size of the w-projection GC kernel is proportional to $2w/\Delta u$ and at least the support size of the gridding kernel G. With the support size known, we use the absolute different criteria with $\eta = 10^{-6}$.

5.2. *Kernel cross-section*

Figure 2 shows a cross section of the w-projection kernel $[GC](u_{\mathrm{pix}}, 0, w)$, the real and imaginary components, and the absolute value, for $0 \leq u_{\mathrm{pix}} \leq 19$ and $0 \leq w \leq 99$. We find that the convolution of C_{H} with $G(u)$ and $G(v)$ creates a smooth varying w-projection kernel in both real and imaginary components. The imaginary component is zero at $w = 0$, which is consistent with Equation 27. We find that the decay in the kernel as a function of w is more extreme with wider fields of view.

We then evoke radial symmetry in the gridding kernel and field of view, and evaluate Equation 34 in Figure 3. We find that the features of the radially symmetric gridding kernel from Equation 32 match the cross section of Equation 34, suggesting little difference between the two kernels. Additionally, when N samples are required to evaluate the 1d radially symmetric kernel, approximately N^2 are required to evaluate the 2d kernel, as shown in Figure 4. This suggests that the symmetric kernel calculation scales with radius, not total area as in the 2d case. This has enormous general implications for computation and storage for w-projection kernels at large fields of view.

5.3. *Numerical equivalence of radially symmetric kernel*

Next, we show that using the radially symmetric gridding kernel is consistent with the non radially symmetric kernel. To test this, we constructed three measurement operators $\boldsymbol{\Phi}_{\mathrm{standard}}$ (standard w-projection kernel), $\boldsymbol{\Phi}_{\mathrm{radial}}$ (symmetric w-projection kernel), and $\boldsymbol{\Phi}_{\mathrm{no-projection}}$ (no w-term), and show that $\boldsymbol{\Phi}_{\mathrm{standard}} \approx \boldsymbol{\Phi}_{\mathrm{radial}}$ within some error (suggesting that they agree), and use $\boldsymbol{\Phi}_{\mathrm{no-projection}}$ as a reference operator.

To show that two operators are equivalent, we need the notion of an operator norm $\| \cdot \|_{\mathrm{op}}$. The operator norm for an operator that maps between Hilbert spaces (ℓ_2) has the property that

$$\|\boldsymbol{\Phi}\boldsymbol{x}\|_{\ell_2} \leq \|\boldsymbol{\Phi}\|_{\mathrm{op}}\|\boldsymbol{x}\|_{\ell_2} \quad \forall \boldsymbol{x} \in \mathbb{R}^N. \quad (40)$$

$\|\boldsymbol{\Phi}\|_{\mathrm{op}}$ is the smallest value for which this is true for all $\boldsymbol{x}$. This allows us to put bounds on the output of $\|\boldsymbol{\Phi}\|_{\mathrm{op}}$ for each input. We also have the properties that $\|\boldsymbol{\Phi}\|_{\mathrm{op}} = \|\boldsymbol{\Phi}^\dagger\|_{\mathrm{op}}$ and $\|\boldsymbol{\Phi}^\dagger\boldsymbol{\Phi}\|_{\mathrm{op}} = \|\boldsymbol{\Phi}\|_{\mathrm{op}}^2$.

The operator norm allows the following statement

$$\begin{aligned} \frac{\|(\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}})\boldsymbol{x}\|_{\ell_2}}{\|\boldsymbol{x}\|_{\ell_2}} \\ \leq \|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}} \quad \forall \boldsymbol{x} \in \mathbb{R}^N. \end{aligned} \quad (41)$$

For every input sky model $\boldsymbol{x}$, the root-mean-squared (RMS) difference between the model visibilities is
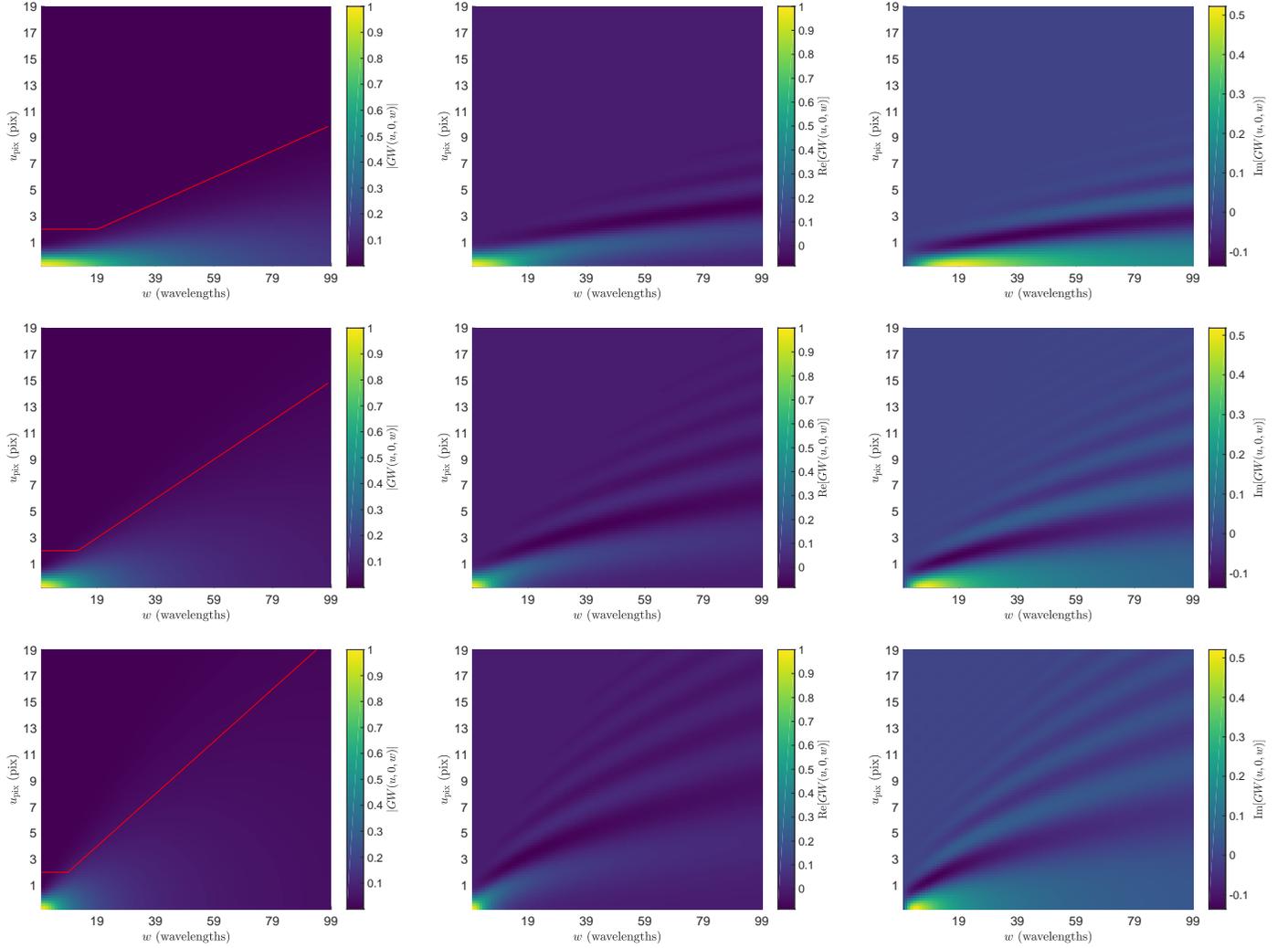
Figure 2. Plot of the kernels calculated using Equation 32, as a function of u_{pix} and w, with $v_{\mathrm{pix}} = 0$, for absolute (left column), real (middle column), and imaginary (right column) values. Each row has a different field of view, $11.3778° \times 11.3778°$ (top), $17.0667° \times 17.0667°$ (middle), and $22.7556° \times 22.7556°$ (bottom). We see that the kernel spreads as a function of increasing w. The support size in pixels increases with field of view, due to a large field increasing the sampling rate of the kernel. It is also clear that the kernel decreases in value with increasing w, faster at wider fields of view. The real and imaginary components both show oscillations. We find the imaginary component is zero at $w = 0$ as expected. The values have been calculated using adaptive quadrature within an absolute error of $\eta = 10^{-6}$. There are 100 uniform samples in each of u_{pix} and w, making 10^4 for each plot. The red line shows $\max(4, 2w/\Delta u)/2$ for reference, which is assumed to be the support size for this work. The features of this kernel are also consistent with w-projection kernels used by ASKAPSoft (Cornwell et al. 2011).

bounded by the product of the RMS of the input sky model and the operator norm $\|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}}$. Additionally, for visibilities $\boldsymbol{y}$

$$
\begin{aligned}
\frac{\|(\boldsymbol{\Phi}^{\dagger}_{\mathrm{standard}} - \boldsymbol{\Phi}^{\dagger}_{\mathrm{radial}})\boldsymbol{y}\|_{\ell_2}}{\|\boldsymbol{y}\|_{\ell_2}} \\
\leq \|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}} \quad \forall \boldsymbol{y} \in \mathbb{R}^M.
\end{aligned} \tag{42}
$$

This statement says that the RMS difference between dirty maps is bounded by the product of the RMS of the input visibilities and the operator norm $\|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}}$. When $\|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}} = 0$, the two operators will clearly be the same.

Since our linear operators map between two Hilbert spaces, the operator norm of $\boldsymbol{\Phi}$ is the square root of the largest Eigenvalue of $\boldsymbol{\Phi}^{\dagger}\boldsymbol{\Phi}$. To calculate the largest Eigenvalue, we use the power method (as used in Pratley et al. (2018)).

First we normalize each operator, such that $\|\boldsymbol{\Phi}\| = 1$, so there is no arbitrary scaling. Then we calculate $\|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{radial}}\|_{\mathrm{op}}$ and $\|\boldsymbol{\Phi}_{\mathrm{standard}} - \boldsymbol{\Phi}_{\mathrm{no-projection}}\|_{\mathrm{op}}$.
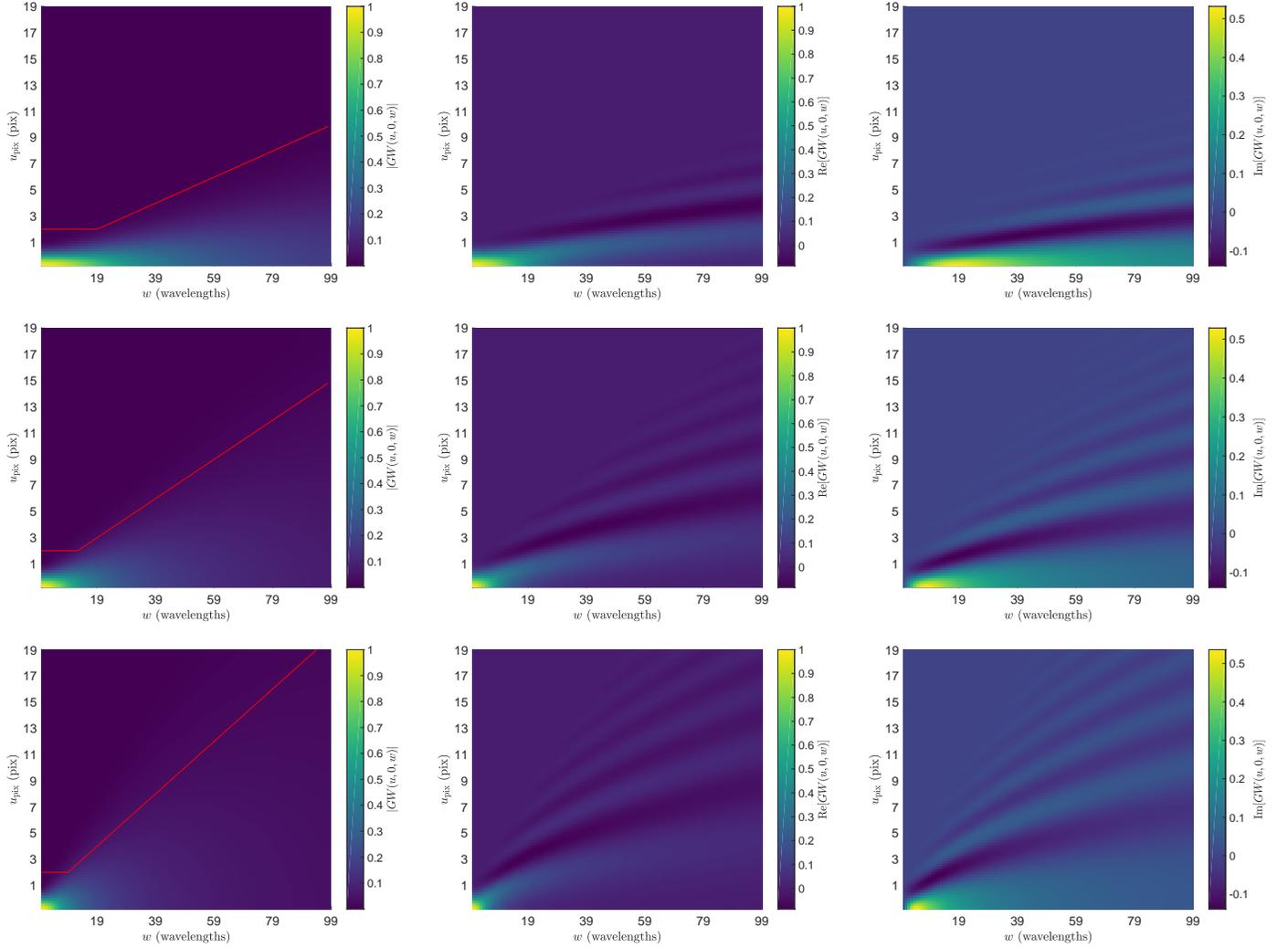
Figure 3. Plot of the kernels calculated from Equation 34, as a function of u_{pix} and w, with $v_{\mathrm{pix}} = 0$, for absolute (left column), real (middle column), and imaginary (right column) values. Each row has a different field of view, $11.3778° \times 11.3778°$ (top), $17.0667° \times 17.0667°$ (middle), and $22.7556° \times 22.7556°$ (bottom). We find the same features in Figure 2, showing that it is consistent with Equation 32. The values have been calculated using adaptive quadrature within an absolute error of $\eta = 10^{-6}$. There are 100 uniform samples in each of u_{pix} and w, making 10^4 for each plot. The red line shows $\max(4, 2w/\Delta u)/2$ for reference.

To construct the measurement operators, we use a variable Gaussian sampling density in (u, v, w), with a root-mean-squared spread of 100 wavelengths. We scale w to have an RMS value of 20 wavelengths. We choose a cell size of 240 arcseconds and an image size of 256 by 256 pixels. This provides a full width field of view of $17.0667° \times 17.0667°$. It is important to note that the w-kernels are a function of the field of view, and not the cell size. The kernel support size is estimated by the w-value for each measurement to be $\min(\max(4, 2w/\Delta u), 40)$. This support has a minimum size of 4 and a largest size of 40, and in between a size of $2w/\Delta u$. The benchmarking was performed on a high performance workstation comprised of two Intel Xeon Processors (E5-2650Lv3)

with 12 cores each with 2 times hyper-threading per core (at 1.8 GHz) and 256 Gigabytes of DDR4 RAM (at 2133 MHz).

We found the construction time of a radially symmetric kernel was almost two orders of magnitude faster to calculate. An absolute difference of 10^{-4} was used for quantifying quadrature convergence. The power method was considered converged with a relative difference of 10^{-6}.

In Figure 5, we show the operator construction time (excluding the normalization), and the operator norm of the difference. Each data point was generated by averaging over 5 realizations. The number of measurements M ranges from only 100 to 1000. From this figure, it is
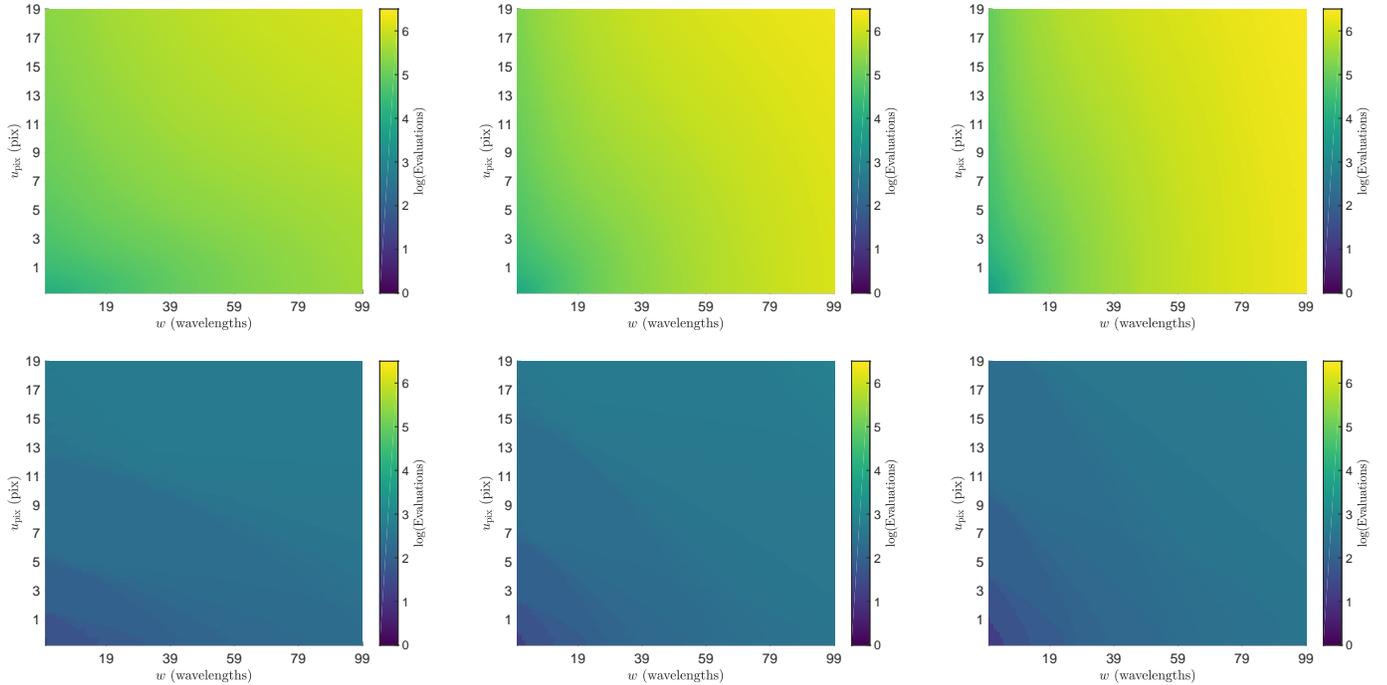
Figure 4. The plots above show the number of function evaluations in the quadrature method required to produce Figures 2 (top row) and 3 (bottom row). Each column corresponds to a field of view of $11.3778° × 11.3778°$ (left), $17.0667° × 17.0667°$ (middle), and $22.7556° × 22.7556°$ (right). The top row shows two times the values in the bottom row, suggesting that if Equation 34 takes N evaluations, then Equation 32 takes N^2 evaluations to compute. This shows the computation of Equation 34 scales with radius vs. the computation of Equation 32 that scales with area. The number of evaluations required can be greatly reduced by increasing the absolute error η.

clear that that the operator difference is consistently on the order of 10^{-3}, suggesting that we have the bounds of $\frac{\|(\boldsymbol{\Phi}^\dagger_{\text{standard}} - \boldsymbol{\Phi}^\dagger_{\text{radial}})\boldsymbol{y}\|_{\ell_2}}{\|\boldsymbol{y}\|_{\ell_2}} \leq 10^{-3}$, which translates to an upper bound dirty map RMS difference of the order of less than 1%. However, the difference will in principle be less. Similar can be said for generating model visibilities.

It is also clear that the construction times are dramatically different between the two. The construction time is greatly improved by the threading, since the kernel construction was performed in parallel. However, due to the small value of M, this improvement has reached saturation. It is clear in this example that construction is hundreds of times faster when using a radial symmetric kernel.

5.4. *Imaging of the directionally dependent w-effect via the zero-spacing*

The previous tests have indirectly verified that the radially symmetric w-projection kernel is consistent with the 2d w-projection kernel, suggesting that the entire degridding and gridding process is self consistent. In this section, we image the generated radially symmetric kernels directly and compare against the theoretically expected values that are independent of implementation.

In the image domain, we expect the w-projection kernel to be a chirp with the form of

$$c(l, m; w) = e^{-2\pi i w(\sqrt{1-l^2-m^2}-1)}, \tag{43}$$

then by only imaging the zero-spacing with an artificial w-component, which can be done by choosing $y(0, 0, w) = 1$ and $\bar{w} = 0$ in the measurement equation, we find that the adjoint application of the measurement operator and then taking the complex conjugate will result in

$$dde_{\text{expected}}(l, m; w) = a(l, m)\frac{c(l, m; w)}{\sqrt{1 - l^2 - m^2}}. \tag{44}$$

It follows that in the discrete setting, gridding a visibility at $(u, v) = (0, 0)$ and $\bar{w} = 0$ will produce the same result

$$dde_{\text{calculated}}(l_i, m_i; w) = \sqrt{N}(\boldsymbol{\Phi}^\dagger_{(u=0,v=0,w)})^*_i. \tag{45}$$

We calculate the average relative difference of *dde* for the imaginary and real parts, using the formula

$$\delta(q, p) = 2\left[\frac{q - p}{|q| + |p|}\right], \tag{46}$$

this suppresses divergences for when q or p are close to zero. We choose $a(l, m) = 1$, and values of $w = 10, 100$
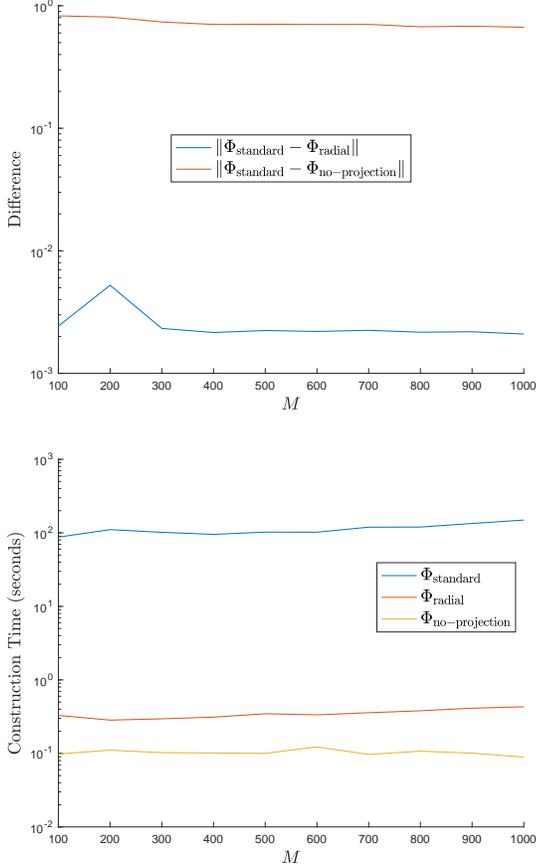
Figure 5. Figures comparing 3 types of measurement operators. One with a standard 2d w-projection kernel $\boldsymbol{\Phi}_{\text{standard}}$, a radially symmetric kernel $\boldsymbol{\Phi}_{\text{radial}}$, and one with no w-projection kernel $\boldsymbol{\Phi}_{\text{no}-\text{projection}}$. The comparisons were performed for 100 to 1000 measurements. (top) The difference in operator norms. We find that the full 2d and radially symmetric kernels are bounded to be the same within about 3×10^{-3}. We find that assuming no w-projection kernel produces a difference close to 1. (Bottom) A plot of the construction time for each operator (excluding normalization). We find that using an analytic expression for the Kaiser-Bessel with no w-projection, $\boldsymbol{\Phi}_{\text{no}-\text{projection}}$, is fastest for two reasons. These are no quadrature integral to calculate, and minimal amount of coefficients to store into memory. The quadrature calculation with variable kernel size means that $\boldsymbol{\Phi}_{\text{radial}}$ will always take more time to calculate, even for $w = 0$, which is computationally cheap for quadrature (see Figure 4). We find $\boldsymbol{\Phi}_{\text{standard}}$ is the most expensive in time to calculate. This is consistent with the number of function evaluations required to calculate each coefficient.

wavelengths using an image with 4096 by 4069 pixels and a pixel height and width of 15 arcseconds. This leads to a field of view of $17.0667° \times 17.0667°$. We compare using a support size linear in w, $\frac{2w}{\Delta u}$, rounded to the nearest pixel. We choose an accuracy of 10^{-6} in absolute and relative error for numerical quadrature.

Figure 6 and 7 show that the radially symmetric w-projection kernel has an error on the order of 1% for both the real and imaginary parts. Where the w-effect goes through zero in the real and imaginary parts the average relative difference diverges. It is clear that the w-projection kernel still matches the expected w-effect, and that these divergences are due to instabilities of the average relative difference for values close to zero.

We find that increasing the support size and reducing the error in numerical quadrature can reduce the average relative difference. We also find that the support size $\frac{2w}{\Delta u}$ and accuracy of 10^{-6} in absolute and relative error for numerical quadrature is sufficient for relative error on the order of 1%. However, if we do not require this accuracy, we can reduce the needed computation by reducing the support size and reducing the accuracy of the numerical quadrature.

6. DISTRIBUTED W-STACKING W-PROJECTION HYBRID ALGORITHM

In this section, we provide a brief demonstration of using radially symmetric w-projection kernels in image reconstruction. We show for the first time that fast and accurate kernel construction, in conjunction with w-stacking, enables the ability for modeling sky curvature and non-coplanar baselines to extremely wide-fields of view for each visibility. The kernels are calculated to an absolute accuracy of 10^{-6}, making the kernel extremely accurate for each w and very wide-fields of view. We present a hybrid of w-stacking and w-projection algorithm that uses the Message Passing Interface (MPI) standard and show its application to image reconstruction of an MWA observation of Puppis A and Vela. This algorithm is made practical with the developments of the previous section and the use of distributed computation.

6.1. w-stacking-w-projection measurement operator

First, we distribute the measurements into w-stacks using MPI. Then, we generate a w-projection kernel for each visibility in a w-stack.

The measurement operator corrects for the average w-value in the w-stack, then applies a further correction to each visibility with the w-projection. Each w-stack $\boldsymbol{y}_k$ has the measurement operator of

$$\boldsymbol{\Phi}_k = \mathbf{W}_k \mathbf{G} \mathbf{C}_k \mathbf{F} \mathbf{Z} \tilde{\mathbf{S}}_k \,. \tag{47}$$

The gridding correction has been modified to correct for the w-stack dependent effects, such as the average $\bar{w}_k$ and $1/n(\boldsymbol{l})$

$$\left[\tilde{\mathbf{S}}_k\right]_{ii} = \frac{a_k(l_i, m_i)\mathrm{e}^{-2\pi i \bar{w}_k\left(\sqrt{1-l_i^2-m_i^2}-1\right)}}{g(l_i^2 + m_i^2)\sqrt{1 - l_i^2 - m_i^2}} \,. \tag{48}$$
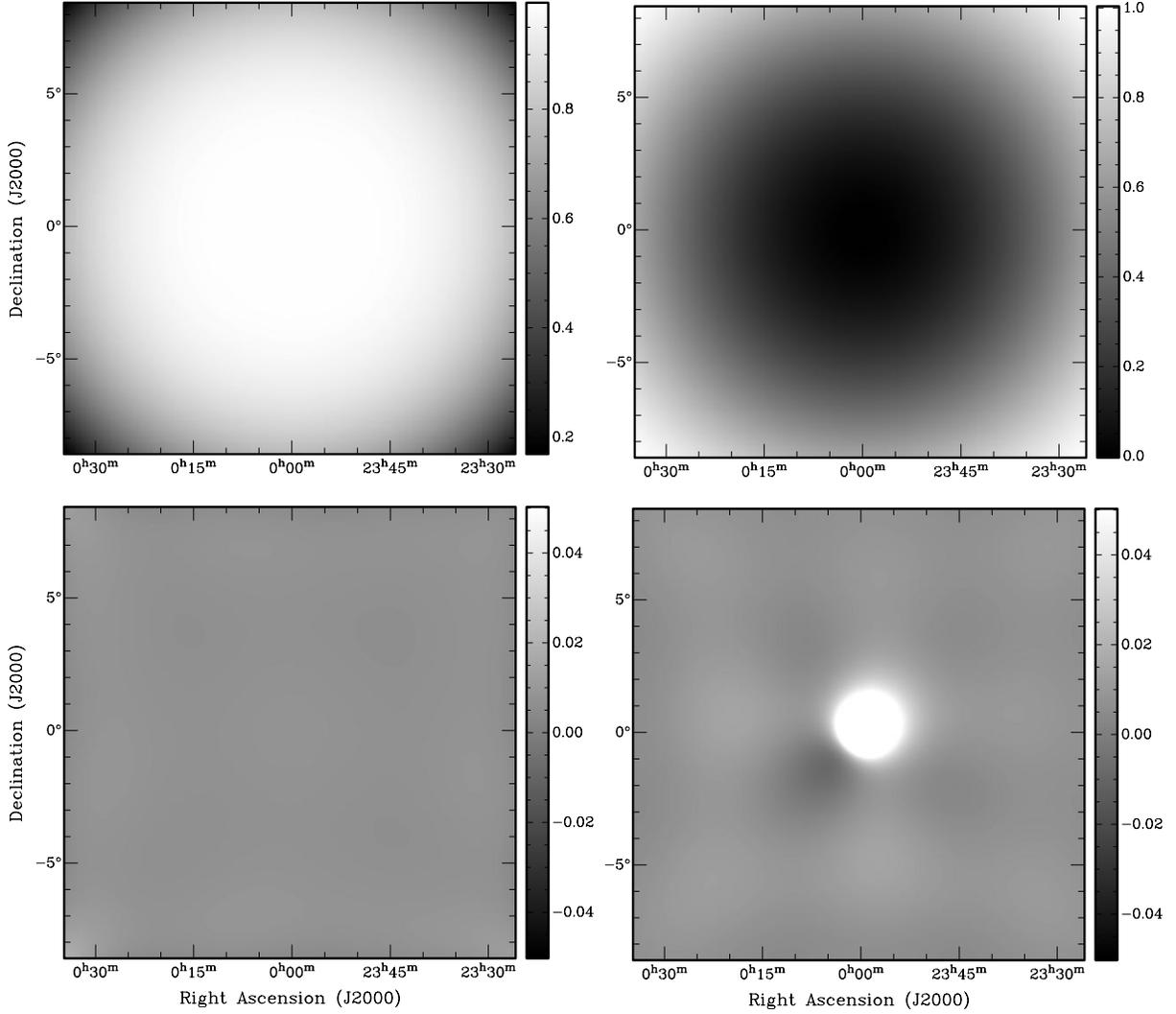
Figure 6. Here we show the calculated radial w-projection chirp in the image domain along with the average relative difference of the expected and calculated chirp for both the real and imaginary parts. The left column displays the real component of the chirp, and the right column the imaginary component. The top row is the radial w-projection chirp in the image domain calculated using $dde_{\mathrm{calculated}}$ with 4096 pixels and a pixel size of 15 arcseconds, calculated for a $w = 10$ wavelengths using a kernel support size of 10 by 10 pixels. The bottom row is the average relative difference $\delta(dde_{\mathrm{expected}}, dde_{\mathrm{calculated}})$. We find that average relative difference is on the order of 1%, excluding where $dde_{\mathrm{calculated}}$ and dde_{expected} are close to zero and the average relative difference diverges. This shows that the radial symmetric w-projection kernel accurately models the directionally dependent w-effect at high resolution over wide-fields of view.

We choose no primary beam effects within the stack $a_k(l_i, m_i)$. This gridding correction shifts the relative w value in the stack. This can reduce the effective w value in the stack, especially when the stack is close to the mean $\bar{w}_k$, i.e. to the value of $w_i - \bar{w}_k$[5]. This reduces the size of the support needed in the w-projection

gridding kernel for each stack,

$$[\mathbf{GC}_k]_{ij} = [GC](\sqrt{(u_i/\Delta u - q_{u,j})^2 + (v_i/\Delta u - q_{v,j})^2}$$
$$, w_i - \bar{w}_k, \Delta u). \tag{49}$$

$(q_{u,j}, q_{v,j})$ represents the nearest grid points. For each stack $\boldsymbol{y}_k \in \mathbb{C}^{M_k}$ we have the measurement equation $\boldsymbol{y}_k = \boldsymbol{\Phi}_k \boldsymbol{x}$.

To cluster the visibilities into w-stacks, it is ideal to minimize the kernel sizes across all stacks, minimizing the memory and computation costs of the kernel. A k-means clustering can be used, which greatly improves

---

[5] Another good choice may be to minimize the median w in a stack rather than the mean w in a stack.
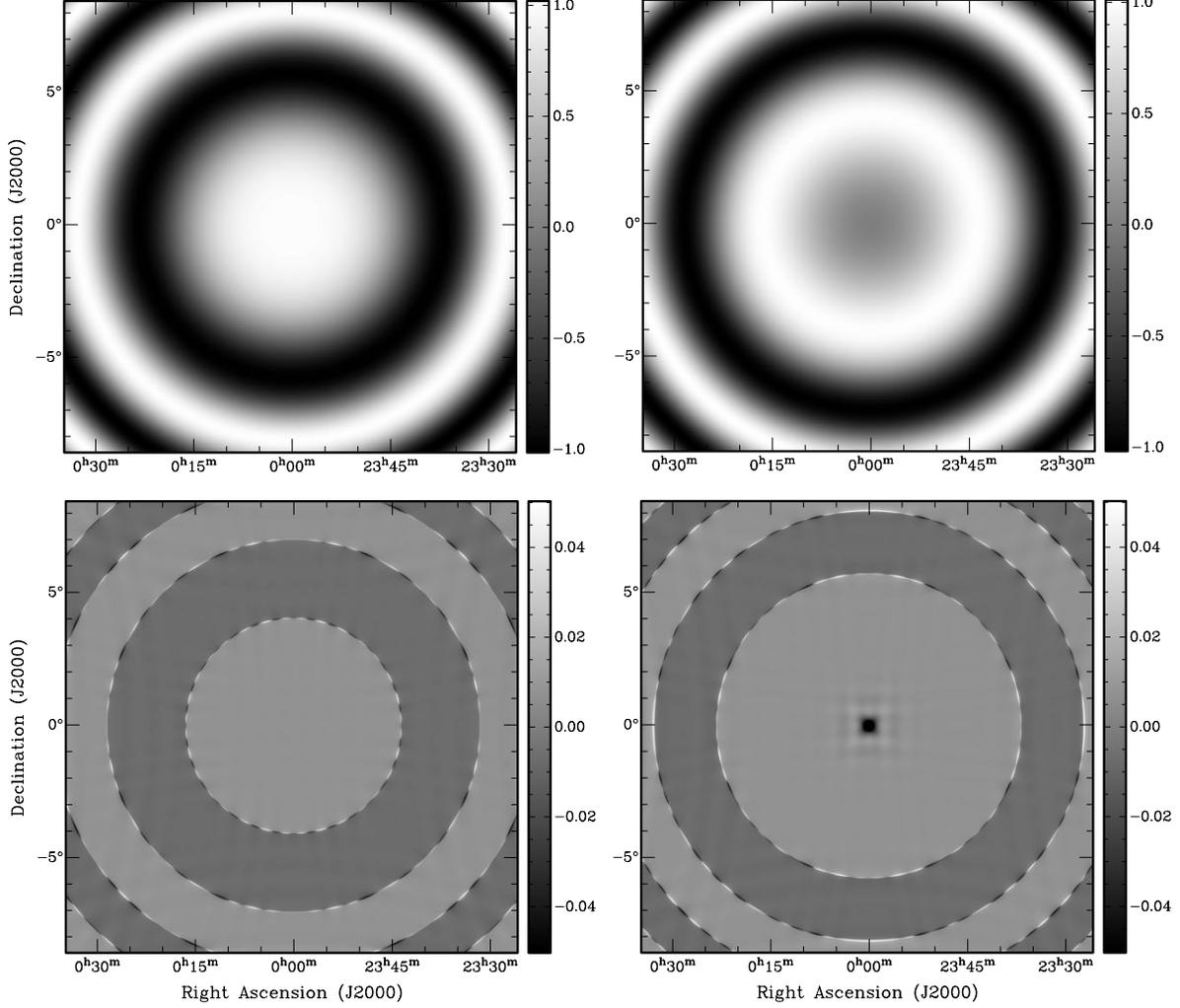
Figure 7. As in Figure 6, but for $w = 100$ wavelengths and using a kernel support size of 118 by 118 pixels. Again we find that average relative difference is on the order of 1%, demonstrating that even for larger w, the radial symmetric w-projection kernel accurately models the directionally dependent w-effect at high resolution over wide-fields of view.

performance by reducing the values of $|w_i - \bar{w}_k|^2$ across the w-stacks.

It is clear that each stack has an independent measurement equation. However, the full measurement operator is related to the stacks in the adjoint operators such that

$$\boldsymbol{x}_{\text{dirty}} = \begin{bmatrix} \boldsymbol{\Phi}_1^\dagger, & \ldots, & \boldsymbol{\Phi}_{k_{\max}}^\dagger \end{bmatrix} \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_{k_{\max}} \end{bmatrix} = \boldsymbol{\Phi}^\dagger \boldsymbol{y}. \quad (50)$$

When applying the w-stacks in parallel, an MPI all reduce can be used to sum over the dirty maps generated from each node. The full operator $\boldsymbol{\Phi}$ can be normalized using the power method.

6.2. *Distributed Image Reconstruction*

For image reconstruction, we use alternating direction method of multipliers as implemented in PURIFY

(ADMM) (Pratley et al. 2018), but built using MPI to operate on a computing cluster. The algorithm solves the same minimisation problem stated in Pratley et al. (2018)

$$\min_{\boldsymbol{x} \in \mathbb{R}^N} \left\| \boldsymbol{\Psi}^\dagger \boldsymbol{x} \right\|_{\ell_1} \quad \text{subject to} \quad \left\| \boldsymbol{y} - \boldsymbol{\Phi} \boldsymbol{x} \right\|_{\ell_2} \leq \epsilon. \quad (51)$$

The term $\left\| \boldsymbol{\Psi}^\dagger \boldsymbol{x} \right\|_{\ell_1}$ is a penalty on the number of non-zero wavelet coefficients, while $\left\| \boldsymbol{y} - \boldsymbol{\Phi} \boldsymbol{x} \right\|_{\ell_2} \leq \epsilon$ is the condition that the measurements fit within a Gaussian error bound ϵ. The wavelet operator $\boldsymbol{\Psi}$ uses a wavelet dictionary of 9 wavelets, which includes a Dirac basis, and Debauches 1 to 8. Each basis in the dictionary $\boldsymbol{\Psi}_k$ has its own node, and is performed in parallel. Like with the adjoint measurement operator, an MPI reduction is performed to sum over the nodes for the forward wavelet

operator[6]

$$\boldsymbol{x} = \left[\boldsymbol{\psi}_1, \; \ldots, \; \boldsymbol{\psi}_9 \right] \begin{bmatrix} \boldsymbol{\alpha}_1 \\ \vdots \\ \boldsymbol{\alpha}_9 \end{bmatrix} = \boldsymbol{\Psi}\boldsymbol{\alpha} \,. \qquad (52)$$

6.3. *MWA observation of Puppis A and Vela*

We use PURIFY (Pratley et al. 2018) and the MPI w-stacking w-projection hybrid algorithm to reconstruct an observation of Puppis A performed with the MWA telescope. The observation is from the Phase 1 configuration of the MWA taken on 16 May 2013. The data was collected with XX and YY linear polarizations and has been calibrated and flagged following the standard MWA data reduction process, more details on this process be found in Offringa et al. (2014). The observation is centered at (RA = 08:19:59.99, DEC = -42:45:00), with a 112 second integration, and a central frequency of 149.115 MHz with a bandwidth of 30.720 MHz. Figure 8 shows a histogram of the visibilities as a function of w, the w-coverage of the observation ranges between ±600 wavelengths. The observation contains on the order of 17 million visibilities, and the XX and YY correlations are combined to generate the Stokes I visibilities.

We use a k-means algorithm with MPI to sort and distribute the visibilities into 50 w-stacks, spread over 25 nodes (2 processes per node, with 1 process per stack), this sorting took approximately 5 seconds. Most w-stacks contain w-values between 0 and ±12 wavelengths, however, some stacks contain w-values of up to 22 wavelengths. The reconstructed image was performed over a $25°$ by $25°$ field of view, using 2048^2 pixels and a pixel width of $45''$. Generating the radial w-projection kernels took close to 40 minutes, this generation time can be changed with more or less w-stacks. Furthermore, the measurement operator was computed in parallel with over 25 nodes, and used in combination with sparse image reconstruction algorithms used in Pratley et al. (2018). We used the Galaxy Supercomputer (located in the Pawsey Supercomputing Centre[7]).

This observation contains the Puppis A and Vela supernova remnants, a mix of many bright compact sources and extended structures of the galactic plane. With PURIFY, we use natural weighting, as it provides the best performance in modeling both extended and compact structures. We do not include primary beam corrections when solving for the reconstructed image.

Figure 9 shows the dirty map, residuals, and the reconstructed image. As described in Pratley et al. (2018), we

---

[6] We use the convention that $\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\alpha}$ and $\boldsymbol{\Psi}^\dagger \boldsymbol{x} = \boldsymbol{\alpha}$.

[7] https://www.pawsey.org.au/our-systems/
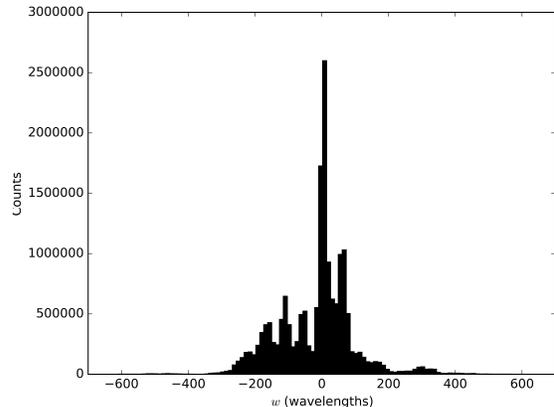
Figure 8. A histogram of the w-coverage of the imaged data using 100 bins. The w-values span over ±600 wavelengths. This w-coverage represents 17,529,644 visibilities after flagging of Radio Frequency Interference (RFI) has been applied.

do not include the restored map, and the reconstructed image is a sky model that is the equivalent to a CLEAN component model. We also follow Pratley et al. (2018) by using the same wavelet dictionary, and scale the epsilon by 275 because the weights are relative not absolute. We can correct the scale of flux due to the field of view by using the Fourier relation $F(\Delta u u_{\rm pix}, \Delta v v_{\rm pix})$ being paired with $\frac{f(l/\Delta u, m/\Delta v)}{\Delta u \Delta v}$.

To convert the dirty map and residual map to Jy/Beam, we image the weights of the visibilities to obtain the peak pixel value of the point spread function, the dirty map is then divided by this value to convert from Jy/Pixel to Jy/Beam. We find that the residual map has a RMS value of approximately 190 mJy/Beam, with many of the extended structures removed from the residuals. The large scale structures of Vela are accurately removed, with only a few positive regions in the residuals where the negative side-lobes of Vela are located. This shows that the majority of the large scale structures and more compact detailed sources such as Puppis A are accurately modeled using PURIFY over a 25 by 25 degree field of view. The dynamic range of the reconstruction is 19,850.

7. CONCLUSION

In this work, we investigate exact analytic expressions for modeling curvature in wide-field interferometry, for extremely wide-fields of view. This expression has traditionally been stated in the (l, m, n) domain. However, this work provides the first exact analytic expression for sky curvature and horizon seen in wide-field interferometry in the (u, v, w) domain. Unlike the previous small field of view approximations, this exact kernel does
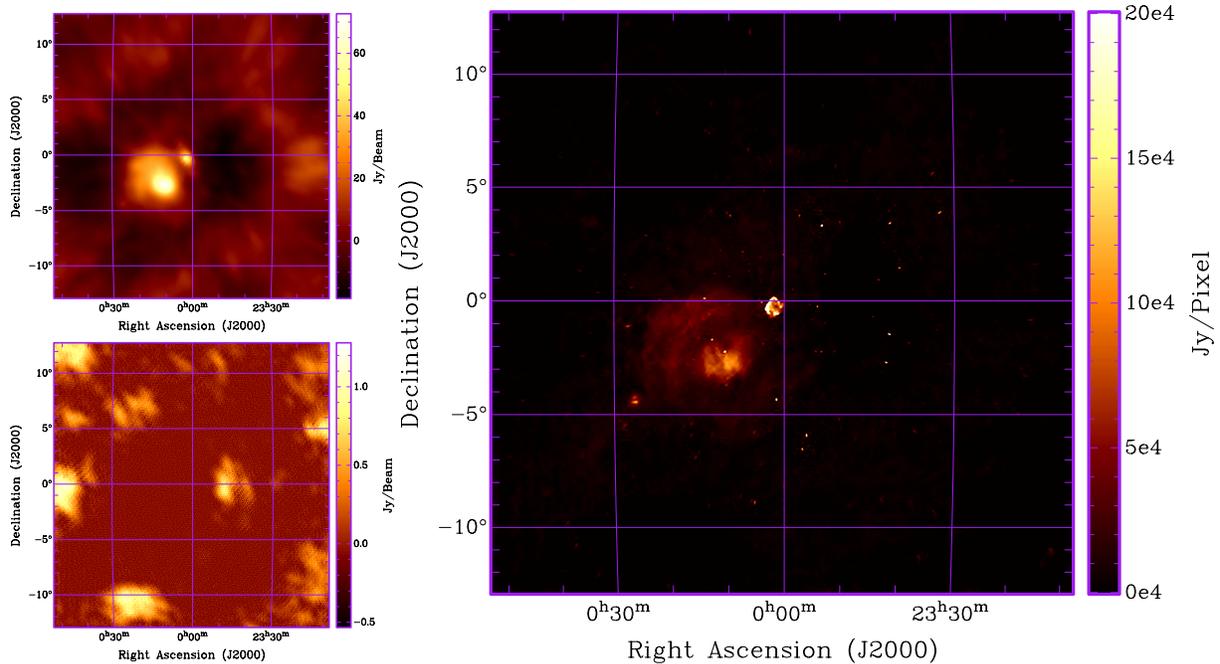
Figure 9. The dirty map (Top Left), residuals (Bottom Left), and sky model reconstruction (Right) of the 112 second MWA Puppis A observation centered at 149.115 MHz, using 17.5 million visibilities and an image size of 2049^2 (each pixel is 45 arcseconds and the field of view is approximately 25 by 25 degrees). This image was reconstructed using the MPI distributed w-stacking-w-projection hybrid algorithm, using the radial symmetric w-projection kernels, in conjunction with the ADMM algorithm. The RMS of the residuals is 0.189 Jy/Beam, the dynamic range of the reconstruction is 19,850.

not diverge and is continuous. Furthermore, it provides more insight and understanding of spherical imaging, i.e. it describes a fundamental resolution limit for the measurement of a visibility from a sphere, and the impact of the horizon window in the (u, v, w) domain. While this expression provides insight, the rapid oscillations due to the spherical sky and large support make calculation difficult. These insights suggest that exact computation of projection kernels is more feasible through a Fourier integral from the (l, m, n) domain.

As described previously, the effect of the w-projection kernel for non-coplanar baselines ($w \neq 0$) becomes greater at larger fields of view. At these extremely wide-fields of view, construction of a w-projection kernel is expensive using FFT based methods. Additionally, in this work, we have found that calculations are extremely fast and accurate using adaptive quadrature to compute a radially symmetric gridding kernel. This dramatically reduces the amount of calculations for a numerically exact kernel calculation, reducing the number of samples in the 2d case from N^2 to N in the radially symmetric case. This immediately makes such a quadrature method computationally competitive. It has low memory usage, it can be distributed in parallel, and scales to extremely wide-fields of view. Furthermore, the calculation is analytic up to a chosen numerical error, allowing

the tuning of speed vs. accuracy that is not possible with FFT based methods for large images.

In this work, we developed a new technique to validate the calculation and application of a DDE. We show that by applying the modeled DDE when gridding the zero-spacing, we provided an image of the DDE model where it can be directly verified. We applied this to the radial w-projection kernel to show the w-effect corrections to be accurate on the order of 1%. This accuracy value is tunable through the support size and the accuracy of the quadrature integration.

These modeling effects are critical not just for imaging, but calibration of instrumental and ionospheric effects, where the w-projection can be used to simulate extremely wide-fields of view. Additionally, any sky model needs to have wide-field of view effects taken into account. Such a sky model maybe critical for physical scientific results. For example, any physical model of the EoR that is to be compared with data collected from a wide-field interferometer needs to have wide-field effects included in the comparison, just as any other instrumental effect (such as the primary beam). This emphasizes that while imaging methods are generally not important for non-imaging experiments, the same process of modeling and correcting for the instrument is still critical in any other analysis.

The fast and exact correction via quadrature using a radially symmetric kernel is new, and makes fast, exact, spherical and non co-planar baseline corrections possible with a w-stacking w-projection hybrid. The process works by first correcting for the average w-value in a stack to reduce kernel size and total computation, then correcting the exact difference for each visibility using quadrature calculated kernels. This method was then demonstrated on an MWA observation of the Puppis A and Vela supernova remnants for a 25 by 25 degree field of view and over 17.5 million measurements.

We have shown that this distributed and paralleled algorithm is extremely powerful for wide-field imaging. Furthermore, these algorithms can be accelerated using multi-threaded parallelism, i.e. General Purpose Graphics Processing Units, in addition to MPI.

With this work, we provide an important step forward in the fast and accurate evaluation of wide-field interferometric imaging, bringing us closer to solving the computational challenges of the SKA and thus realizing its enormous scientific potential.

REFERENCES

Ables, J. G. 1974, A&AS, 15, 383

Berntsen, J., Espelid, T. O., & Genz, A. 1991, ACM Trans. Math. Softw., 17, 437, doi: 10.1145/210232.210233

Bhatnagar, S., Cornwell, T. J., Golap, K., & Uson, J. M. 2008, A&A, 487, 419, doi: 10.1051/0004-6361:20079284

Birkinshaw, M. 1994, in Astronomical Society of the Pacific Conference Series, Vol. 61, Astronomical Data Analysis Software and Systems III, ed. D. R. Crabtree, R. J. Hanisch, & J. Barnes, 249

Carrillo, R. E., McEwen, J. D., & Wiaux, Y. 2014, MNRAS, 439, 3591, doi: 10.1093/mnras/stu202

Cornwell, T. J., & Evans, K. F. 1985, A&A, 143, 77

Cornwell, T. J., Golap, K., & Bhatnagar, S. 2008, IEEE Journal of Selected Topics in Signal Processing, 2, 647, doi: 10.1109/JSTSP.2008.2005290

Cornwell, T. J., Humphreys, B., Lenc, E., Voronkov, V., & Whiting, M. 2011, ASKAP-SW-0020: ASKAP Science Processing, Askap memorandum

Dabbech, A., Onose, A., Abdulaziz, A., et al. 2018, MNRAS, 476, 2853, doi: 10.1093/mnras/sty372

DeBoer, D. R., Parsons, A. R., Aguirre, J. E., et al. 2017, PASP, 129, 045001, doi: 10.1088/1538-3873/129/974/045001

Ekers, R. D., & Rots, A. H. 1979, in Astrophysics and Space Science Library, Vol. 76, IAU Colloq. 49: Image Formation from Coherence Functions in Astronomy, ed. C. van Schooneveld, 61

Ernst, R. 1989, International Journal for Numerical Methods in Engineering, 28, 1335, doi: 10.1002/nme.1620280608

Fessler, J. A., & Sutton, B. P. 2003, IEEE Transactions on Signal Processing, 51, 560, doi: 10.1109/TSP.2002.807005

Genz, A., & Malik, A. 1980, Journal of Computational and Applied Mathematics, 6, 295 , doi: https://doi.org/10.1016/0771-050X(80)90039-X

Greisen, E. W. 1979, The Effects of Various Convolving Functions on Aliasing and Relative Signal-to-Noise Ratios, VLA SCIENTIFIC MEMORANDUM 131, National Radio Astronomy Observatory, Charlottesville, Virginia

Högbom, J. A. 1974, A&AS, 15, 417

Jackson, J. I., Meyer, C. H., Nishimura, D. G., & Macovski, A. 1991, IEEE Transactions on Medical Imaging, 10, 473, doi: 10.1109/42.97598

Johnston-Hollitt, M., Govoni, F., Beck, R., et al. 2015, Advancing Astrophysics with the Square Kilometre Array (AASKA14), 92. https://arxiv.org/abs/1506.00808

Koopmans, L., Pritchard, J., Mellema, G., et al. 2015, Advancing Astrophysics with the Square Kilometre Array (AASKA14), 1. https://arxiv.org/abs/1505.07568

McEwen, J. D., & Scaife, A. M. M. 2008, MNRAS, 389, 1163, doi: 10.1111/j.1365-2966.2008.13690.x

Merry, B. 2016, MNRAS, 456, 1761, doi: 10.1093/mnras/stv2761

Offringa, A. R., McKinley, B., Hurley-Walker, N., et al. 2014, MNRAS, 444, 606, doi: 10.1093/mnras/stu1368

Onose, A., Carrillo, R. E., Repetti, A., et al. 2016, MNRAS, 462, 4314, doi: 10.1093/mnras/stw1859

Parseval, M. A. 1805, Mémoire sur les séries et sur l'intégration complète d'une équation aux différences partillies linéaires du second ordre, á coé'fficiens constans. https://books.google.com.au/books?id=D1s3AQAAMAAJ

Pawsey, J. L., Payne-Scott, R., & McCready, L. L. 1946, Nature, 157, 158, doi: 10.1038/157158a0

Poisson, S. D. 1820, Mémoire sur l'intégration de quelques équations linéaires aux différences partielles, et particuliérement de l'équation générale du mouvement des fluides élastiques, Vol. 3 (Firmin Didot). https://books.google.com.au/books?id=TZ8AAAAAYAAJ

Pratley, L., & Johnston-Hollitt, M. 2016, MNRAS, 462, 3483, doi: 10.1093/mnras/stw1377

Pratley, L., McEwen, J. D., d'Avezac, M., et al. 2018, MNRAS, 473, 1038, doi: 10.1093/mnras/stx2237

Price, D. C., & Smirnov, O. M. 2015, MNRAS, 449, 107, doi: 10.1093/mnras/stv137

Ryle, M., & Vonberg, D. D. 1948, Proceedings of the Royal Society of London Series A, 193, 98, doi: 10.1098/rspa.1948.0036

Scaife, A. 2015, PDR.02.05.03 Imaging Pipeline, Ska sdp scientific memorandum

Schwarz, U. J. 1978, A&A, 65, 345

Smirnov, O. M. 2011, A&A, 531, A159, doi: 10.1051/0004-6361/201116764

Steer, D. G., Dewdney, P. E., & Ito, M. R. 1984, A&A, 137, 159

Tasse, C., van der Tol, S., van Zwieten, J., van Diepen, G., & Bhatnagar, S. 2013, A&A, 553, A105, doi: 10.1051/0004-6361/201220882

Thompson, A. R. 1999, in Astronomical Society of the Pacific Conference Series, Vol. 180, Synthesis Imaging in Radio Astronomy II, ed. G. B. Taylor, C. L. Carilli, & R. A. Perley, 11

Thompson, A. R., Moran, J., & Swenson, G. 2008, Interferometry and Synthesis in Radio Astronomy (Wiley)

Tingay, S. J., Goeke, R., Bowman, J. D., et al. 2013, PASA, 30, 7, doi: 10.1017/pasa.2012.007

van der Tol, S., Veenboer, B., & Offringa, A. 2018, Astronomy & Astrophysics, doi: 10.1051/0004-6361/201832858

van Haarlem, M. P., Wise, M. W., Gunst, A. W., et al. 2013, A&A, 556, A2, doi: 10.1051/0004-6361/201220873

Vembu, S. 1961, The Quarterly Journal of Mathematics, 12, 165

Zernike, F. 1938, Physica, 5, 785, doi: 10.1016/S0031-8914(38)80203-2