WSCLEAN: an implementation of a fast, generic wide-field imager for radio astronomy

A. R. Offringa,[1,2]⋆ B. McKinley,[1,2] N. Hurley-Walker,[3] F. H. Briggs,[1,2]
R. B. Wayth,[2,3] D. L. Kaplan,[4] M. E. Bell,[2,5] L. Feng,[6] A. R. Neben,[6] J. D. Hughes,[1]
J. Rhee,[1,2] T. Murphy,[5,2] N. D. R. Bhat,[3] G. Bernardi,[7] J. D. Bowman,[8]
R. J. Cappallo,[9] B. E. Corey,[9] A. A. Deshpande,[10] D. Emrich,[3] A. Ewall-Wice,[6]
B. M. Gaensler,[5,2] R. Goeke,[6] L. J. Greenhill,[11] B. J. Hazelton,[12] L. Hindson,[13]
M. Johnston-Hollitt,[13] D. C. Jacobs,[8] J. C. Kasper,[14,11] E. Kratzenberg,[9] E. Lenc,[5,2]
C. J. Lonsdale,[9] M. J. Lynch,[3] S. R. McWhirter,[9] D. A. Mitchell,[15,2] M. F. Morales,[12]
E. Morgan,[6] N. Kudryavtseva,[3] D. Oberoi,[16] S. M. Ord,[3,2] B. Pindor,[17] P. Procopio,[17]
T. Prabu,[10] J. Riding,[17] D. A. Roshi,[18] N. Udaya Shankar,[10] K. S. Srivani,[10]
R. Subrahmanyan,[10,2] S. J. Tingay,[3,2] M. Waterson,[3,1] R. L. Webster,[17,2]
A. R. Whitney,[9] A. Williams[3] and C. L. Williams[6]

Affiliations are listed at the end of the paper

ABSTRACT

Astronomical wide-field imaging of interferometric radio data is computationally expensive, especially for the large data volumes created by modern non-coplanar many-element arrays. We present a new wide-field interferometric imager that uses the w-stacking algorithm and can make use of the w-snapshot algorithm. The performance dependences of CASA's w-projection and our new imager are analysed and analytical functions are derived that describe the required computing cost for both imagers. On data from the Murchison Widefield Array, we find our new method to be an order of magnitude faster than w-projection, as well as being capable of full-sky imaging at full resolution and with correct polarization correction. We predict the computing costs for several other arrays and estimate that our imager is a factor of 2–12 faster, depending on the array configuration. We estimate the computing cost for imaging the low-frequency Square Kilometre Array observations to be 60 PetaFLOPS with current techniques. We find that combining w-stacking with the w-snapshot algorithm does not significantly improve computing requirements over pure w-stacking. The source code of our new imager is publicly released.

Key words: instrumentation: interferometers – methods: observational – techniques: interferometric – radio continuum: general.

1 INTRODUCTION

Visibility data from non-coplanar interferometric radio telescopes that observe large fractions of the sky at once cannot be accurately imaged with a two-dimensional fast Fourier transform (FFT). Instead, the imaging algorithm needs to account for the 'w-term' during inversion, which is the term that describes the deviation of the array from a perfect plane (Perley 1999). The image degradation effects of the w-term are amplified for telescopes with wide fields of view (FOV), making this a significant issue for low-frequency telescopes that by nature are wide-field instruments.

There are several methods to deal with the w-term during imaging: faceting (Cornwell & Perley 1992); a three-dimensional Fourier transform (Perley 1999); w-projection (Cornwell, Golap & Bhatnagar 2008); w-stacking (Humphreys & Cornwell 2011); and warped snapshots (Perley 1999). Hybrid methods are sometimes useful, such as with the w-snapshots method (Cornwell, Voronkov & Humphreys 2012).

⋆ E-mail: andre.offringa@anu.edu.au

A new generation of wide-field observatories is producing data sets that are orders of magnitude larger than before. Examples of such telescopes include the Murchison Widefield Array (MWA; Lonsdale et al. 2009; Tingay et al. 2013), the upgraded Jansky Very Large Array and the Low-Frequency Array (LOFAR; van Haarlem et al. 2013). The Common Astronomy Software Applications (CASA; McMullin et al. 2007; Jaegar 2008) have an efficient implementation of the *w*-projection algorithm, with many available features such as multiscale clean and spectral-shape fitting during deconvolution. However, with the MWA we have seen that imaging a 2-min snapshot observation away from zenith can take up to tens of wall-clock hours with CASA's *w*-projection algorithm, because of the larger *w*-terms for off-zenith observations. Imaging with larger image sizes or at higher zenith angles can be impossible because the size and number of *w*-kernels become too large to hold in memory.

Another option exists for imaging MWA data: the Real-Time System (RTS; Mitchell et al. 2008; Ord et al. 2010). This has been designed as an efficient calibration and imaging pipeline specifically for MWA data. It can use graphical processing units (GPUs) to improve efficiency. Snapshot imaging is performed to deal with the *w*-term, which implies that slight variations in tile elevation cause some decorrelation on the longer baselines. As the RTS was designed as a single-pass stream processor, standard iterative deconvolution algorithms are not available. Compact emission can be subtracted and peeled from visibilities using a sky model and calibration updates, but updates to the sky model need to be realized using separate forward-modelling routines (Bernardi et al. 2011; Pindor et al. 2011).

To reach high dynamic ranges, it can be necessary to deal with direction-dependent effects (DDEs). This is especially true for wide-field telescopes. One way to correct for known DDEs is by using the *a*-projection technique, which convolves the data during gridding with a kernel that corrects the DDEs (Bhatnagar et al. 2008). One particular DDE is the effect of the ionosphere. For the MWA it can be assumed that the ionosphere has the same effect on all antennas, because the maximum baseline length is relatively small (2.9 km) and smaller than the typical size of ionospheric structure (Lonsdale 2004). This is not the case for LOFAR, making it necessary to correct the direction-dependent ionospheric effects per station before gridding the data. The AWIMAGER (Tasse et al. 2013) has been written to perform these corrections, and uses a hybrid of *a*-projection, *w*-projection and *w*-stacking. The *a*-projection technique can only be applied directly for deterministic effects such as the correction of the primary beam. Effects like the ionosphere require separate calibration or estimation before *a*-projection can be applied.

Once the Square Kilometre Array (SKA) begins its operation, the required computational power for wide-field imaging will become an even bigger challenge. Cornwell et al. (2012) argues that the *w*-snapshots algorithm is the most efficient approach for the SKA.

In this article, we present a new implementation of a generic wide-field imager that is significantly faster than CASA's *w*-projection implementation. To obtain the increase in speed, the implementation uses the *w*-stacking method for correcting the *w*-terms, optionally combined with a new technique for *w*-snapshot imaging. We named the new imager 'WSCLEAN', as an abbreviation for '*w*-Stacking Clean'. Our new imaging implementation, which in our experience is anywhere from 2 to 12 times faster than the CASA *w*-projection imager, is publicly released.[1]

This paper is structured as follows: The *w*-stacking algorithm is described in Section 2. Details of implementing the *w*-stacking and *w*-snapshots algorithms are described in Section 3. The performance and accuracy will be analysed in Section 4. The conclusions are presented in Section 5.

2 THE *W*-STACKING TECHNIQUE

In this section, we will describe the *w*-stacking algorithm from a mathematical point of view. Instead of applying a convolution in *uv*-space, the *w*-stacking method grids visibilities on different *w*-layers and performs the *w*-corrections after the inverse Fourier transforms (Humphreys & Cornwell 2011).

An interferometer samples the complex visibility function

$$V(u, v, w) = \int \int \frac{A(l, m)I(l, m)}{\sqrt{1 - l^2 - m^2}}$$
$$\times e^{-2\pi i\left(ul+vm+w(\sqrt{1-l^2-m^2}-1)\right)} dl dm, \qquad (1)$$

where u, v, w is a baseline coordinate in the coordinate system of the array, A is the primary-beam function, I is the sky function and l, m are cosine sky coordinates. We will use $I'(l, m)$ to denote the sky function before primary-beam correction, $I'(l, m) = A(l, m)I(l, m)$. We will not discuss calibration, but assume V has been calibrated before imaging. In the case of a polarized measurement, the symbols become 2×2 matrices and beam correction is more complicated, but without loss of generality we will ignore polarization and treat inversion as a scalar problem. Imaging consists of inverting equation (1), i.e. to find I' from V.

For small FOVs, the term $\sqrt{1 - l^2 - m^2}$ is approximately of unit size, making equation (1) approximately an ordinary invertible two-dimensional Fourier transform. A common rule is that this is valid when

$$\forall w, l, m : w\left(\sqrt{1 - l^2 - m^2} - 1\right) \ll 1. \qquad (2)$$

To derive the *w*-stacking technique, equation (1) is rewritten to

$$V(u, v, w) = \int \int \frac{I'(l, m)e^{-2\pi iw(\sqrt{1-l^2-m^2}-1)}}{\sqrt{1 - l^2 - m^2}}$$
$$\times e^{-2\pi i(ul+vm)} dl dm.$$

This is an ordinary two-dimensional Fourier transform going from u, v space to l, m space, and can be inverted to get

$$\frac{I'(l, m)}{\sqrt{1 - l^2 - m^2}} = e^{2\pi iw(\sqrt{1-l^2-m^2}-1)} \int \int V(u, v, w)$$
$$\times e^{2\pi i(ul+vm)} du dv.$$

Integrating both sides over $w_{\min}$ to $w_{\max}$, the minimum and maximum value of w, results in

$$\frac{I'(l, m)(w_{\max} - w_{\min})}{\sqrt{1 - l^2 - m^2}} = \int_{w_{\min}}^{w_{\max}} e^{2\pi iw\left(\sqrt{1-l^2-m^2}-1\right)}$$
$$\times \int \int V(u, v, w)e^{2\pi i(ul+vm)} du dv dw. \qquad (3)$$

The final step is to make the u, v, w parameters discrete, so that the integration over u and v can become an inverse FFT and the integration over w becomes a summation. This shows that the sky function can be reconstructed by: (i) gridding samples with equal w-value on a uniform grid; (ii) calculating the inverse FFT;

---

[1] The WSCLEAN source code can be found at: http://sourceforge.net/p/wsclean

(iii) applying the direction-dependent phase shift $e^{2\pi i w(\sqrt{1-l^2-m^2}-1)}$;
(iv) repeating this for all w-values and adding the results together;
(v) applying the final scaling.

In practice, the final scaling will be different from $(w_{\max} - w_{\min})/\sqrt{1 - l^2 - m^2}$ suggested by equation (3), because the individual w-layers will not be completely filled with samples. Therefore, each pixel is divided by the weighted number of samples. Additionally, it might be required to divide out the effect of a possible convolution kernel, the primary beam and correct for other DDEs. In equally-polarized baselines (e.g. XX, YY, LL or RR), a correlated baseline is the complex conjugate of the reversed baseline, and the relation $V(u, v, w) = \overline{V(-u, -v, -w)}$ holds. The right-hand side of equation (3) with only positive w-value samples then becomes the complex conjugate of the one with only negative w-value samples. In this case, we can therefore calculate the image for $w < 0$ from the image with $w > 0$. That allows us to set $w_{\min}$ to the minimum absolute w-value, which requires half the number of layers. In any case, the input to the two-dimensional inverse FFT is not generally a Hermitian-symmetric function, and hence the inverse FFT is always performed as a complex-to-complex transform.

The inverse of imaging, i.e. to calculate the visibility from a model image, can be done by reversing the w-stacking algorithm: (i) multiply the image with the appropriate factor; (ii) copy the image to several layers; (iii) inverse apply the direction-dependent phase shift for each layer; (iv) FFT each layer; and (v) sample a required visibility from the correct w-layer. We will refer to this operation as prediction.

2.1 Discretization of w

While the discretization of u and v is similar to conventional imaging, the discretization of w defines the number of w-layers that need to be processed. For this, one can use a rule similar to (2), and make sure that the phase difference for two subsequent discretized w-values, w_A and w_B, is less than one radian. This results in the constraint

$$\left| (w_A - w_B) 2\pi \left(\sqrt{1 - l^2 - m^2} - 1 \right) \right| \ll 1. \tag{4}$$

This suggests that a uniform discretization in w is optimal. This is in contrast to Cornwell et al. (2008), where $\sqrt{w}$ tabulation is suggested. From equation (4), the required number of layers can be derived and is given by

$$N_{w\text{-lay}} \gg 2\pi (w_{\max} - w_{\min}) \max_{l,m} \left(1 - \sqrt{1 - l^2 - m^2} \right). \tag{5}$$

Actual values for the right-hand side can be very different, depending on the observation. The value of $w_{\max} - w_{\min}$ is influenced by the coplanarity of the array, the zenith angle (ZA) and the wavelength, while the value of the max $_{l,m}$ term is influenced by the angular size of the image. For the MWA, a typical value of $w_{\max} - w_{\min}$ is $\sim$10 at zenith, and reaches $\sim$400 at a ZA of 30°. For a typical full-FOV image of an MWA observation of 3072×3072 pixels of 0.75 arcmin size, the max $_{l,m}$ term is 0.68. This implies that tens of w-layers are required at zenith and hundreds at lower elevations. The number of w-layers has a large effect on the performance of the w-stacking algorithm, and will be discussed further in the next sections.

To grid the visibilities, the w-values are rounded to the w-value of the nearest w-layer. This discretization can cause noticeable aliasing when using too few w-layers, and results in decorrelation of sources far from the phase centre in the longer baselines. Additionally, this w-aliasing can cause ghost sources to appear in the
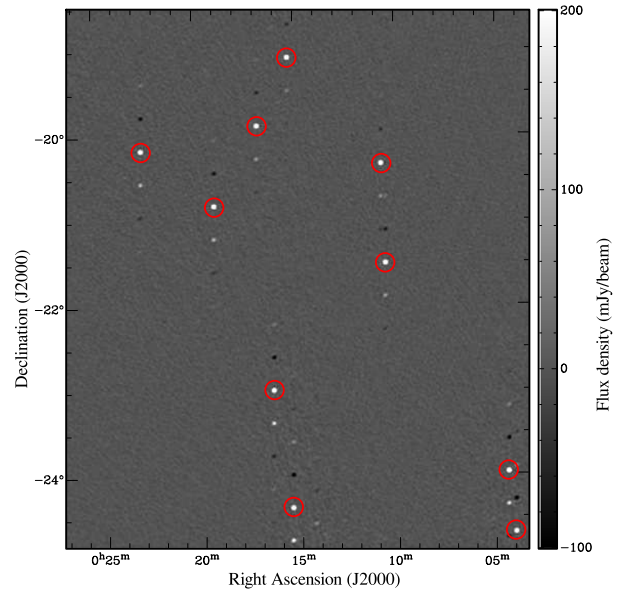
Figure 1. Aliasing artefacts caused by insufficient w-layers in a simulated field. WSCLEAN was set to use 12 w-layers. The centre of the image is at 10° ZA, which would normally require $\sim$195 w-layers. Sources are 1 Jy (red circles), ghost sources are approximately 0.2 Jy. Each source produces two ghost sources, but because they reappear after a major cleaning cycle, they are eventually cleaned and produce more ghost sources.

image. An (extreme) example of this effect is shown in Fig. 1. When Cotton–Schwab cleaning includes prediction with too few w-layers, incorrect values will be subtracted from the visibilities even when no aliased sources are cleaned during minor iterations. Therefore, accurate prediction is more important than accurate imaging, because aliasing artefacts are attenuated by cleaning as long as the model is subtracted accurately.

2.2 Computational complexity of w-stacking

It is useful to analyse the time complexity of w-stacking and compare it with w-projection, to understand which algorithm performs better in a given situation. We will use the following symbols: $N_{w\text{-lay}}$ is the number of w-layers for w-stacking, N_{pix} is the number of pixels in the image along each side, N_{vis} is the number of visibilities, N_{kern} is the size of the anti-aliasing kernel (see Section 3.4), $N_{w\text{-kern}}$ is the size of the w-kernel for w-projection, $w_{\max}$ is the maximum w value and α_{FOV} is the imaging FOV.

Table 1 shows how the computational costs scale for the operations that dominate the imaging in the w-stacking and w-projection algorithms. For comparison, we can assume the anti-aliasing kernel can be neglected and the terms $N_{w\text{-lay}}$ and $N_{w\text{-kern}}$ follow

Table 1. Scaling of the computational cost for various imaging steps, with $N_{w\text{-lay}}$ the number of w-layers, N_{pix} the number of pixels along each side, N_{vis} the number of visibilities, N_{kern} the size of the anti-aliasing kernel, $N_{w\text{-kern}}$ the size of the w-kernel, $w_{\max}$ the maximum w value and α_{FOV} the imaging FOV.

Operation	w-stacking	w-projection
Fourier transform(s)	$N_{w\text{-lay}} N_{\text{pix}}^2 \log N_{\text{pix}}$	$N_{\text{pix}}^2 \log N_{\text{pix}}$
w-term corrections	$N_{w\text{-lay}} N_{\text{pix}}^2$	$N_{\text{vis}} N_{w\text{-kern}}^2$
Gridding	$N_{\text{vis}} N_{\text{kern}}^2$	$N_{\text{vis}} N_{\text{kern}}^2$

approximately $w_{\max} \sin \alpha_{\mathrm{FOV}}$. The time complexity for w-stacking is then given by

$$\mathrm{TC}_{w-\mathrm{stacking}} = \mathcal{O}\left(N_{\mathrm{pix}}^2 \log N_{\mathrm{pix}} w_{\max} \sin \alpha_{\mathrm{FOV}} + N_{\mathrm{vis}}\right), \qquad (6)$$

and for w-projection it is

$$\mathrm{TC}_{w-\mathrm{projection}} = \mathcal{O}\left(N_{\mathrm{pix}}^2 \log N_{\mathrm{pix}} + N_{\mathrm{vis}} w_{\max}^2 \sin^2 \alpha_{\mathrm{FOV}}\right). \qquad (7)$$

From these bounds it can be concluded that in the limiting behaviour, the w-stacking method will be faster when the gridding of the visibilities is the dominating cost of the algorithm. The w-projection algorithm will be faster when the inverse FFTs are the dominant expense. In Section 4, we will determine which method is faster in practice for different parameters.

2.3 *W-snapshot imaging*

W-snapshot imaging is a technique that combines warped-snapshot imaging with a w-correcting technique, such as w-projection or w-stacking (Cornwell et al. 2012). In the warped-snapshot imaging technique, the w-term is neglected, which results in an image with distorted coordinates (Perley 1999; Ord et al. 2010). Additionally, if the positions of the array elements are not perfectly planar or multiple timesteps are integrated to the same grid, visibilities will decorrelate and this can cause imaging artefacts. The original w-snapshot algorithm as described in Cornwell et al. (2012) corrects such artefacts by gridding visibilities on a tilted best-fitting plane in uvw-space, and performs w-corrections towards that plane using w-projection or w-stacking.

We have looked into implementing snapshot imaging with w-corrections in a slightly different way. Instead of performing transforms of tilted planes, our method consists of phase-rotating the visibilities such that the phase centre of the observation is towards the zenith direction, i.e. the direction with minimal w-values. During imaging, the w-layers are recentred from zenith to the direction of interest by phase-shifting the visibilities, thereby translating the image over the tangent plane. This method results in w-corrected snapshots which need to be regridded before further integration, similar to the w-snapshots algorithm as described by Cornwell et al. (2012). However, instead of creating warped images by performing the FFT over a tilted plane in uvw-space, this method transforms planes with constant w-values and produces images in zenith projection.

An image can be recentred from (l, m) to $(\hat{l}, \hat{m}) = (l + \Delta l, m + \Delta m)$ by performing the substitution $(l, m) \to (\hat{l}, \hat{m})$ in equation (3):

$$\frac{I'(\hat{l}, \hat{m})\,(w_{\max} - w_{\min})}{\sqrt{1 - \hat{l}^2 - \hat{m}^2}} = \int_{w_{\min}}^{w_{\max}} \mathrm{e}^{2\pi \mathrm{i} w\left(\sqrt{1 - \hat{l}^2 - \hat{m}^2} - 1\right)}$$

$$\times \iint \mathrm{e}^{2\pi \mathrm{i}(u\Delta l + v\Delta m)} V(u, v, w) \mathrm{e}^{2\pi \mathrm{i}(ul + vm)} \mathrm{d}u \mathrm{d}v \mathrm{d}w. \qquad (8)$$

In words, recentring an image involves accounting for the position shift during the w-correction and final scaling, and shifting the visibilities in phase by multiplication with $\mathrm{e}^{2\pi \mathrm{i}(u\Delta l + v\Delta m)}$ prior to the imaging. By doing the inverse corrections during prediction, a recentred visibility set can be cleaned with Cotton–Schwab iterations similar to a non-recentred set.

Our main reason for developing this method is that it is easier to implement, because no changes are required to the performance-critical gridding step. Another benefit of our method is that the resulting image has a circular synthesized beam, i.e. the resolution in l and m directions matches the intrinsic resolution of the instrument, which is desirable for cleaning. Regridding a recentred image is also more straightforward compared to regridding a warped snapshot, because in the latter case there is no analytic solution to the coordinate conversion (Perley 1999). A benefit of warped snapshots is that the w-term errors are zero at image centre and get worse with the distance from image centre. With our approach, the effect gets worse with the distance from zenith. Our technique has a similar computational cost compared to the w-snapshot algorithm, although the required number of w-layers has a different dependence on ZA, FOV and the non-coplanarity.

Another method to shift an image is by using the periodicity of the FFT function. Since sources outside the FOV will be aliased back into the field, the image can be transformed to have the alias in the centre. This complicates gridding during imaging, because the anti-aliasing kernel needs to have a pass-band shape instead of a low-pass shape. Therefore, we chose to implement the former method of recentring the phase centre before imaging.

3 THE WSCLEAN IMAGER IMPLEMENTATION

With the purpose of testing new algorithms for the imaging of MWA data, we have written a new imager around the w-stacking algorithm. The imager is not specialized for the MWA, and has been successfully used for imaging Very Large Array (VLA) and Giant Metrewave Radio Telescope (GMRT) data.

The new imager, called 'WSCLEAN', is written in the C++ language. It reads visibilities from CASA measurement sets and writes output images to Flexible Image Transport System (FITS) files. Several steps are multithreaded using the threading module of the C++11 standard library. These are: reading and writing; gridding from and to different w-layers; performing the FFTs; and Högbom Clean iterations (peak-finding and image subtraction; Högbom 1974). For the latter, intrinsics are used as well. Because of these optimisations, minor Cleaning iterations with an image size of 3072×3072 are performed at a rate of hundreds per second. This is fast enough to make the Clark Clean optimisation (Clark 1980), which consists of considering only a subset of pixels with a reduced point-spread function (PSF), less relevant. For very large images this optimisation might still be useful, but we have not implemented it in WSCLEAN. Because the PSF varies for imaging with non-zero w-values, subtracting a constant PSF in image space leads to inaccuracies. After a number of minor iterations, it is therefore beneficial to invert the model back to the visibilities via prediction, and subtract the model directly from the visibilities (Cornwell et al. 2008). This is similar to the Cotton–Schwab cleaning method (Schwab 1984). WSCLEAN allows cleaning individual polarizations, or can jointly deconvolve the polarizations. In the latter case, peak finding can be performed in the sum of squared Stokes parameters, $I^2 + Q^2 + U^2 + V^2$, or in $pp^2 + 2pq\overline{(pq)} + qq^2$ space, where p and q are the two polarizations and $\overline{pq} = qp$ is the complex conjugate of pq. After a peak has been found, the PSF is subtracted from the individual polarizations with different factors.

It will not be possible to store all w-layers in memory when creating large images or when many w-layers are used. For example, our test machine, with 32 GB of memory, can store 227 w-layers of 3072×3072. In more demanding imaging configurations, the implementation performs several passes over the measurement set and will grid a subset of w-layers in each pass.

3.1 Full-sky imaging

For low-frequency telescopes, it can be of interest to do full-sky (i.e. horizon to horizon) imaging. In the case of MWA, the tile beam can have strong sidelobes at a distance of more than $90°$ from the pointing centre. Imaging these sidelobes might be relevant because they are scientifically of interest (e.g. when searching for transients). Full-sky imaging can also be useful for self-calibration or deconvolution. For example, self-calibration using full-sky clean components has been found to give good results in imaging the resolved FR-II radio source Fornax A (McKinley et al., in preparation). An example of a full-sky MWA image is shown in Fig. 2.

To efficiently image the full sky, an observation is split into short snapshots, their phase centres are changed to zenith and the snapshots are subsequently imaged, with an appropriate number of pixels and resolution. Depending on desired resolution and dynamic range, this might require very large images. To make an image at the MWA resolution, the image needs to have approximately 10 k pixels along each side. The w-values will be very small at zenith, because at zenith only the vertical offsets of the antennas will contribute to the w-value. For the 128-tile MWA, the maximum differential elevation between tiles is 8.5 m. The tiles are in fact on a slight slope, and by fitting a plane to the antennas and changing the phase centre towards the normal of that plane, the maximum w-value decreases to 5.5 m/λ. In the following sections, when discussing the MWA zenith direction we are referring to this optimal w-direction.

3.2 Implementation of *w*-snapshot imaging

As discussed, all-sky snapshot imaging with zenith as phase centre is quite efficient. However, if one is not interested in imaging the whole sky, and the direction of interest is far from zenith, the computational overhead of making all-sky images is undesirable. For these cases, the recentring technique described in Section 2.3 was implemented in WSCLEAN. This allows making smaller images with minimal w-values that are recentred on the direction of interest. The implementation is generic and supports interferometric data from any telescope.

A recentred image is in the projection of the zenith tangent plane and, like warped snapshots, will need an additional regridding step before multiple snapshots can be added together. A recentred image can be stored in a FITS file with the orthographic (SIN) projection normally used in interferometric imaging (Calabretta & Greisen 2002), by setting the centre of tangent projection with the `CRPIXi` keyword (Greisen & Calabretta 2002). Common viewers such as KVIS (Gooch 1996) and DS9 support such FITS files and display their coordinates correctly. Unlike warped snapshots, recentred images do not need the generalized-SIN-projection keywords `PV2_1` and `PV2_2`.

Our implementation supports cleaning of recentred images in the same modes that normal images can be cleaned. The PSF used during minor cleaning iterations is created by multiplying the weights with the recentring corrections, such that the PSF represents a source in the middle of the image.

An example of the difference in projection between non-recentred and recentred imaging is given in Fig. 3, which displays a 13-min MWA observation imaged with WSCLEAN. The processing steps performed for this image were: (i) pre-processing of the observation with the Cotter pre-processing pipeline, which includes time averaging and RFI flagging with the AOFLAGGER (Offringa et al. 2010; Offringa, van de Gronde & Roerdink 2012b); (ii) calibration using Hydra A without direction dependence using a custom

implementation of the RTS full-polarization calibration algorithm (Mitchell et al. 2008); and (iii) imaging of the seven 112 s intervals separately. Cleaning an image that is in a different projection yields slightly different results, but qualitatively the two images are clearly of equal accuracy. The wall-clock time for imaging is 60 min and 41 min for normal projection and the recentred image, respectively. Of the 41 min, 3 min is spent on phase rotating the visibilities.

3.3 Beam correction for the MWA

Because the MWA consists of fixed, beam-formed dipole antennas, the voltage beam of an MWA tile is described by a complex, non-diagonal Jones matrix. Alternatively, Mueller matrices can be used as well, but in general these result in more computations (Smirnov 2011). When assuming all individual antennas of the tiles are working properly, all tiles have the same beam. The beam can then be corrected in image space for all tiles at once with

$$I(l, m) = J(l, m)^{-1} I'(l, m) \left(J(l, m)^* \right)^{-1}, \qquad (9)$$

where each term is a 2×2 complex matrix, with J the voltage beam and $*$ denoting the conjugate transpose. Because J is complex and non-diagonal, all four combinations of the cross-correlated polarisations[2] p and q including the imaginary part of the pq and qp are required to calculate any of the Stokes parameters. The pp and qq have no imaginary part due to the uv-symmetry. To make proper MWA beam correction possible, WSCLEAN can output both real pp, pq, qp, qq images and imaginary pq and qp images. This requires four runs of the algorithm. Once these images are created, a separate program is used to perform the correction of equation (9). The method is demonstrated in Fig. 4, which displays a detection of pulsar J0437−4715 in Stokes V. The image was made with an initial pipeline for the MWA radio-sky monitor project that uses the MWA to search for transient and variable sources. The pipeline includes WSCLEAN to make the Stokes-parameter images.

Our image-based beam-correction method avoids expensive beam-correcting kernels during gridding, but requires snapshot imaging, because the MWA beam changes over time because of Earth rotation, and only works because all tiles have the same beam. To apply the same method on heterogeneous arrays such as LOFAR, each set of correlations with a different combination of station beams will have to be imaged separately, which will increase the cost of the algorithm excessively unless an a-projection kernel is used. The AWIMAGER uses an intermediate method, and corrects a common dipole factor in image space and the phased-array beam factor in uv-space, which allows the gridding kernel to be smaller compared to correcting both in uv-space (Tasse et al. 2013).

3.4 Gridding

A gridding convolution kernel improves the accuracy of gridding in uv-space (Schwab 1983). A common kernel function is a windowed sinc function, which acts as a low-pass filter. This decreases the flux of sources outside the FOV, and thus helps to attenuate aliased

---

[2] The instrumental polarizations of the MWA are informally often referred to as *X* and *Y*, mainly because many software treats the two polarizations as such. However, this is somewhat confusing because they are not necessarily orthogonal.
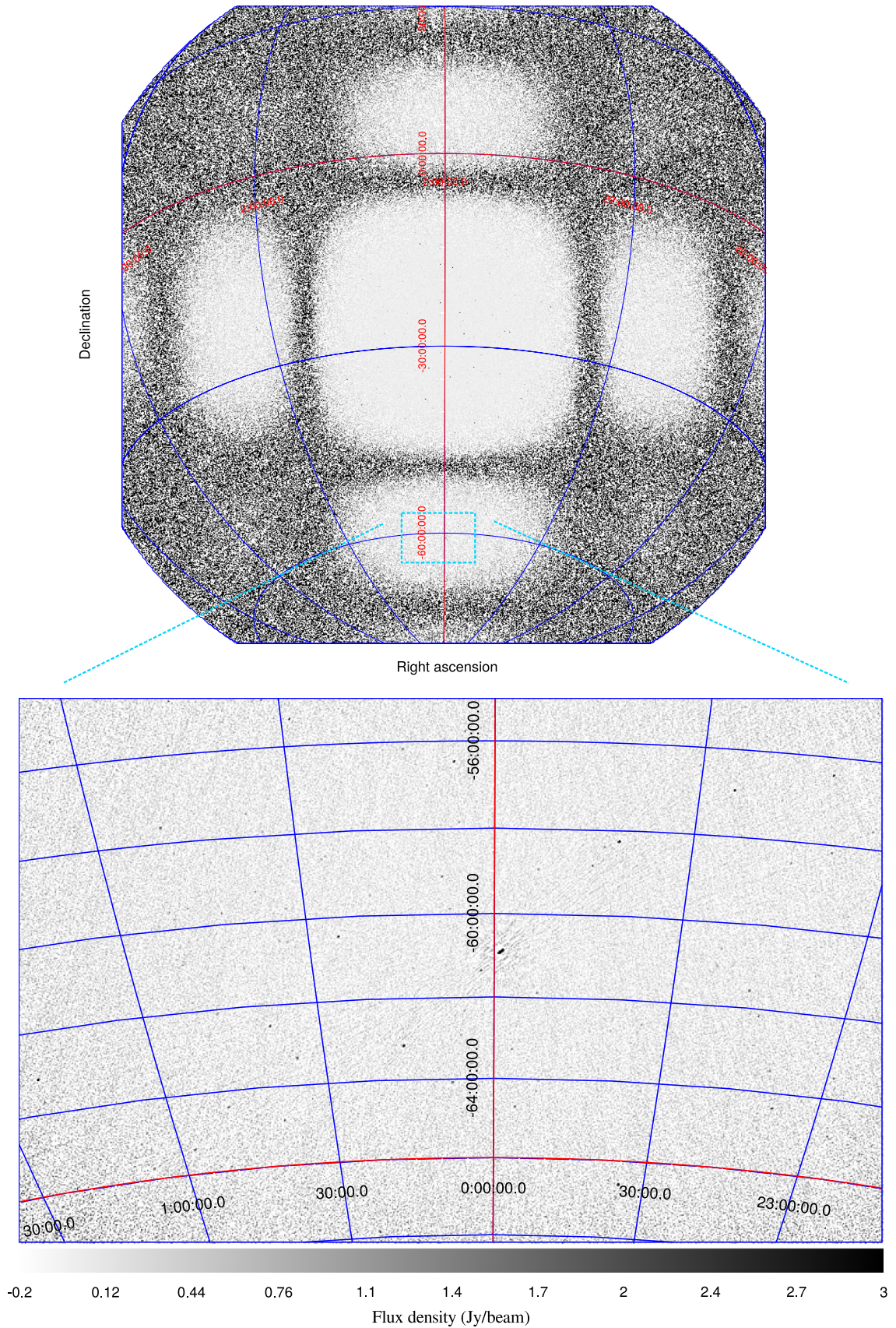
Figure 2. A beam-corrected MWA image of a 112 s zenith observation at 180 MHz that covers almost the full sky. The lower image shows a zoom-in on the southern side lobe. PKS J2358−6054 (∼100 Jy) is visible in the centre of the southern side lobe and resolved. The noise level in the side lobe is about 200 mJy beam^{-1}. PKS J2358−6054 has been cleaned, but some artefacts remain because no direction-dependent calibration has been performed.
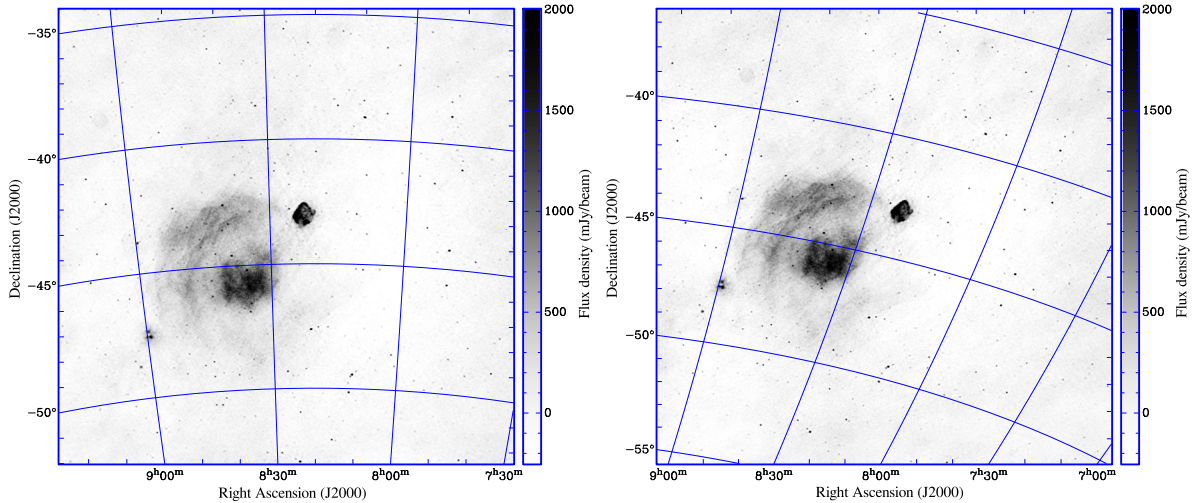
Figure 3. 13-min MWA observation of supernova remnants Vela and Puppis A with a centre frequency of 149 MHz. Left: normal projection centred on Puppis A, right: phase-rotated to zenith to reduce w-values, recentred on Puppis A during imaging. Both Stokes-I images have been made with WSCLEAN using the Cotton–Schwab clean algorithm. No beam correction was applied. Imaging computing cost was 60 min for the normal projected image and 41 min for the recentred image.
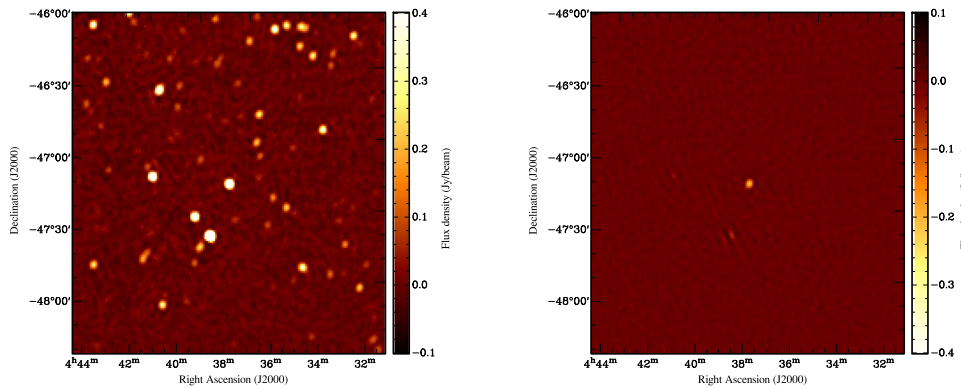

Figure 4. Stokes I (left-hand panel, total power) and Stokes V (right-hand panel, circular polarization) images of PSR J0437−4715, demonstrating the full-polarization beam correction capability for MWA observations with WSCLEAN. This observation was performed in drift-scan mode with a centre frequency of 154 MHz. The pulsar (centre of image) displays 17 per cent circular polarization. Due to inaccuracies in the current beam model, other (unpolarized) sources show ∼1 per cent leakage into Stokes V.

ghost sources and sidelobes (Offringa, de Bruyn & Zaroubi 2012a). By supersampling, a convolution kernel also makes it possible to place samples more accurately at their uv position, thereby lowering decorrelation.

The prolate spheroidal wavefunction (PSWF) is generally considered to be the optimal windowing function for gridding (Jackson et al. 1991). CASA's gridder implementation convolves samples with a PSWF of seven pixels total width during gridding. When using a variable kernel size, a PSWF is quite complicated and computationally expensive to calculate. WSCLEAN currently uses a Kaiser–Bessel (KB) window function, which is easy and fast to compute, and is a good approximation of the PSWF (Jackson et al. 1991).

The type of window function has no effect on the gridding performance, but the size of the kernel affects gridding performance quadratically. Fig. 5 shows that this quadratic relation becomes significant for kernel sizes $\gtrsim 10$ pixels. Decreasing the kernel size to values below seven pixels has little effect on performance, because for small kernels the visibility-reading rate is lower than the gridding rate. We have not noticed much benefit of larger kernels except in rare cases where a bright source lies just outside the imaged FOV.
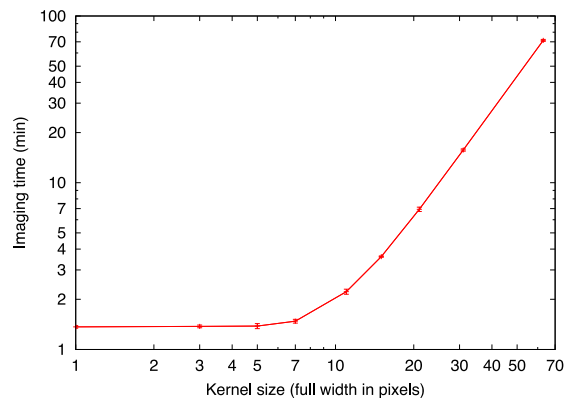
Figure 5. Gridding kernel size plotted against imaging time, using common MWA settings.

Therefore, WSCLEAN uses a default of seven pixels for the gridding kernel. It can be increased when necessary. Of course, different hardware might give slightly different results because of different reading and calculation performance.

3.5 Lowering the resolution of inversion

Typically, for cleaning it is desired that images have a pixel size (side length) at least five times smaller than the synthesized-beam width. This improves the accuracy of the clean algorithm, because the positions of image maxima will be closer to the actual source positions. This factor of 5 is normally taken into account in the overall imaging resolution, i.e. the image size is increased during inversion. For the w-projection method, the cost of inversion with an increased resolution is small, because it does not increase the size of the w-kernels. It will affect the inverse FFT, but the relative cost of this step is negligible in the total cost. The w-stacking method is however significantly affected by the image resolution, because it performs many inverse FFTs.

The spatial frequencies in the output image are band-limited by the synthesized beam. Therefore, as long as the uv-plane is sampled with at least the Nyquist frequency, a high-resolution image can be perfectly reconstructed from an inversion at lower resolution. Cleaning can be performed on the high-resolution image that is reconstructed from the low-resolution image. After the high-resolution image has been cleaned, a model is created at the same (high) resolution. Because the model consists of delta functions, the spatial frequencies in the model are not band-limited. Consequently, lowering the resolution of the model image will remove information from the model. However, only the low spatial frequencies of this image will be used in the prediction, because in uv-space only visibilities up to the corresponding maximum baseline length will be sampled. Therefore, as long as lowering the resolution does not modify the low-frequency components, the output of the prediction step will not change.

We have implemented an option in WSCLEAN to automatically decrease the inversion resolution to the Nyquist limit,

$$N_{\text{Nyquist}} = 2 \frac{N_{\text{pix}} S_{\text{pix}}}{S_{\text{synth beam}}}, \tag{10}$$

where N_{Nyquist} is the resolution in pixels used during inversion, N_{pix} is the requested number of pixels of the image along one side, S_{pix} is the requested angular pixel scale and $S_{\text{synth beam}}$ is the minimum angular size of the synthesized beam. Before cleaning, the low-resolution image is interpolated with a procedure consisting of: (i) a low-resolution FFT; (ii) zero padding; and (iii) a high-resolution inverse FFT. Such interpolation assumes the image to be periodic, but this is already assumed during the inversion process. After cleaning, the model is decimated by the inverse procedure, consisting of: (i) a high-resolution FFT; (ii) truncation; and (iii) a low-resolution inverse FFT. With common settings, this procedure decreases the inversion imaging resolution by a factor of 2.5. Without further correction, such a decrease would lower the positional accuracy with which visibilities are gridded on to the uv-plane. This can be corrected by increasing the oversampling rate, which has almost no effect on performance.

Recreating Fig. 3 using both the snapshot method and this optimization lowers the computational cost from 41 to 24 min. Of the 24 min, 77 per cent is spent on cleaning approximately 100 000 components. The outputs with and without lowering the inversion and prediction resolution do not visibly differ, but the difference between the residual images has an rms of 18 mJy beam^{-1}. This can be compared to a noise level of 67 mJy beam^{-1} in the original residual image and a peak flux in the restored image of 10.3 Jy beam^{-1}. Because the difference is mostly noise like, the difference could be caused by the non-linear behaviour of clean.

Table 2. Parameter values used during benchmarks, unless otherwise mentioned.

Array	MWA
Number of elements	128
Image size	3072×3072
Angular pixel size	$0.72'$
Number of visibilities	3.5×10^8
Time resolution	2 s
Frequency resolution	40 kHz
Observation duration	112 s
Bandwidth	30.72 MHz (768 channels)
Central frequency	182 MHz
ZA at phase centre	$10°$
Max w-value for phase centre	172λ (283 m)
Number of polarizations in set	4
Imaged polarization	pp ($\sim$XX)
Imaging mode	multifrequency synthesis
Weighting	uniform
Data size	18 GB

4 ANALYSIS

We will now analyse the performance and accuracy of our implementation, and compare it with the w-projection implementation in CASA. For the analysis, we use imaging parameters common for MWA imaging. When a specific parameter value is not mentioned, the settings from Table 2 are used. The number of w-projection planes in w-projection is kept equal to the number of w-layers in w-stacking, and is set to the right-hand side of equation (5). This yields 195 w-planes/layers at $10°$ ZA. Several configurations with large w-values fail to image with CASA, because CASA crashes during the imaging, presumably because the w-kernels become too large.

The software version of CASA is 'stable release 42.0, revision 26 465', which was released on 2013 September. For WSCLEAN, version 1.0 from 2014 February was used. The tests were run on a high-end desktop with 32 GB of memory and a 3.20-GHz Intel Core i7-3930K processor with six cores that can perform 138 giga-floating point operations per second (GFLOPS). The data are stored on a multidisc array with five spinning hard discs, which has a combined read rate of about 450 MB s^{-1}.

4.1 Accuracy

To assess the accuracy of WSCLEAN and CASA's clean task, we simulate an MWA observation with 100 sources of 1 Jy in a $20°$ diameter area, without adding system noise. A unitary primary beam is assumed. We image the simulated set with WSCLEAN and CASA using Cotton–Schwab cleaning to a threshold of 10 mJy. The two imagers calculate slightly different restoring (synthesized) beams, hence to avoid bias the restoring beams are fixed. Other imaging parameters are given in Table 2. The AEGEAN program (Hancock et al. 2012) is used to perform source detection on the produced images. Sidelobe noise of the residual 10 mJy source structures triggers a few false detections. These are ignored.

Table 3 lists the measured root mean square (rms) in the residual image and the standard errors of the source brightnesses as detected by AEGEAN. WSCLEAN is more accurate: it shows 2–33 per cent lower errors in the source fluxes and produces 0–49 per cent lower rms noise compared to CASA. The large residual rms for CASA at zenith is caused by the fact that we keep the number of w-projection planes in w-projection equal to the number of w-layers in w-stacking, resulting in only 12 w-planes at zenith. This evidently has a stronger effect on the w-projection algorithm. However, the

Table 3. Results on imaging accuracy measurements.

	WSCLEAN	WSCLEAN + recentre	CASA
Zenith angle 0° (12 w-layers/planes)			
Source flux standard error	1.31 per cent	–	1.34 per cent
rms in residual image	0.94 mJy b^{-1}	–	1.90 mJy b^{-1}
Computational time	8.5 min	–	19.3 min
Zenith angle 0° (128 w-layers/planes)			
Source flux standard error	1.39 per cent	–	2.08 per cent
rms in residual image	0.94 mJy b^{-1}	–	0.94 mJy b^{-1}
Computational time	10.3 min	–	19.6 min
Zenith angle 10° (195 w-layers/planes)			
Source flux standard error	1.75 per cent	1.40 per cent	2.41 per cent
rms in residual image	0.90 mJy b^{-1}	1.03 mJy b^{-1}	1.07 mJy b^{-1}
Computational time	15.3 min	6.6 min	178.2 min

computational performance of the w-projection algorithm is hardly affected by the number of w-projection planes, and in practical situations one would always use more w-projection planes. When 128 w-projection planes are used in CASA, the residual rms is equal to WSCLEAN with 12 w-layers, but the flux density measurements are less accurate. We do not know why this parameter needs to be higher in CASA to reach the same rms. We are using enough w-planes to cover the sources: equation (5) results in $N_{w\text{-lay}} \gg 1$ for the source furthest from the phase centre. Also unexpected is that the source flux density becomes worse by increasing the number of planes. For WSCLEAN, both values stay approximately the same when the number of w-layers is increased.

In the ZA $= 10°$ case, CASA is slightly less accurate. Extra noise can be seen in the images, as shown in Fig. 6. An image resulting from the technique of recentring a zenith phase-centred visibility set, as described in Section 2.3, is also made and analysed. As can be seen in Table 3, the source fluxes in the recentred image have smaller errors compared to the normal projection, but the residual noise is higher. The recentring technique needs on average fewer w-layers to reach the same level of accuracy. We have performed the tests with and without the optimization of Section 3.5. They yield identical numbers.

WSCLEAN is faster in all tested cases. Both imagers perform five major iterations for these results, and the inversions and predictions

dominate the computing time. We will look more closely at the differences in performance in the next section.

4.2 Performance analysis

We measure the performance of the imagers using several MWA data sets. Each specific configuration is run five times and standard deviations are calculated. The variation in duration between runs is typically a few seconds. In each benchmark, the wall-clock time is measured that is required to produce the synthesized PSF and the image itself. No cleaning or prediction is performed, and the optimization of Section 3.5 is not used. The results are given in Fig. 7.

Fig. 7(a) shows the dependence on the size of the visibility set. Results for imaging at zenith and 10° ZA are shown. The size of the visibility set was varied by changing the time resolution, which affects the number of visibilities to be gridded without changing the maximum w-value. For larger sets, both methods show a linear time dependence on the number of visibilities. This implies that gridding or reading dominates the cost. In that situation, the w-stacking implementation is 7.9 times faster than CASA at ZA $= 10°$ and 2.6 times faster at ZA $= 0°$. With small data volumes, the FFTs start to dominate the cost, visible in Fig. 7(a) as a flattening towards the left. At that point, WSCLEAN is in both cases approximately three times faster. At zenith, the two methods are expected to perform almost identically. The factor of 2–3 difference in Fig. 7(a) could be due to different choices in optimizations.

In Fig. 7(b), the computational cost as a function of ZA is plotted. It shows that, as expected from Section 2.2, compared to w-stacking the w-projection algorithm is more affected by the increased w-values, leading to differences of more than an order of magnitude at ZAs of $\gtrsim 20°$. Additionally, at higher ZAs, the w-kernels become too large to be able to make images of 3072 pixels or larger. Using the recentring technique with WSCLEAN to decrease the w-values makes the computational cost approximately constant. However, the additional time required to rephase and regrid the measurement set makes recentring only worthwhile for images ≥ 3072 pixels and ZA $\geq 15°$. This is dependent on the speed of rotating the phase centre and regridding. Our code to change the phase centre is currently not multithreaded, so its performance can be improved. Furthermore, the relative cost of changing the phase centre is lower when performing multiple major iterations.
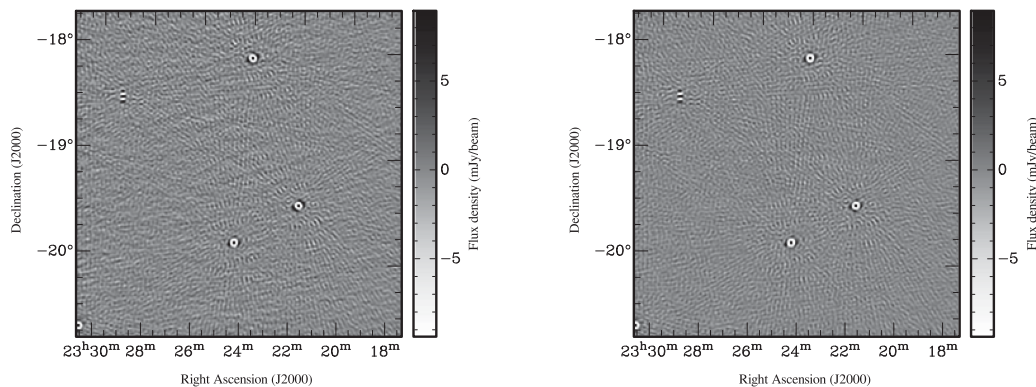
Figure 6. Residual images after Cotton–Schwab cleaning of a simulated 10°-ZA MWA observation using CASA (left) and WSCLEAN (right), using similar inversion and cleaning parameters. The panels show a small part of the full images. The full field contains 100 simulated sources over 20°. Sources in the image produced with CASA are slightly less accurately subtracted, leading to a residual noise level of 1.07 mJy beam^{-1} and some visible artefacts, whereas WSCLEAN reaches 0.90 mJy beam^{-1} rms noise. Since the effect is stronger away from the phase centre, it is likely that this is caused by the finite size of the w-kernels used in w-projection, which leads to inaccuracies. The ring-shaped residuals are caused by imperfect deconvolution.
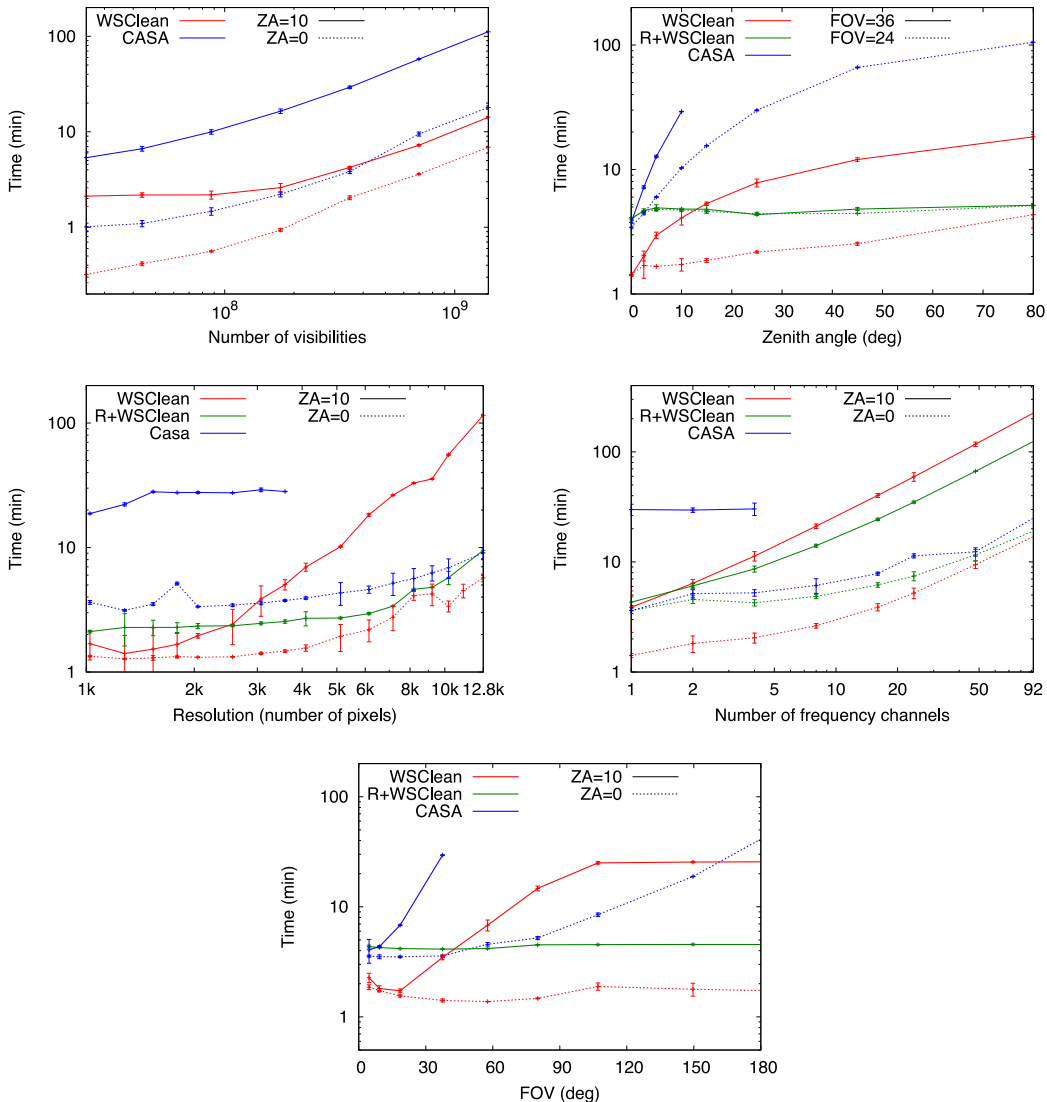
Figure 7. Imaging performance as a function of several parameters. Error bars show 5σ level. Unfinished lines indicate the imager could not run the specific configuration successfully. The label 'R+wsclean' refers to using wsclean with the recentring technique described in Section 2.3.

In Fig. 7(c), the number of pixels in the image is changed without changing the FOV. The performance of w-stacking is more affected by the size of the image, but is still significantly faster in making 12.8 K images at $0°$ ZA than w-projection. Imaging the MWA primary beam requires approximately an image size of 3072 pixels for cleaning. If the small-inversion optimization of Section 3.5 is used, the inversion can be performed at an image size of 1500 pixels. At ZA=10°, this saves about a factor of 2 in computing cost. The benefit of the optimization increases with resolution: at an image size of 10 000 pixels, cost is decreased by an order of magnitude.

The cost of spectral imaging, i.e. imaging multiple frequencies, is the cost of making an image from fewer visibilities multiplied by the number of desired frequencies. Fig. 7(d) shows performance versus the spectral output resolution. As can be expected from the previous results, FFTs become the dominant cost for such small visibility sets, and the number of imaged frequencies affect performance linearly.

As can be seen in Fig. 7(e), the FOV has a large effect on performance when imaging off-zenith. This plot was created by varying the size of a pixel in the output image, such that the number of

pixels in the image did not change. The FOV is calculated as the angle subtended between the left- and right-most pixel in the image. casa is not able to make off-zenith images larger than ∼30°, and its imaging cost increases much more rapidly compared to the cost of wsclean, which implies that gridding is the major cost in this scenario. The image recentring technique is clearly beneficial for FOVs larger than ∼30°, and can even make a difference of an order of magnitude at very large FOVs of ∼90°.

4.3 Derivation of computing cost formulae

Based on the expected cost terms described in Sec. 2.2, we derive analytical functions for the casa and wsclean measurements using least-squares fitting. Several functions with different free parameters were tested, and formulae with minimum number of parameters are selected that still follow the trend of the measurements and have reasonably small errors. Measurements are weighted with the inverse standard deviation instead of the variance, because we found that the latter results in too much weight on fast configurations, leading to functions that represent the general trend less well. The

single outlying measurement for $N_{pix} = 1792$ with CASA in Fig. 7(c) is removed.

For the WSCLEAN time-cost function $t_{WSCLEAN}$ we find

$$t_{WSCLEAN}(w_{max}, N_M\text{vis}, N_{freq}, N_{kpix})$$

$$= N_{freq}\left(0.526N_{kpix}^2\log_2 N_{kpix}(w_{max} + 0.715) + 0.535\right)$$

$$+ 0.248N_M\text{vis} \qquad (11)$$

and for CASA, we find

$$t_{CASA}(w_{max}, N_{vis}, N_{freq}, N_{pix})$$

$$= N_{freq}\left(0.965N_{kpix}^2\log_2 N_{kpix}\right.$$

$$\left. + 0.0106N_M\text{vis}(w_{max}^2 + 40.1)\right) + 40.8. \qquad (12)$$

Parameter w_{max} is the maximum w-value and is estimated with

$$w_{max} = \frac{1}{\lambda}\left[(D\sin ZA + \Delta z_{max}\cos ZA + \xi)\right.$$

$$\left. \times \left(1.0 - \cos\left(\frac{1}{2}FOV\right)\right)\right], \qquad (13)$$

where D is the maximum baseline length, Δz_{max} is the maximum height difference between antennas and λ is the wavelength, all in metres, and ξ is an extra parameter for fitting the CASA measurements, $\xi_{CASA} = 28.4$ and $\xi_{WSCLEAN} = 0$. These functions follow the trend of the measurements well, with an absolute error of 14.8 and 20.7 per cent for the WSCLEAN and CASA functions, respectively.

4.4 Optimal snapshot duration

Snapshot imaging, such as the recentring technique described in Section 2.3, can be implemented with either w-projection or w-stacking. Using the derived formulae, we can estimate the cost of making snapshots with both techniques. Snapshot imaging effectively removes the dependence on ZA from w_{max}. Given the snapshot duration $\Delta\tau_{snapshot}$ and the total integration time $\Delta\tau_{total}$, the total time cost of inversion becomes

$$t_{snapshot}(\Delta\tau_{snapshot})$$

$$= \frac{\Delta\tau_{total}}{\Delta\tau_{snapshot}}t_{imaging}(w'_{max}, N'_{vis}, N_{freq}, N_{pix}), \qquad (14)$$

with

$$w'_{max} = \frac{1}{\lambda}\max\Delta z\left(1.0 - \cos\left(\frac{1}{2}(FOV + \omega_E\Delta\tau_{snapshot})\right)\right),$$

$$(15)$$

ω_E the rotational speed of the Earth and $N'_{vis} = N_{vis}\frac{\Delta\tau_{total}}{\Delta\tau_{snapshot}}$. We have excluded the cost of phase shifting the visibilities and gridding. The cost for regridding with simple nearest neighbour or bilinear interpolation is indeed negligible, although more accurate interpolation (e.g. Lanczos interpolation) can be expensive. Also, our current implementation of the phase-changing program does take a non-negligible time, but this implementation is not optimized and can in theory be implemented in a pre-processing pipeline or on-the-fly during imaging.

The function $t_{snapshot}$ can be minimized to find the optimal snapshot duration. For WSCLEAN with MWA parameters, we find that this function decreases but no minimum is reached for $\Delta\tau_{snapshot} < 24$ h.

Therefore, from a performance perspective, the snapshot duration should be as large as possible. In practice, the beam needs to be corrected on small time-scales. A snapshot duration of more than a few minutes is therefore not possible. For w-projection in CASA, we find an optimal snapshot duration of $\sim$2 min for MWA observations.

4.5 Combination of w-projection and w-stacking

To combine the w-projection and w-stacking algorithms, small w-corrections are made before the FFTs using a w-correcting kernel and large w-terms are corrected after the FFTs by gridding on to several w-layers. This allows using small w-kernels during the w-projection stage and limits at the same time the number of FFTs and required memory that pure w-stacking would require. The AWIMAGER has been applied in this way, which led to better results compared to w-projection without w-stacking (Tasse et al. 2013). For this scenario, equation (12) can be used to determine the optimal number of w-layers, or more generally the optimal distance between layers, by assuming that the cost of calculating a single w-layer equals the cost of imaging the data set with a correspondingly smaller w_{max} value and smaller N_{vis}. If Δw is the distance between w-layers, then

$$t(\Delta w) = \frac{w_{max}}{\Delta w}t_{CASA}(w'_{max}, N'_{vis}, N_{freq}, N_{pix}), \qquad (16)$$

with $N'_{vis} = N_{vis}\frac{\Delta w}{w_{max}}$ and $w'_{max} = \Delta w$. For MWA observations, the optimal value for Δw is about unity, and improves the speed of w-projection by a factor of 4. However, the performance of the hybrid method is still approximately a factor of 2 lower than our pure-stacking implementation. This could again be the difference in optimization choices between CASA and WSCLEAN, but it does show that the effort of implementing a hybrid over pure stacking might not be worthwhile.

An optimization suggested by Tasse et al. (2013) is to not grid visibilities with w-values larger than some value, because this is where most of the computational cost resides when using w-projection, while for LOFAR there is little benefit in gridding these samples. In w-stacking, the speed gain associated with this optimization is less significant. Limiting the w-values also lowers the snapshot resolution in one direction significantly, because the long baselines in one direction are no longer gridded, which is often not desirable for the MWA.

4.6 Estimated computing cost for other telescopes

We use the derived functions to estimate the computational cost of the algorithms for configurations of several telescopes. We estimate the cost of imaging an hour of ZA=20° data for the following surveys or array configurations: The Galactic and Extragalactic MWA survey (GLEAM); the Evolutionary Map of the Universe (EMU) ASKAP survey (Norris et al. 2011); the low and high bands of the LOFAR Multifrequency Snapshot Sky Survey (MSSS);[3] all Dutch LOFAR stations (van Haarlem et al. 2013); the Amsterdam–ASTRON Radio Transients Facility and Analysis Centre (AART-FAAC) project;[4] The VLA Low-Frequency Sky Survey (VLSS; Cohen et al. 2007); The MeerKAT; and the low-frequency Phase 1 aperture arrays of the SKA, with the full core (3 km) and the core+arms

---

[3] See www.astron.nl/radio-observatory/lofar-msss/lofar-msss

[4] See www.aartfaac.org

Table 4. Configurations for which the computational cost is predicted. Columns: λ = wavelength; FOV = field of view; Beams = number of beams; Ant = number of elements; Res = angular resolution; D = maximum baseline; max Δz = maximum differential elevation; Δt = correlator dump time; BW = bandwidth; and $\Delta \nu$ = correlator frequency resolution.

Configuration	λ (m)	FOV (FWHM)	Beams	Ant	Res	D (km)	max Δz (m)	Δt (s)	BW (MHz)	$\Delta \nu$ (kHz)
GLEAM	2	24.7°	1	128	2′	2.9	5	2	32	40
EMU	0.2	1°	30	36	10″	6	0.2	10	300	20
MSSS low	5	9.8°	5	20	100″	5	2	10	16	16
MSSS high	2	3.8°	5	40	120″	5	2	10	16	16
LOFAR LBA NL	5	4.9°	1	38	3″	180	20	1	96	1
AARTFAAC	5	45°	1	288	20′	0.3	0.5	1	7	24
VLSS	4.1	14°	1	27	80″	11.1	0.2	10	1.56	12.2
MeerKAT	0.2	1°	1	64	6″	8	1	0.5	750	50
SKA1 AA core	2	5°	1	866	3′	3	5	10	250	1
SKA1 AA full	2	5°	1	911	5″	100	50	0.6	250	1

Table 5. Predicted computational costs for configurations listed in Table 4, based on multifrequency synthesis with five major iterations observing for 1 h at ZA = 20° with 1 polarization. Predictions are for w-projection with CASA; w-stacking with WSCLEAN; w-stacked snapshots with WSCLEAN using optimal snapshot duration; and a hybrid between w-projection and w-stacking using CASA with optimal Δw. All have approximately equal accuracy.

	Predicted computing cost on test computer				min	best
Configuration	w-projection	w-stacking	w-snapshot	hybrid	computing	FLOPS/float
GLEAM	65h 25m	8h 04m	8h 03m	14h 10m	1.1 TFLOPS	6.8×10^2
EMU	5.2 d	2.9 d	2.9 d	4.4 d	9.7 TFLOPS	2.1×10^4
MSSS low	2h 16m	0h 57m	0h 57m	2h 10m	130 GFLOPS	3.4×10^3
MSSS high	7h 16m	3h 52m	3h 52m	5h 44m	530 GFLOPS	3.4×10^3
LOFAR NL	50.2 d	7.3 d	7.2 d	12.9 d	24 TFLOPS	7.2×10^2
AARTFAAC	2.5 d	1.2 d	1.2 d	2.5 d	4.1 TFLOPS	6.8×10^2
VLSS	0h 09m	0h 01m	0h 01m	0h 08m	2.2 GFLOPS	1.1×10^3
MeerKAT	10.8 d	6.2 d	6.2 d	10.8 d	21 TFLOPS	6.8×10^2
SKA1 AA core	1581 d	911 d	911 d	1570.8	3.0 PFLOPS	6.8×10^2
SKA1 AA full	643 yr	48.8 yr	48.8 yr	84.1 yr	59 PFLOPS	6.8×10^2

configurations (Dewdney et al. 2013). The configurations are summarized in Table 4. In multibeam configurations such as the EMU and LOFAR configurations, we assume each beam is imaged separately, i.e. the FOV of a single beam is used and the computing cost is multiplied with the number of beams. The selected wavelengths are approximately the central wavelength available for each instrument. The image size is set to two times the FOV width divided by the resolution, as described by equation (10). Therefore, this assumes that the optimization of Section 3.5 to compute the inverse at lower resolution is used. The total required computing power is calculated by multiplying the estimated computing time on our test machine with the performance of the machine (138 GFLOPS).

The results are summarized in Table 5. WSCLEAN is in most situations predicted to be 2–3 times faster than CASA. WSCLEAN has the largest benefit on the full LOFAR, MWA and the full SKA, where WSCLEAN is 7, 8 and 12 times faster, respectively. The w-snapshot method does not improve the performance much over normal WSCLEAN operation in any of the cases. This might seem to differ from some of the results in Fig. 7, where the w-snapshot method does show improvement in certain cases. This is because Fig. 7 tests somewhat more extreme parameters, in which the w-snapshot method shows more benefit. The w-snapshot might become more valuable at higher ZAs or when the image size needs to be larger than the half-power beam width. The hybrid method is in all situations approximately a factor of 2 slower than WSCLEAN. All configurations listed in Table 4, with the exception of VLSS and GLEAM,

have optimal values for Δw much smaller than one, suggesting the w-corrections should be done entirely by w-stacking instead of the w-stacking/projection hybrid method.

Imaging the full FOV with the full SKA becomes very expensive due to the high frequency and time resolution, and image size of 7.2 k × 7.2 k pixels. This translates to a computing power requirement of ∼60 PetaFLOPS.

5 CONCLUSIONS

We have shown that the w-stacking algorithm is well suited for imaging MWA observations. The WSCLEAN w-stacking implementation is faster than CASA's w-projection algorithm in all common MWA imaging configurations, giving up to an order of magnitude increase in speed at a relatively small ZA of 10°, and results in slightly lower imaging errors. Roughly speaking, for ZAs >15° or FOVs >35° snapshot imaging becomes faster. This can lead to performance improvements of a factor of 3 for the MWA, but only in the most expensive imaging configurations that are less commonly used. Considering the extra regridding step required, which complicates issues such as calculating the integrated beam shape, recentring snapshots is worthwhile for the MWA only at very low elevations or with large FOV. Extrapolation of the computing cost predicts that this holds for most arrays, including SKA low. A hybrid between w-stacking and w-projection does not improve performance over a pure w-stacking implementation, but does improve

a pure w-projection implementation significantly. Our optimization of lowering the image resolution during inversion and prediction increases performance by a factor of 2–10 and has no noticeable effect on the accuracy in either our test simulations, which reach approximately a dynamic range of 1:1000, or on the complicated field of Fig. 3.

The available SKA computing power is estimated to be around ∼100 PetaFLOPS. Extrapolation of our results shows that the current imagers require 3–60 petaFLOPS for SKA1 low alone. Although we have measured our performance in FLOPS, it is likely that memory bandwidth will be a limiting factor, because memory bandwidth is increasing less quickly compared to the floating point performance (e.g. Romein 2012). Cornwell et al. (2012) suggests that the w-snapshots algorithm improves the imaging speed for the SKA situation, but our results show that w-snapshot imaging does not improve SKA imaging performance over w-stacking. Clearly it remains challenging to perform wide-field imaging with acceptable performance, but some optimizations could be made for the SKA. For example, a performance gain of factors of a few can be achieved by averaging shorter baselines to their lowest time and bandwidth resolution before imaging. Moreover, although longer baselines make the imaging more expensive, imaging the full FOV will likely not (always) be required when using the longest baselines.

Deconvolution is not an issue for achieving low to intermediate dynamic ranges with the MWA. Because of sufficient instantaneous uv-coverage and near-confusion snapshot noise level, snapshots can be cleaned individually to deep levels. For example, cleaning to a 5σ level results in a cleaning threshold of approximately 100 mJy in 112 s observations. However, MWA's Epoch of Reionization (EoR) observations (Bowman et al. 2013) require more advanced deconvolution techniques.

So far, we have corrected the (full-polarization) beam in image space, which is possible because of the homogeneity of the MWA tiles. This is computationally cheap for our current snapshot time of 112 s, but this might be infeasible when it is required to calculate the beam on very short time-scales. Accurate MWA beam models are still being developed (Sutinjo et al., in preparation). When beam corrections are required on short time-scales, calculating a-projection kernels or performing snapshot imaging also becomes more expensive, and it would be interesting to find out where the balance lies between these methods. When only a few thousand components need to be deconvolved, an approach with direct Fourier transforms is accurate and affordable. For wide-field arrays this can be combined with calibration of the direction dependence, e.g. with the SAGECAL (Kazemi et al. 2011) or RTS peeling (Mitchell et al. 2008) calibration techniques. This is not possible for fields with diffuse emission or faint point sources when no model is known a priori.

WSCLEAN is currently not able to perform multiscale clean. Results of applying CASA's multiscale clean (Cornwell 2008) on MWA data show that especially the Galactic plane is significantly better deconvolved using multiscale clean, but it is very computationally expensive. Imaging a one-minute observation takes 30 h of computational time without w-projection. We plan to implement some form of multiscale cleaning in WSCLEAN. It is likely that additional optimizations need to be made to be able to do this with acceptable performance. Combining multiscale with wide-band deconvolution techniques is a possible further improvement (Rau & Cornwell 2011).

By using the w-stacking algorithms, some computational cost is transferred from gridding to performing FFTs. WSCLEAN uses the FFTW library for calculating the FFTs (Frigo & Johnson 2005). Further performance improvement can be made by using one of the available FFT libraries that make use of GPUs.

REFERENCES

Bernardi G., Mitchell D. A., Ord S. M., Greenhill L. J., Pindor B., Wayth R. B., Wyithe J. S. B., 2011, MNRAS, 413, 411
Bhatnagar S., Cornwell T. J., Golap K., Uson J. M., 2008, A&A, 487, 419
Bowman J. D. et al., 2013, PASA, 30, e031
Calabretta M., Greisen E., 2002, A&A, 395, 1077
Clark B. G., 1980, A&A, 89, 377
Cohen A. S., Lane W. M., Cotton W. D., Kassim N. E., Lazio T. J. W., Perley R. A., Condon J. J., Erickson W. C., 2007, AJ, 134, 1245
Cornwell T., 2008, IEEE J. Sel. Top. Signal Process., 2, 793
Cornwell T. J., Perley R. A., 1992, A&A, 261, 353
Cornwell T. J., Golap K., Bhatnagar S., 2008, IEEE J. Sel. Top. Signal Process., 2, 647
Cornwell T. J., Voronkov M. A., Humphreys B., 2012, Proc. SPIE Conf. Ser. Vol. 8500, Image Reconstruction from Incomplete Data VII. SPIE, Bellingham, p. 85000L
Dewdney P., Turner W., Millenaar R., McCool R., Lazio J., Cornwell T. J., 2013, SKA office, available at: https://www.skatelescope.org/uploaded/21705_130_Memo_Dewdney.pdf
Frigo M., Johnson S., 2005, Proc. IEEE, 93, 216
Gooch R., 1996, in Jacoby G. H., Barnes J., eds, ASP Conf. Ser. Vol. 101, Astronomical Data Analysis Software and Systems V. Astron. Soc. Pac., San Francisco, p. 80
Greisen E., Calabretta M., 2002, A&A, 395, 1061
Hancock P. J., Murphy T., Gaensler B. M., Hopkins A., Curran J. R., 2012, MNRAS, 422, 1812
Högbom J. A., 1974, A&AS, p. 417
Humphreys B., Cornwell T. J., 2011, SKA MEMO 132, available at: http://www.skatelescope.org/PDF/memos/132_Memo_Humphreys.pdf

Jackson J., Meyer C., Nishimura D., Macovski A., 1991, IEEE Trans. Med. Imaging, 10, 473

Jaegar S., 2008, in Argyle R. W., Bunclark P. S., Lewis J. R., eds, ASP Conf. Ser. Vol. 394, Astronomical Data Analysis Software and Systems. Astron. Soc. Pac., San Francisco, p. 623

Kazemi S., Yatawatta S., Zaroubi S., Labropoulos P., de Bruyn A. G., Koopmans L. V. E., Noordam J., 2011, MNRAS, 414, 1656

Lonsdale C., 2004, MIT Haystack, Tech. Rep. LFD memo 015

Lonsdale C. J. et al., 2009, Proc. IEEE 97, 1497

McMullin J. P., Waters B., Schiebel D., Young W., Golap K., 2007, in Shaw R. A., Hill F., Bell D. J., eds, ASP Conf. Ser. Vol. 376, Astronomical Data Analysis Software and Systems XVI. Astron. Soc. Pac., San Francisco, p. 127

Mitchell D., Greenhill L., Wayth R., Sault R., Lonsdale C., Cappallo R., Morales M., Ord S., 2008, IEEE J. Sel. Top. Signal Process., 2, 707

Norris R. P. et al., 2011, PASA, 28, 215

Offringa A. R., de Bruyn A. G., Biehl M., Zaroubi S., Bernardi G., Pandey V. N., 2010, MNRAS, 405, 155

Offringa A. R., de Bruyn A. G., Zaroubi S., 2012a, MNRAS, 422, 563

Offringa A. R., van de Gronde J. J., Roerdink J. B. T. M., 2012b, A&A, 539, A95

Ord S. M. et al., 2010, PASP, 122, 1353

Perley R. A., 1999, in Taylor G. B., Carilli C. L., Perley R. A., eds, ASP Conf. Ser. Vol. 180, Imaging in Radio Astronomy II, A Collection of Lectures from the Sixth NRAO/NMIMT Synthesis Imaging Summer School, Astron. Soc. Pac., San Francisco, p. 383

Pindor B., Wyithe J. S. B., Mitchell D. A., Ord S. M., Wayth R. B., Greenhill L. J., 2011, PASA, 28, 46

Rau U., Cornwell T. J., 2011, A&A, 532, A71

Romein J. W., 2012, Proc. 26th ACM Int. Conf. Supercomputing ICS '12, An Efficient Work-Distribution Strategy for Gridding Radio-Telescope Data on GPUs. ACM, New York, NY, p. 321

Schwab F. R., 1983, in Roberts J., ed., Proc. Int. Symp. Indirect Imaging, Optimal Gridding of Visibility Data in Radio Interferometry. Cambridge Univ. Press, Cambridge, p. 333

Schwab F. R., 1984, AJ, 89, 1076

Smirnov O. M., 2011, A&A, 527, A106

Tasse C., van der Tol S., van Zwieten J., van Diepen G., Bhatnagar S., 2013, A&A, 553, A105

Tingay S. J. et al., 2013, PASA, 30, e007

van Haarlem M. P. et al., 2013, A&A, 556, A2

[1]*Research School of Astronomy and Astrophysics, Australian National University, Canberra, ACT 2611, Australia*

[2]*ARC Centre of Excellence for All-sky Astrophysics (CAASTRO), Australian National University, Canberra, ACT 2611, Australia*

[3]*International Centre for Radio Astronomy Research, Curtin University, Bentley, WA 6102, Australia*

[4]*Department of Physics, University of Wisconsin–Milwaukee, Milwaukee, WI 53201, USA*

[5]*Sydney Institute for Astronomy, School of Physics, The University of Sydney, NSW 2006, Australia*

[6]*Kavli Institute for Astrophysics and Space Research, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*

[7]*Square Kilometre Array South Africa (SKA SA), Cape Town 7405, South Africa*

[8]*School of Earth and Space Exploration, Arizona State University, Tempe, AZ 85287, USA*

[9]*MIT Haystack Observatory, Westford, MA 01886, USA*

[10]*Raman Research Institute, Bangalore 560080, India*

[11]*Harvard-Smithsonian Center for Astrophysics, Cambridge, MA 02138, USA*

[12]*Department of Physics, University of Washington, Seattle, WA 98195, USA*

[13]*School of Chemical & Physical Sciences, Victoria University of Wellington, Wellington 6140, New Zealand*

[14]*University of Michigan, Ann Harbor, MI 48109, USA*

[15]*CSIRO Astronomy and Space Science, Marsfield, NSW 2122, Australia*

[16]*National Centre for Radio Astrophysics, Tata Institute for Fundamental Research, Pune 411007, India*

[17]*School of Physics, The University of Melbourne, Parkville, VIC 3010, Australia*

[18]*National Radio Astronomy Observatory, Charlottesville and Greenbank, Charlottesville, VA 22903-2475, USA*

This paper has been typeset from a TeX/LaTeX file prepared by the author.