

TDP CALIBRATION & PROCESSING GROUP CPG
MEMO
COMPUTATIONAL COSTS OF RADIO IMAGING
ALGORITHMS DEALING WITH THE NON-COPLANAR
BASELINES EFFECT: I

Mark Yashar and Athol Kemball

October 19, 2009

Contents

1	Introduction	2
2	Visibility Data Rates	2
3	Imaging Algorithms	4
3.1	3-D Imaging	5
3.1.1	3-D Direct Fourier Transform (Fourier Sums)	6
3.1.2	3-D Fourier Transform implemented with FFT	7
3.1.3	UVW-space facets (Polyhedron Method)	9
3.1.4	W-Projection	10
3.1.5	Hybrid UVW-Space Facets and W-Projection Approach	12
4	Comparisons and Plots	13

Abstract

We summarize the computational costs in forming dirty images of a number of existing (wide-field) radio interferometry imaging algorithms used to deal with non-coplanar baselines in wide-field radio interferometry, including Fourier Sums, 3-D FFT, facets/polyhedron imaging, w-projection, and a hybrid facets and w-projection approach. We will consider computational costs associated with deconvolution in a future memo.

1 Introduction

A number of imaging algorithms for dealing with the non-coplanar baselines effect in wide-field radio synthesis imaging have been proposed and implemented [13]. For new radio telescope projects such as the Square Kilometer Array (SKA), the computational costs are predicted to be large relative to current practice. The computational costs will depend on the data rate, the computational intensity, and the efficiency of the parallel computing implementation, η , by way of the following equation

$$\text{computational cost} = (\text{data rate}) \times (\text{computational intensity}) \times \eta^{-1}, \quad (1)$$

where the units for computing intensity are in operations per byte [1, 2]. In this memo we estimate the computational costs and loads for the implementation of the direct (3-D) Fourier Transform (Fourier Sum), 3-D Fast Fourier Transform (FFT), UVW-Space Facets, and the W-Projection imaging algorithms, along with the use of UVW-Space Facets together with the W-Projection algorithm, in the formation of dirty images.

2 Visibility Data Rates

The total number of baselines in an array of N_{ant} antennas is given by

$$N_{bas} = \frac{N_{ant}(N_{ant} - 1)}{2} + N_{ant}, \quad (2)$$

comprising a sum of terms for cross-correlation and auto-correlation data, respectively. Measured visibilities are sampled along uv tracks (i.e., the trajectory that a baseline defined by (u, v, w) generates due to the earth's rotation), with coordinates of projected baseline (u, v, w) , frequency channel, and time. The intrinsic data rates for Large- N_{ant} Small Diameter (LNSD) SKA array designs are inherently large. For an array consisting of N_{ant} antennas amongst which all possible correlations are made for each of N_{pol} polarizations (and dropping auto-correlation considerations here and for the rest of the memo since $N_{ant} \ll N_{ant}(N_{ant} - 1)$ for large N_{ant}), over N_{chan} spectral channels, and using a correlator integration time of t_{int} seconds, the visibility output rate in bytes per second can be expressed as [5, 6, 8]

$$\dot{V} = \frac{N_{chan} N_{pol} N_{ant} (N_{ant} - 1) b_v}{2t_{int}} \quad (3)$$

where b_v is the number of bytes per visibility. It is assumed that each complex visibility value is written as two floating point numbers with an extra floating point number needed for visibility weight. For the customary assumption of floating point single-precision (IEEE-754 32-bit), $b_v = 12$. For compressed representations (with an associated reduction in dynamic range) b_v can be reduced to $\sim 4 - 6$ bytes per visibility. In terms of terabytes per second (and taking N_{pol} to be 4), we have

$$\dot{V} = \frac{2N_{chan}N_{ant}(N_{ant} - 1)b_v}{10^{12}t_{int}} \left(\frac{TB}{s} \right). \quad (4)$$

To avoid bandwidth smearing, the employment of narrow frequency channels becomes necessary. As discussed in Section 6.3 of [9], for a source angular offset from the field center, $\theta \ll 1$ radian, synthesized beam-width θ_b , and a bandwidth $\Delta\nu$ centered on frequency ν_0 , control of bandwidth smearing requires:

$$\frac{\theta}{\theta_b} \frac{\Delta\nu}{\nu_0} \ll 1. \quad (5)$$

The synthesized beam-width is given by $\theta_b \sim \frac{\lambda}{B_{max}}$, where B_{max} is the maximum baseline length. For wide-field imaging considered here $\theta \sim \frac{\lambda}{D}$, where D is the diameter of the antenna. Adopting a quality factor Q_ν (with increasing values of Q_ν for lower bandwidth smearing) and a channel count N_{chan} over the bandwidth $\Delta\nu$, Eq. (5) becomes:

$$\frac{B_{max}}{D} \frac{\Delta\nu}{\nu} \frac{1}{N_{chan}} = \frac{1}{Q_\nu}. \quad (6)$$

This defines the required channel count:

$$N_{chan}^{bw} = Q_\nu \frac{B_{max}}{D} \frac{\Delta\nu}{\nu}. \quad (7)$$

To minimize time-smearing requires [9]:

$$\frac{\theta}{\theta_b} \omega_e t_{int} \ll 1, \quad (8)$$

where $\omega_e = 7.27 \times 10^{-5}$ rad/sec is the angular rotation rate of the earth. Adopting a similar quality factor for time-smearing, Q_t , and substituting for θ and θ_b as above, we obtain:

$$\frac{1}{t_{int}} = \frac{B_{max}}{D} \omega_e Q_t. \quad (9)$$

Substituting Eqs. (7) and (9) into Eq. (4), setting $Q_\nu = Q_t = 10$, assuming uniform resolution across the field, and assuming $b_v \sim 10$, we obtain the following estimate for the visibility output rate in terms of maximum baseline length, dish diameter, frequency ν , and bandwidth $\Delta\nu$ [5, 6, 8]:

$$\dot{V} \sim 10^{-14} \left(\frac{N_{ant}B_{max}}{D} \right)^2 \left(\frac{\Delta\nu}{\nu} \right) \left(\frac{TB}{s} \right). \quad (10)$$

Multiplying Eq. (10) by the observing time t_{obs} gives the total amount of data (in bytes) in time t_{obs} .

Assuming a constant-sensitivity array, the number of antennas scales as $N_{ant} \propto \frac{1}{D^2}$, and Eq. (10) becomes:

$$\dot{V} \sim 10^{-14} \frac{B_{max}^2}{D^6} \left(\frac{\Delta\nu}{\nu} \right) \left(\frac{TB}{s} \right). \quad (11)$$

For spectral line observing modes, the observed spectral channel count N_{chan}^{obs} over the bandwidth $\Delta\nu$ may be higher than that required to avoid bandwidth smearing alone (as defined by Eq. (7)). In this case, Eq. (10) takes the form:

$$\dot{V} \sim 10^{-15} \left(\frac{B_{max}}{D} \right) N_{ant}^2 N_{chan}^{obs} \left(\frac{TB}{s} \right). \quad (12)$$

For SKA spectral line surveys, the spectral line count is assumed to be $N_{chan}^{obs} = 16384$ [10]. In the remainder of this memo, we will consider computational costs for SKA continuum observing modes only, with visibility output rate given by Eq. (10) and (11), but these can be scaled to spectral line observing modes to first order by multiplying by the ratio between Eq. (12) and (10), $\frac{N_{chan}^{obs}}{10 \frac{B_{max} \Delta\nu}{D \nu}}$.

3 Imaging Algorithms

Generating a sky image itself translates into reconstructing the sky brightness I at a given frequency and as a function of angular coordinates (l, m) on the sky using the measured visibility data. The relationship between the 2-dimensional sky brightness $I(l, m)$ and the 3-dimensional visibility $V(u, v, w)$ is given by [9]

$$V(u, v, w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{A_N(l, m) I(l, m) e^{[-2\pi i(u l + v m + w(\sqrt{1-l^2-m^2}-1))]} }{\sqrt{1-l^2-m^2}} dldm, \quad (13)$$

where $A_N(l, m)$ is the normalized primary beam pattern of an antenna and l, m, n are direction cosines. These are the visibilities measured by an interferometer of baseline $\vec{b} \equiv (u, v, w)$. When the term $2\pi w(\sqrt{1-l^2-m^2}-1) \ll 1$ (also corresponding to $l^2+m^2 \ll 1$) or $w = 0$, corresponding to a narrow field of view in which all the baselines are in the same (u, v) plane, we use the 2D form of the visibility-brightness relation,

$$V(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{A_N(l, m) I(l, m) e^{[-2\pi i(u l + v m)]}}{\sqrt{1-l^2-m^2}} dldm, \quad (14)$$

which shows that $V(u, v)$ and $I(l, m)$ are now a 2D Fourier pair. Therefore, we can recover the brightness $I(l, m)$ from a 2-D inverse Fourier transform of $V(u, v)$ and a deconvolution process. In this memo on image formation costs we ignore visibility-plane or image-plane calibration. The details of the deconvolution process and associated computing loads and costs will be covered in a future memo.

For the more general case of non-coplanar baselines, in which $w \neq 0$ and $2\pi w(\sqrt{1-l^2-m^2}-1) \gtrsim 1$, a 2-D FT cannot be used. This is the case for the SKA for which wide-field,

low-frequency imaging will constitute a large part of the science case. In the following subsections, we discuss the computational loads and costs of imaging algorithms that deal with 3-D imaging and the non-coplanar baselines effect.

3.1 3-D Imaging

Eq. (13) may be embedded in a 3-dimensional space with axes (l, m, n) [25, 14, 13]

$$V(u, v, w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{A_N(l, m) I(l, m) \delta(n - \sqrt{1 - l^2 - m^2}) e^{[-2\pi i(ul + vm + wn)]}}{n} dl dm dn. \quad (15)$$

The 3-D FT of the sampled visibility defines a 3-D dirty image I_3^D as follows:

$$I_3^D(l, m, n) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(u, v, w) V(u, v, w) e^{[2\pi i(ul + vm + wn)]} du dv dw, \quad (16)$$

where $W(u, v, w)$ is a weighted, 3-D sampling function (see below). This can be expressed as a convolution function in the image plane [25, 14]

$$I_3(l, m, n) = \left[\frac{A_N(l, m) I(l, m) \delta(n - \sqrt{1 - l^2 - m^2})}{n} \right] * B^{3D}(l, m, n) \quad (17)$$

where $B^{3D}(l, m, n)$ is the 3-D PSF

$$B^{3D}(l, m, n) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(u, v, w) e^{[2\pi i(ul + vm + wn)]} du dv dw \quad (18)$$

convolved with a 3-D image cube $F(l, m, n)$ [25, 14, 24],

$$F(l, m, n) = \frac{A_N(l, m) I(l, m) \delta(n - \sqrt{1 - l^2 - m^2})}{n}, \quad (19)$$

as $I_3^D = F * B_D^3$. The basis of the 3-D imaging method is to form $I_3^D(l, m, n)$ and $B^{3D}(l, m, n)$ and then to solve for $F(l, m, n)$ using any number of deconvolution algorithms. After deconvolution, we must perform the additional operation of projecting all points in the image volume along the celestial sphere onto the 2-D tangent plane and onto an appropriate (l, m) grid to recover (an estimate of) the 2-D sky brightness distribution, $I(l, m)$ [24]. In conjunction with or in addition to this projection operation, an estimate of $I(l, m)$ can also be obtained, for example, by performing a least squares fit of a Gaussian function to a deconvolved and restored estimate of $F(l, m, n)$ (Eq. (20)) for slices in n , centered at $n = \sqrt{1 - l^2 - m^2}$ [11]. In this manner, then, we can obtain an estimate of the sky brightness using Eq. (17), i.e., the deconvolution is performed in the 3-D image volume, constrained to a unit sphere in the 3-D volume, and the result projected onto an appropriate (l, m) grid [25].

The observed visibilities are recorded at (u, v, w) points (u_k, v_k, w_k) , $k = 1, \dots, M$ (where M is the total number of visibility data points). We also take into account the number of

planes required in the n axis direction with a maximum field of view angular size $\theta_{max} = \sqrt{\frac{\lambda}{B_{max}}}$ and $\theta_{FWHM} \sim \frac{\lambda}{D}$ [11, 14, 25, 15]

$$N_{planes} = \left(\frac{\theta_{FWHM}}{\theta_{max}} \right)^2 = \left(\frac{\lambda}{D} \sqrt{\frac{B_{max}}{\lambda}} \right)^2 = \frac{\lambda B_{max}}{D^2}, \quad (20)$$

which is calculated from the requirement that the portion of the celestial sphere of interest must lie inside a 3-dimensional image cube [11] (and which also insures that critical sampling has been achieved), as described further in Section 3.1.1 below.

3.1.1 3-D Direct Fourier Transform (Fourier Sums)

For numerically approximating the 3-dimensional Fourier Transform (FT), we can use the 'direct FT' approach, which approximates the dirty image $I_3^D(l, m, n)$ for a discrete number of visibilities by brute force evaluation of the sum (and not an integral) via [16]

$$I_3^D(l, m, n) = \frac{1}{M} \sum_{k=1}^M V'(u_k, v_k, w_k) e^{2\pi i(u_k l + v_k m + w_k n)}. \quad (21)$$

where $V'(u_k, v_k, w_k) = W(u_k, v_k, w_k)V(u_k, v_k, w_k)$. This sum is then calculated for each pixel in the map. So, if there are M total visibility points in a data set, it is possible to image a $N_l \times N_m \times N_n$ (where N_i is the total number of pixels along the l, m, n axis) cube in this way via the direct FT. Therefore, the dirty image is calculated via a discrete representation $I_3^D(l, m, n)$ on a uniform cube. Then, if evaluated at every point of the $N_l \times N_m \times N_n$ cube, the number of complex operations required for the direct FT is $\sim MN^3$.

In order to assess the computational loads and costs of the 3-D direct FT approach, we must also estimate the associated computational intensity, which is defined as [1, 2]

$$computational\ intensity = \frac{total\ number\ of\ operations}{number\ of\ input\ and\ output\ points} \quad \left(\frac{operations}{data\ point} \right), \quad (22)$$

where the number of input and output data points is essentially the same as the number of data transmitted. Then, we estimate the computational intensity for this case to be N^3 complex operations per visibility, where $N^3 = N_l N_m N_n$. For this particular approach, N_{planes} is equivalent to the number of 2-D direct Fourier Transforms along the n axis. Then, for the 3-D direct FT, the total number of operations per visibility is $N_l N_m N_n$, where $N_n \sim \frac{\lambda B_{max}}{D^2}$ is the number of planes in the n direction. Now, given that the minimum angular resolution (i.e., the size of the synthesized beam in the n direction) is $\theta_{min} \sim \frac{\lambda}{B_{max}}$ and the field of view is given by solid angle $\Omega \sim \left(\frac{\lambda}{D}\right)^2$, the number of resolution elements $\sim \frac{\Omega}{\theta_{min}^2} \sim \left(\frac{B_{max}}{D}\right)^2$, and then the total number of volume pixels to be calculated, and hence the complex operations per visibility, is [25, 11, 14]

$$N_l N_m N_n \sim \frac{\Omega}{\theta_{min}^2} \times N_{planes} \sim \frac{4\lambda B_{max}^3}{D^4}, \quad (23)$$

where $N_l = N_m \sim \frac{2\theta_{FWHM}}{\theta_{max}} \sim 2\frac{B_{max}}{D}$. We include a factor of 2 here for minimizing aliasing effects for Nyquist-sampled imaging [7]. We also assume that we are imaging only the main lobe of the antenna power pattern [7]. Now, from Eqs. (1), (10), and (23), the computational costs (in terms of floating point operations per second (FLOPS)) for the 3-D direct Fourier Transform method without taking deconvolution into account will be:

$$10^{-2} \left(\frac{N_{ant} B_{max}}{D} \right)^2 \left(\frac{\Delta\nu}{\nu} \right) N_l N_m N_n \times \eta^{-1} \sim 4 \times 10^{-2} N_{ant}^2 \left(\frac{\Delta\nu}{\nu} \right) \frac{\lambda B_{max}^5}{D^6} \times \eta^{-1}, \quad (24)$$

3.1.2 3-D Fourier Transform implemented with FFT

The 3-D FT can be carried out using a Fast Fourier Transform in all axes, or if the range in n is small, an FFT can be used in (l, m) and a discrete Fourier Transform in n [25]. In order for the Fast Fourier Transform (FFT) algorithm to be used, the irregularly sampled visibility data $V(u, v, w)$ needs to be assigned to a regular, rectangular matrix or grid of cells, usually with a power-of-two number of points along each side [9]. This operation requires an interpolation procedure to be used to assign visibility values at the grid points, based on the observed values [9, 16]. In conventional gridding [9], the weighted, sampled visibility measurement distribution V^W are convolved with a gridding convolution function (GCF) C , and re-sampled at the center of each cell of the grid.

Following [17], we start with a model \hat{I} of source intensity capable of representing all visibility data. This model consists of a 3-D cube of δ -functions with strength $\hat{I}(p\Delta l, q\Delta m, r\Delta n)$, where $\Delta l, \Delta m, \Delta n$ are separations of 3-D grid elements in 3 orthogonal sky coordinates. The visibility \hat{V} predicted by this model is given by

$$\hat{V}(u, v, w) = \sum_{p=1}^{N_l} \sum_{q=1}^{N_m} \sum_{r=1}^{N_n} \hat{I}(p\Delta l, q\Delta m, r\Delta n) e^{-2\pi i(pu\Delta l + qv\Delta m + rw\Delta n)}. \quad (25)$$

To simplify the notation, we let $\hat{I}(p\Delta l, q\Delta m, r\Delta n) \equiv \hat{I}_{pqr}$. Then, the dimensions of the (u, v, w) space that contain the data are $(N_l\Delta u) \times (N_m\Delta v) \times (N_n\Delta w)$. In the (l, m, n) (map) space, the points are spaced Δl in l , Δm in m , and Δn in n , and the 3-D map dimensions are $(N_l\Delta l) \times (N_m\Delta m) \times (N_n\Delta n)$. The grid spacings $\Delta l, \Delta m, \Delta n$ and the number of pixels on each axis N_l, N_m, N_n must allow representation of all these scales. In terms of the range of (u, v, w) points sampled, the Nyquist sampling theorem requires that $\Delta l \leq \frac{1}{2u_{max}}$, $\Delta m \leq \frac{1}{2v_{max}}$, $\Delta n \leq \frac{1}{2w_{max}}$ (where $w_{max} = \frac{B_{max}}{\lambda}$), $N_l\Delta l \geq \frac{1}{u_{min}}$, $N_m\Delta m \geq \frac{1}{v_{min}}$, $N_n\Delta n \geq \frac{1}{w_{min}}$ in order to map an angular region of the sky of size $N_l\Delta l, N_m\Delta m, N_n\Delta n$. Then, from the sampling theorem, the dimensions in the two domains are related by:

$$\Delta u = (N_l\Delta l)^{-1}, \Delta v = (N_m\Delta m)^{-1}, \Delta w = (N_n\Delta n)^{-1} \quad (26)$$

$$\Delta l = (N_l\Delta u)^{-1}, \Delta m = (N_m\Delta v)^{-1}, \Delta n = (N_n\Delta w)^{-1}. \quad (27)$$

Now, the $N_m \times N_l \times N_n$ FFT algorithm generates a discretely sampled periodic version of I^D via a convolution relation in the image plane [17],

$$I_{p,q,r}^D = \sum_k^M W(u_k, v_k, w_k) \text{Re}[V(u_k, v_k, w_k) e^{2\pi i(pu_k\Delta l + qv_k\Delta m + rw_k\Delta n)}], \quad (28)$$

where $W(u, v, w)$ is the weighted, sampling function given by

$$W(u, v, w) = \sum_k^M W_k \delta(u - u_k, v - v_k, w - w_k), \quad (29)$$

where the grid spacings satisfy $N_l \Delta u = \frac{1}{\Delta \theta_l}$, $N_m \Delta v = \frac{1}{\Delta \theta_m}$, $N_n \Delta w = \frac{1}{\Delta \theta_n}$ wavelengths for imaging a 3-D rectangular region of dimensions $N_l \Delta \theta_l$ radians in l , $N_m \Delta \theta_m$ radians in m , and $N_n \Delta \theta_n$ radians in n . Moreover, the region over which I_3^D is computed (i.e., the primary field of view) is given by: $|l| < \frac{N_l \Delta \theta_l}{2}$, $|m| < \frac{N_m \Delta \theta_m}{2}$, $|n| < \frac{N_n \Delta \theta_n}{2}$. Following Eq. (17), Eq. (28) can also be expressed as

$$I_3^D(l, m, n) = \sum_k W_k V(u_k, v_k, w_k) e^{2\pi i(u_k l + v_k m + w_k n)}. \quad (30)$$

The 3-D FFT itself needs about $N_l N_m N_n [\log_2(N_l N_m N_n)]$ FLOPS to complete [2, 3, 4], where $N_l N_m N_n \sim \frac{4\lambda B_{max}^3}{D^4}$. As discussed previously, the gridding operations involve convolving the weighted, sampled measurement distribution V^W with a GCF, C , and sampling this convolution at the center of each cell of the grid. In practice, C is taken to be identically zero outside some small, bounded region A_C [9]. Since V^W is a linear combination of M δ functions (where M is the number of visibility data points), this convolution $C * V^W$, evaluated at the grid point (u_c, v_c, w_c) , is given by [9, 16]:

$$\sum_{k=1}^M C(u_c - u_k, v_c - v_k, w_c - w_k) V^W(u_k, v_k, w_k). \quad (31)$$

Since the region A_C is quite small in area, there are generally many fewer than M nonzero terms in this sum. The number of arithmetic operations required for gridding is $O(Mm^3)$, where the GCF is truncated to m grid cells so that $C(u) = 0$ for $|u| > \frac{m\Delta u}{2}$, where m is typically 7 to 9 [13]. We define $N_{GCF, points}$ to be the support of the gridding convolution on each of the u and v axes as N_{GCF} , such that $N_{GCF}^2 = m^2$. Then, following the same approach we took in the previous section for the 3-D direct FT, we have that the cost for the 3-D FFT continuum imaging method (including gridding convolution but not including deconvolution or parallel computing considerations at this time) scales as

$$\sim 10^{-2} \left(\frac{N_{ant} B_{max}}{D} \right)^2 \left(\frac{\Delta \nu}{\nu} \right) N_{GCF}^2 \frac{\lambda B_{max}}{D^2} + \frac{4\lambda B_{max}^3}{D^4} \log_2 \left(\frac{4\lambda B_{max}^3}{D^4} \right). \quad (32)$$

We note here that there are other ways of carrying out 3-D imaging with the FFT algorithm for solving the general problem of non-coplanar baselines (as described, for example, in [11]). In general, however, the 3-D FFT approach is wasteful for wide fields, since most of the cube or rectangular region computed will be empty. Moreover, the memory required is a factor $N_{planes} \sim \frac{\lambda B_{max}}{D^2}$ greater than for the case of the 2-D FFT for two dimensional processing [11].

3.1.3 UVW-space facets (Polyhedron Method)

The 2-D FFT is a good approximation to the full relationship between visibility and brightness for a sufficiently small field of view (FOV). One way of exploiting this is to estimate $I(l, m)$ in the neighborhood of a number of points on the celestial sphere [11, 17]. Thus, the celestial sphere is approximated by a “celestial polyhedron” – a unit sphere with small flat planes (“facets”), each of which stays close to the sphere’s surface. The image width or size of each facet is small enough that the 2-D approximation of the measurement equation is valid on each facet and so non-coplanar imaging is not needed for each facet. For each facet, the entire data set must be phase-shifted for the facet-center, and the (u, v, w) coordinates recomputed for the new orientation. Thus, for a given direction (l_0, m_0, n_0) the appropriate visibility function is given by phase rotating V [11]:

$$V^{PR} = V(u, v, w)e^{2\pi i[u(l_0-l)+v(m_0-m)+w(n_0-n)]} \quad (33)$$

and also re-calculating the spatial coordinates (u, v, w) for that direction. $I(l, m)$ can then be approximated by a 2-D FFT for V^{PR} with respect to the re-projected coordinate (u_0, v_0) . Calculating the phase error involved in this approach, [11] estimate that the number of facets required is

$$N_{facets} = \frac{2\lambda B_{max}}{fD^2}, \quad (34)$$

where f is the maximum acceptable pixel separation between the plane of the facet and the celestial sphere, measured in pixels; for high dynamic range imaging, $f \ll 1$ [11], here taken to be $f \sim 0.2$. For each facet of the polyhedron, the sampling of the V function produces a convolution of the true sky brightness with a locally shift-invariant point spread function (PSF). Correspondingly, the image or Fourier plane will be covered by a series of facets, in each of which the 2-D FFT holds. So, we decompose the visibilities into a summation of re-phased faceted visibilities predicted from the number of facets (i.e., a summation of 2-D FFTs):

$$V(u, v, w) = \sum_k V_k(u, v) \frac{e^{2\pi i(u_k l_k + v_k m_k + w_k(\sqrt{1-l_k^2-m_k^2}-1))}}{\sqrt{1-(l-l_k)^2-(m-m_k)^2}}, \quad (35)$$

where the visibility for the k th facet is

$$V_k(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I_k(l, m) e^{2\pi i(u_k(l-l_k)+v_k(m-m_k))} dl dm. \quad (36)$$

The total number of pixels required for the polyhedron method is approximately [14, 21]

$$\frac{\Omega}{\theta_{min}^2} \sim 4 \left(\frac{B_{max}}{D} \right)^2, \quad (37)$$

where the factor of 4 is included here for the purpose once again of minimizing aliasing effects [7]. So, the total number of pixels per facet is roughly

$$N_{l,facet} N_{m,facet} \sim 4 \frac{\left(\frac{B_{max}}{D} \right)^2}{N_{facets}} = \frac{0.4 B_{max}}{\lambda} \quad (38)$$

(for sampling, $f = 0.2$) and the 2-D FFT per facet requires $\frac{0.4B_{max}}{\lambda} \log_2 \left(\frac{0.4B_{max}}{\lambda} \right)$ floating point operations on $\frac{0.4B_{max}}{\lambda}$ complex words to complete. The gridding convolution operations for each of N_{facets} has a computational intensity of $N_{GCF,points}^2$ over all data. Then, proceeding as in previous sections, the computational costs for the facet imaging method (we must multiply the computational intensity by the total number of facets as well) goes as (in units of FLOPS) [13]

$$\left(\frac{N_{ant} B_{max}}{D} \right)^2 \left(\frac{\Delta \nu}{\nu} \right) \times \left(\frac{\lambda B_{max}}{D^2} \right)^2 N_{GCF,points}^2 + \left(\frac{10\lambda B_{max}}{D^2} \right)^2 \left[\left(\frac{0.4B_{max}}{\lambda} \right) \log_2 \left(\frac{0.4B_{max}}{\lambda} \right) \right] \quad (39)$$

where $N_{GCF,points}$ is the support of the GCF in one axis and goes as $\frac{B_{max}\lambda}{D^2}$ and is typically 7 or 9 [13].

3.1.4 W-Projection

In this approach, the goal is to project w out of the problem, thus allowing a 2-D FFT to a single image to be used. Re-projection to and from any position in (u, v, w) space from and to the $w = 0$ plane can be carried out by a convolution with a known kernel. In this approach, Eq. (13) is expressed as [13]

$$V(u, v, w) = \int \int \frac{A_N(l, m) I(l, m) G(l, m, w) e^{[-2\pi i(ul+vm)]}}{\sqrt{1-l^2-m^2}} dl dm \quad (40)$$

$$= \tilde{G}(u, v, w) * V(u, v, w = 0), \quad (41)$$

where Eq. (41) is a convolution between the FT of $I(l, m)$ and the FT of an image plane phase term parameterized by w [13],

$$G(l, m, w) = e^{-2\pi i[w(\sqrt{1-l^2-m^2}-1)]}, \quad (42)$$

where the FT of $G(l, m, w)$ over (l, m) is [13]

$$\tilde{G}(u, v, w) = \frac{i}{w} e^{-\pi i \left(\frac{u^2+v^2}{w} \right)}. \quad (43)$$

Thus, each (u, v, w) point is projected onto the $(u, v, w = 0)$ plane using the w projection function $\tilde{G}(u, v, w)$. These gridded visibilities can be transformed to the image plane using a 2-D FFT, thus obtaining the dirty image on the tangent plane. To limit aliasing, an additional tapering function $T(l, m)$ is required in the image plane as [13]:

$$G_T(l, m, w) \equiv T(l, m) G(l, m, w). \quad (44)$$

The effective convolution function is then \tilde{G}_T . For fast evaluation, $\tilde{G}_T(u, v, w)$ (i.e., the GCF) is pre-computed for a discrete set of w with uniform sampling in \sqrt{w} such that aliasing effects are less than the required dynamic range. This effectively increases the limit before which errors due to the non-coplanar array dominates.

To evaluate the dirty (non-deconvolved) image, the w-projection algorithm proceeds as follows [13]:

1. For each sample visibility, evaluate the convolution for gridded $(u, v, w = 0)$ plane.
2. Perform 2-D inverse FFT to obtained tapered $T(l, m)I^D(l, m)$.
3. Divide out image plane tapering function $T(l, m)$ to obtain dirty image $I^D(l, m)$.

A spheroidal gridding function with, for example, 9×9 pixel support in (u, v) can be used as the image-plane tapering function, and the work involved in convolving with \tilde{G}_T increases in direct proportion to its extent in (u, v) , i.e., increasing the spread of \tilde{G}_T in (u, v) directly leads to more computing [13]. The number of pixels needed in the convolution kernel \tilde{G}_T along each of the u and v axes (i.e., the size of the support region) is denoted $N_{w,project}$ and goes as [13]:

$$N_{w,project} \sim \left(\frac{\lambda}{D}\right)^2 \left(\frac{B_{max}}{\lambda}\right) = \frac{\lambda B_{max}}{D^2} \quad (45)$$

and $N_{w,project}^2$ is typically between about 30×30 and 70×70 pixels [13]. We take $N_{GCF,points}$ to be the size of the normal spheroidal gridding convolution function in one axis (typically 9).

Each visibility V_i at location (u, v, w) is mapped to the $w = 0$ plane with the phase shift proportional to the distance from the V_i point to the plane. Each V_i will be mapped to all points lying within a cone whose full angle is the same as the FOV of the desired map, $\theta \sim \frac{2\lambda}{D}$ for a full-field image [21]. Once we have projected the grid $V(u, v, w)$ points to $V(u, v, w = 0)$ in this way, we then compute the 2-D FFT for each projection of baseline b onto a fairly small number P_w of planes (usually between about 10 and 30) parallel to the (u, v) plane and sum the results to obtain an image [22]. The number of facets N_{facets} (in polyhedron imaging) and the number of w planes P_w are roughly the same (both being proportional to $\frac{B_{max}\lambda}{D^2}$), being determined by the number of antenna areas in a Fresnel zone, but note that $N_{facets} \sim 5N_{w,project}$ for high dynamic range imaging. In faceted approaches, we have to grid the entire data set for each facet and the gridding kernel extends over, for example, a 7×7 region in the (u, v) plane. In the w -projection approach, we have to grid only once, but the gridding kernel extends over more pixels in the (u, v) plane, roughly as the number of antennas across Fresnel zones [13].

Now, proceeding as in the case of polyhedron imaging in the previous section, we estimate the number of pixels per w plane to be, roughly,

$$N_{l,w\ plane} N_{m,w\ plane} \sim \frac{4\left(\frac{B_{max}}{D}\right)^2}{\frac{B_{max}\lambda}{D^2}} = \frac{4B_{max}}{\lambda}. \quad (46)$$

Then, the number of operations required by the 2-D FFT per w plane is:

$$\frac{4B_{max}}{\lambda} \log_2\left(\frac{4B_{max}}{\lambda}\right). \quad (47)$$

This is the number of floating point operations on $N_{l,w\ plane} N_{m,w\ plane}$ words. The computational intensity of the 2-D FFT operations will then be $\log_2\left(\frac{4B_{max}}{\lambda}\right)$. Now, assuming that the size of the normal gridding convolution function $N_{GCF,points}^2$ and the size of the \tilde{G}_T

gridding convolution function $N_{w,project}^2$ add in quadrature [13] and taking $P_w \sim \frac{B_{max}\lambda}{D^2}$, our estimate for the computational load for the w-projection algorithm without considering deconvolution operations or parallel computing efficiency and overhead is

$$\sim 10^{-2} \left(\frac{N_{ant} B_{max}}{D} \right)^2 \left(\frac{\Delta\nu}{\nu} \right) \left(\left(\frac{\lambda B_{max}}{D^2} \right)^2 + N_{GCF,points}^2 \right) + \left(\frac{\lambda B_{max}}{D^2} \right)^2 \left[\left(\frac{4B_{max}}{\lambda} \right) \log_2 \left(\frac{4B_{max}}{\lambda} \right) \right]. \quad (48)$$

3.1.5 Hybrid UVW-Space Facets and W-Projection Approach

For w-projection imaging, one image is made and the area of the corresponding gridding function goes as $\frac{\lambda B_{max}}{D^2}$, and in facet-based imaging, the gridding function is constant but the number of pixels goes as $\frac{\lambda B_{max}}{D^2}$ [23]. Therefore, the number of operations (as we saw in the previous two sections) required to grid the data for both of these algorithms goes asymptotically as $\frac{\lambda N_{ant}^2 B_{max}^3}{D^4}$ [23]. The cost of convolution with \tilde{G}_T can become quite large. The very large sizes of \tilde{G}_T found for a few hundred km baselines at meter wavelengths may require hybrid w-projection/uvw facets approaches. Such an approach might involve projection of the visibilities onto a small number of facets rather than onto a single plane. Whether this is so may depend on the architecture of the machines on which such processing will be carried out [13]. In this approach, the number of facets can correspond to the number of discrete values of w for which the GCF is evaluated. For example, 256 samples along the w-axis may be sufficient. A hybrid method would allow for a smaller number of planes to be used in the w-projection GCF (e.g., $P_w = 32$) as well as a small number of facets along each axis in the main image (e.g., 3) in order to conserve memory if the image size approached the computer memory limit. However, there could be a larger penalty in computing execution time.

In a hybrid method we can break down a large field of view into smaller fields using facets and then apply the w-projection algorithm in each of those facets [20]. Putting it another way, a very large image can be constructed of facets, each of which is made using w-projection [19]. Thus, an analytical equation for computational loads associated with a hybrid w-projection and facet approach could just involve essentially multiplying the expression for the w-projection computational load (Eq. (48)) with N_{facets} . Taking $N_{facets} \sim \frac{2\lambda B_{max}}{fD^2} \sim \frac{10\lambda B_{max}}{D^2}$ (where $f \sim 0.2$) and taking the number of operations required by the 2-D FFT per facet for this case to be $\frac{0.4B_{max}}{\lambda} \log_2 \left(\frac{0.4B_{max}}{\lambda} \right)$ (corresponding to a computational intensity of $\log_2 \left(\frac{0.4B_{max}}{\lambda} \right)$), we then have

$$\sim 10^{-1} \left(\frac{N_{ant} B_{max}}{D} \right)^2 \left(\frac{\Delta\nu}{\nu} \right) \left(\left(\frac{\lambda B_{max}}{D^2} \right)^2 + N_{GCF,points}^2 \right) \frac{\lambda B_{max}}{D^2} + \left(\frac{\lambda B_{max}}{D^2} \right)^2 \frac{10\lambda B_{max}}{D^2} \left[\left(\frac{0.4B_{max}}{\lambda} \right) \log_2 \left(\frac{0.4B_{max}}{\lambda} \right) \right]. \quad (49)$$

4 Comparisons and Plots

We now generate plots of computational cost vs. antenna diameter using the above derived computational cost equations (without consideration of deconvolution operations and parallel computing efficiency here) for typical (e.g., [13]) gridding convolution function sizes and for interferometer array parameters that have been considered for the SKA project. These parameters are listed in Table 1.

We emphasize here that we are considering the case in which each of the imaging algorithms is applied to the same given visibility data-set to form essentially equivalent images with the same dynamic range.

Table 1: Values and Parameters used in Computational Cost vs. Antenna Diameter plots in Fig. 1

λ (wavelength)	0.21m
ν (observing frequency)	$1420 \times 10^6 Hz$
$\Delta\nu$ (contiuum bandwidth)	$400 \times 10^6 Hz$
N_{GCF} (support of gridding convolution in 1 axis)	9
B_{max} (max. baseline length)	35 km
N_{ant} (number of antenna)	3000

Inserting the values in the table into the five computational cost equations for the five imaging algorithms (with units of computational cost being FLOPS in these plots), we generate 5 semi-log plots (Fig. 1).

There is a clear trend, as expected, for the computational costs to decrease with increasing antenna diameter for all imaging algorithms when all other parameters are kept constant. We confirm that the gridding operations and the size of the gridding convolution functions have a more significant relative impact on the computational costs than the FFT operations themselves, as pointed out by, e.g., [23] and [13].

Overall, as can be seen in all figures, the computational costs and loads are minimized for all imaging algorithms when, keeping all other parameters constant, the antenna diameter is maximized and the sizes of the gridding convolution functions are minimized. We will discuss computational costs associated with deconvolution operations and parallel efficiency in a future memo. In all cases the effective computational cost is a strong function of the efficiency with which the algorithm can be mapped to the underlying computing architecture, including the computational intensity, memory footprint, and memory bandwidth requirements.

References

- [1] Hockney, R. W., and Jesshope, C.R. 1986, "*Parallel Computers, Second Edition*", Johns Hopkins Press.

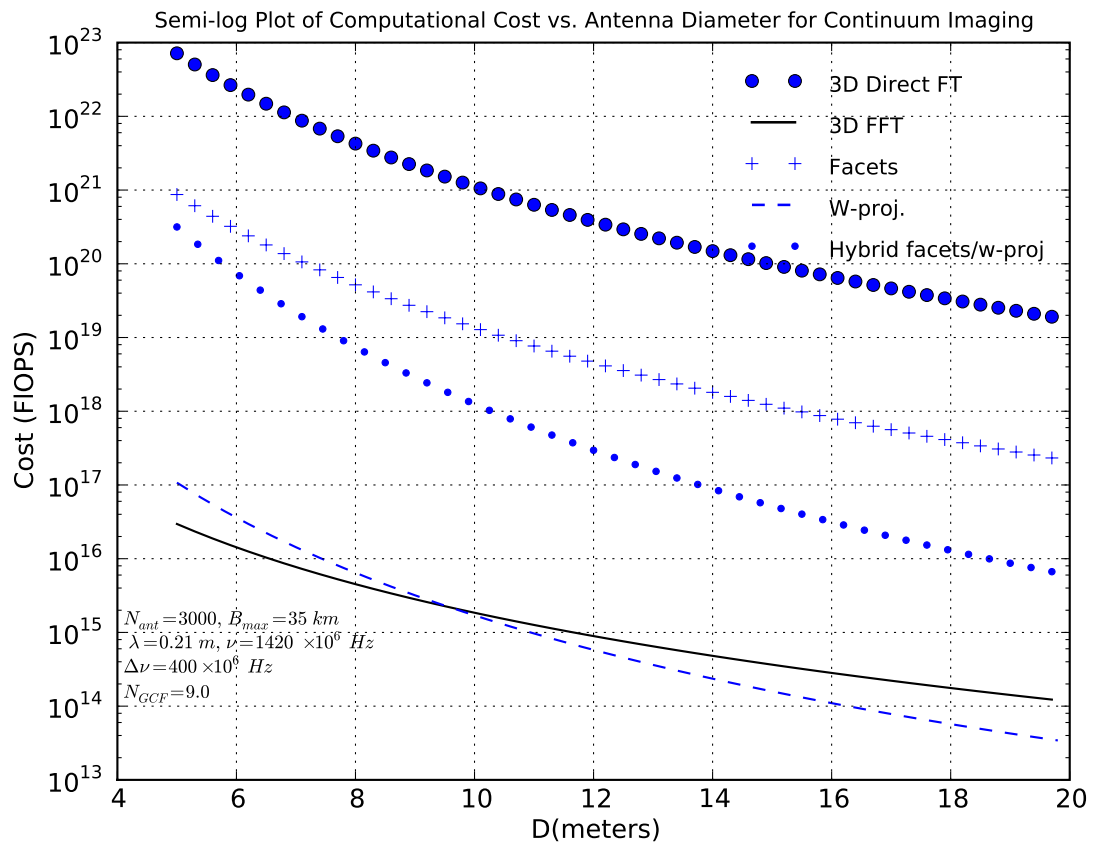


Figure 1: Semi-log x plots of computational costs (without consideration of deconvolution and parallel computing efficiency η) vs. antenna diameter D for continuum imaging for the 3-D direct FT, 3-D FFT, facets, w-projection, and hybrid facets/w projection imaging algorithms.

- [2] Miles, D. 1993, "Compute Intensity and the FFT", Proceedings of the 1993 ACM/IEEE conference on Supercomputing, Pages 676-684, <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01263522>.
- [3] Nukada, A. et al. 2008, "Bandwidth Intensive 3-D FFT kernel for GPUs using CUDA", Proceedings of the 2008 ACM/IEEE conference on Supercomputing, Article No. 5
- [4] Moreland, K. & Angel, E., 2003, "The FFT on a GPU", SIGGRAPH/Eurographics Workshop on Graphics Hardware 2003 Proceedings, pp. 112-119, eds. M. Doggett, W. Heidrich, W. Mark, A. Schilling
- [5] Perley, R. & Clark, B. 2003, "Scaling Relations for Interferometric Post-Processing", EVLA Memo 63, <http://www.nrao.edu>.

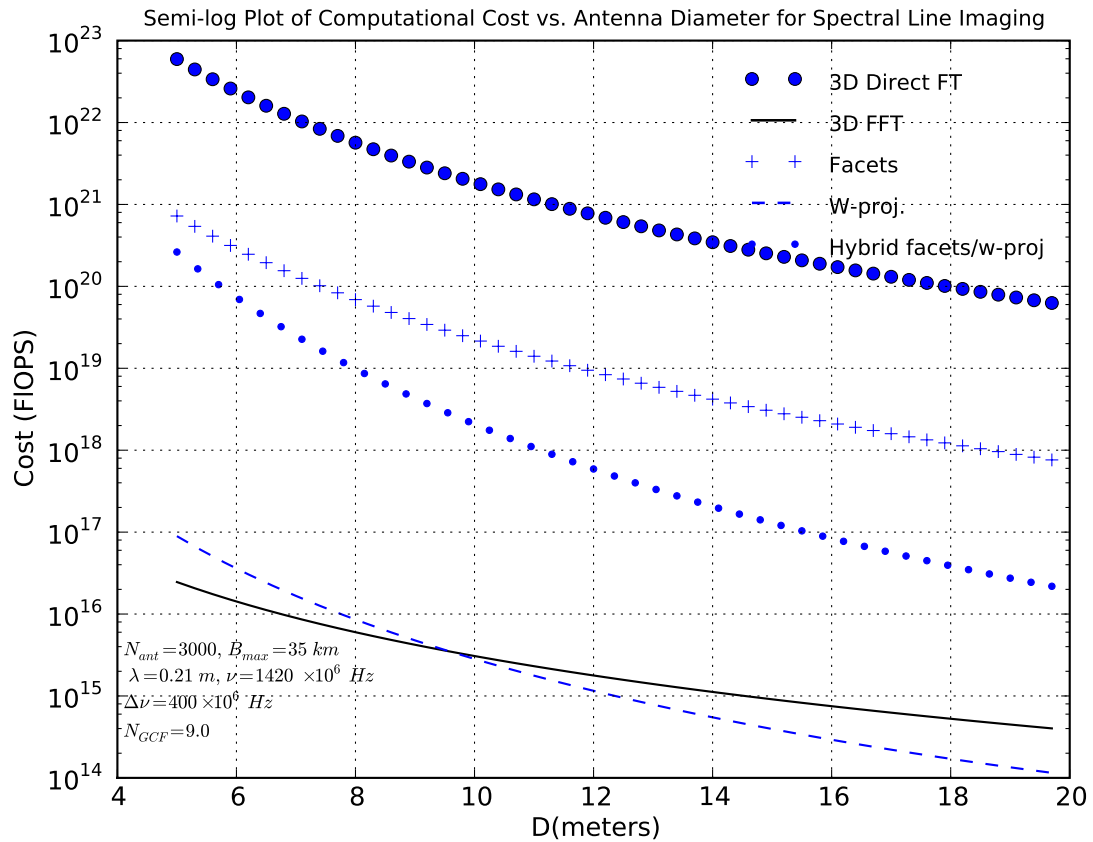


Figure 2: Same as Fig. 1 except spectral line imaging costs are considered instead of continuum imaging costs.

- [6] Perley, R. 2004, "Wide-Field Imaging with the EVLA: WIDAR Correlator Modes and Output Data Rates", EVLA Memo 64, <http://www.nrao.edu>.
- [7] Bhatnagar, S. 2009, "Report on the findings of the CASA Terabyte Initiative: Single-node tests", EVLA Memo 132, <http://www.nrao.edu>.
- [8] Kembball, A. 2008, "Petascale Computing Challenges for the SKA", TDP Calibration and Processing Group Memo 1
- [9] Thompson, A., Moran, J., & Swenson, G. 2004, "Interferometry and Synthesis in Radio Astronomy". Wiley-VCH.
- [10] Schilizzi, R.T., 2007, "Preliminary Specifications for the Square Kilometer Array", SKA Memo 100, http://www.skatelescope.org/PDF/memos/100_Memo_Schilizzi.pdf

- [11] Cornwell, T.J. & Perley, R.A. 1992, "*Radio-interferometric imaging of very large fields: The problem of non-coplanar arrays*", *Astron. Astrophys.* 261,353.
- [12] Golap, K., Shankar, N. 2001, "*Deconvolution of Wide-field Images from a Non-Coplanar T-Array*", *J. Astrophys. Astr.*, **22**,251
- [13] Cornwell, T.J., Golap, K, & Bhatnagar, S. 2008, "*The Noncoplanar Baselines Effect in Radio Interferometry: The W-Projection Algorithm*", *IEEE Journal of Selected Topics in Signal Processing*", Vol. 2, No. 5.
- [14] Perley, R.A. 1999 "*Imaging with Non-coplanar Arrays*", in *Synthesis Imaging in Radio Astronomy II*, ASP Conference Series, Vol. 180, eds. G.B. Taylor, C.L. Carilli, and R.A. Perley.
- [15] Wright, M. 2005, "*Real Time Imaging*", SKA Memo 60, http://www.skatelescope.org/PDF/memos/http://www.skatelescope.org/PDF/memos/60_Wright.pdf
- [16] Briggs, D.S., Schwab, F.R., and Sramek, R.A., "*Imaging*", in *Synthesis Imaging in Radio Astronomy II*, ASP Conference Series, Vol. 180, eds. G.B. Taylor, C.L. Carilli, and R.A. Perley, 1999.
- [17] Cornwell, T.J., Braun, R., and Briggs, D.S., "*Deconvolution*", in *Synthesis Imaging in Radio Astronomy II*, ASP Conference Series, Vol. 180, eds. G.B. Taylor, C.L. Carilli, and R.A. Perley, 1999.
- [18] Taylor, G.B., Carilli, C.L., & Perley, R.A., Eds. 1999, "*Astron. Soc. Pac. Conf. Ser. 180: Synthesis Imaging in Radio Astronomy II*".
- [19] Cornwell, T.J. 2004, "*The Correction of Near Field Effects in Radio Interferometric Geolocation*", <http://www.atnf.csiro.au/people/Tim.Cornwell/geolocation.pdf>
- [20] Datta, A. & Carilli, C.L. 2007, "*Imaging of PAPER data using CASA*", <https://safe.nrao.edu/wiki/pub/Main/PaperDocs/PAPER.imaging.pdf>
- [21] Perley, R. 2008, "*Wide-Field Imaging I: Non-Coplanar Visibilities*", presented at the Eleventh Synthesis Imaging Workshop, <http://www.aoc.nrao.edu/events/synthesis/2008/lectures/WideFieldImaging2008.pdf>
- [22] Varbanescu, A.L. et al. 2009, "*Building high-resolution sky images using the Cell/B.E*", *Scientific Programming*, **17**, 113.
- [23] Cornwell, T.J. 2004, "*EVLA and SKA Computing Costs for Wide-Field Imaging*", *Experimental Astronomy*, **17**, 329.
- [24] Bhatnagar, S. 2003, "*Wide Field Imaging*", Notes from summer school on low frequency astronomy at NCRA, Pune, June 21-July 17 1999, http://www.gmrt.ncra.tifr.res.in/gmrt_hpage/Users/doc/WEBLF/LFRA/index.html

- [25] Perley, R.A., and Cornwell, T.J., "*Imaging with Non-coplanar Data*", in *Radio Interferometry: Theory, Techniques, and Applications*, IAU Coll. 131, ASP Conference Series, Vol. 19, eds. T.J. Cornwell and R.A. Perley, 1991.

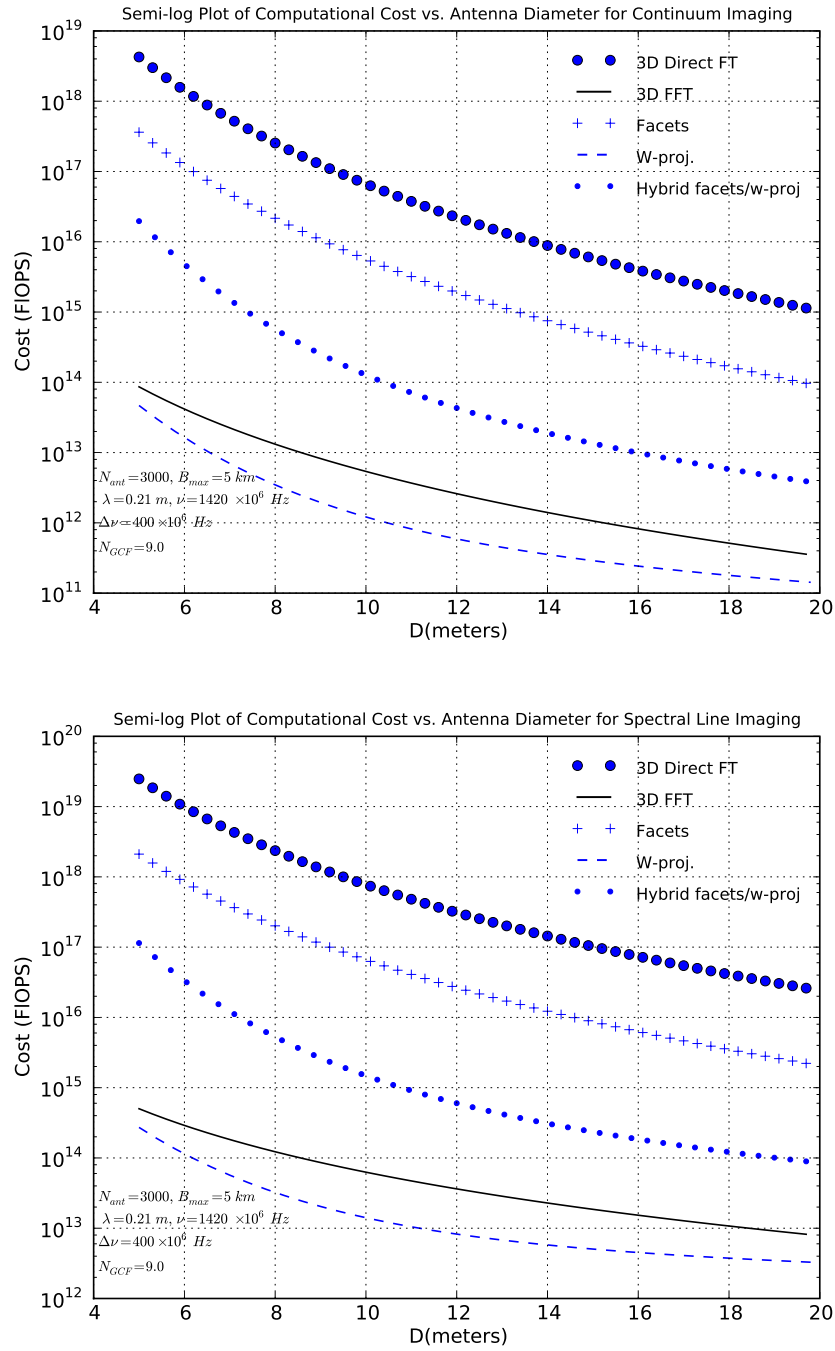


Figure 3: Additional computational cost curves for B_{max} set to 5 km for continuum imaging (upper panel) and spectral line imaging (lower panel).