ngVLA Data Rates and Computational Loads (Update)
ngVLA Computing Memo 11

Rafael Hiriart

August 30, 2024

1 Introduction

This memo provides an update to the size-of-computing metrics estimated in [1]. We improve these preliminary estimations by considering explicitly the effect of the subarray configuration in the distribution of the w coordinate for each visibility, and the size of the support function for each value of w in the calculation of the computational load that will be required to image each science case of the ROP and the EOP.

2 The Reference Observing Program and the Envelope Observing Program

We consider as input two observational programs defined by the ngVLA SciOps group, the Reference Observing Program (ROP) and the Envelope Observing Program (EOP). The ROP defines a set of 27 observations that will be necessary to achieve the ngVLA key science goals. The EOP consists of 63 observations, defining what a typical year would look like. The ROP input table is presented in Table 1, while the EOP table can be found in Table 6 in the Appendix. The columns are:

- `scnum`: science case number
- `scname`: science case name
- `frac`: science use fraction
- `fov`: field of view, in arcseconds
- `psf`: point spread function FWHM (or desired resolution), in milli-arcseconds
- `dr`: dynamic range
- `freq`: center frequency, in Hz
- `bw`: bandwidth, in Hz
- `chan`: science channel width, in Hz
- `dump`: maximum dump time, in seconds
- `sub-array`: required sub-array

Table 1: Reference Observing Program.

scnum	scname	frac	fov	psf	dr	freq	bw	chan	dump	sub-array
1	KSG1 Driving Cont Band 6 eg Taurus disk	0.09	5	10	1000	1.000000e+11	FULL	120000000	1.0000	Main
2	KSG1 Driving Cont Band 4 eg Taurus disk	0.04	5	10	1000	2.725000e+10	FULL	120000000	1.0000	Main
3	KSG2 Driving Line Band 5 eg Sgr B2(N)	0.04	60	100	1000	4.050000e+10	4000000000	13500	1.0000	Main
4	KSG2 Driving Line Band 4 eg Sgr B2(N)	0.01	60	100	1000	2.730000e+10	4000000000	9100	2.0000	Main
5	KSG2 Driving Line Band 3 eg Sgr B2(N)	0.01	60	100	1000	1.640000e+10	4000000000	5500	2.0000	Main
6	KSG3 Driving Line Band 5 eg COSMOS	0.04	FULL	1000	100	4.050000e+10	FULL	675000	1.0000	Plains+Core
7	KSG3 Driving Line Band 4 eg COSMOS	0.01	FULL	1000	100	2.730000e+10	FULL	455000	2.0000	Plains+Core
8	KSG3 Driving Line Band 3 eg COSMOS	0.01	FULL	1000	100	1.640000e+10	FULL	273300	2.0000	Plains+Core
9	KSG3 Driving Line Band 6 eg Spiderweb galaxy	0.02	5	100	1000	7.200000e+10	240000000	7200000	1.0000	Main
10	KSG3 Driving Line Band 5 eg Spiderweb galaxy	0.01	5	100	1000	3.600000e+10	120000000	3600000	1.0000	Main
11	KSG3 Driving Line Band 4 eg Spiderweb galaxy	0.01	5	100	1000	2.770000e+10	92300000	2770000	2.0000	Main
12	KSG3 Driving Line Band 6 eg Virgo Cluster	0.07	FULL	100	1000	1.125000e+11	6000000000	375000	1.0000	Plains+Core
13	KSG3 Driving Line Band 1 eg M81 Group	0.11	FULL	1000	1000	1.420000e+09	7000000	4730	2.0000	Plains+Core
14	KSG3 Driving Line Band 1 eg M81 Group	0.13	FULL	60000	1000	1.420000e+09	7000000	47300	2.0000	Core
15	KSG5 Driving Cont Band 1 OTF Find LIGO event	0.07	FULL	1000	5000	2.350000e+09	FULL	2000000	0.5000	Main
16	KSG5 Driving Cont Band 4 OTF Find LISA event	0.07	FULL	1000	5000	2.730000e+10	FULL	5000000	0.5000	Plains+Core
17	KSG5+4 Driving Cont Band 2 OTF Find BHs+Possib...	0.04	FULL	1000	5000	7.900000e+09	FULL	5000000	0.5000	Plains+Core
18	KSG5 Driving Cont eg Band 2 Followup from OTF	0.00	1	10	5000	7.900000e+09	FULL	120000000	2.0000	Main+LBA
19	KSG5 Driving Cont Band 3 Gw170817@200Mpc	0.24	1	1	100	1.640000e+10	FULL	120000000	2.0000	LBA
20	KSG3 Supporting Cont Band 6 eg Virgo Cluster	0.00	FULL	1000	5000	9.300000e+10	FULL	5000000	1.0000	Main
21	KSG3 Supporting Cont Band 5 eg Virgo Cluster	0.00	FULL	1000	5000	4.050000e+10	FULL	5000000	1.0000	Main
22	KSG3 Supporting Cont Band 4 eg Virgo Cluster	0.00	FULL	1000	5000	2.730000e+10	FULL	5000000	2.0000	Main
23	KSG3 Supporting Cont Band 3 eg Virgo Cluster	0.00	FULL	1000	5000	1.640000e+10	FULL	5000000	2.0000	Main
24	KSG3 Supporting Cont Band 2 eg Virgo Cluster	0.00	FULL	1000	5000	7.900000e+09	FULL	5000000	2.0000	Main
25	KSG5 Driving Cont Band 1 PTA timing 5 subs 1 P...	0.16	0.1	10	100	2.400000e+09	FULL	500000	2.0000	Mid
26	KSG4 Driving Cont Band 3 GC search 1 sub 10 PSFs	0.05	0.1	10	100	1.640000e+10	FULL	1000000	0.0001	Main
27	KSG4 Driving Cont Band 3 GC timing 1 sub 10 PSFs	0.00	0.1	10	100	1.640000e+10	FULL	1000000	2.0000	Main

We start by calculating several quantities that are necessary to estimate data rates and computational loads, in a similar manner as it was done in the ngVLA Quantitative eXchange Model [3]. These are:

- The actual bandwidth for the rows where this quantity was defined as "FULL". We lookup the center frequency in ngVLA band definitions.
- Likewise, the FoV has been defined as "FULL" in several rows. For these, we calculate the FoV as $\theta_{HPBW} \approx 1.02 \cdot \lambda / D$ in radians. We use the lower frequency for each band, which gives the larger beamwidth, for the purpose of estimating upper bounds for the processing load.
- We estimate the image linear dimension by dividing the FoV by the resolution and assuming that the bandwidth of the synthesized beam is 4 pixels.
- We calculate the channel width required to prevent bandwidth smearing *for the required resolution*. Using the resolution achieved by the subarray synthesized beam (i.e., λ / B_{max}, with B_{max} being the maximum baseline in the subarray) gives a number of channels that is prohibitively expensive. Assuming that imaging will use the required resolution instead (which is equivalent to dropping the visibilities corresponding to the larger baselines with corresponding decrease in sensitivity), we calculate the channel width as described in [2] using $\beta = 0.5$, and take of minimum of this value and the value provided in the ROP/EOP tables.
- We calculate the dump time required to prevent loss of power due to time-average smearing as described in [2], and we take the mininum of this value and the one defined in the ROP/EOP table.
- For the purposes of this memo, we are only interested in the data rates and computational load for the synthesis imaging science cases. Pulsar science cases are processed in the Pulsar Engine, internal to the CSP. Hence, the pulsar science cases are removed from the calculations. For the ROP they correspond to science cases 25, 26 and 27, and for the EOP these are 50 and 59.

These derived parameters for the ROP and EOP are shown in Tables 2 and 7, respectively.

3 Data Rates

The data rates, in GVisibilities per second, are calculated as

$$\lambda \text{ [GVis/sec]} = \frac{n_{pol} \cdot n_{ant} \cdot (n_{ant} + 1) \cdot n_{channel}}{2 \cdot t_{int} \cdot 10^9}$$

where n_{pol} is the number of polarization products, n_{ant} is the number of antennas, $n_{channel}$ is the number of channels, t_{int} is the integration duration in seconds, and auto-correlations are included. Visibilities are represented in single precision (32 bits, or 4 bytes), so we use 8 bytes per visibility, given that visibilities are complex numbers. From the image linear size and the number of channels we calculate the image and cube sizes, assuming the pixels are stored in single precision as well. These rates are summarized in Table 3 below. We include an estimation of the High Level Data Product (HLDP) data rate, assumming an

Table 2: ROP derived parameters.

scnum	scname	bandwidth	fov	img	eff_channel_hz	eff_num_channels	dump
1	KSG1 Driving Cont Band 6 eg Taurus disk	20.0000	5.000000	1500	9.000000e+07	222	1.0
2	KSG1 Driving Cont Band 4 eg Taurus disk	13.5000	5.000000	1500	2.050000e+07	658	1.0
3	KSG2 Driving Line Band 5 eg Sgr B2(N)	4.0000	60.000000	1800	1.350000e+04	296296	1.0
4	KSG2 Driving Line Band 4 eg Sgr B2(N)	4.0000	60.000000	1800	9.100000e+03	439560	2.0
5	KSG2 Driving Line Band 3 eg Sgr B2(N)	4.0000	60.000000	1800	5.500000e+03	727272	2.0
6	KSG3 Driving Line Band 5 eg COSMOS	20.0000	114.967269	344	6.750000e+05	29629	1.0
7	KSG3 Driving Line Band 4 eg COSMOS	13.5000	170.632686	511	4.550000e+05	29670	2.0
8	KSG3 Driving Line Band 3 eg COSMOS	8.2000	285.081440	855	2.733000e+05	30003	2.0
9	KSG3 Driving Line Band 6 eg Spiderweb galaxy	0.2400	5.000000	150	7.200000e+06	33	1.0
10	KSG3 Driving Line Band 5 eg Spiderweb galaxy	0.1200	5.000000	150	3.600000e+06	33	1.0
11	KSG3 Driving Line Band 4 eg Spiderweb galaxy	0.0923	5.000000	150	2.770000e+06	33	2.0
12	KSG3 Driving Line Band 6 eg Virgo Cluster	6.0000	32.022847	960	3.750000e+05	16000	1.0
13	KSG3 Driving Line Band 1 eg M81 Group	0.0070	2475.468907	7426	4.730000e+03	1479	2.0
14	KSG3 Driving Line Band 1 eg M81 Group	0.0070	2475.468907	123	4.730000e+04	147	2.0
15	KSG5 Driving Cont Band 1 OTF Find LIGO event	2.3000	2922.084755	8766	2.053329e+05	11201	0.5
16	KSG5 Driving Cont Band 4 OTF Find LISA event	13.5000	170.632686	511	5.000000e+06	2700	0.5
17	KSG5+4 Driving Cont Band 2 OTF Find BHs+Possib...	8.8000	1001.857630	3005	1.746755e+06	5037	0.5
18	KSG5 Driving Cont eg Band 2 Followup from OTF	8.8000	1.000000	300	1.750000e+07	502	2.0
19	KSG5 Driving Cont Band 3 Gw170817@200Mpc	8.2000	1.000000	3000	6.150000e+06	1333	2.0
20	KSG3 Supporting Cont Band 6 eg Virgo Cluster	20.0000	42.247009	126	5.000000e+06	4000	1.0
21	KSG3 Supporting Cont Band 5 eg Virgo Cluster	20.0000	114.967269	344	5.000000e+06	4000	1.0
22	KSG3 Supporting Cont Band 4 eg Virgo Cluster	13.5000	170.632686	511	5.000000e+06	2700	2.0
23	KSG3 Supporting Cont Band 3 eg Virgo Cluster	8.2000	285.081440	855	5.000000e+06	1639	2.0
24	KSG3 Supporting Cont Band 2 eg Virgo Cluster	8.8000	1001.857630	3005	1.746755e+06	5037	2.0

average SB length of 4 hours, which produces 4 data products: the restored image, the model image, the residual image, and the PSF image. We also include an estimation of the number of pixels, calculated as the image size squared times the number of polarizations and the number of channels.

Table 3: Summary of data rates for ROP and EOP.

	ROP	EOP
Average Data Rate [GVis/sec]	1.934710	2.636177
Average Data Rate [GBytes/sec]	15.477683	21.089419
Average Data rate [PBytes/month]	40.118156	54.663773
Peak Data Rate [GVis/sec]	33.461785	40.448311
Peak Data Rate [GBytes/sec]	267.694278	323.586490
Median Data Rate [GVis/sec]	0.032188	0.118245
90% Quantile Data Rate [GVis/sec]	2.013198	5.478574
Average Image Size (1 Plane) [MBytes]	57.072190	69.815383
Max Image Size (1 Plane) [MBytes]	307.371024	1605.925476
Average Cube Size [TBytes]	2.380373	5.842852
Max Image Cube Size [TBytes]	37.701780	215.746452
Median Image Cube Size [TBytes]	0.175634	0.091198
90% Quantile Cube Size [TBytes]	12.801836	13.379510
Average Number of Pixels [GPixels]	595.093231	1460.713077
Average HLDP Data Rate [GBytes/sec]	0.661215	1.623015
Ratio HLDP DR / Vis DR [%]	4.272052	7.695871

The distributions of visibility data rates and cube sizes, across the science cases defined in the ROP and EOP are shown in Figures 1 and 2. Both distributions are similar, with most of the science cases generating relatively modest data rates while a few extreme cases skew the metrics higher. In fact, 90% of the science cases require less than 2 GVis/sec for the ROP, and 5.4 GVis/sec for the EOP, while the rest 10% generate much larger data rates, peaking at 33.4 Gvis/sec for the ROP and 40.4 Gvis/sec EOP. The same can be said for the cube sizes, with 90% of them being less than 12.8 TBytes for the ROP and 13.3 TBytes for the EOP.

4 Gridding Processing Load

As described in [1], the number of FLOPS to grid the visibilities in one major cycle is

$$C = N_{op} \times \sum_{\nu,w} N_{vis}(\nu, w) \times S^2(\nu, w) \tag{1}$$

where N_{op} is the number of FLOPs per multiply-add operation, which is

$$N_{op} = \begin{cases} 20 & \text{for complex gridding (W-proj and A-proj)} \\ 12 & \text{for real (standard) gridding} \end{cases} \tag{2}$$

For standard gridding the support function doesn't depend on either w or ν. For A-proj the support function depends on ν, but not on w. In the case of W-proj the support function depends on both ν and w. The dependence on ν can be subsumed into w when this quantity

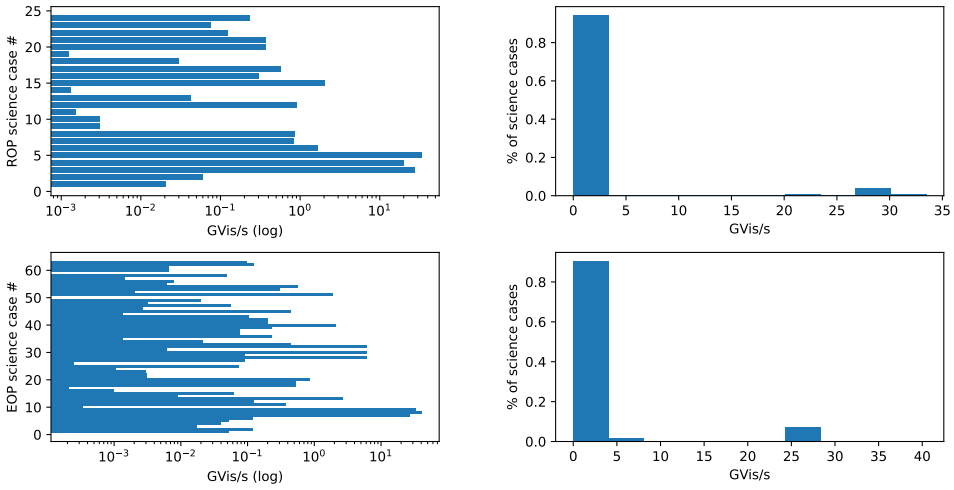

Figure 1: Distribution of data rates for the ROP and EOP.

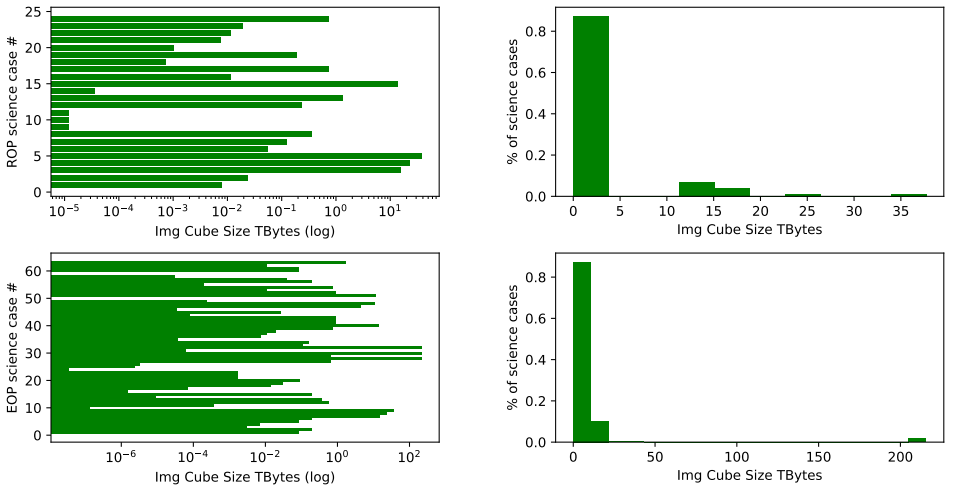

Figure 2: Distribution of cube sizes for the ROP and EOP.

is measured in wavelengths. N_{vis} is the number of visibilities in the intervals $[\nu, \nu + \Delta\nu]$ and $[w, w + \Delta w]$ for some way to partition the (ν, w) domain. S is the support size of the convolution function. This convolution function is the result of combining the prolate spheroidal function with the A-proj kernel and the W-proj kernel, with the choice of which kernels are combined depending on the science case. When more than one convolution kernel is combined, the final convolution function is the result of convolving the individual kernels (as convolution is a distributive operation), although in practice they are usually computed by applying the Fourier transform to the kernels, multiplying them, and applying the inverse Fourier transform to return to the (u, v) space. Both approaches should yield the same result, so we assume that the support size of the combined kernel is roughly the sum of the support sizes of the kernels being convolved.

To estimate the computational load, we normalize N_{vis} by the total number of visibilities on the major cycle, divide by the time that it took to receive these visibilities and introduce an "expansion" factor K_e to extrapolate from the computational load of one major cycle to the whole observation:

$$CL \left[\frac{\text{FLOP}}{\text{sec}}\right] = \lambda \left[\frac{\text{Vis}}{\text{sec}}\right] \times 2 \times N_{op} \times K_e \times \sum_{\nu,w} \bar{N}_{vis}(\nu, w) \times S^2(\nu, w) \tag{3}$$

The sum at the left side represents the number of multiply-add operations per visibility. The factor of 2 accounts for the fact that algorithms performs gridding and de-gridding operations at the same time. The factor K_e accounts for the number of major cycles and several other factors and algorithmic options that increase the number of FLOPs:

$$\begin{aligned} K_e =& 1.1 \times N_{iter} \times N_{stk} \times (12 \text{ if Full-Mueller}) \times (3 \text{ if multi-term}) \times \\ & (1.18 \text{ if multi-scale}) \times (N_{selfcal} \text{ if self-calibration}) \end{aligned} \tag{4}$$

These are from left to right:

- The cost of the deconvolution needs to be accounted for. Unfortunately we don't have an easy way to account for these FLOPs, given that the number of minor cycle iterations will depend on the observed brightness distribution and parameters such as the threshold used when cleaning. We can assume that it will be a relatively minor percentage of the cost of gridding, say 10%.

- The algorithm is iterative so the same visibilities are gridded multiple times. We assume that the number of major cycle iterations N_{iter} falls somewhere in the interval [10, 50].

- The N_{op} FLOPs are for each Stokes component, so we need to introduce the number of Stokes components N_{stk}.

- If Full-Mueller is needed, it is necessary to introduce an additional factor of 12, as imaging is performed independently for each element in the 4×4 Mueller matrix with the exception of the diagonal elements.

- If the Multi-Scale algorithm is needed, an additional factor of 3 needs to be introduced.

- If Multi-Term algorithm is needed, the cost of deconvolution needs to be scaled by the number of scales, say 30% instead of 10% (i.e. a factor of $1.3/1.1 = 1.18$).

- The number of self-calibration iterations $N_{selfcal}$ needs to be factored in when self-calibration is performed (it is expected that it will be necessary for bands 1-4). $N_{selfcal}$ is usually $2-3$ iterations.

These factors could combine in the worst case to raise K_e to 93,456 (!) In the best case $K_e = 22$. At this time, we don't have enough information to determine when these algorithmic options are required for each case. Assuming that a typical observation will demand a number of iterations of 20, all Stokes parameters, and 3 self-cal iterations, we arrive to a typical K_e value of 264.

We can justify the assumption of the relatively minor role that deconvolution plays in the computing load by estimating an upper bound for this operation. From Table 3, the average number of pixels N_{pix} is approximately 1,000 GPixels. When performing deconvolution, two operations are relevant: searching for the peak in the residual image and substracting the PSF image from the residual image. Assuming a worst-case linear search for the first operation, the number of operations involved in finding the peak is N_{pix}. Substracting the two images is also of order N_{pix}, so one iteration of the minor cycle should be of the order N_{pix}^2. If we assume that *every pixel* in the image generates a minor cycle, then the cost of deconvolution becomes $N_{pix}^3 \sim 10^9$ GPixels. On the other hand, from Table 5 the average computational load for gridding is ≈ 60 PFLOPs/sec. To estimate one major cycle we divide this figure by $K_e = 264$ to obtain $60.0/264.0 = 0.23$ PFLOPs/sec. For an SB of 4 hours this generates $4 \times 3600 \times 0.23 = 3,272$ PFLOPs, or $\sim 10^{18}$ FLOPs, which is in the same order than the volume of operations for deconvolution. As only a fraction of pixels in an image are actually substracted during a minor cycle, then this fraction should be more or less the ratio between the deconvolution operations in one minor cycle and the gridding operations in one major cycle.

When performing only standard gridding, assuming a support function of size 7×7, equation [3] becomes

$$CL \left[\frac{\text{FLOP}}{\text{sec}}\right] = \lambda \left[\frac{\text{Vis}}{\text{sec}}\right] \times 1176 \left[\frac{\text{FLOP}}{\text{Vis}}\right] \times K_e.$$

For convenience, we define a factor K_c (the "complexity factor") as the ratio of the number of FLOPs per visibility for a given science case and the number of FLOPs per visibility for the standard gridding algorithm, i.e.,

$$K_c = \frac{2 \times N_{op} \times \sum_{\nu,w} \bar{N}_{vis}(\nu, w) \times S^2(\nu, w)}{1176} \tag{5}$$

This factor is equal to 1 in the case of standard gridding, increasing when a science case requires A-proj, W-proj, or both.

With this, we can simplify equation 3, after correcting the value 1176 for computing overheads as reported in [1] (instead of 1176 FLOPs for standard gridding, we measured 1280). We assume the overheads cancel out in K_c, so we can estimate this parameter using Equation 5. On the other hand, we do consider the measured overheads in the calculation of CL, as shown below:

$$CL \left[\frac{\text{FLOP}}{\text{sec}}\right] = \lambda \left[\frac{\text{Vis}}{\text{sec}}\right] \times (1280 \times K_c) \left[\frac{\text{FLOP}}{\text{Vis}}\right] \times K_e. \tag{6}$$

In order to evaluate K_c for the different science cases in the ROP and EOP, it is necessary to evaluate the distribution of w for the different ngVLA sub-arrays, assuming some sky

coordinates; and estimate the support size for the different kernels. These calculations are discussed in the following sections.

4.1 The w-term distribution

The w-term distribution is calculated purely from geometry [6]. We use the ngVLA array configuration rev. D, and adopt a target with a declination close to zenith at 30 degrees, observed during 8 hours, from an hour angle of -4.0 hours to +4.0 hours. The normalized w-distribution that results from this observation is shown in Figure 3. Figure 4 shows the corresponding UV-distributions.

The w-term is quantized in *w-planes* following a w^2 mapping, so small values of w are quantized more finely than larger values. This has been shown empirically to provide better numerical accuracy and is the practice adapted by CASA. Incidentally, this is the reason why the support function appears to follow a quadratic function when plotted against the w-planes (the integer numbers enumerating the planes, not the value w value). The support size actually doesn't seem to be quadratic when plotted against w, as shown in the next section.

4.2 Support size for W-proj

The W-proj convolution kernel is the Fourier transform of the function

$$G(l,m) = e^{2\pi i w\sqrt{1-l^2-m^2}-1} \tag{7}$$

The support size depends on w, the FoV and the frequency, as the pixel size in the (u,v) space is the reciprocal of the FoV, and w is measured in wavelengths. Some example kernels are shown in Figure 7 in the Appendix. We could find out the support size numerically by applying the Fourier transform to [7] and counting pixels until the convolution function decreases belows a certain threshold, but an analytic expression for the support size has been reported in [4], and as Figure 8 shows, is in good agreement with the numerical approach. Hence, we use the expression

$$S(w,\theta_{fov}) = 2\theta_{fov}\sqrt{\left(\frac{2\theta_{fov}}{2}\right)^2 + \frac{w^{3/2}\theta_{fov}}{2\pi\eta}}. \tag{8}$$

4.3 Support size for A-proj

The A-proj kernel is the Fourier transform of the multplication of the primary beam functions of each antenna participating in a baseline. We assume that the primary beam can be approximated by a Gaussian[1] function with a half-power beamwidth θ_{HPBW}, which has the form

$$A(\theta) = A_0 \exp\left[-4\ln 2\left(\frac{\theta}{\theta_{HPBW}}\right)^2\right] \tag{10}$$

[1]Which is convenient because the Fourier transform of a Gaussian is a Gaussian:

$$\mathscr{F}\{e^{-ax^2}\}(u) = \sqrt{\frac{\pi}{a}}e^{-\pi^2u^2/a} \tag{9}$$

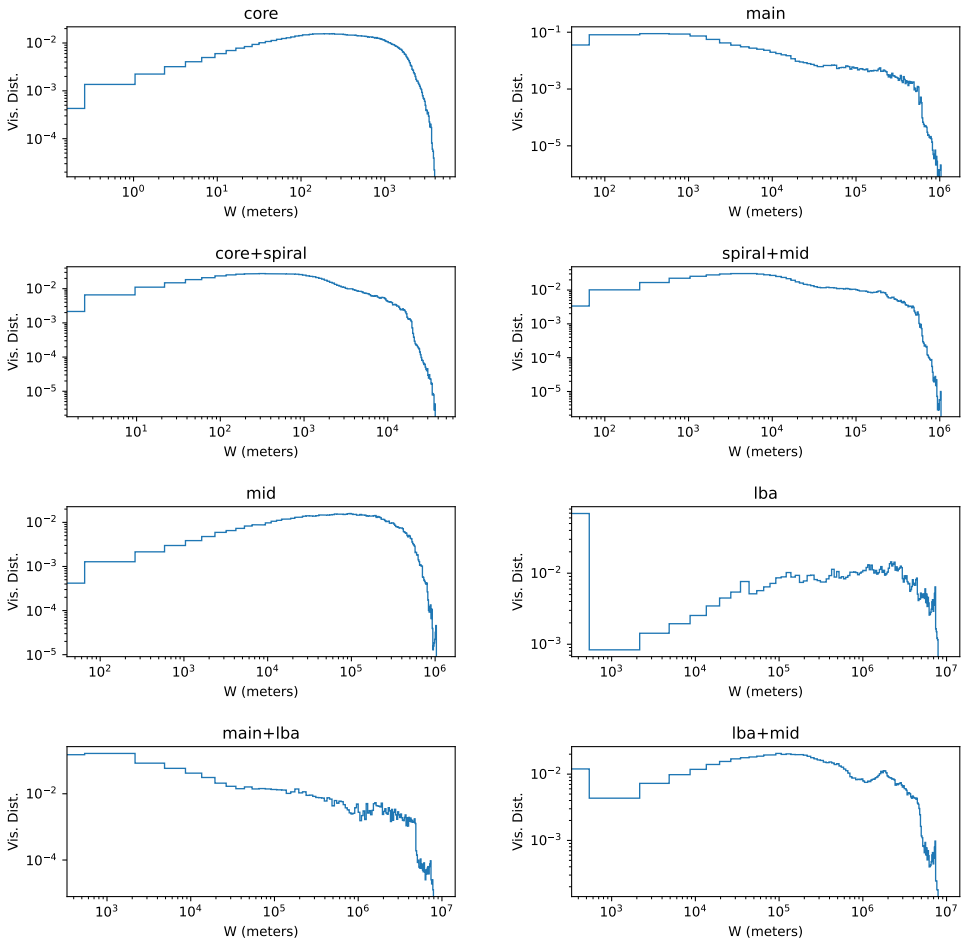

Figure 3: Normalized w-term distributions for the subarrays that are referenced in the ROP and EOP.

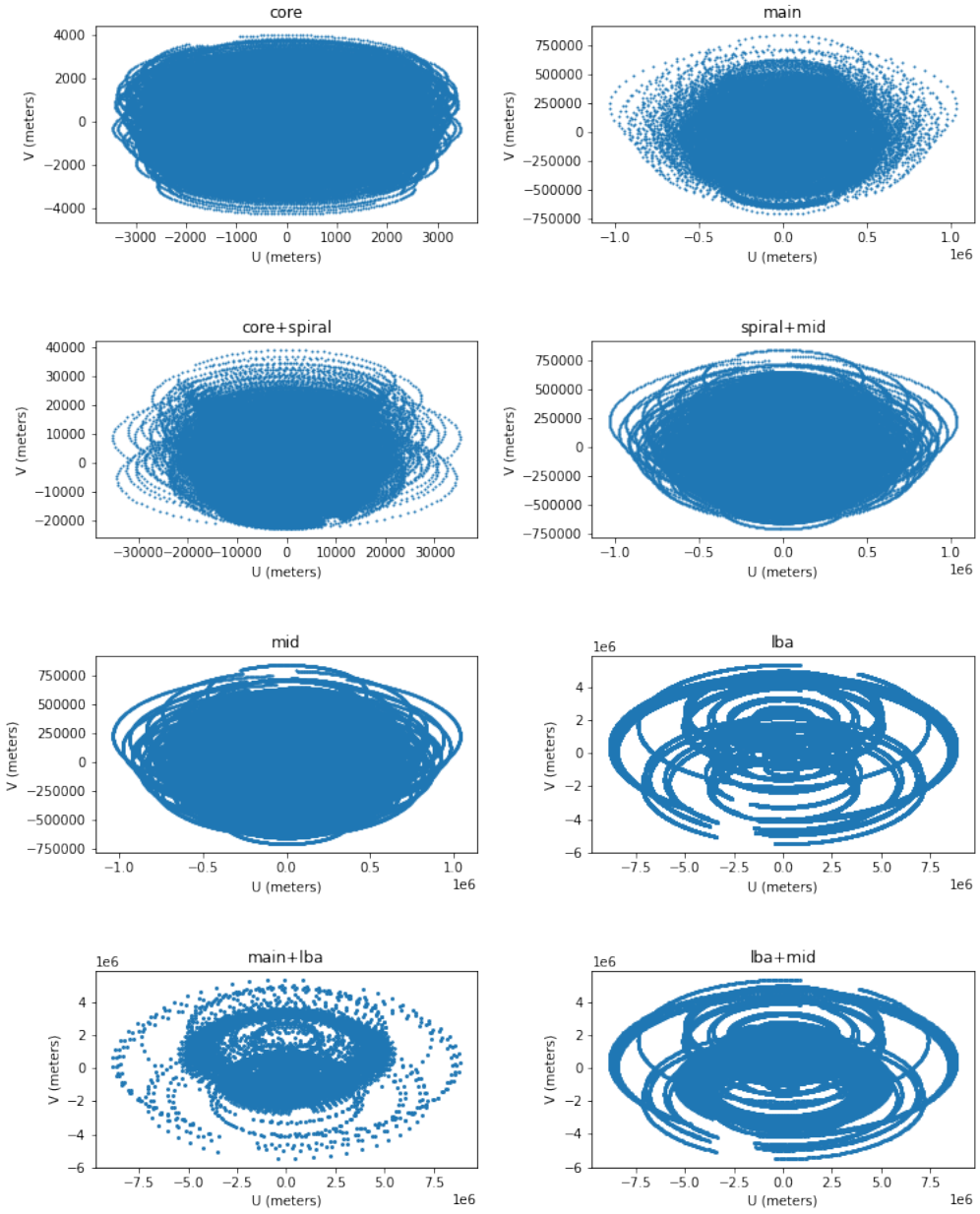

Figure 4: UV tracks for the subarrays that are referenced in the ROP and EOP.

Assuming all antennas have the same primary beam, the multiplication of two of them is another Gaussian:

$$A^2(\theta) = A_0^2 \exp\left[-8\ln 2\left(\frac{\theta}{\theta_{HPBW}}\right)^2\right] \tag{11}$$

We can derive a expression for the standard deviation of $\mathscr{F}\{A^2\}$:

$$\sigma = \frac{2\sqrt{\ln 2}}{\pi\theta_{HPBW}} = \frac{2D}{\pi\lambda}\sqrt{\ln 2} \tag{12}$$

where we approximated $\theta_{HPBW} \approx \lambda/D$.

As the size of the pixels $\Delta u = 1/\theta_{fov}$, we approximate the A-proj support size as 3σ:

$$S(\theta_{fov}, \nu) = \frac{3\sigma}{\Delta u} = \frac{6D\theta_{fov}\nu}{\pi c}\sqrt{\ln 2} \tag{13}$$

Note that the size in both [8] and [13] are only one side of the function so it needs to be multiplied by 2 to get the full support size of the convolution function.

4.4 Estimation of the complexity factor K_C

Using the results of the last three last sub-sections we estimate the complexity factor K_c calculating the weighed sum from Equation [5]. The range of w values is partitioned in 128 w-planes, and the bandwidth for each science case is partitioned in 10 intervals. The values of w in meters are converted to wavelengths for each one of these frequencies. Figure 5 shows how the factor K_c varies with FoV and frequency for the main subarray. It is readily apparent how the computational cost grows very fast for the wide-field science cases. A few science cases surpass a value of K_c of 100. For these W-proj will probably be too computationally expensive, and other approaches such as w-stacking, w-snapshots, or faceting maybe less computationally expensive than pure W-proj. Similar plots for the other subarrays referenced in the ROP and EOP can be found in the appendix.

Figure 5 shows in addition the usual criteria to decide when an observation requires W-proj or A-proj. An observation requires W-proj when

$$\frac{B_{max} \cdot \theta_{fov}^2}{\lambda} > 1 \tag{14}$$

with B_{max} being the maximum baseline of the subarray. This relation is shown as a segmented orange line in the plot. All science cases above this line would require wide-field correction. The green segmented line shows the criteria for A-proj, which is

$$\theta_{fov} > 0.5 \cdot \frac{\lambda}{D} \tag{15}$$

with D being the antenna diameter. Also shown in the plot as a segmented blue line is the primary beam FWHM. It can be seen that several science cases are above this line. These correspond to the cases where the FoV was specified as "FULL" in the ROP/EOP, and as explained above the *low frequency* was used to calculate the FoV. The decision of using the low frequency to calculate the FoV for these cases was verified with SciOps and represents a worst-case calculation. It is evident in the plot that if the middle or end frequency were used instead, the cost of computing for the cases demanding large FoV would decrease in an

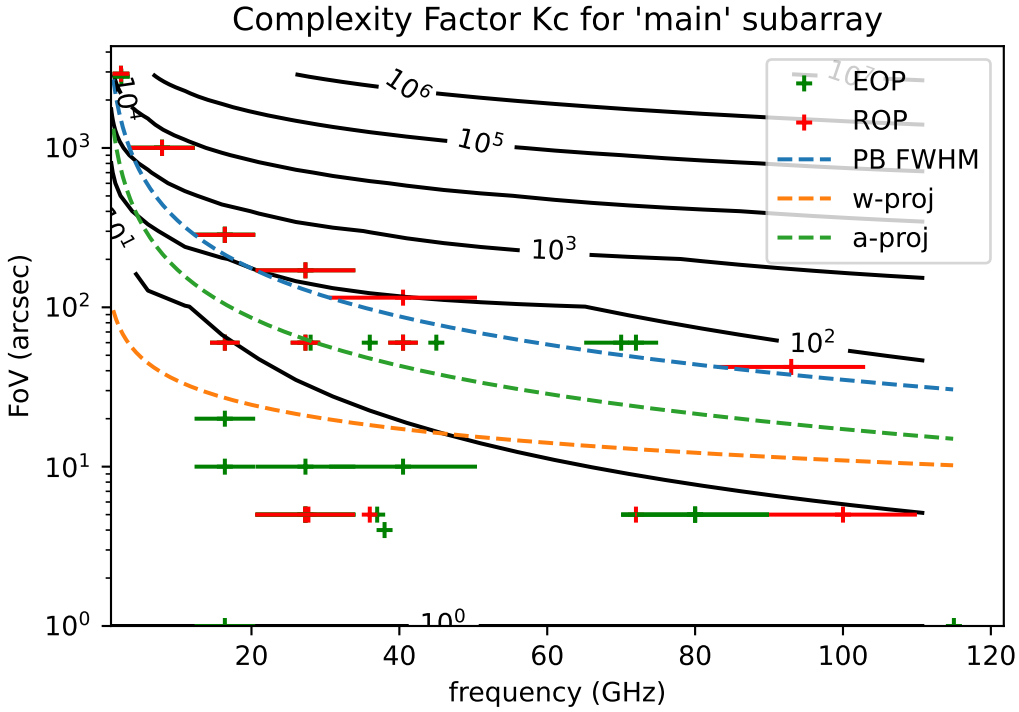

Figure 5: Dependency of K_c with FoV and frequency for the main subarray, along with the science cases for the ROP and EOP. The horizontal bars represent the bandwidth required by each science case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

order of magnitude. These are the observations that are driving the average computational load high.

For the purposes of this memo, we attempt to reduce the load of these cases by applying an hybrid algorithm that combines faceting with W-projection. This is illustrated in Figure 6 for ROP science case 15. We assume a number of facets ranging from 1 (i.e. no faceting at all) to a number of facets such that processing each one won't require W-projection. We calculate this maximum number of facets using equation 14. For each number of facet, we plot N^2, K_c and the multiplication of both, which is the computational cost of applying the hybrid algorithm. This function has a minimum, which becomes the value of K_c to use for these science cases.

The values of K_c, along with the computational load for each science case are presented in Table 4 for the ROP and Table 8 for the EOP. The averages are presented in Table 5.

5 Computational efficiencies

The metrics presented in Table 5 correspond to the computational loads generated by the observations in the ROP and EOP. In order for the processing system to image the incoming data without accumulating visibilities that will never be processed, the system throughput needs to be greater than the average computational load (handling the peak computational load is discussed in the next section). This is a basic requirement for any queueing system:

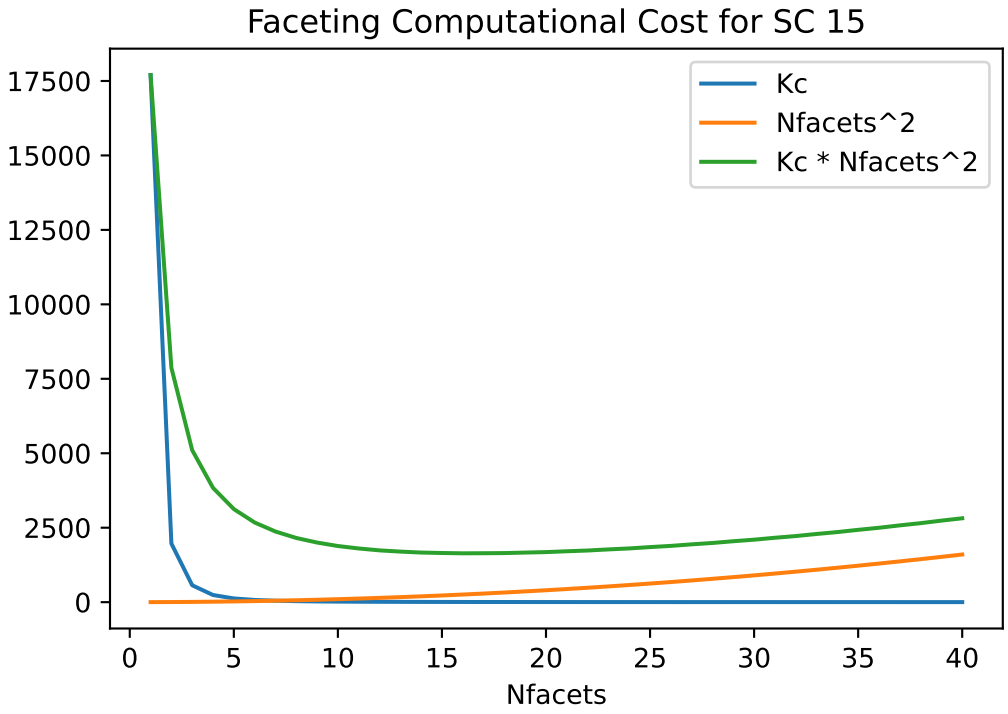

Figure 6: Example application of an hybrid algorithm that combines facets and W-proj to science case #15 of the ROP. The image is divided N^2_{facets} facets, and each one is processed using W-proj. The number of facets starts at 1 (no facetting) and ends with the number of facets such that processing each one doesn't require W-proj. The relative cost of processing all the facets is $K_c \cdot N^2_{facets}$. The minimum of this expression replaces K_c without faceting in the calculation of the computing load.

Table 4: Computing loads for ROP.

scnum	scname	frac	Kc	nchannels	GVis_per_sec	PFLOPs_per_sec
1	KSG1 Driving Cont Band 6 eg Taurus disk	0.09	1.148355	222	0.020428	0.007927
2	KSG1 Driving Cont Band 4 eg Taurus disk	0.04	1.009439	658	0.060549	0.020654
3	KSG2 Driving Line Band 5 eg Sgr B2(N)	0.04	14.256831	296296	27.265158	131.354488
4	KSG2 Driving Line Band 4 eg Sgr B2(N)	0.01	8.848658	439560	20.224156	60.473028
5	KSG2 Driving Line Band 3 eg Sgr B2(N)	0.01	4.834698	727272	33.461785	54.667898
6	KSG3 Driving Line Band 5 eg COSMOS	0.04	5.402650	29629	1.682453	3.071593
7	KSG3 Driving Line Band 4 eg COSMOS	0.01	6.450366	29670	0.842391	1.836165
8	KSG3 Driving Line Band 3 eg COSMOS	0.01	8.785482	30003	0.851845	2.528950
9	KSG3 Driving Line Band 6 eg Spiderweb galaxy	0.02	1.098031	33	0.003037	0.001127
10	KSG3 Driving Line Band 5 eg Spiderweb galaxy	0.01	1.024359	33	0.003037	0.001051
11	KSG3 Driving Line Band 4 eg Spiderweb galaxy	0.01	1.009525	33	0.001518	0.000518
12	KSG3 Driving Line Band 6 eg Virgo Cluster	0.07	3.183811	16000	0.908544	0.977478
13	KSG3 Driving Line Band 1 eg M81 Group	0.11	53.469595	1479	0.041992	0.758726
14	KSG3 Driving Line Band 1 eg M81 Group	0.13	10.250466	147	0.001313	0.004547
15	KSG5 Driving Cont Band 1 OTF Find LIGO event	0.07	1640.204834	11201	2.061432	1142.565236
16	KSG5 Driving Cont Band 4 OTF Find LISA event	0.07	6.450366	2700	0.306634	0.668372
17	KSG5+4 Driving Cont Band 2 OTF Find BHs+Possib...	0.04	56.332445	5037	0.572042	10.889312
18	KSG5 Driving Cont eg Band 2 Followup from OTF	0.00	1.000000	502	0.030010	0.010141
19	KSG5 Driving Cont Band 3 Gw170817@200Mpc	0.24	1.000000	1333	0.001240	0.000419
20	KSG3 Supporting Cont Band 6 eg Virgo Cluster	0.00	18.084185	4000	0.368080	2.249340
21	KSG3 Supporting Cont Band 5 eg Virgo Cluster	0.00	76.947115	4000	0.368080	9.570805
22	KSG3 Supporting Cont Band 4 eg Virgo Cluster	0.00	71.379288	2700	0.124227	2.996416
23	KSG3 Supporting Cont Band 3 eg Virgo Cluster	0.00	110.718614	1639	0.075410	2.821407
24	KSG3 Supporting Cont Band 2 eg Virgo Cluster	0.00	648.204980	5037	0.231752	50.763370

Table 5: Computational load summary for ROP and EOP.

	ROP	EOP
Average Computational Load [PFLOPs/sec]	85.478609	59.898202
Peak Computation Load [PFLOPs/sec]	1142.565236	1353.931735
80% Percentile [PFLOPs/sec]	3.018969	3.543039

the system throughput needs to be greater than the incoming input rate, or the queue becomes unstable.

In order to estimate how many computational resources will be needed in order to keep up with the computational load, we need to consider the fact that only a fraction of the maximum computing capability of a given processor is available to be used by a given algorithm. Let's define this fraction as the *computational efficiency*. This is a critical driver for the cost of the processing system.

Two effects need to be considered: the efficiency in a single processor, which we define as the *core efficiency*; and the efficiency achieved on a system of multiple processors, the *parallelization efficiency*. The overall computational efficiency is the multiplication of both.

The observed computing throughput of a single processor is

$$X^{obs,single} = \frac{\text{FLOPs}}{\text{Exec. time for a single proc.}}$$

On the other hand, the throughput of a parallel execution is

$$\begin{aligned} X^{obs,par} &= \frac{\text{FLOPs}}{\text{Exec. time for a parallel system}} \\ &= X^{obs,single} \cdot \frac{\text{Exec. time for a single proc.}}{\text{Exec. time for a parallel system}} \\ &= X^{obs,single} \cdot Sp(N) \end{aligned}$$

where $Sp(N)$ is the speedup, as a function of the number of processors N.

The observed processor throughput differs from the nominal maximum processor throughput by the core efficiency factor ϵ_c:

$$X^{obs,single} = \epsilon_c X^{single}.$$

Matching the load generated by gridding with the observed performance of the parallel system (otherwise the system becomes unstable)

$$CL = 1280 \cdot \lambda \cdot K_c \cdot K_e = X^{obs,par} = \epsilon_c X^{single} Sp(N)$$

Multiplying and dividing by the number of processors N and defining the parallelization efficiency $\epsilon_p = Sp(N)/N$ as the ratio of the real speedup and the ideal speedup (N):

$$CL = N \cdot X^{single} \epsilon_c \epsilon_p$$

This expression is convenient for calculating the cost of the system, as X^{single} is the maximum computing capacity of a processor as reported in its specification. The total number of processors that are needed to keep up with a given computational load is then

$$N_{proc} = \frac{CL}{\epsilon_c \epsilon_p X^{single}}. \tag{16}$$

If the cost of a processor is C_{proc}, then the cost of the SDP computing systems is

$$C_{sdp} = N_{proc} \cdot C_{proc} = \frac{CL}{\epsilon_c \epsilon_p} \cdot \frac{C_{proc}}{X^{single}} \tag{17}$$

However, C_{proc}/X^{single} is just the ($/FLOPs/sec) ratio that we can probably find out looking around at the HPC market, and we can define $CL' = CL/(\epsilon_c \epsilon_p)$ as an "effective" computing load that incorporates the efficiencies. With this

$$C_{sdp} = CL' \cdot \left(\frac{\$}{\text{FLOPs/sec}} \right) \tag{18}$$

For example, an NVIDIA A100 is capable of sustaining 9.7 TFLOPs/sec and costs around $10,000. This gives 103,092.8 $/PFLOPs/sec. Assuming both efficiencies as 1, we could build the SDP for $7.73MM. On the other hand, if the core efficiency is 0.1 and the parallelization efficiency is 0.8, then we would need to pay $96.64MM. This illustrates the importance of optimizing the efficiencies in the design and implementation. We don't have currently any estimation for these important parameters, so we can only say that the cost of the system will be more than $7.73MM!

In practice, an important factor limiting the core efficiency is the machine balance compared with the arithmetic intensity of the algorithm running in a machine. The *machine balance* for a given computer hardware is the total number of FLOPs that can be executed in the machine divided by the memory bandwidth. The *arithmetic intensity* for a given application measures the number of FLOPs executed per memory operation. Both are measured in FLOPs/Byte or FLOPs/Word. If the arithmetic intensity is lower than the machine balance, then the processor cannot achieve its computational capacity because it is not being fed the data that it needs for the computation at a high enough rate. Note that although the machine balance is defined in terms of the memory bandwidth (and many profiling tools based on the roofline model do so as well), the data needs to be fed into memory from disk in the first place, so it would be more proper to compare arithmetic intensities with a *system balance* that takes into consideration the full memory/storage hierarchy. Memory bandwidths can reach thousands of GBytes/sec, while data can be read from SSD disks only at a few GBytes/sec. One way to provide a more realistic system balance would be to consider the entire memory and storage hierarchy, the memory sizes at each level of the hierarchy, and the probability of cache hits given the usage patters of the algorithm and the size of the cache lines. See for example [7].

As pointed out in [1], the total number of bytes that need to be read to grid a single visibility point are

- 2 complex numbers for two polarizations: $2 \times 2 \times 4 = 16$ bytes,
- 2 weight floats: $2 \times 4 = 8$ bytes,
- 3 uvw coordinate doubles: $3 \times 8 = 24$ bytes,

For a total of 48 bytes. On the other hand, the number of floating point operations is 40 FLOPs $\times S^2(w, \nu)$ for complex gridding, and 24 FLOPs $\times 7^2$ for standard gridding. This gives an arithmetic intensity of

$$AI = \frac{40}{48} \cdot S^2(w, \nu) = 0.83 \cdot S^2(w, \nu) \tag{19}$$

for complex gridding, while for standard gridding $AI = 24.5$ FLOPs/Byte. Note that this calculation does **not** include the convolution functions. If these need to be loaded then the AI falls to very low numbers. We assume that they will be computed "on-the-fly" on the GPU or CPU, and loaded into GPU memory using a high-bandwidth interconnect, to the point that it doesn't affect the efficiency of the computations. This is an important aspect for DMS to verify, design and implement.

For example, an NVIDIA V100 GPU is capable of sustaining 6.6 TFLOPs/sec in FP64. Assuming that we feed this GPU from a single SSD disk with a bandwidth of 2.0 GBytes/sec, the "system balance" would be 3,300. The AI would need to be higher than this number in order to fully use the GPU. For standard gridding we are hopeless. At this bandwidth, we can only achieve a maximum of 24.5 FLOPs/Byte $\times$ 2.0 GBytes/sec = 49 GFLOPs/sec, a core efficiency of only $49/6,600 = 0.0074$, not even 1%! For complex gridding the support function would need to be at least 63 for processing not to be constrained by I/O. As most of the workloads in the ROP/EOP don't require A-proj or W-proj, we should highlight that solving the bandwidth problem is critical for reducing the cost of the system.

Let's switch now our attention to the parallelization efficiency. As explained above

$$\epsilon_p = \frac{\text{Real Speedup}}{\text{Ideal Speedup}} = \frac{Sp(N)}{N} \tag{20}$$

The behavior of this parameter depends on how the problem scales. There are two different ways of characterizing the scaling behaviour:

- **Strong scaling** is defined as how the speedup behaves as the number of processors increases for a fixed problem size.
- **Weak scaling** is defined as how the speedup behaves as the number of processors increases for a fixed problem size per processor.

In other words, in a strong scaling problem we maintain the size of the problem the same as we add processors, while in a weak scaling problem the size of the problem increases as we add processors.

If the problem is divided in the parts that can be parallelized, and the parts that can't, assigning a fraction s to the former and a fraction p to the latter ($s + p = 1$), then strong scalability follows Amdahl's law

$$\begin{aligned} Sp(N) &= \frac{s+p}{s+\frac{p}{N}} \\ &= \frac{1}{s+\frac{p}{N}} \xrightarrow[N\to\infty]{} \frac{1}{s} \end{aligned}$$

and the speedup saturates as N becomes large. This fact will affect the latency that it is possible to achieve. For a strongly scaling problem there is a point where adding more processors doesn't improve the latency, and the parallelization efficiency becomes very small:

$$\epsilon_p = \frac{1}{N \cdot s + p} \xrightarrow[N\to\infty]{} 0$$

On the other hand, weak scaling follows Gustafson's Law

$$Sp(N) = \frac{s + p \cdot N}{s + p}$$

$$= s + p \cdot N$$

$$\epsilon_p = \frac{s + p \cdot N}{N} \xrightarrow[N \to \infty]{} p$$

and the efficiency approaches the fraction of the problem that can be parallelized when the number of processors is large.

Does our problem exhibit weak or strongly scaling behaviour? Both, actually. When doing cube imaging, each spectral channel can be processed fairly independently. As we add more spectral planes, the total problem size just becomes bigger. This is a weak scaling problem, or how it is sometimes called, an *embarrasingly parallel* problem. On the other hand, when doing continuous imaging, MFS, or the processing of each spectral channel in a cube; there are portions of the problem that can be parallelized (gridding, FFT), while other parts of the algorithm are more difficult to parallelize (e.g. the deconvolution). This is strongly parallel problem. Which one of these scaling law dominates will depend on the science case. When the size of the problem is determined by the number of channels and we want to do cube imaging (e.g. ROP KSG 2) then we have a weak scaling problem. On the other hand, if the size of the problem is determined by the size of the support function (e.g. ROP 15) and we are doing continuous imaging, then we could expect the problem to exhibit a strongly scaling behaviour.

In practical terms, if the problem scales strongly we would assign just enough processors to maintain the parallelization efficiency to a reasonable level. This may affect the latency that such cases can achieve. As explained in the following section, this doesn't affect the total cost of the system, as the total number of processors required to maintain the average throughput is unaffected by this assignment. On the other hand, not being able to achieve a sufficiently high parallelization efficiency could affect the processing system's ability to deliver products for triggered observations with their required timeliness.

6 Conclusions

The data rates for the ROP and EOP were calculated and summarized in Table 3. The EOP data rates are larger than the ROP (by 36% for the average). Both distributions are similar, with most of the science cases generating relatively modest data rates (more than 80% of the science cases are less than 5 GVis/sec), with a few cases producing much larger data rates (as large as 40 GVis/sec).

A simple model was developed to characterize the computational load that the science cases in the ROP and EOP will impose on the system, from the visibility data rate (equation 6). Several processing options (including self-calibration) and the iterative nature of the algorithms where included as factors over the processing of a standard gridding major cycle. The impact of having to compensate for wide-field distortions and the effect of the primary beam on the computational load was captured in the complexity factor (K_c), which was defined as the weighted average of the squared support function. This parameter increases very rapidly with the field of view and the frequency of each observation. Most of the science cases defined in the ROP and EOP have modest complexity factors, but for a few cases where this parameter is significatively higher, due to the combination of their required field of view and frequency. These cases drive the required computational capacity for the processing system an order of magnitude higher than otherwise would be required. Science cases that generate high data rates also drive the computational load high, but

their impact is secondary. The required computational loads for the ROP and EOP are summarized in Table 5.

In order to specify the system in terms of number of processors and size of storage buffers; the efficiency of executing the software implementation of the algorithms on a physical hardware platform need to be considered. We developed a simple model based on two parameters: the core efficiency and the parallelization efficiency. The cost of the processing system directly depend on the values of these parameters. As these are at this moment largely unknown, it is not possible to estimate the cost of the processing system. These efficiencies not only depend on the quality of an implementation, but on system design parameters such as the system aggregated bandwidth and the proportion of code that has/can been parallelized. Low efficiencies can increase the cost of the processing system by orders of magnitude. If the cost for the processing system is capped, low efficiencies will constraint the ability of the ngVLA to deliver the science it was designed for. We suggest to mitigate this risk by allocating resources to develop prototype implementations and evaluate their performance, with the objective of providing estimations of these parameters as close as possible to the real values. This work should inform the system design and algorithm development activities. In general, with the exception of the measurements presented in [1] — which only included FLOPs/sec and I/O rate for a few cases — the model used in this memo to extrapolate the computing loads to the scale of ngVLA relies solely on theoretical arguments. These should be validated by measurement, and until this is done, the figures resulting from this model should be considered only as order-of-magnitude estimations.

It is worthwhile to point out that the estimations presented in this document may represent a worst-case model, that could be overestimating the computational load. There are several opportunities for reducing the computational load that were not introduced in the calculations, as they require either further analysis or requirement clarifications. These are:

- As discussed in section 4.4, several science cases in the ROP and EOP have defined their required FoV as "FULL". This requirement is ambiguous, as the calculation of the FoV depends on a frequency that is not well defined for an observation with a sizable bandwidth. We have followed the guideline of using the lower frequency in the bandwidth as a worst-case estimation, but as Figure 5 shows, a significant reduction in the computational load can be achived if these cases can define a smaller FoV.

- As pointed out in [5], the subarray used by each science case in the ROP and EOP includes baselines that are longer than necessary for the required angular resolution. Substantial reductions in data rates and computational load can be achieved if the subarrays assigned to each case are tailored to their resolution, discarding baselines that won't be gridded. Decreasing the number of antennas implies a reduction in sensitivity that needs to be compensated with longer observation times.

- We have applied equations 14 and 15 to determine whether each science case require W-proj and/or A-proj. However, these decisions may need to be reconsidered in light of other science requirements. One example is ROP-15, which generates a very high imaging computational load given its large FoV, even though the images will be used for quick-look purposes. Would it be possible to decrease the amount of processing generated by W-proj in this case, and maybe other similar cases?

- We have applied an hybrid algorithm that combines facetting and W-proj to reduce the computational load of wide-field cases. Other algorithms may be able to optimize

processing even further, e.g. w-stacking or w-snapshots. These possibilities need to be investigated and quantified.

Nevertheless, the estimations presented in this memo do represent our current understanding of the data rate and computational requirements from the information that we have currently available. We recommend that further analysis of the above topics be prioritized, and the parameters necessary to introduce these optimizations be added to the ROP and EOP input tables prepared by SciOps. Once this has been done, the data rate and computational load estimations can be updated accordingly.

References

[1] S. Bhatnagar, R. Hiriart, and M. Pokorny. ngVLA Computing Memo 4, Size-of-Computing Estimates for ngVLA Synthesis Imaging. Technical report, NRAO ngVLA, 2021.

[2] A. H. Bridle and F. R. Schwab. Bandwidth and Time-Average Smearing. In G. B. Taylor, C. L. Carilli, and R. A. Perley, editors, *Synthesis Imaging in Radio Astronomy II*, volume 180 of *Astronomical Society of the Pacific Conference Series*, page 371, January 1999.

[3] J. Kern, R. Selina, D. Arancibia, J. Hibbard, C. Langley, and S. Myers. ngVLA Quantitative eXchange Model. Technical report, NRAO ngVLA, 2017.

[4] Bruce Merry. Approximating W projection as a separable kernel. *Monthly Notices of the Royal Astronomical Society*, 456(2):1761–1766, 12 2015.

[5] Jan-Willem Steeb. , ngVLA Computing Memo 8, Time Averaging Limits for the ngVLA. Technical report, NRAO, 2022.

[6] A. Richard Thompson, James M. Moran, and Jr. Swenson, George W. *Interferometry and Synthesis in Radio Astronomy, 3rd Edition.* 2017.

[7] Charlene Yang, Thorsten Kurth, and Samuel Williams. Hierarchical roofline analysis for gpus: Accelerating performance optimization for the nersc-9 perlmutter system. *Concurrency and Computation: Practice and Experience*, 32, 11 2019.

A Additional Tables and Figures

Table 6: Envelope Observing Program.

scnum	scname	frac	fov	psf	dr	freq	bw	chan	dump	sub-array
1	Protoplanetary disks (100 looks)	4.8600	5.00E+00	5.00	100.0	8.000000e+10	FULL	120000000	1.0000	Main
2	8 protoplanetary disks	4.4500	5.00E+00	5.00	100.0	2.730000e+10	FULL	120000000	1.0000	Main
3	2 protoplanetary disks (substructures)	0.0948	5.00E+00	15.00	100.0	8.000000e+10	FULL	120000000	1.0000	Main
4		5.5000	5.00E+00	15.00	100.0	2.730000e+10	FULL	120000000	1.0000	Main
5	2 protoplanetary disks (circumplanets)	0.2210	5.00E+00	5.00	1000.0	8.000000e+10	FULL	120000000	1.0000	Main
6		5.2100	5.00E+00	5.00	1000.0	2.730000e+10	FULL	120000000	1.0000	Main
7	6 hot cores (5 frequency settings)	6.9500	6.00E+01	100.00	1000.0	4.050000e+10	4000000000	13500	1.0000	Main
8	6 hot cores (3 frequency settings)	0.4170	6.00E+01	100.00	1000.0	2.730000e+10	4000000000	9100	1.0000	Main
9	6 hot cores (2 frequency settings)	0.2090	6.00E+01	100.00	1000.0	1.640000e+10	4000000000	5500	2.0000	Main
10	3 protoplanetary disks (NH3 snowlines)	1.9600	5.00E+00	400.00	100.0	2.400000e+10	500000	80000	1.0000	Plains+Core
11	Protoplanetary disk (organic lines)	2.4300	5.00E+00	200.00	100.0	3.700000e+10	100000000	24700	1.0000	Main
12	Hot core (organic lines)	0.5790	6.00E+01	35.00	1000.0	4.500000e+10	200000000	150000	1.0000	Main
13	Protostellar cores (deuterated lines)	4.8600	6.00E+01	200.00	100.0	7.000000e+10	10000000000	350000	1.0000	Main
14	Comet (NH3)	0.0521	2.00E+01	1000.00	10.0	2.400000e+10	1280000	8000	1.0000	Plains+Core
15	Uranus (bright polar regions)	2.2300	2.00E+01	20.00	1000.0	1.640000e+10	FULL	120000000	2.0000	Main
16	500 massive stars (wind clumping)	0.5680	1.00E-01	8.00	100.0	1.640000e+10	FULL	120000000	2.0000	Plains+Mid
17	50 red giant stars (2D velocities in globular ...	1.0600	1.00E-01	1.50	100.0	2.730000e+10	FULL	120000000	1.0000	LBA
18	CO galaxy search (8 pointings)	2.6400	FULL	2000.00	100.0	4.050000e+10	FULL	675000	1.0000	Core
19		1.0300	FULL	2000.00	100.0	2.730000e+10	FULL	455000	1.0000	Core
20		0.7210	FULL	2000.00	100.0	1.640000e+10	FULL	273000	2.0000	Plains+Core
21	4 CO galaxies (lines z~2)	3.1900	6.00E+01	100.00	1000.0	7.200000e+10	240000000	7200000	1.0000	Main
22		0.3710	6.00E+01	100.00	1000.0	3.600000e+10	120000000	3600000	1.0000	Main
23		0.2780	6.00E+01	100.00	1000.0	2.800000e+10	92300000	2800000	1.0000	Main
24	50 dusty galaxy redshifts (CO z~2)	0.6680	4.00E+00	2000.00	100.0	3.840000e+10	76800000	1280000	1.0000	Core
25	3 galaxies' dense gas (HCN z~2)	0.5070	4.00E+00	2000.00	100.0	2.900000e+10	4000000000	967000	1.0000	Core
26	CO galaxy environs (lines)	0.0888	6.00E+01	1500.00	1000.0	3.600000e+10	120000000	8400000	1.0000	Core
27	2 JWST deep fields	1.5200	FULL	190.00	100000.0	1.640000e+10	FULL	120000000	2.0000	Main
28		0.5000	FULL	150.00	100000.0	7.900000e+09	FULL	5000000	0.5000	Main
29		1.5200	FULL	190.00	100000.0	1.640000e+10	FULL	120000000	2.0000	Main
30		0.5000	FULL	150.00	100000.0	7.900000e+09	FULL	5000000	0.5000	Main
31	3 AGN outflows (CO z~0.5-2)	1.1300	4.00E+00	50.00	1000.0	3.800000e+10	253000000	3800000	1.0000	Main
32	Obscured AGN in 1 deg^2 (62 pointings)	0.7760	FULL	150.00	100000.0	7.900000e+09	FULL	5000000	0.5000	Main
33	CO in nearby galaxies (17 pointings)	5.2200	FULL	100.00	1000.0	1.150000e+11	6000000000	767000	1.0000	Plains+Core
34	HI in 5 group galaxies	2.5600	FULL	2000.00	1000.0	1.420000e+09	7100000	9470	2.0000	Plains+Core
35	HI around group galaxies	1.2200	FULL	60000.00	1000.0	1.420000e+09	7100000	47300	2.0000	Core
36	2 nearby galaxies (5 pointings each)	1.5900	FULL	1000.00	5000.0	4.050000e+10	FULL	5000000	1.0000	Plains+Core
37	2 nearby galaxies (7 pointings each)	1.0700	FULL	1000.00	5000.0	2.730000e+10	FULL	5000000	2.0000	Plains+Core
38	2 nearby galaxies (3 pointings each)	0.4170	FULL	1000.00	5000.0	1.640000e+10	FULL	5000000	2.0000	Main
39	2 nearby galaxies (1 pointing each)	0.0863	FULL	1000.00	5000.0	7.900000e+09	FULL	5000000	2.0000	Main
40	2 nearby galaxies (1 pointing each)	0.1440	FULL	1000.00	5000.0	2.350000e+09	FULL	2000000	0.5000	Main
41	Massive SF in local galaxy	0.2160	1.00E+01	6.00	1000.0	4.050000e+10	FULL	120000000	1.0000	Main
42		0.2160	1.00E+01	6.00	1000.0	2.730000e+10	FULL	120000000	1.0000	Main
43		0.4860	1.00E+01	6.00	1000.0	1.640000e+10	FULL	120000000	2.0000	Main
44	NH3 in local galaxy (10 pointings)	0.6240	FULL	2500.00	100.0	2.300000e+10	115000000	767000	2.0000	Core
45	CO in 2 local group galaxies (1 pointing each)	2.9600	FULL	200.00	100.0	1.150000e+11	6000000000	767000	1.0000	Plains+Core
46	HCN in nearby galaxy (10 pointings)	3.5600	FULL	1000.00	100.0	8.800000e+10	440000000	2930000	1.0000	Core
47	Globular star clusters in 3 nearby galaxies	1.3700	FULL	100.00	1000.0	1.640000e+10	FULL	5000000	2.0000	Plains+Mid
48	Parallaxes of 50 SiO masers in Galactic bulge	0.6820	1.00E+00	0.15	10000.0	8.600000e+10	500000000	287000	1.0000	LBA

Continued on next page

scnum	scname	frac	fov	psf	dr	freq	bw	chan	dump	sub-array
49	Proper motions of 50 galaxies in Virgo Cluster	0.9240	1.00E-01	1.00	1000.0	4.050000e+10	FULL	120000000	1.0000	Main+LBA
50	Pulsar search near Sgr A* (30 looks)	2.0900	1.00E-01	20.00	100.0	1.640000e+10	FULL	1000000	0.0001	Main
51	Localize 10 LIGO events (2 looks each)	2.0600	FULL	1000.00	5000.0	2.400000e+09	FULL	2000000	0.5000	Main
52	Proper motions of 3 LIGO events (2 looks each)	1.0000	1.00E+00	0.60	100.0	1.640000e+10	FULL	120000000	2.0000	LBA
53	Localize 10 LISA events (2 looks each)	3.6400	FULL	1000.00	5000.0	2.730000e+10	FULL	5000000	0.5000	Plains+Core
54	Find candidate BHs and pulsars in inner Galaxy	1.7000	FULL	1000.00	5000.0	7.900000e+09	FULL	5000000	0.5000	Plains+Core
55	Astrometry of 50 BH candidates	1.9300	1.00E+00	10.00	100.0	1.640000e+10	FULL	120000000	2.0000	Main
56	Parallax of a BH (5 looks)	0.2060	1.00E+00	1.00	100.0	1.640000e+10	FULL	120000000	2.0000	LBA+Mid
57	Orbital wobble of Cyg X-1 (20 looks)	0.0590	1.00E-01	0.17	1000.0	4.050000e+10	FULL	120000000	1.0000	LBA
58	SMBH masses in 12 nearby galaxies (CO)	0.5210	1.00E+00	50.00	1000.0	1.150000e+11	2000000000	3830000	1.0000	Main
59	Time 10 pulsars (52 looks each)	0.7060	1.00E-01	100.00	100.0	2.350000e+09	FULL	500000	2.0000	Mid
60	Jet of EHT primary (2 looks)	0.6150	1.00E-01	0.10	10000.0	8.000000e+10	FULL	120000000	1.0000	LBA+Mid
61	Jet of EHT primary (2 looks)	0.6150	1.00E-01	0.10	10000.0	8.000000e+10	FULL	120000000	1.0000	LBA+Mid
62	2 primaries plus 4 others (52 looks each)	0.9390	FULL	1000.00	10000.0	2.730000e+10	FULL	5000000	2.0000	Main
63	Find ejected SMBHs	2.6000	2.00E+01	15.00	1000.0	7.900000e+09	FULL	5000000	2.0000	Plains+Mid

Table 7: EOP derived parameters.

scnum	scname	bandwidth	fov	img	eff_channel_hz	eff_num_channels	dump
1	Protoplanetary disks (100 looks)	20.00000	5.000000	3000	3.500000e+07	571	1.0
2	8 protoplanetary disks	13.50000	5.000000	3000	1.027500e+07	1313	1.0
3	2 protoplanetary disks (substructures)	20.00000	5.000000	1000	1.050000e+08	190	1.0
4		13.50000	5.000000	1000	3.082500e+07	437	1.0
5	2 protoplanetary disks (circumplanets)	20.00000	5.000000	3000	3.500000e+07	571	1.0
6		13.50000	5.000000	3000	1.027500e+07	1313	1.0
7	6 hot cores (5 frequency settings)	4.00000	60.000000	1800	1.350000e+04	296296	1.0
8	6 hot cores (3 frequency settings)	4.00000	60.000000	1800	9.100000e+03	439560	1.0
9	6 hot cores (2 frequency settings)	4.00000	60.000000	1800	5.500000e+03	727272	2.0
10	3 protoplanetary disks (NH3 snowlines)	0.00050	5.000000	37	8.000000e+04	6	1.0
11	Protoplanetary disk (organic lines)	0.10000	5.000000	75	2.470000e+04	4048	1.0
12	Hot core (organic lines)	0.20000	60.000000	5142	1.500000e+05	1333	1.0
13	Protostellar cores (deuterated lines)	10.00000	60.000000	900	3.500000e+05	28571	1.0
14	Comet (NH3)	0.00128	20.000000	60	8.000000e+03	160	1.0
15	Uranus (bright polar regions)	8.20000	20.000000	3000	6.150000e+06	1333	2.0
16	500 massive stars (wind clumping)	8.20000	0.100000	37	1.200000e+08	68	2.0
17	50 red giant stars (2D velocities in globular ...	13.50000	0.100000	200	1.200000e+08	112	1.0
18	CO galaxy search (8 pointings)	20.00000	114.967269	172	6.750000e+05	29629	1.0
19		13.50000	170.632686	255	4.550000e+05	29670	1.0
20		8.20000	285.081440	427	2.730000e+05	30036	2.0
21	4 CO galaxies (lines z~2)	0.24000	60.000000	1800	7.200000e+06	33	1.0
22		0.12000	60.000000	1800	3.600000e+06	33	1.0
23		0.09230	60.000000	1800	2.800000e+06	32	1.0
24	50 dusty galaxy redshifts (CO z~2)	0.07680	4.000000	6	1.280000e+06	60	1.0
25	3 galaxies' dense gas (HCN z~2)	4.00000	4.000000	6	9.670000e+05	4136	1.0
26	CO galaxy environs (lines)	0.12000	60.000000	120	8.400000e+06	14	1.0
27	2 JWST deep fields	8.20000	285.081440	4501	4.098829e+06	2000	2.0
28		8.80000	1001.857630	20037	2.620133e+05	33586	0.5
29		8.20000	285.081440	4501	4.098829e+06	2000	2.0
30		8.80000	1001.857630	20037	2.620133e+05	33586	0.5
31	3 AGN outflows (CO z~0.5-2)	0.25300	4.000000	240	3.800000e+06	66	1.0
32	Obscured AGN in 1 deg^2 (62 pointings)	8.80000	1001.857630	20037	2.620133e+05	33586	0.5
33	CO in nearby galaxies (17 pointings)	6.00000	31.308051	939	7.670000e+05	7822	1.0
34	HI in 5 group galaxies	0.00710	2475.556289	3713	9.470000e+03	749	2.0
35	HI around group galaxies	0.00710	2475.556289	123	4.730000e+04	150	2.0
36	2 nearby galaxies (5 pointings each)	20.00000	114.967269	344	5.000000e+06	4000	1.0
37	2 nearby galaxies (7 pointings each)	13.50000	170.632686	511	5.000000e+06	2700	2.0
38	2 nearby galaxies (3 pointings each)	8.20000	285.081440	855	5.000000e+06	1639	2.0
39	2 nearby galaxies (1 pointing each)	8.80000	1001.857630	3005	1.746755e+06	5037	2.0
40	2 nearby galaxies (1 pointing each)	2.30000	2922.084755	8766	2.053329e+05	11201	0.5
41	Massive SF in local galaxy	20.00000	10.000000	5000	9.150000e+06	2185	1.0
42		13.50000	10.000000	5000	6.165000e+06	2189	1.0
43		8.20000	10.000000	5000	3.690000e+06	2222	2.0
44	NH3 in local galaxy (10 pointings)	0.11500	152.838693	183	7.670000e+05	149	2.0
45	CO in 2 local group galaxies (1 pointing each)	6.00000	31.308051	469	7.670000e+05	7822	1.0
46	HCN in nearby galaxy (10 pointings)	0.44000	39.946476	119	2.930000e+06	150	1.0
47	Globular star clusters in 3 nearby galaxies	8.20000	285.081440	8552	2.157278e+06	3801	2.0
48	Parallaxes of 50 SiO masers in Galactic bulge	0.50000	1.000000	20000	2.870000e+05	1742	1.0

Continued on next page

scnum	scname	bandwidth	fov	img	eff_channel_hz	eff_num_channels	dump
49	Proper motions of 50 galaxies in Virgo Cluster	20.00000	0.100000	300	1.200000e+08	166	1.0
51	Localize 10 LIGO events (2 looks each)	2.30000	2805.201365	8415	2.228004e+05	10323	0.5
52	Proper motions of 3 LIGO events (2 looks each)	8.20000	1.000000	5000	3.690000e+06	2222	2.0
53	Localize 10 LISA events (2 looks each)	13.50000	170.632686	511	5.000000e+06	2700	0.5
54	Find candidate BHs and pulsars in inner Galaxy	8.80000	1001.857630	3005	1.746755e+06	5037	0.5
55	Astrometry of 50 BH candidates	8.20000	1.000000	300	6.150000e+07	133	2.0
56	Parallax of a BH (5 looks)	8.20000	1.000000	3000	6.150000e+06	1333	2.0
57	Orbital wobble of Cyg X-1 (20 looks)	20.00000	0.100000	1764	2.592500e+07	771	1.0
58	SMBH masses in 12 nearby galaxies (CO)	2.00000	1.000000	60	3.830000e+06	522	1.0
60	Jet of EHT primary (2 looks)	20.00000	0.100000	3000	3.500000e+07	571	1.0
61	Jet of EHT primary (2 looks)	20.00000	0.100000	3000	3.500000e+07	571	1.0
62	2 primaries plus 4 others (52 looks each)	13.50000	170.632686	511	5.000000e+06	2700	2.0
63	Find ejected SMBHs	8.80000	20.000000	4000	1.312500e+06	6704	2.0

Table 8: EOP computing loads.

scnum	scname	frac	Kc	nchannels	GVis_per_sec	PFLOPs_per_sec
1	Protoplanetary disks (100 looks)	4.8600	1.114071	571	0.052543	0.019781
2	8 protoplanetary disks	4.4500	1.009448	1313	0.120822	0.041214
3	2 protoplanetary disks (substructures)	0.0948	1.114071	190	0.017484	0.006582
4		5.5000	1.009448	437	0.040213	0.013717
5	2 protoplanetary disks (circumplanets)	0.2210	1.114071	571	0.052543	0.019781
6		5.2100	1.009448	1313	0.120822	0.041214
7	6 hot cores (5 frequency settings)	6.9500	14.256831	296296	27.265158	131.354488
8	6 hot cores (3 frequency settings)	0.4170	8.848658	439560	40.448311	120.946056
9	6 hot cores (2 frequency settings)	0.2090	4.834698	727272	33.461785	54.667898
10	3 protoplanetary disks (NH3 snowlines)	1.9600	1.000000	6	0.000341	0.000115
11	Protoplanetary disk (organic lines)	2.4300	1.026087	4048	0.372497	0.129158
12	Hot core (organic lines)	0.5790	16.301693	1333	0.122663	0.675708
13	Protostellar cores (deuterated lines)	4.8600	29.333975	28571	2.629103	26.061085
14	Comet (NH3)	0.0521	1.000044	160	0.009085	0.003070
15	Uranus (bright polar regions)	2.2300	1.395569	1333	0.061331	0.028923
16	500 massive stars (wind clumping)	0.5680	1.000000	68	0.000987	0.000334
17	50 red giant stars (2D velocities in globular ...	1.0600	1.000000	112	0.000208	0.000070
18	CO galaxy search (8 pointings)	2.6400	3.830441	29629	0.529174	0.684953
19		1.0300	4.039493	29670	0.529906	0.723336
20		0.7210	8.785482	30036	0.852782	2.531731
21	4 CO galaxies (lines z~2)	3.1900	30.491170	33	0.003037	0.031288
22		0.3710	12.311254	33	0.003037	0.012633
23		0.2780	9.111984	32	0.002945	0.009067
24	50 dusty galaxy redshifts (CO z~2)	0.6680	1.000000	60	0.001072	0.000362
25	3 galaxies' dense gas (HCN z~2)	0.5070	1.000000	4136	0.073869	0.024962
26	CO galaxy environs (lines)	0.0888	2.176776	14	0.000250	0.000184
27	2 JWST deep fields	1.5200	110.718614	2000	0.092020	3.442839
28		0.5000	648.204980	33586	6.181167	1353.931735
29		1.5200	110.718614	2000	0.092020	3.442839
30		0.5000	648.204980	33586	6.181167	1353.931735
31	3 AGN outflows (CO z~0.5-2)	1.1300	1.004324	66	0.006073	0.002061
32	Obscured AGN in 1 deg^2 (62 pointings)	0.7760	648.204980	33586	6.181167	1353.931735
33	CO in nearby galaxies (17 pointings)	5.2200	3.174746	7822	0.444164	0.476504
34	HI in 5 group galaxies	2.5600	53.474631	749	0.021266	0.384273
35	HI around group galaxies	1.2200	10.251194	150	0.001339	0.004640
36	2 nearby galaxies (5 pointings each)	1.5900	5.402650	4000	0.227136	0.414674
37	2 nearby galaxies (7 pointings each)	1.0700	6.450366	2700	0.076658	0.167093
38	2 nearby galaxies (3 pointings each)	0.4170	110.718614	1639	0.075410	2.821407
39	2 nearby galaxies (1 pointing each)	0.0863	648.204980	5037	0.231752	50.763370
40	2 nearby galaxies (1 pointing each)	0.1440	1640.204834	11201	2.061432	1142.565236
41	Massive SF in local galaxy	0.2160	1.246283	2185	0.201064	0.084677
42		0.2160	1.164317	2189	0.201432	0.079253
43		0.4860	1.086349	2222	0.102234	0.037530
44	NH3 in local galaxy (10 pointings)	0.6240	3.108607	149	0.001331	0.001398
45	CO in 2 local group galaxies (1 pointing each)	2.9600	3.174746	7822	0.444164	0.476504
46	HCN in nearby galaxy (10 pointings)	3.5600	2.913313	150	0.002679	0.002637
47	Globular star clusters in 3 nearby galaxies	1.3700	191.871322	3801	0.055191	3.578396
48	Parallaxes of 50 SiO masers in Galactic bulge	0.6820	1.143003	1742	0.003240	0.001251

Continued on next page

scnum	scname	frac	Kc	nchannels	GVis_per_sec	PFLOPs_per_sec
49	Proper motions of 50 galaxies in Virgo Cluster	0.9240	1.000000	166	0.019847	0.006707
51	Localize 10 LIGO events (2 looks each)	2.0600	1544.373458	10323	1.899845	991.480957
52	Proper motions of 3 LIGO events (2 looks each)	1.0000	1.000000	2222	0.002066	0.000698
53	Localize 10 LISA events (2 looks each)	3.6400	6.450366	2700	0.306634	0.668372
54	Find candidate BHs and pulsars in inner Galaxy	1.7000	56.332445	5037	0.572042	10.889312
55	Astrometry of 50 BH candidates	1.9300	1.000000	133	0.006119	0.002068
56	Parallax of a BH (5 looks)	0.2060	1.000000	1333	0.007801	0.002636
57	Orbital wobble of Cyg X-1 (20 looks)	0.0590	1.000000	771	0.001434	0.000485
58	SMBH masses in 12 nearby galaxies (CO)	0.5210	1.000000	522	0.048034	0.016232
60	Jet of EHT primary (2 looks)	0.6150	1.000000	571	0.006683	0.002258
61	Jet of EHT primary (2 looks)	0.6150	1.000000	571	0.006683	0.002258
62	2 primaries plus 4 others (52 looks each)	0.9390	71.379288	2700	0.124227	2.996416
63	Find ejected SMBHs	2.6000	1.381034	6704	0.097342	0.045427

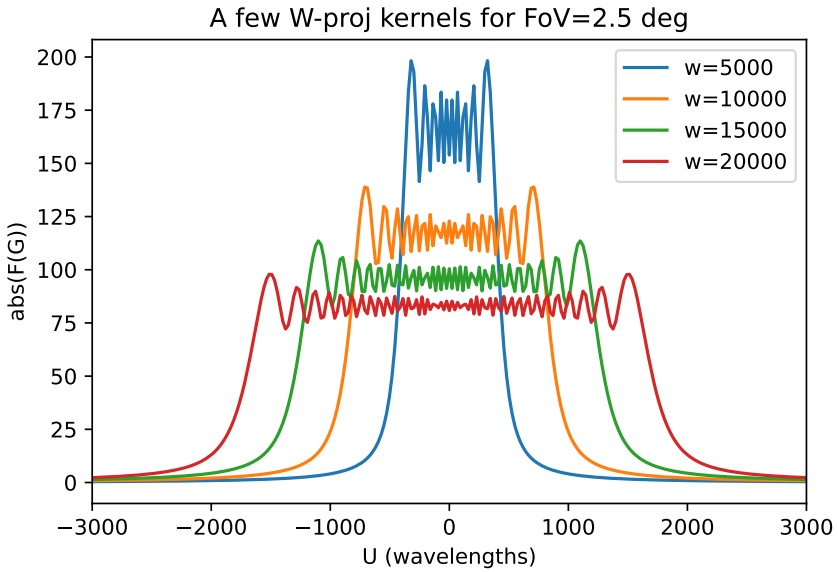

Figure 7: Example W-proj kernel functions for a FoV of 2.5 degrees.

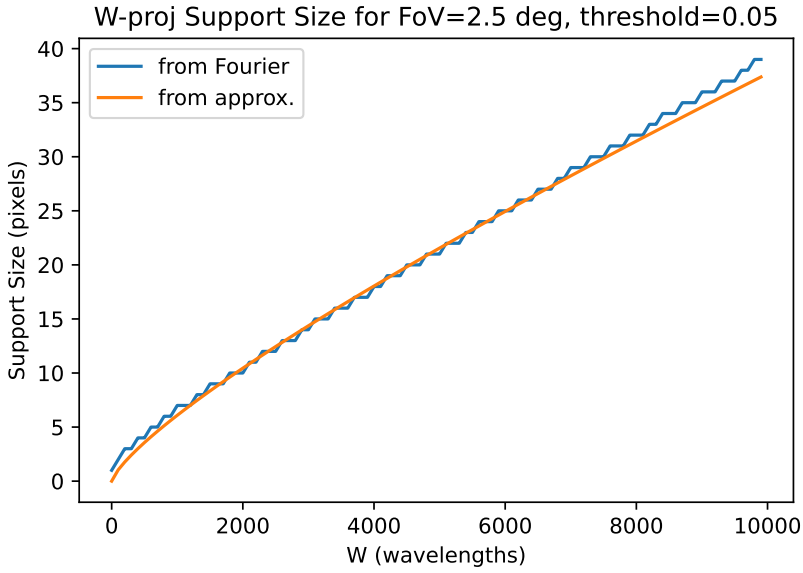

Figure 8: Comparison of support functions calculated numerically (blue line) and using the analytical approximation defined in [4] (orange line).

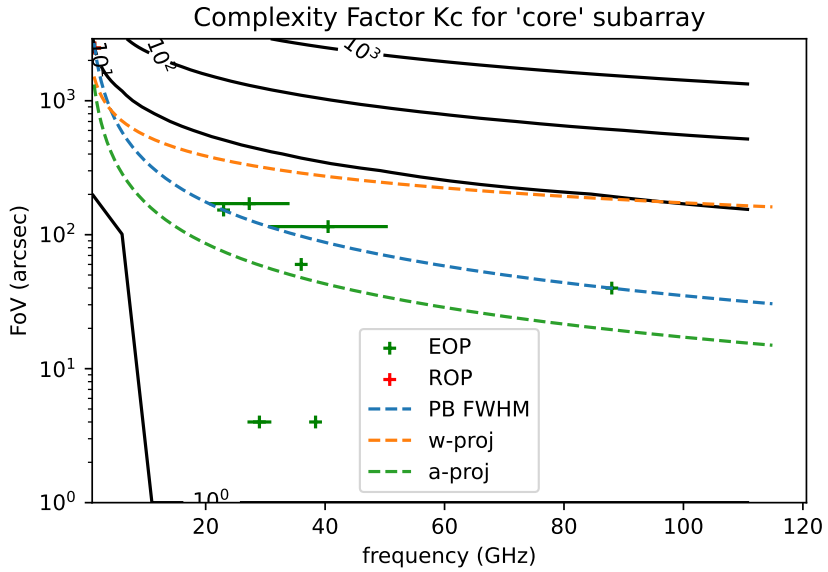

Figure 9: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the core subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

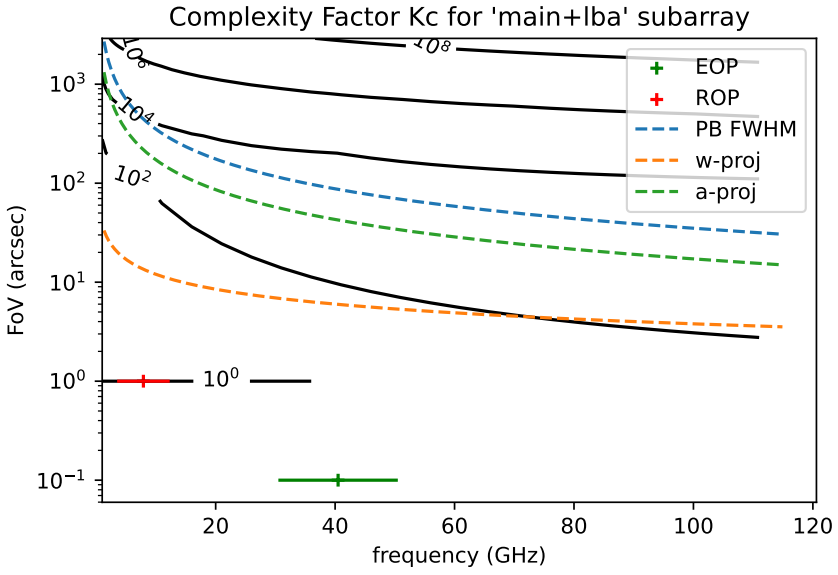

Figure 10: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the main + lba subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

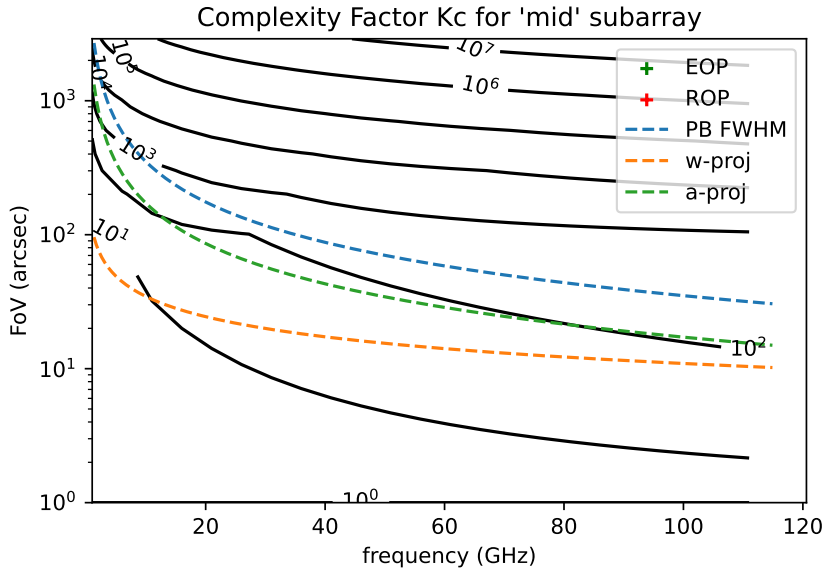

Figure 11: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the mid subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

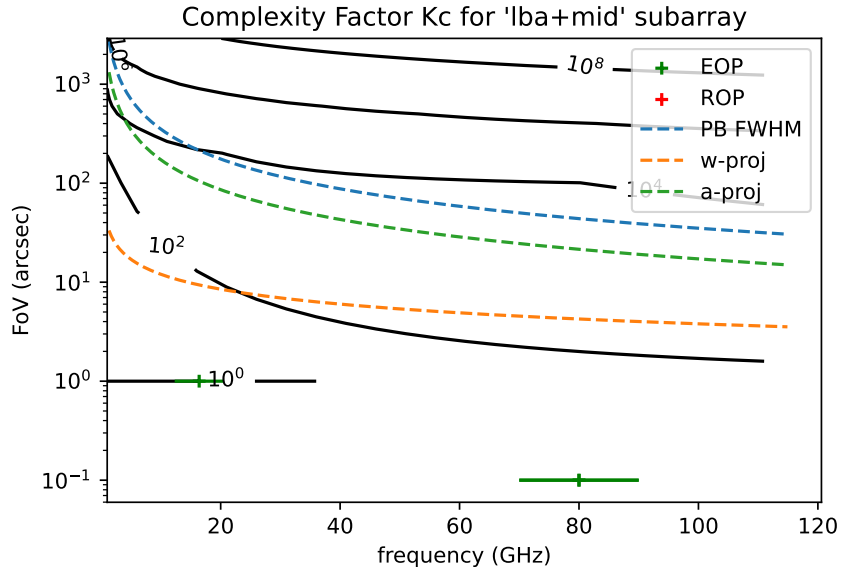

Figure 12: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the mid + lba subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

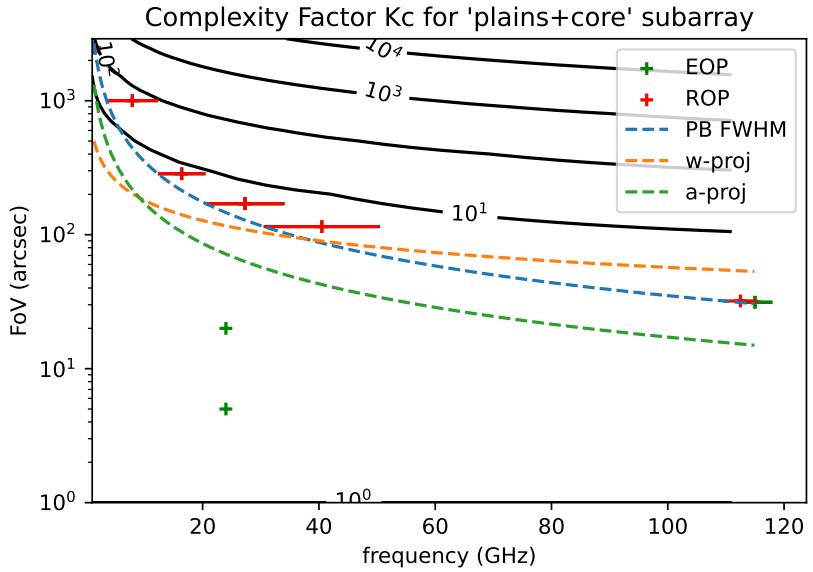

Figure 13: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the spiral + core subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.

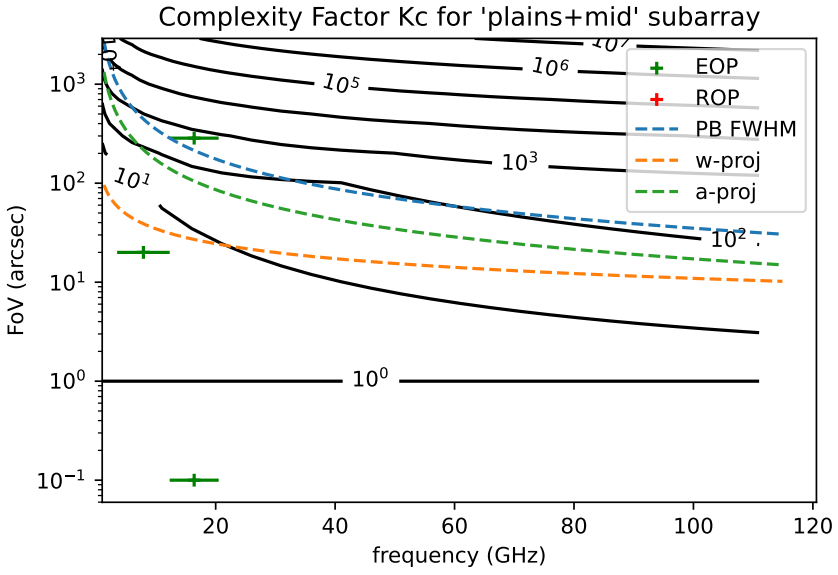

Figure 14: Estimation of the complexity factor K_c as a function of the field of view and frequency, for the spiral + lba subarray. Also shown are the science cases from the ROP and EOP, with the horizontal bar representing the bandwidth required by each case. The cases above the orange segmented line would require W-proj, and the ones above the green line would require A-proj.