MultiGPU-Based Visibility Gridding for Faceting

W. D. Cotton (NRAO), May 19, 2023

***Abstract*—GPUs have proven useful for the convolutional gridding used in making images from radio interferometer array data using the faceting technique to account for sky curvature. Since each facet is gridded independently but using the same data for all facets, it is straightforward to split this process across several GPU devices by allocating a set of facets to each device. An implementation of multiGPU gridding using faceting in Obit is described and examples are given of the application to a MeerKAT data set showing substantial performance gain. However, the scaling of gridding performance was somewhat less than linear with the number of GPUs. There was an additional per GPU overhead of about 20% of unknown cause.**

***Index Terms*—multiple GPUs, Interoferometric Synthesis**

I. INTRODUCTION

RADIO interferometric “visibilities” are samples of the spatial coherence function of an incoming wave-front at random locations in the aperture plane. It is desirable to convert these onto a regular grid to allow using FFT algorithms to transform to the image plane. This is usually done by a convolutional gridding of the randomly spaced samples onto a regular grid [1] [2]. Each visibility is convolved with a kernel with desirable properties and summed onto the regular grid. This operation is one of the dominant compute intensive portions of imaging, especially in the iterative deconvolution schemes such as CLEAN [3][4] .

Graphics Processing Units (GPUs) are widely available and relatively cheap and have potentially enormous computing power with typically, many thousands of cores. GPU based gridding is described in [5] as well as a general outline of the imaging process. Even with the implementation described in [5], gridding visibility data is a major contributor to run time of imaging. This memo extends the technique using multiple GPUs in the Obit package [6] [1]. Examples using data from the MeerKAT array are described.

II. FACETED IMAGING

Imaging interferometric data results in a flat map of the curved sky [7] [8]. Away from the location where the image intersects the sky, the image can become increasingly defocused. The curvature of the sky must be accommodated in the imaging process. Faceting using multiple segments of tangent planes to represent the curved sky is the technique used in Obit [9]. Faceting is ideal for parallel implementations as the same data, with slightly differing processing parameters, is used in all facets. Thus, facets are formed independently of each other.

In the implementation of GPU based gridding described in [5], multiple facets are gridded in parallel in the single GPU. Dividing the facets among multiple GPUs is largely a bookkeeping problem. The visibility data must be copied to all GPUs and the gridded data retrieved from the appropriate GPU.

III. MULTI-GPU GRIDDING IN OBIT

The principal broadband continuum imaging task in Obit is MFImage. MFImage will accept a list of GPUs to use in parallel and the imaging class will divide up the facets among the devices. Each GPU device is managed from its own thread to improve execution efficiency. The implementation of gridding is essentially that of [5] except that visibility is read in blocks of 8192 records that are passed to the GPU using 4 streams to overlap the transfer of data to the device and computation.

IV. TIMING EXAMPLES

A comparison of execution times was made for gridding using various numbers of GPUs for a modest MeerKAT L band dataset on a low surface brightness supernova remnant. A two resolution, multiresolution CLEAN was used to help recover the low surface brightness features. Testing used Obit task MFImage which allows control of the number of GPUs used in gridding. A single GPU was always used for the “degridding”. Except for the number of GPUs used in the gridding, the data, software and control parameters were the same in all tests.

A. Test Data

The test MeerKAT dataset has 591,016 visibilities with 8 spectral windows of 119 channels each. Imaging was done in Stokes I to a radius of 1.0° using 147 facets and 14 × 5% fractional bandwidth frequency bins. Since the test dataset was on a supernova remnant, multiresolution CLEAN was used with two resolutions and a total of 294 facets. CLEANing used up to 100,000 CLEAN components or a minimum peak residual of 80 μJy/beam and autoWindowing [10] to set the CLEAN window. An initial CLEAN window mask was also used.

The field of view contained a very extended, low surface brightness source which required many cycles to CLEAN. The run is dominated by the time taken for gridding.

B. Test Machines

Timing tests were run on two different hosts. One machine is cheeta at NRAO which has 72 (hyperthreaded) cores of Intel Xenon CPU E5-2695 v4 @ 2.1 GHz with 256 GByte of RAM, 150 GBytes of which were in a RAM disk for output image and scratch files. The other disk used was SSD. Cheeta has a 256 bit memory bus and supports AVX2. This machine

---

National Radio Astronomy Observatory, 520 Edgemont Rd., Charlottesville, VA, 22903 USA email: bcotton@nrao.edu

[1] http://www.cv.nrao.edu/∼bcotton/Obit.html

TABLE I
TIMING TESTS

Machine	no. GPU	1st min.	Run min.	no. image	Maj. Cyc.
cheeta	1	20.7	184.3	4530	26
cheeta	2	13.5	138.4	4830	27
com2	1	16.0	158.9	4924	26
com2	2	10.5	116.6	4940	28
com2	3	8.5	99.0	4786	26
com2	4	7.5	91.1	4766	25
com2	4 A	7.6	91.8	4523	26
com2	4 B	7.7	88.4	4382	26
com2	4 C	7.7	91.8	4493	26
com2	4 D	7.4	88.0	4354	26

Notes:
“no. GPU” is the number of GPUs used in the gridding, “A”...”D” show the results of repeats of the 4 GPU test.
“1st” is the wall clock time for the initial imaging of the 294 images+beams.
“Run” is the total wall clock time.
“images” is the total number of images made.
“Maj. Cyc.” is the number of major cycles used in the CLEAN.

has two GPUs, both NVIDIA GeForce GTX 1080 with 20 Multiprocessors with 128 cores each (2560 cores total) and a clock speed of 1508 kHz. The CUDA capability is 6.1 with 7 GByte of global memory.

The SARAO cluster machine com2.science.kat.ac.za has a CPU similar to cheeta also with 1 TByte of NVME disk but with 4 GPUs. Output image and scratch files used the NVME disk; the other disk was SSD. The GPUs are NVIDIA GeForce GTX 1081 TI devices with 28 Multiprocessors with 128 cores each (3584 cores total) and a clock speed of 1508 kHz. The CUDA capability is 6.1 with 10 GByte of global memory.

C. Timing Tests

Timing tests were run using different numbers of GPUs for gridding. A single GPU was used in the degridding. The test involves a seriously nonlinear component, CLEAN deconvolution, meaning the total amount of work was not constant as differences in the order of numeric operations will cause the numerical results to differ enough to take different paths to the final solution. This is especially true using GPUs for gridding as the order of execution of threads is not defined. The various test results are given in Table I. The resultant images are visually nearly indistinguishable from each other and from one derived in a test run using only CPU gridding when blinked against each other.

A simple comparison is the wall clock time to make the initial set of 294 images and beams, this is the same in all tests and can be determined from the log files. This is given in Table I as “1st”. This timing also includes the FFT and correction for the gridding convolution which is done in the CPU and will take the same time independent of the number of GPUs used in the gridding. The entire test process is dominated by the gridding.

The metric of interest for using multiple GPUs is the relative run times of the different tests. The runtime ratios are shown in Figure 1 for both the time needed to make the initial set of images and beams, and the total run time. The central portion of the field of view showing the supernova remnant is given in Figure 2.

V. DISCUSSION

The results of using multiple GPUs for gridding is most easily seen in Figure 1 where the execution times are shown to decrease significantly with increasing numbers of GPUs. Neither of the panels in this Figure reflect a pure measure of the gridding performance - which ideally should be close to linear with the number of GPUs - as they contain contributions from computations which are not being parallelized.

The “First imaging” plot is the simpler as it is just for gridding, FFTing and correction of the images for the effects of convolutional gridding (all scalar computations). In the cheeta tests, this part of the process is estimated to be 1.7 of the 20.7 or 13.5 min (1 or 2 GPUs). Applying this correction to the measured 1st Image timings gives a ratio of 0.62 rather than the 0.5 expected from a purely linear relationship. A similar test was done on com2; the measured time for the nongridding portion of the first imaging was 1.87 min. The ratio of the achieved runtime of gridding to that of a linear scaling for the 2, 3 and 4 GPU tests were 1.22, 1.42 and 1.60. There appear to be per GPU additional overheads of about 20% not hidden by handling each device in a separate thread. Monitoring the GPUs using the utility nvidia-smi showed that all GPUs included were close to 100% utilized most of the time they were in use; the individual efficiency does not appear to drop significantly when multiple GPUs are used.

Gridding the visibilities is still a substantial component of the total run time in these tests. Using a second GPU reduced the total run time by about a quarter on both test machines. Using 4 GPUs reduced the run time by almost half; however, Figure 1 shows diminishing returns as the scalar components become more dominant (Amdahl’s Law). The actual performance enhancement in a given run will depend on the fraction of the total work in the gridding which depends on the details of the data and target field.

CLEAN is a seriously nonlinear process and changing the order of computations can change the results enough to affect the path of the processing. In GPU processing, the computation is divided up onto thousands of threads but the order of thread execution is not defined. Each cell in the visibility grid can be the sum of hundreds, or thousands of components each computed in a separate thread. Due to the finite precision (32 bit) the result of the summation will differ slightly if the order of the numeric operations is changed.

The results shown in Table I show that even though the same data was processed by the same software on the same (or identical) processors, the details of the processing were not identical. This table gives the total number of facet (and beam) images formed and the number of CLEAN major cycles used. These show small but significant variations among the various runs on the same machine. To emphasize this effect the last

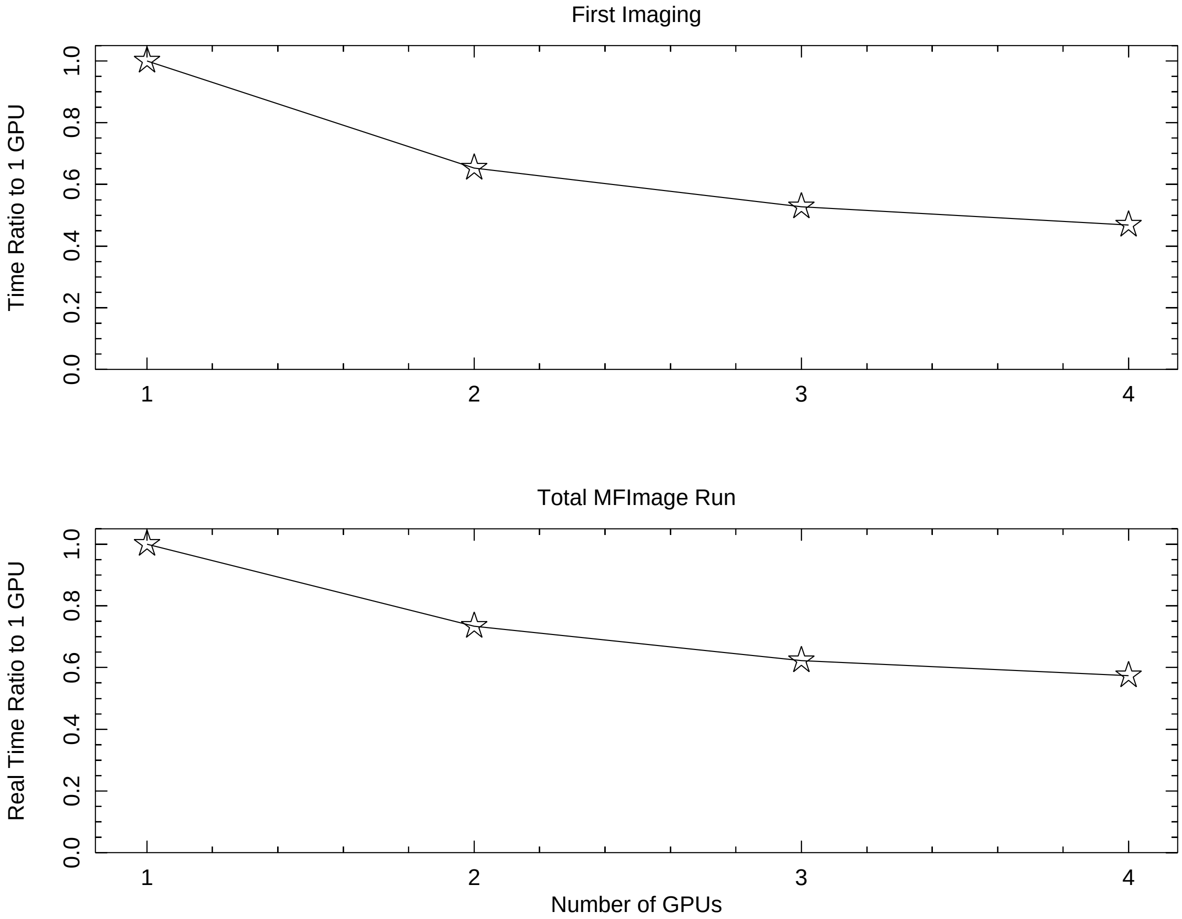

Fig. 1. Timings of MFImage on com2 relative to one GPU. The upper panel is the ratio of the run time for the initial imaging of all facets and beams for a variety of numbers of GPUs to one GPU. The lower panel is the ratio of total runtime to that for a single GPU.

5 entries in Table I show the results of 5 identical runs; the number of images formed and CLEAN major cycles illustrate this effect. The number of images formed was largely driven by differences in the decision of which resolution to use for each major cycle.

VI. ACKNOWLEDGMENT

I would like to thank the South African Radio Astronomy Observatory (SARAO) for use of their computing hardware and for the test dataset used.

REFERENCES

[1] F. R. Schwab, “Optimal Gridding of Visibility Data in Radio Interferometry,” in *Indirect Imaging. Measurement and Processing for Indirect Imaging. Australia, Cambridge University Press, Cambridge, England, New York, NY, L C # QB51.3.E43 I53 1984. ISBN # 0-521-26282-8. P.333, 1983*, 1983, pp. 333–+.

[2] ——, “Optimal Gridding of Visibility Data in Radio Interferometry,” in *Indirect Imaging. Measurement and Processing for Indirect Imaging*, J. A. Roberts, Ed., 1984, pp. 333–+.

[3] J. A. Högbom, “Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines,” *A&A Suppl.*, vol. 15, p. 417, Jun. 1974.

[4] B. G. Clark, “An efficient implementation of the algorithm ’CLEAN’,” *A&A*, vol. 89, pp. 377–+, Sep. 1980.

[5] W. D. Cotton, “GPU-Based Visibility Gridding for Faceting,” *Obit Development Memo Series*, vol. 73, pp. 1–4, 2022. [Online]. Available: https://www.cv.nrao.edu/∼bcotton/ObitDoc/GPUGridv2.pdf

[6] W. D. Cotton, “Obit: A Development Environment for Astronomical Algorithms,” *PASP*, vol. 120, pp. 439–448, 2008.

[7] R. A. Perley, “Imaging with Non-Coplanar Arrays,” in *Synthesis Imaging in Radio Astronomy II*, ser. Astronomical Society of the Pacific Conference Series, G. B. Taylor, C. L. Carilli, and R. A. Perley, Eds., vol. 180, 1999, pp. 383–+.

[8] ——, “Imaging with Non-Coplanar Arrays,” in *Synthesis Imaging in Radio Astronomy II*, ser. Astronomical Society of the Pacific Conference Series, G. B. Taylor, C. L. Carilli, and R. A. Perley, Eds., vol. 180, 1999, pp. 383–+.

[9] W. D. Cotton, “Multi-facet CLEANing in Obit,” *Obit Development Memo Series*, vol. 15, pp. 1–5, 2009. [Online]. Available: https://www.cv.nrao.edu/∼bcotton/ObitDoc/MultiClean.pdf

[10] ——, “Performance Enhancement of the autoWindow technique,” *Obit Development Memo Series*, vol. 9, pp. 1–3, 2009. [Online]. Available: https://www.cv.nrao.edu/∼bcotton/ObitDoc/autoWin2.pdf

Fig. 2. The supernova remnant, G16.2-2.7, dominating the field of view as imaged in the multiGPU tests. The “snakebite” on the eastern rim is where a pair of strong background sources were “Peeled” from the data.