



HAL
open science

Parallelisation of the Wide-Band Wide-Field Spectral Deconvolution Framework DDFacet on Distributed Memory HPC System

Nicolas Monnier, Erwan Raffin, Cyril Tasse, Jean-François Nezan, Oleg M Smirnov

► To cite this version:

Nicolas Monnier, Erwan Raffin, Cyril Tasse, Jean-François Nezan, Oleg M Smirnov. Parallelisation of the Wide-Band Wide-Field Spectral Deconvolution Framework DDFacet on Distributed Memory HPC System. ADASS, Nov 2020, Grenada, Spain. hal-02611036v3

HAL Id: hal-02611036

<https://hal.science/hal-02611036v3>

Submitted on 1 Dec 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Parallelization of the wide-band wide-field spectral deconvolution framework DDFacet on Distributed memory HPC system

Nicolas Monnier,^{1,2} Erwan Raffin,¹ Cyril Tasse,³ Jean-François Nezan,⁴ and Oleg Smirnov⁵

¹*Center for Excellence in Performance Programming, Atos, Rennes, France*

²*Univ Paris-Saclay, Centralesupelec, L2S, Gif sur Yvette, France*

³*GEPI, Observatoire de Paris, CNRS, Univ Paris Diderot, Meudon, France*

⁴*Univ Rennes, INSA Rennes, IETR, CNRS, Rennes, France*

⁵*SKA South Africa, Pinelands, South Africa*

Abstract. The next generation of radio telescopes, such as the Square Kilometer Array (SKA), will need to process an incredible amount of data in real-time. In addition, the sensitivity of SKA will require a new generation of calibration and imaging software to exploit its full potential. The wide-field direction-dependent spectral deconvolution framework, called DDFacet, has already been successfully used in several existing SKA pathfinders and precursors like MeerKAT and LOFAR. However, DDFacet has been developed and optimized for single node HPC systems. DDFacet is a good candidate for being integrated into the SKA computing pipeline and should, therefore, have the possibility to be run on a large multi-node HPC system for real-time performance. The objective of this work is to study the potential parallelism of DDFacet on multi-node HPC systems. This paper presents a new parallelization strategy based on frequency domains. Experimental results with a real data set from LOFAR show an optimal parallelization of the calculations in the frequency domain, allowing to generate a sky image more than four times faster. This paper analyses the results and draws perspectives for the SKA use case.

1. Introduction

The Square Kilometer Array (SKA) is the World's largest mega-science project, and also the World's largest Big Data project and the largest international computing collaboration. Thousands of dishes and the million antennas of the SKA equipment will generate unprecedented scale of data to compute in order to convert them into science for the astronomers. As an intermediate step towards SKA, LOFAR and MeerKAT are radiotelescopes already operational, delivering real data to process. Compared with previous telescopes, such instruments show that this new generation of radiotelescopes must take into account direction dependant signal distortions in the computing of data in order to benefit from the telescope sensitivity. A new generation of calibration and imaging algorithms are studied by astronomers and implemented in the DDFacet (Direction Dependent Facet) library. This paper presents the results obtained when distributing DDFacet on a multi-node HPC system. We show how the parallelization

of the wideband wide-field direction dependent spectral deconvolution algorithms of DDFacet has been implemented. The challenge is to distribute data and computations on the distributed memory architecture of a multi-node HPC system.

2. DDFacet: facet based imager

2.1. Wideband widefield direction-dependent spectral deconvolution framework

It is extremely challenging to synthesize high-resolution images with the new generation of radiotelescope like LOFAR. This latter operates at very low frequency, observing a wide field of view and combining short and long baselines. The estimation of the true sky, assuming the Jones terms constant, is done by the imager DDFacet (Direction Dependant Facet)(Tasse & al.). A specificity of DDFacet framework is to solve the imaging problems due to the DDE by using the faceting approach. The purpose of faceting is to approximate a wide field of view with many small narrow-field images. One independent grid is used per-facet and a constant DDE is applied to each facet. Experience says that we need to split the sky model into a few tens of direction to describe the spatial variation of the direction-dependent Jones matrices (Shimwell et al.). DDFacet also incorporates a few deconvolution algorithms that natively incorporate and compensate for the DDE. For the next of this paper, every mention of the deconvolution algorithm will reference the Hybrid Matching Pursuit (HPM) algorithm described in (Tasse & al.).

2.2. Multi-Core parallel processing and profiling

DDfacet is currently implemented as a functional software based on python and C. A naive description of the imaging framework is described by Fig 1a. In this example, we are working with a set of measured visibility V_ν covering the frequency band ν , where ν can be divided into several frequency channels ν_i . The imaging framework consists of calculating through an FFT and degrading the predicted visibilities for a specific frequency channel from the sky model $\widehat{\pi}_{\nu_i}$. Those are then subtracted from the measured visibilities V_{ν_i} producing residual visibilities. The residual visibilities are then gridded onto a 2D uv-plan with a different plan for each channel. All these plans are then transformed into the time domain through an FFT^{-1} , forming a residual image $\widehat{\delta y}$. The deconvolution aims to reconstruct the sky iteratively (minor cycle) from the residual image. After this step, an updated sky model, $\widehat{\pi}_{+1}$, is generated. This entire process is repeated in a Major Cycle until the global stopping criteria is reached.

The framework has a native singlenode multiprocess, based on DDFacet's faceting image processing, except for the deconvolution part where all the facets are merged. Each facet φ can be processed independently of the others. As shown in Fig. 1, the different facets are computed in parallel on the available processor cores. We performed a profiling of DDFacet to study the distribution of the total execution time. This profiling has been done using 24 Measurement Sets (or MS file) from the LOFAR survey. Each MS file contains Visibilities and metadata for a specific frequency channel ν_i . The size of the sky image generated was 10.000 x 10.000 pixels. The framework DDFacet has been run on an Atos Bull dual-socket compute node equipped with Intel Skylake processors (Intel Xeon Gold 6130) and 192GB RAM DDR4 memory at 3200 MT/s. A total of 64 cores (1 process per core) could run in parallel on the node. The profile obtained in such conditions is given in figure 1b. This profile shows that the gridding and

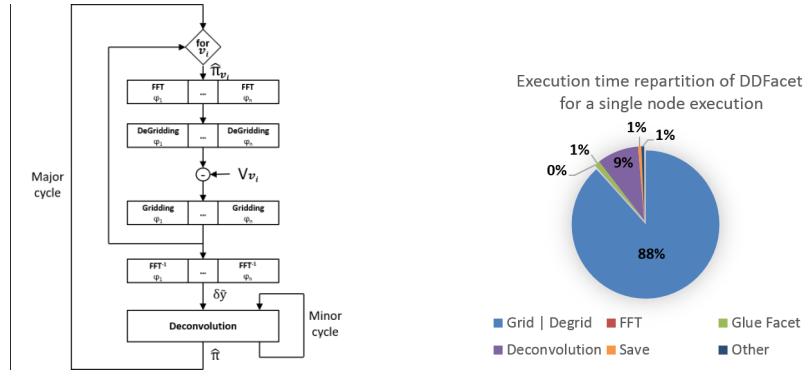


Figure 1. *a*-DDFacet single node representation. *b*-Distribution of the total execution time of DDFacet running on a single node.

degridding are the most compute-intensive part of the execution, even with the efficient intra-node multicore parallelization (Tasse & al.).

3. Parallelization on a distributed memory system

We decided to use the natural frequency channel-independent computation as the base of our distributed memory parallelization scheme. Indeed, since visibilities are stored in specific MS files for specific frequency channels, MS files are computed independently. Fig.2 represents the multi-node strategy parallelization. This figure shows an example where DDFacet is parallelized on k nodes, and visibilities for the frequency band ν are subdivided into N_ν frequency channels. Each node computes its specific visibilities, from the frequency channels ν_i , where $\nu_i \subset \nu$, ν_i may be composed by one or several frequency channels, to generate a sub-residual image $\delta\hat{y}_{1..k}$ specific to the node. These sub-residual images are then collected and summed by a single node to create a single residual image $\delta\hat{y}$. This residual image is identical to the one generated by the single-node version of DDFacet. The node is also in charge of the deconvolution step to update the sky model. The model is then broadcasted to all the nodes. It is important to note that the "per facet" intra-node parallelization is still the same. The software implementation of this level of parallelism has been done using MPI through mpi4py on python (Gonzalez 2019).

4. Experimental results

The experiments have been performed on the same type of nodes as for the profiling presented in subsection 2.2 with dual-socket Intel Xeon Gold 6130. The nodes were interconnected using a Mellanox EDR interconnect. The performance tests have been done on a number of nodes multiple of the number of MS files used, which corresponds in our case to 2, 3, 4, 6, 8, 12, and 24 nodes, representing a total of 1536 logical cores. The profiling of one node depending on the number of nodes used for the experiment is shown by 3a. This result shows that gridding/degridding steps are efficiently distributed over nodes since on 24 nodes. This result is confirmed observing the speedup of only the gridding/degridding over nodes in figure 3b.

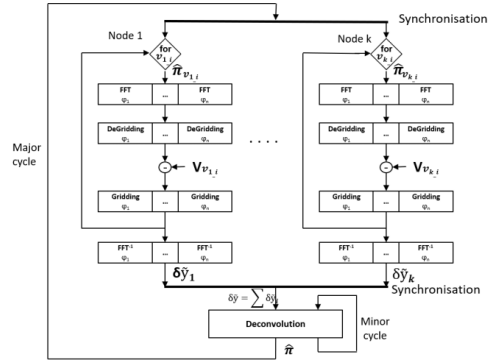


Figure 2. Multi-node parallelization on independent frequency channels.

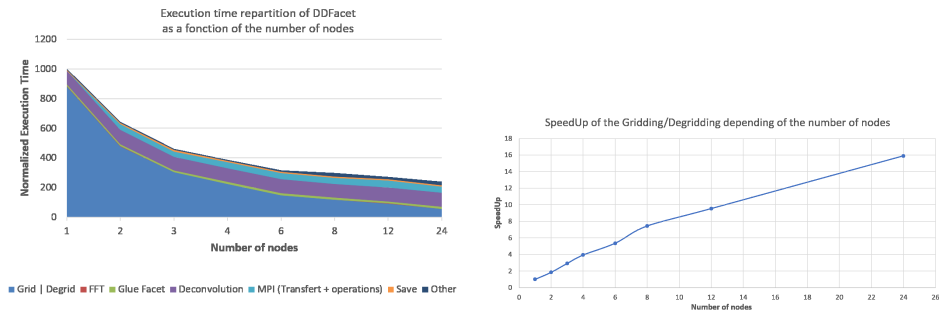


Figure 3. *a*- Profiling of DDFacet depending on the number of nodes. *b*- Gridding/DeGridding speedup of the distributed version of DDFacet.

5. Conclusion and Future Work

The results of our MPI parallelization over several nodes on the existing shared memory parallel implementation with a real data set from LOFAR have been analyzed. The hybrid parallelization has speedup the image generation of the radio sky up to a factor of 4.2 for the whole program and up to 16 for the Gridding/DeGridding kernel on 24 compute nodes. The acceleration of the whole program is limited by the size of the data set used for this study, which won't be the case with the large data sets of SKA. We showed in this paper that the parallelization potential of DDFacet on multi-node HPC systems is very high. This potential is a crucial feature for the selection of DDFacet in the SKA final computing system. Futur work will aim to distribute intensive kernels on co-processors like GPUs.

References

- Gonzalez, J. e. a. 2019, *Astronomical Data Analysis Software and Systems*, 321, 515
 Shimwell, T. W., Tasse, C., & al.622, *A1. 1811.07926*, URL <http://arxiv.org/abs/1811.07926>
 Tasse, C., & al.611, *A87*. URL <https://www.aanda.org/10.1051/0004-6361/201731474>