

Calibration and Imaging Pipeline Processing Baseline-dependent Averaged Visibilities

Chiara Salvoni^{*(1)}, Rob R. van den Bergh⁽¹⁾, Tammo Jan Dijkema⁽²⁾, Jakob Maljaars⁽³⁾, Maik Nijhuis⁽⁴⁾, André R. Offringa⁽²⁾⁽⁵⁾, Sebastiaan van der Tol⁽²⁾, Mark de Wever⁽³⁾, Stefan J. Wijnholds⁽²⁾

(1) CGI, Rotterdam, The Netherlands

(2) ASTRON, Dwingeloo, The Netherlands

(3) Science & Technology, Delft, The Netherlands

(4) TriOpSys, Utrecht, The Netherlands

(5) Kapteyn Astronomical Institute, Groningen, The Netherlands

Abstract

In this paper, we present a calibration and imaging pipeline whose processing functions directly operate on visibilities to which Baseline-Dependent Averaging (BDA) has been applied. This avoids the need to expand BDA-ed visibilities to full resolution. We demonstrate that this reduces the computational needs for calibration (both visibility prediction and solving) and (de)gridding by processing fewer visibilities without causing unexpected degradation of the results. This first prototype of a BDA-enabled calibration and imaging pipeline allows further exploration of BDA-enabled processing, for which we raise several questions.

1 Introduction

The visibilities (crosscorrelated data) acquired by a radio interferometer need to be sampled at a sufficiently high temporal and spectral resolution to avoid unacceptable levels of time and bandwidth smearing [1] on the longest baselines of the interferometer array. This implies that the time and frequency resolution of the visibility data and, hence, the visibility data volume, are driven by the needs of the longest baseline. On shorter baselines, longer integration times per sample and wider frequency channels would be tolerable. Cotton [2] and Skipper [3] have therefore considered baseline-dependent averaging (BDA) in the context of the Very Large Array (VLA) [4] and the Square Kilometre Array (SKA) [5]. In a detailed assessment of the impact of BDA [6], it was concluded that BDA should not have a detrimental effect on calibration and imaging apart from an expected and small increase in coherence loss on the shorter baselines. Besides reducing the raw data volume, BDA can also be used to tune the spatial filtering during imaging [7, 8] and tools have been developed to apply such advanced BDA schemes [9].

BDA can potentially also reduce the computational burden of calibration and imaging by reducing the number of data points that these processing functions need to process. Unfortunately, most production-ready radio astronomical processing functions do not operate on BDA-ed visibilities di-

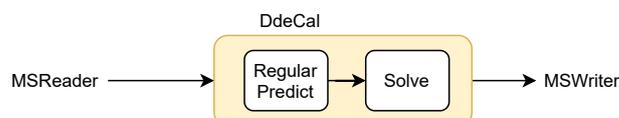


Figure 1. Standard workflow for DD calibration in DP3.

rectly. This is circumvented by expanding the BDA-ed visibilities first to regularly sampled visibilities at the highest resolution before the calibration and imaging functions process the visibility data. This approach preserves the benefits of reduced storage and I/O requirements, but does not reduce the compute load of calibration and imaging functions. In this contribution, we present a rudimentary calibration and imaging pipeline avoiding the visibility expansion step by processing the BDA-ed visibilities directly. This pipeline includes a direction-dependent (DD) calibration step in the Default Pre-Processing Pipeline (DP3)¹ and an imaging step in WSClean² [10]. We discuss the steps taken to create these BDA-enabled processing functions and show preliminary performance results highlighting the advantages of processing functions that support BDA natively.

2 BDA-enabled DD calibration

We used DP3 for DD calibration, for which it provides the DdeCal data processing step. The standard workflow for this processing step is shown in Fig. 1. In this workflow, the MSReader reads the visibility data from a stored Measurement Set (MS) along with the meta-data such as flags and (u, v) -coordinates associated with them. The (u, v) -coordinates are used to predict the visibilities based on an initial source model. DP3 can either predict visibilities by taking the fast Fourier transform of an image of the source model followed by a degridding step using Image Domain Gridding (IDG) [11] or by a direct Fourier transform of a specified list of discrete sources. For the experiments in this paper, we used (and adapted) the latter approach, because

¹<https://git.astron.nl/RD/DP3>

²<https://wsclean.readthedocs.io/>

the direct Fourier transform is usually faster for the source models with limited complexity typically used in DD calibration and it is easier to adapt to BDA. The predicted and observed visibilities are then fed into a Solver that estimates the direction-independent (DI) and DD gains. The DD gains are stored into an `h5parm` file that can be used to correct for them in the imaging process while the DI gains are used to correct the visibilities, which are then written to the output MS by the `MSWriter`. In the simulation described in Sec. 4, we used DI gains only and applied the appropriate corrections to the visibilities.

To turn this into a BDA-enabled workflow, we adapted the `MSReader` and `MSWriter` to read and write BDA-ed visibilities, enabled prediction of BDA-ed visibilities and adjusted the Solver to estimate gains based on visibilities with baseline-dependent integration times and channel widths. As mentioned in the introduction, BDA-ed visibilities can already be processed by functions operating on regular visibilities by expanding the BDA-ed visibilities. We therefore started following that route and adapted the `MSReader` and `MSWriter` to support reading and writing of BDA-ed visibilities. To store the meta-data needed to interpret the BDA-ed visibilities stored in the MS, we defined a few additional MS keywords. In addition, a `BdaBuffer` class was defined for in-memory storage of the BDA-ed visibilities after reading them from the MS.

The second step was to directly predict the BDA-ed visibilities. Initially, we expected that the fastest way would be to use the direct Fourier transform to predict the values at the (u, v) -coordinates of the BDA-ed visibilities. That approach, however, turned out to be slower than predicting visibilities at full resolution followed by BDA. This counter-intuitive result was caused by the fact that the geometric delays and DD gain of the primary beam need to be calculated per station first after which the two complex values associated with a specific baseline need to be multiplied. To make this work, we had to divide the baselines into groups with identical averaging intervals and evaluate the station-based factors for each of these baseline groups. This caused an increase in compute time spent on beam evaluations that exceeded the decrease in compute time spent on computing model visibilities. We therefore decided to predict the visibilities at full resolution followed by a BDA step. This has the additional advantage that the small increase in coherence loss on the shorter baselines is automatically taken into account in the visibility prediction.

The last and most challenging step is to adapt the calibration solve algorithm (`Solver`). The main challenge here is the organisation of the data. The visibility data and weights should also be ordered in such a way that the solver is not slowed down by memory accesses. These technicalities are handled by the `BdaSolverBuffer` class. The BDA-enabled workflow resulting from the steps above is sketched in Fig. 2.

A complicating factor in calibration is that different fac-

tors require different solution intervals. For example, ionospheric corruptions in LOFAR are typically solved for on time scales of a few seconds while solving for the band-pass, beam errors and Faraday rotation is performed with longer time intervals [12]. A future direction to look into is thus to expand and average the data at specific points in the pipeline to make maximal use of the computational and storage benefits of BDA, while still allowing short solution intervals.

3 BDA-enabled imaging

The top panel of Fig. 3 shows our calibration and imaging pipeline after the adjustments made to DP3. In this situation, we still have to expand the visibilities to full resolution before storing them in a MS suitable for WSClean. To avoid this step, we made similar changes to the MS I/O functions of WSClean as we did for DP3. The internal data structures of WSClean can quite naturally handle BDA-ed visibilities, because WSClean already supports visibility averaging prior to gridding to reduce the compute load of gridding. Supporting BDA input data therefore mainly consists of implementing a BDA MS reader and ensuring that the BDA-ed visibilities got properly organised into the data structures of WSClean. This BDA-enabled workflow is shown in the bottom panel of Fig. 3.

4 Results

We ran the BDA-enabled pipeline on a simulated data set derived from an actual MS of an observation with the Dutch LOFAR stations on the Boötes field. The visibility data in this MS were replaced by simulated visibilities, predicted based on a source model composed of a 9-by-9 rectangular grid of 1-Jy point sources spaced 0.5 degrees from each other. As the original MS was pre-averaged, the integration time per visibility was already 8 s. Since WSClean does not yet support BDA over frequency channels, we defined a BDA scheme integrating over time only with integration over integer multiples of the nominal 8 s integration with transitions to the next integration interval at baseline lengths of 20/1, 20/2, 20/3, ... km up to a maximum integration time of 96 s on the shortest baselines. This BDA scheme provided a reduction of the visibility data volume to 28.7% of the original volume, which already was only 12.5% of the raw data volume produced by the LOFAR correlator due to pre-averaging from 1 s to 8 s, i.e., with just BDA over time, already only 3.6% of the raw visibility data volume was left.

To assess calibration, we multiply all visibilities by 4 to emulate a direction-independent gain of 2. As the simulated data are noise-free and generated with the same DP3 visibility predict step, this gives perfect results (down to numerical accuracy) for the regular data. The relative average (over stations) calibration error for the BDA-ed data is shown in Fig. 4 as a function of solution interval for a few representative frequency channels. The errors are (typically

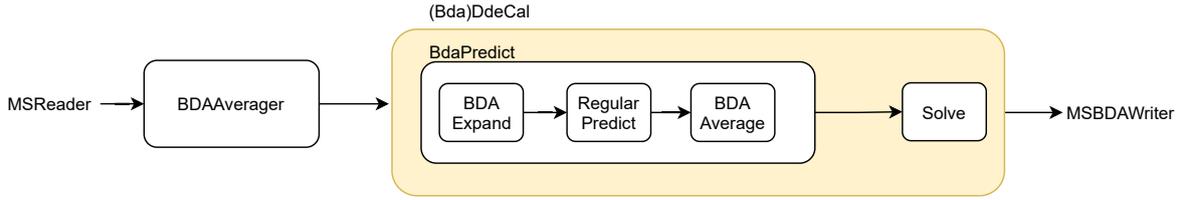


Figure 2. BDA-enabled workflow for DD calibration in DP3.

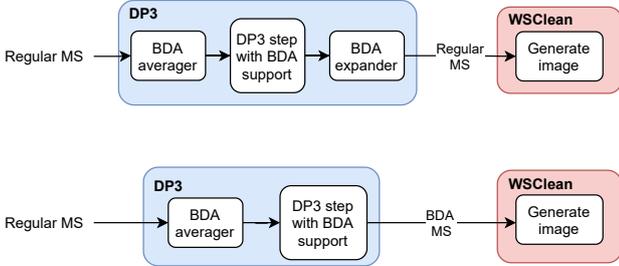


Figure 3. Calibration and imaging pipeline with BDA-enabled calibration step prior to (top) and after (bottom) enabling WSClean to read BDA-ed visibilities.

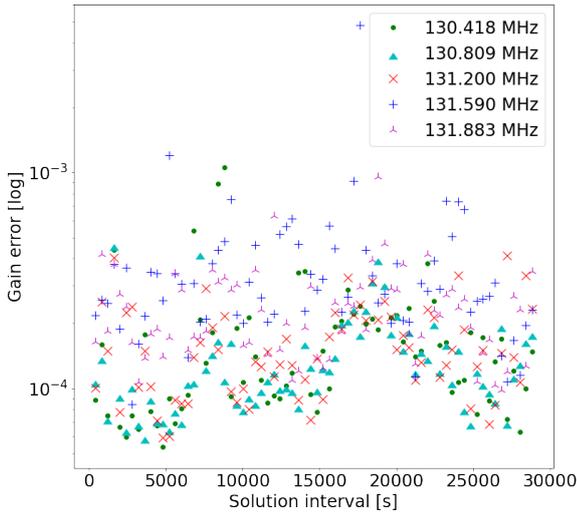


Figure 4. Average relative gain error as a function of solution interval with BDA. The colors of the dots represent results for different frequency channels.

well) below 0.5% as expected [6] with a few significant outliers at specific solution intervals. This indicates that the combination of averaging factors in the BDA scheme and length of the solution interval should be chosen carefully. Our choice causes a situation in which many averaging intervals crossed solution interval boundaries, which is likely suboptimal. This can be avoided by ensuring that all averaging intervals are integer divisors of the solution interval.

Fig. 5 shows the upper left quadrant of the images obtained from the regular data and the BDA-ed data. In the image reconstructed from the regular data, all sources appear as well-defined point sources of unit power as expected (our simulation predicted the 8-s averaged visibilities at their

step	regular	BDA	% reduction
DP3 - DdeCa1 (predict)	2:45	1:38	40%
DP3 - DdeCa1 (solve)	3:30	1:18	62%
WSClean - inversion	10:29	08:21	20%
WSClean - prediction	08:16	06:24	22%
WSClean - deconvolution	05:21	06:55	-29%
overall	30:21	24:36	18%

Table 1. Compute performance of the BDA-enabled workflow and the regular workflow.

mid-points without taking the small effect of decorrelation into account for the off-center sources). The image reconstructed from the BDA-ed data clearly shows the effect of increasing time smearing with increasing distance from the phase center. Fortunately, these artefacts only appear at a very low level as signified by the strongly zoomed in color scale needed to make them visible.

Table 1 shows the CPU time taken by the various processing steps for the pipeline working on the regular data as well as for the pipeline working on BDA-ed data. As expected, the DdeCa1 step in DP3 and the inversion and prediction steps in WSClean benefit from the smaller visibility data volume of the BDA-ed data set. The deconvolution step in WSClean, however, needs careful configuration to avoid an excessive number of minor cycles in an attempt to deconvolve "source structures" caused by time smearing at the few hundred μJy level. In this case, we found that 9 major cycles was a reasonable stopping criterion, balancing the growth of the number of minor cycles per major cycles against the depth of cleaning. If deeper imaging is needed, a less aggressive BDA scheme should be used.

5 Conclusions

In this paper, we present a BDA-enabled calibration and imaging pipeline with a DD calibration, imaging and deconvolution step that work directly with BDA-ed visibilities. We demonstrate the advantage of operating directly on BDA-ed visibilities in terms of computational resources needed for calibration, prediction, deconvolution and inversion. Our results also indicate that certain algorithm settings produce suboptimal results, either in terms of scientific quality or in terms of computational performance. With growing experience, we will be able to avoid such settings.



Figure 5. Image reconstructed using the standard workflow (top) and using the BDA-enabled workflow (bottom) based on the same data set. The linear grey scale runs from $-5 \cdot 10^{-5}$ to $5 \cdot 10^{-4}$ to make the imaging artefacts visible (the sources have unit power).

6 Acknowledgements

This work was conducted as part of the Square Kilometre Array (SKA) project. The authors acknowledge the resources provided by the SKA Observatory (SKAO) in producing this work. This work was also supported by the Netherlands Organisation for Scientific Research.

References

[1] A. H. Bridle and F. R. Schwab, “Bandwidth and Time-Average Smearing,” in *Synthesis Imaging in Radio Astronomy II*, ser. ASP Conference Series, G. B. Taylor,

C. L. Carilli, and R. A. Perley, Eds. Astronomical Society of the Pacific, 1999, vol. 180, ch. 18, pp. 371–382.

- [2] W. D. Cotton, “Effects of Baseline Dependent Time Averaging of UV Data,” National Radio Astronomy Observatory, Tech. Rep., Jun. 2009, Obbit Development Memo Series No. 13.
- [3] C. Skipper, “Time and channel averaging,” University of Southampton, Tech. Rep., 2014.
- [4] R. Perley *et al.*, “The Expanded Very Large Array,” *Proceedings of the IEEE*, vol. 97, no. 8, pp. 1448–1462, Aug. 2009.
- [5] P. E. Dewdney, P. J. Hall, R. T. Schilizzi, and T. J. L. W. Lazio, “The Square Kilometre Array,” *Proceedings of the IEEE*, vol. 97, no. 8, pp. 1482–1496, Aug. 2009.
- [6] S. J. Wijnholds, A. G. Willis, and S. Salvini, “Baseline-dependent averaging in radio interferometry,” *Monthly Notices of the Royal Astronomical Society*, vol. 476, no. 2, pp. 2029–2039, Feb. 2018.
- [7] M. T. Atemkeng, O. M. Smirnov, C. Tasse, G. Foster, and J. Jonas, “Using baseline-dependent window functions for data compression and field-of-interest shaping in radio interferometry,” *Monthly Notices of the Royal Astronomical Society*, vol. 462, no. 3, pp. 2542–2558, Nov. 2016.
- [8] M. Atemkeng *et al.*, “Baseline-dependent sampling and windowing for radio interferometry: data compression, field-of-interest shaping, and outer field suppression,” *Monthly Notices of the Royal Astronomical Society*, vol. 477, no. 4, pp. 4511–4523, Jul. 2018.
- [9] M. Atemkeng, S. Perkins, J. Kenyon, B. Hugo, and O. Smirnov, “Xova: Baseline-Dependent Time and Channel Averaging for Radio Interferometry,” in *Astronomical Data Analysis Software and Systems (ADASS)*, 8-12 Nov. 2020. [Online]. Available: <https://arxiv.org/abs/2101.11270>
- [10] A. R. Offringa *et al.*, “WSClean: an implementation of a fast, generic wide-field imager for radio astronomy,” *Monthly Notices of the Royal Astronomical Society*, vol. 444, no. 1, pp. 606–619, Oct. 2014.
- [11] S. van der Tol, V. B., and A. R. Offringa, “Image Domain Gridding: a fast method for convolutional resampling of visibilities,” *Astronomy & Astrophysics*, vol. 616, no. A27, Aug. 2018.
- [12] F. De Gasperin *et al.*, “Systematic effects in lofar data: A unified calibration strategy,” *Astronomy & Astrophysics*, vol. 622, no. A5, 2019.