Astronomy & Astrophysics

Compression of interferometric radio-astronomical data

A. R. Offringa

Netherlands Institute for Radio Astronomy (ASTRON), PO Box 2, 7990 AA Dwingeloo, The Netherlands
e-mail: offringa@astron.nl

ABSTRACT

Context. The volume of radio-astronomical data is a considerable burden in the processing and storing of radio observations that have high time and frequency resolutions and large bandwidths. For future telescopes such as the Square Kilometre Array (SKA), the data volume will be even larger.

Aims. Lossy compression of interferometric radio-astronomical data is considered to reduce the volume of visibility data and to speed up processing.

Methods. A new compression technique named "Dysco" is introduced that consists of two steps: a normalization step, in which grouped visibilities are normalized to have a similar distribution; and a quantization and encoding step, which rounds values to a given quantization scheme using a dithering scheme. Several non-linear quantization schemes are tested and combined with different methods for normalizing the data. Four data sets with observations from the LOFAR and MWA telescopes are processed with different processing strategies and different combinations of normalization and quantization. The effects of compression are measured in image plane.

Results. The noise added by the lossy compression technique acts similarly to normal system noise. The accuracy of Dysco is depending on the signal-to-noise ratio (S/N) of the data: noisy data can be compressed with a smaller loss of image quality. Data with typical correlator time and frequency resolutions can be compressed by a factor of 6.4 for LOFAR and 5.3 for MWA observations with less than 1% added system noise. An implementation of the compression technique is released that provides a Casacore storage manager and allows transparent encoding and decoding. Encoding and decoding is faster than the read/write speed of typical disks.

Conclusions. The technique can be used for LOFAR and MWA to reduce the archival space requirements for storing observed data. Data from SKA-low will likely be compressible by the same amount as LOFAR. The same technique can be used to compress data from other telescopes, but a different bit-rate might be required.

Key words. techniques: interferometric – methods: observational – methods: data analysis – radio continuum: general

1. Background

Data from interferometric radio observatories consists of correlated signals from all the pairs of antennae that are part of the interferometric array. These correlated signals are measured in spectral channels, and integrated over short timesteps before they are recorded to disk (Thompson et al. 2001).

Correlators in modern interferometric radio observatories output data at high spectral and temporal resolutions (Perley et al. 2011; Wilson et al. 2011; Tingay et al. 2013; van Haarlem et al. 2013). It is often not desirable to decrease the resolution before archiving, because resolution in both dimensions provides scientific value. High temporal resolution is for example desirable for variability studies (Stappers et al. 2011), advanced calibration methods that solve for the ionospheric disturbances and instrumental variabilities (Kazemi et al. 2013; Smirnov 2011; van Weeren et al. 2016) and to avoid time decorrelation. On the other hand, high frequency resolution is necessary for spectral-line studies (Favre et al. 2011; Morabito et al. 2014; Offringa et al. 2016), accurate removal of man-made interference (Offringa et al. 2013) and bandwidth decorrelation. The required time and frequency resolutions can lead to data volumes of several petabytes, which have to be supported by complex network architectures to allow acceptable transfer rates. While current interferometers consist of up to a few hundred of antenna elements, future instruments will have thousands of antenna elements that will produce data at even higher resolutions or with multiple beams, thereby increasing the data volume and associated storage costs by several orders of magnitudes (van Cappellen & Bakker 2010; Jiwani et al. 2013; Heywood et al. 2016). For these reasons, compression of correlated interferometric data is desirable.

In radio astronomy, linear quantization of the captured antenna voltages prior to correlation is common and well understood (Thompson et al. 2007). Further quantization of correlated data is however not common, mainly because it has not been explored what the effects of quantization are on the final science products, such as images or spectra. Since many interferometric projects need to reach large dynamic ranges, compression techniques that would limit the dynamic range are undesirable.

In this paper I consider compressing correlator output samples that consist of complex values. These data commonly have a low signal-to-noise ratio (S/N), especially at high resolutions (Thompson et al. 2001), and noise-like signals are inherently hard to compress without loss of information (Shannon & Weaver 1949). In radio data, each complex component is normally stored as a 32-bit IEEE 754 single-precision floating-point value (see e.g. "FITS Interferometry Data Interchange Format", AIPS memo #114, Sect. 4.1; and "MeasurementSet definition version 2.0", CASA Memo #229). Loss-less

compression is only effective for the 8-bit exponents of the floating point values, because the mantissa and sign follow closely a uniform distribution and are incompressible without loss. Therefore, loss-less compression can only reduce the volume of visibilities to approximately 75% of the original size. For these reasons, this paper will explore compression with a lossy encoding.

There are a few existing implementations of compression that are related to compressing visibilities. Firstly, AIPS can write visibilities as 16-bit values with uniform quantization. Secondly, the Flexible Image Transport System (FITS) file format (Wells et al. 1981) allows Rice, GZIP, HCompress, and PLIO compression[1]. Thirdly, for compression of integer data recorded with the CHIME telescope, Masui et al. (2015) showed that those data can be compressed to 28% of its original size using linear quantization and applying the BITSHUFFLE technique before LZ77 encoding (Ziv & Lempel 1977). FITS compression was designed particularly with image compression in mind, and is not designed for visibility data. FITS compression uses exponential quantization (the mantissa is rounded). It supports subtractive dithering (Pence et al. 2010). AIPS and BITSHUFFLE do not use dithering.

In this work, I describe a new compression method that is particularly designed for noise-dominated visibilities. It makes use of the Gaussian distribution of noise. I will test the effects of this compression method and will determine the maximal compression factor for this technique given a maximal increase in the system noise. In Sect. 2, I will describe the new technique. Section 3 presents the implementation. The result of this technique are shown in Sect. 4 using four test sets. The results are discussed in Sect. 5 and suggestions for future work are given in Sect. 6. Final conclusions are presented in Sect. 7.

2. Methods

The new method consists of two steps: a normalization step, in which grouped visibilities are normalized to have a similar distribution; and a quantization and encoding step, which rounds values to a given quantization scheme using a dithering scheme. The new compression technique consisting of the combination of these two steps is named "Dynamical Statistical Compression" (Dysco). In the next subsections, we will describe the two steps in detail.

2.1. Normalization

In observations, the variance of the data can be different for visibilities of different times, antennae, polarizations and frequencies. To encode visibilities accurately, this variation in variance should be normalized so that the visibilities follow the same distribution. The AIPS uvfits format allows compression of visibilities, and in this format visibilities are normalized by row: each row of visibilities is scaled by a single scaling factor to match the data to the quantization. A row contains an array of visibilities with different frequencies and polarizations, but of the same timestep and antennas (Greisen 2016). This normalization method will be referred to as "row normalization". The downside of this method is that visibilities from different frequencies and polarizations are assumed to follow the same noise distribution. This can be an issue when RFI has not been flagged before encoding, as well as when the bandpass has a large dynamic range

---

[1] See https://heasarc.gsfc.nasa.gov/docs/software/fitsio/compression.html

or when the statistics of the differently-polarized visibilities differ strongly.

Two additional ways to normalize the visibilities are tested. For the second method, each visibility is normalized by three terms: f_{ch}, a channel scaling term, and f_a and f_b, a scaling term for the antennae that were correlated. Hence, for a given visibility, the corresponding normalization factor σ can be calculated with $\sigma = f_{ch} f_a f_b$. Different timesteps and polarizations are encoded independently, so will have different normalization factors, while the real and imaginary values of a visibility are assumed to follow the same distribution. Consequently, for the visibilities corresponding to a given timestep and polarization, $N_{ch} + N_{ant}$ factors are stored, as well as $2 N_{ch} N_{ant} (N_{ant} + 1)/2$ quantized values. Such a group of visibilities will be referred to as a timeblock, and this way of normalization as antenna-frequency (AF) normalization. For each timeblock, the channel and antenna factors are stored separately as 32-bit floating point values.

AF-normalization does not work well on auto-correlations, because these have different statistics, and its standard deviation can generally not be described by the multiplication of two antenna factors. Auto-correlations of astronomical interferometers are seldomly used, and therefore in this work the auto-correlations are set to zero and are ignored during compression. If auto-correlations are to be saved, a separate scaling factor could be computed for auto-correlations. Our implementation does not support this at this point. The other normalization methods do preserve the auto-correlations.

The third normalization method, row-frequency normalization (RF-normalization), uses two terms: f_{ch}, a channel scaling term and f_{row}, a row scaling term. Each polarization also has its own channel and row factors. The final normalization factor is $\sigma = f_{ch} f_{row}$. This is somewhat similar to the row normalization method (as used in AIPS) but with some differences: an additional normalization term per frequency is added and independent per-row factors for different polarizations are stored, because the polarizations can have different distributions.

For each of these methods, the data is normalized such that the maximum normalized value is equal to the maximum quantizable value, that is, the maximum value can be represented exactly and never needs trimming. An alternative is to normalize the data such that a certain range (e.g. all data $<3\sigma$) fits inside the quantization, and all excess values are trimmed. Initially, for this work the latter method was tested as well. In the case of noise-dominated data, it was found that trimming does not cause a bias, but in case some part of the data has excess values due to real signal, it causes a bias. Hence, trimming data is not considered in this work. Normalization scales the data to fit within the quantization minimum and maximum values.

In typical situations, in particular for data outputted by the LOFAR or MWA correlators, the extra metadata (the scaling factors) stored by any of the normalization methods, is negligible.

The result of these normalization methods is demonstrated in Fig. 1 for an observation with strong RFI. The original spectrum is shown in the top-left. The other dynamic spectra show the difference between before and after compression using the three normalization techniques. With per-row normalization, when one of the channels contain large values, the noise in the other channels increases strongly. With AF and RF normalization, which have an extra channel normalization factor, the noise in the data without RFI is not significantly increased.

Figure 2 shows another example of the properties of normalization. These scatter plots show a single baseline of a LOFAR observation. This particular observation was averaged
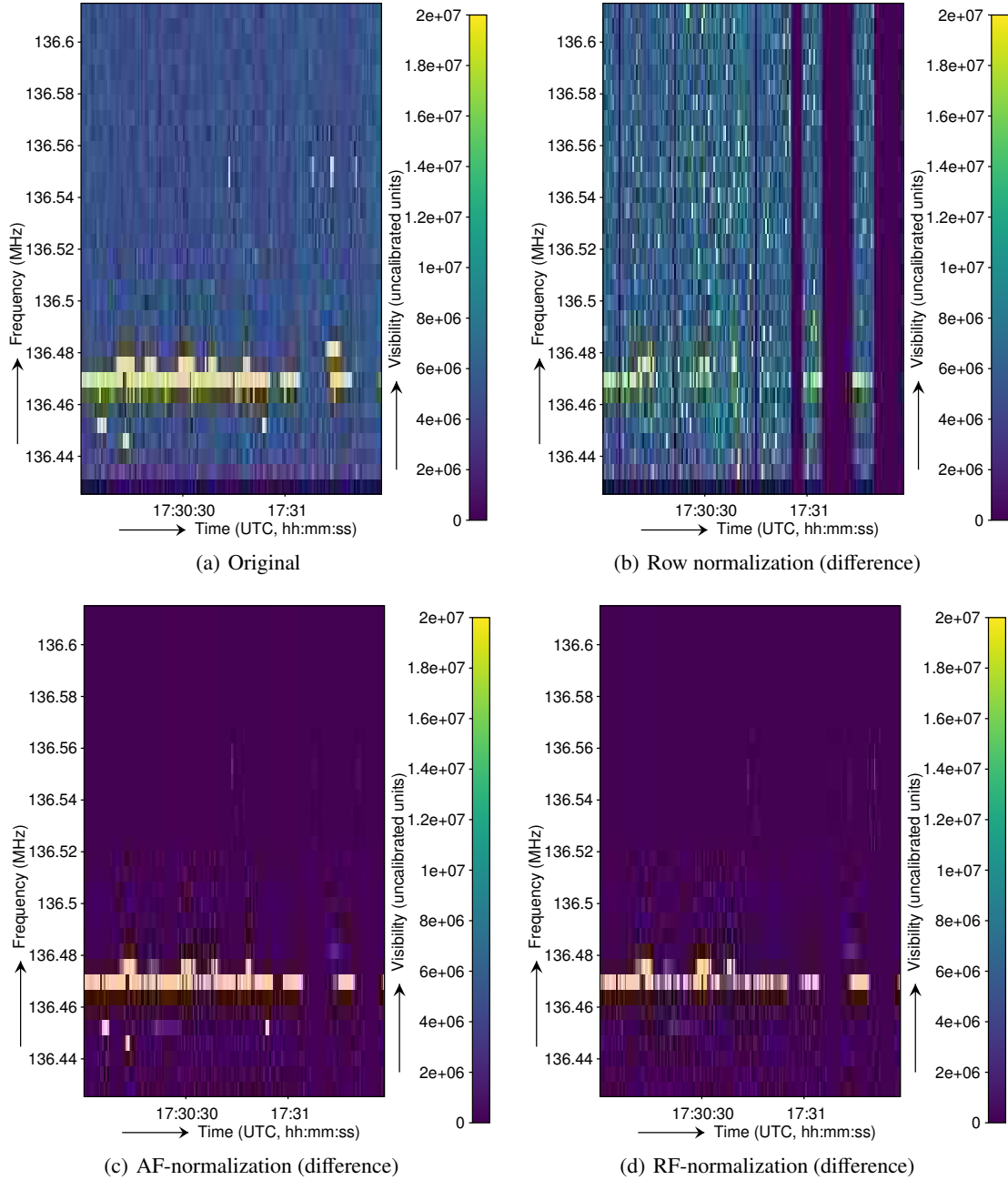
Fig. 1. Dynamic spectra of 1.5 min of a LOFAR observation at high frequency and time resolutions (6 kHz and 0.5 s) with strong RFI, compressed to 8 bits per float with different kinds of normalization. Compression with per-row normalization increases the noise significantly in all channels. The RF and AF normalizations, which normalize the channels individually before encoding, behave more robust in the presence of RFI.

to a resolution of 12 s and 200 kHz, and because of this it has a high S/N in the *XX* and *YY* polarizations. All three normalizations visibly add noise to the data, but in different ways. The assumption for AF-normalization is that the data is noise dominated. Because this is not the case in this example, the noise added to the *XX* and *YY* polarizations is larger than for the other normalizations. Row normalization does not store an independent scaling factor per polarization. Because of this, the accuracy of the *XY* and *YX* quantization strongly depends on the *XX* and *YY* signals. We note that the quantization is unbiased; while the quantization adds certain structures to the raw visibilities (most clearly visible in row normalization), it will be shown in later sections that the added imaging noise is unstructured and unbiased.

2.2. Quantization

After normalization, the visibilities are encoded with a nonlinear quantization scheme. The encoding is a-priory optimized for complex samples with a given distribution with zero mean. The distribution of correlated samples will approximate a normal distribution when the S/N is low and the observation does not contain interfering sources. Several encoding schemes are tested that are optimized for different distributions.

The quantization table is created by uniform sampling of the inverse of the corresponding cumulative distribution function. Dithering is applied to avoid possible systematic bias when the signal-to-noise ratio in the input data is high. Dithering is implemented as follows: for each float to be encoded, the two closest
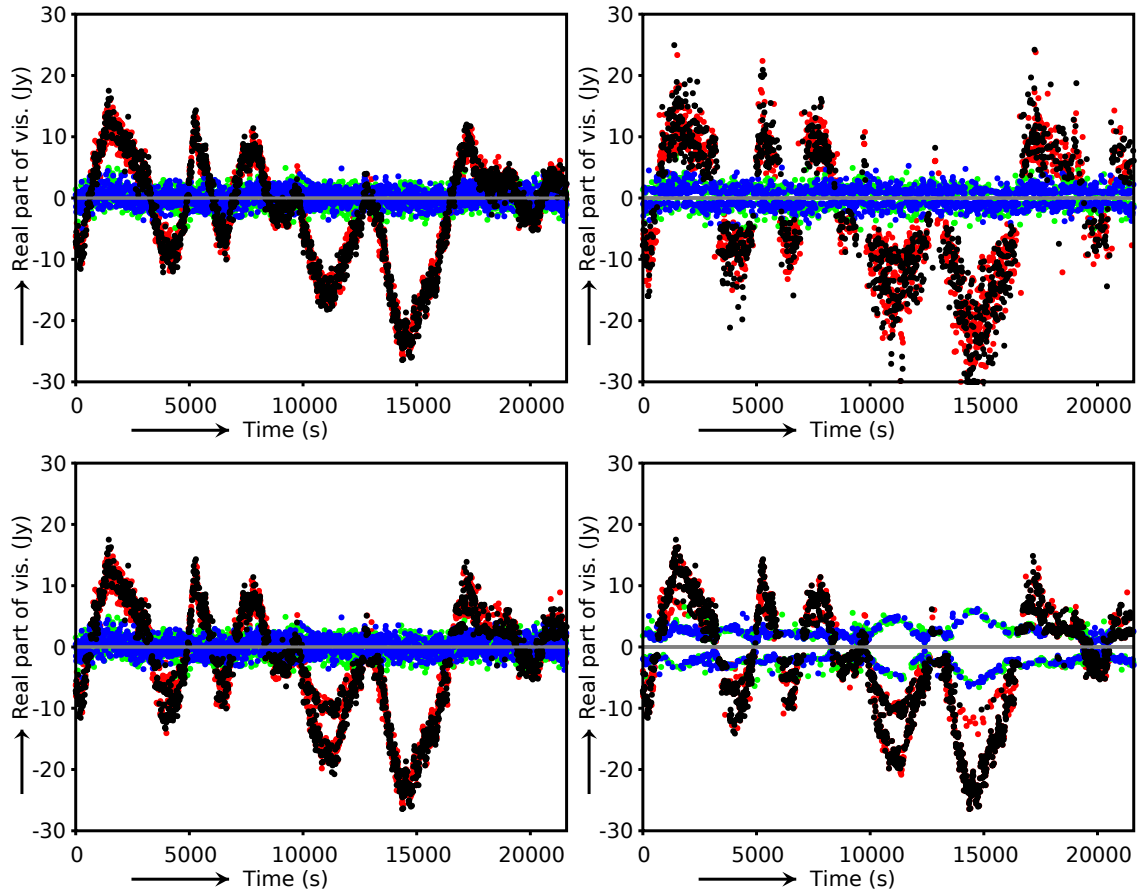
Fig. 2. Compression example of a single baseline of a LOFAR observation with high S/N. *Top-left plot*: original data; *top-right plot*: compression with AF-normalization; *bottom-left plot*: compression with RF-normalization; *bottom-right plot*: compression with row normalization. Compression was performed with the 2.5σ-truncated Gaussian 3-bit quantization. The red, green, blue and black dots represent the *XX*, *XY*, *YX* and *YY* correlations, respectively.

quantization values are found. The chosen encoding quantity is selected randomly, giving each a probability dependent on its distance to the original value. A quantized value will always be selected when it equals the original value, while if the original value is halfway between the two quantized values, both quantities are equal likely to be selected. This is equivalent to adding uniformly-distributed random noise to the input value, which is a common method of applying dithering, except that the bounds of the distribution vary per sample, depending on the distance of the value to the quantization values. Dithering ensures that the average of encoding a value many times asymptotically approaches that value. An example of the quantization is shown in Fig. 3.

The quantized values are converted to binary values and bit-packed. The number of bits can be set by the user, and define the number of quantities per encoding symbol. One encoding quantity is reserved for storing the special value "not-a-number" (NaN). An odd number of quantities remains for encoding finite values. Since the quantized values are symmetric around zero, the value 0 can be encoded precisely. This is beneficial, because it implies that no noise will be added when encoding the value zero. This way, an antenna that has produced zeros can be more easily identified. As an example, when the user asks for 8 bit encoding, a visibility is quantized to 256 values: values <127 correspond with negative visibilities; 127 corresponds with zero; values >127 and <255 correspond with positive visibilities; and 255 corresponds with the value NaN.

3. Implementation

The Casacore table system that is used to access CASA measurement sets uses so-called "storage managers" to access visibilities on disk (van Diepen 2015). The storage manager system is transparent to the client program, and storage managers can be located in an external software library. Any client program that uses Casacore (including CASA) can access measurement sets that were stored with a custom storage manager, without any changes to or recompilation of the client software. For this project, such a Casacore storage manager was designed to allow the transparent compression of data in a Casacore measurement set with the Dysco technique. The source code of the Dysco implementation is publicly released[2].

In my implementation, the compressed data is stored in an extra file inside the measurement set. Inside the file, the time-blocks are stored consecutively. Given that a timeblock has a fixed number of visibilities, as well as that the encoded representation has a fixed size, the position of a timeblock inside the file can be directly calculated. This has implementational advantages, such as not having to maintain an index table, as well as being able to replace a timeblock without the possibility that the new timeblock will not fit.

---

[2] The source code and manual of the implementation can be found at https://github.com/aroffringa/dysco. Commit 393cd97758 (8 Aug. 2016) was used.
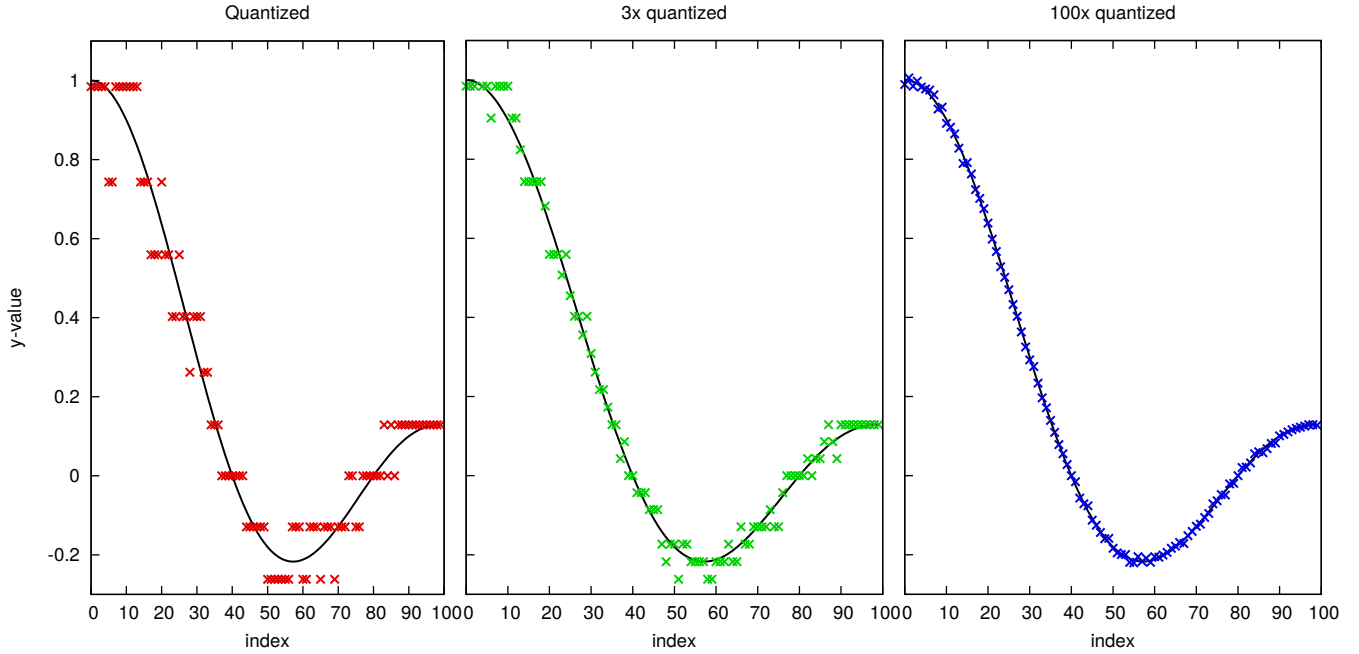
Fig. 3. A quantization example using the Gaussian-optimized least-squares quantization scheme with dithering to quantize a sinc function. 4-bit quantization and a single scaling factor were used. *Left plot*: result of encoding and decoding. Because the quantization is optimized for Gaussian distributed values, the quantization steps are smaller near zero. *Central and right plot*: average of 3 and 100 times encoding and decoding respectively.

Table 1. Sets used for testing the compression.

Set	Observation	Id	Date and size	Δt	$\Delta \nu$	Description, processing
A	LOFAR HBA 3C 196, 6 h, 117 MHz	L431602	2016-02-28 6.1 GB	4 s	36 kHz (5 ch)	Single subband, calibrated after compression.
B	LOFAR HBA 3C 196, 10 min, 133 MHz	L323164	2015-04-01 6.0 GB	0.5 s	6 kHz (32 ch)	Single subband, calibrated before compression.
C	MWA Vela / Puppis A, 2 min, 149 MHz	1052736496	2013-05-16 2.3 GB	4 s	80 kHz (384 ch)	Calibrated before compression.
D	MWA Hydra A, 2 min, 154 MHz	1077974936	2014-03-04 65 GB	0.5s	40 kHz (768 ch)	Calibrated after compression, recompressed after applying solutions.

Because each timeblock is compressed independently in each of the normalization and quantization methods, random access and streaming access of the timeblocks is possible. When a timeblock is requested or written, the implementation will keep that timeblock in the cache until another timeblock is written or read. This avoids encoding or decoding the timeblock when the client accesses different visibilities (table rows) from the same block multiple times. Once a different timeblock is requested, the cached timeblock is queued for writing if it has been changed. This system implies that accessing the visibilities in time-major order requires encoding or decoding each time block only once. This access pattern is the most common way of accessing a measurement set – calibration and imaging access the data in this way. One exception to this is flagging (Offringa et al. 2013), for which a reordering of the data is performed. Random access to measurement sets will lead to encoding or decoding the same block multiple times, but in practice this access pattern is rare.

AF-normalization is implemented as follows: first, the channel standard deviations are divided out to make each channel have a standard deviation of unity. Then, the antenna factors are found by solving $\Sigma = f_{ant} \otimes f_{ant}$, where $\otimes$ is the outer product,

Σ is a matrix with the measured variance of a baseline at each element after channel normalization, and f_{ant} is a vector with the antenna factors to be solved for. After dividing out the antenna factors, such that the variance of each baseline is approximately unity, the algorithm continues by iteratively selecting and maximizing the channel factor or antenna factor that increases the sum of the absolute values the most when maximized. This is continued until the gain is smaller than some threshold.

In the RF-normalization algorithm, the standard deviations of the channels are also first normalized, but afterwards each row is directly maximized. After that, each channel is maximized.

The encoding of timeblocks is multi-threaded. Typical nodes can encode and write visibilities faster than writing the uncompressed visibilities.

4. Results

In this section, Dysco is tested using several real observations. Table 1 lists the sets that have been used for testing. LOFAR HBA and MWA observations were used with different time and frequency resolutions. Because the compression method adds

noise that is relative to the variance of the data, sets with different signal-to-noise ratios were used for testing. For both telescopes, a high-resolution set was used, which is the typically resolution for archiving, and a set at lower resolution was used that is the typical resolution used during processing.

Images were produced for each of the sets and for various compression configurations. Reference images were made by skipping the compression step. Typical imaging parameters for the field of interest were used. Two quantities were measured: the rms of the difference between the reference and the dirty images of the compressed set; and the absolute rms levels of the images after cleaning. The differential rms was measured over the entire image. The absolute rms was measured in a rectangular part of the image in which no sources were visible. Each test set was tested with a different processing strategy, in order to test the effect of compression in combination with different processing steps.

LOFAR test set A of the 3C 196 field was RFI flagged and compressed with one of the described methods. Subsequently, calibration and subtraction of 3C 196 was performed on compressed data. Calibration was performed using the Mitchcal tool introduced by Offringa et al. (2016). The corrected visibilities were written without compression, so that the quantization noise was only added once. These uncompressed visibilities were then imaged. Because 3C 196 is a strong source at these frequencies (~100 Jy) while the imaging noise is only a few mJy, the set also tested compression of visibilities for imaging with high dynamic range.

Test set B is a LOFAR measurement set of the 3C 196 field with high time and frequency resolutions. It was RFI flagged before compression. After compression, it was averaged 8× in frequency and 16× in time, calibrated using the LOFAR calibration tool NDPPP (Dijkema et al., in prep.) and imaged.

Test set C is an MWA measurement set targeting the Vela and Puppis A supernova remnants. It was calibrated and averaged before compression. Calibration was performed by transferring solutions using CASA. The imaging was performed on the compressed visibilities. It has a relatively low time and frequency resolution, which is typical in MWA data processing to decrease the data volume. This set has a high S/N, in particular on the smaller baselines, because some of these see the Galactic plane.

Test set D is an MWA measurement set of calibrator Hydra A. Compression tests were performed on the original MWA time and frequency resolutions outputted by the MWA correlator. The compressed data was calibrated using Mitchcal, and the corrected visibilities were again stored with compression. This implies the compression noise was added twice.

All imaging was performed with WSCLEAN (Offringa et al. 2014), and all sets were imaged using uniform weighting. In uniform weighting, more weight is applied to the long baselines. Therefore, compression errors on the long baselines dominate the compression errors in image space. This is a realistic use-case for data from these telescopes.

4.1. Compression accuracy

Figure 4 shows the differential imaging error (reference dirty image minus dirty image using compression) as a function of the number of bits used in compression, for different normalization and quantization techniques. Continuous lines are normalized using AF-normalization, dashed lines using RF-normalization and dot-dashed lines using row normalization. The colour indicates the quantization distribution. The horizontal grey line

indicated the Stokes V noise, which is similar to the system noise. Hence, when the imaging error is well below this noise level, it will have an insignificant effect.

The Gaussian and 3.5σ truncated Gaussian quantizations produce relatively larger errors compared to other quantizations if more than approximately 8 bits are used in compression. An intuitive explanation for this is that at higher bit levels, these distributions reserve more quantization values for outlying values. The visibilities are scaled such that the maximum visibility value matches the maximum quantization value. Because the number of visibilities per time block are limited, visibilities rarely have outlying values, and therefore do not match the quantization distribution.

With the exception of the Gaussian and 3.5σ-truncated Gaussian quantizations schemes, methods that perform relatively well with few bits also perform relatively well with more bits, which allows one to generalize the results of one bit-rate to other bit rates. Table 2 shows the 8-bit compression imaging errors for the three normalization methods and for the four sets. Additionally, it shows the average over the normalized sets for each unique combination of normalization and quantization method, as well as the average over the normalization methods and quantization methods. In order to make each test set equally important in the averaging, normalization is performed by dividing the results of a particular test set by the average result of that test set, and multiplying it by the average error over all test sets. Confidence intervals are given at 2σ boundaries. Confidence intervals of average values are calculated by assigning each measurement a standard error equal to the standard deviation of the measurements that are averaged over, and these errors are propagated to the average value.

The results show that the combination of row-normalization with 1.5σ-truncated Gaussian quantization achieves the lowest error (153 ± 56 μJy). The combination with second-lowest error is row-normalization with uniformly distributed quantization (177 ± 91 μJy). This is the "AIPS compression" technique, but with 8-bits instead of 16-bits quantization, and with dithering. As was shown in Fig. 1, per-row normalization is not robust and performs badly in the case of RFI or when there are other reasons for large differences between channels. The combination with lowest error excluding row-normalization, is AF-normalization with a 2.5σ-Gaussian quantization distribution (184 ± 31 μJy).

When taking the average over the quantization methods, AF-normalization performs slightly but insignificantly better on average of the three normalization methods tested. When comparing the average error of the quantization methods by averaging over the normalization methods, the 1.5σ-truncated and 2.5σ-truncated Gaussian quantizations achieve the overall lowest errors. The Gaussian and 3.5σ-truncated Gaussian quantizations produce significantly larger errors than the other three methods.

In Fig. 4, for test sets A, B and C it is clear that the compression error decreases by a constant factor for each bit that is added. The fact that the decrease in error is larger at the very low bit-rates of 2–4 bits (which makes the error as a function of bit-rate slightly steeper in Fig. 4), is because one encoding quantity is reserved for the special value NaN. The error ratio of 8-bit compression over 16-bit compression is 258, 246 and 247 for set sets A, B and C respectively. Therefore, every added bit decreases the error by approximately a factor of 2. For test set D, the same 8-bit/16-bit ratio is only 5.8. This is likely because the compression error is not the dominating error in the imaging of this set. This set has more than ten times more visibilities than the other sets, and the large number of visibilities will increase
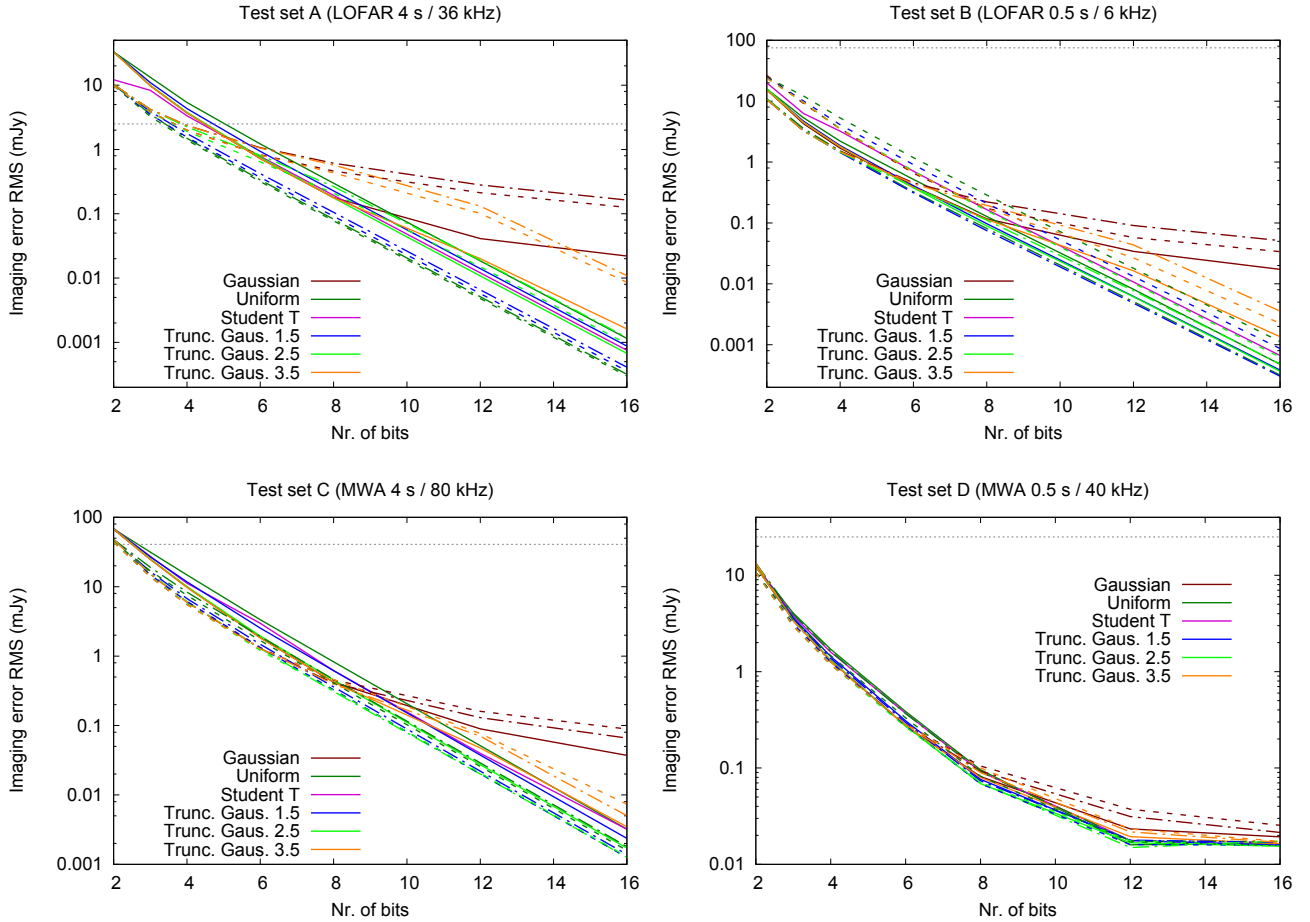
Fig. 4. Compression noise rms values for different quantization schemes and test sets, determined by measuring the rms in the difference between images processed with and without compression. The horizontal dashed grey line indicates the system noise level.

the effect of quantization noise during the calibration and imaging due to the use of 32-bit floats in those operations. The small numerical changes caused by the compression will trigger a different instantiation of quantization noise during the imaging, and this dominates the error. Because the error is not decreased when going from 12 to 16 bits, the implication is that storing more than 12 bits is uneconomical for this test set, because the imaging error is already dominated by the numerical precision of the imaging.

So far, results were based on the error, that is, the difference between the original dirty image and the dirty image after compression. To analyse the actual implications of compression after deconvolution, Fig. 5 shows the fractional noise changes calculated from cleaned Stokes-I images. Noise values are measured in a relatively empty square near the centre of the images. These plots mostly show the same trends as Fig. 4. For test set B, increasing the number of bits does not improve the sensitivity of the deconvolved image: even at the lowest bit-rate, the deconvolved image is dominated by system noise.

4.2. Properties of compression noise

Besides producing low average errors, a visibility compression technique should also not add structural noise to an image, and should not change the flux of sources. To show that the compression technique indeed behaves properly, compression examples are shown in Figs. 6 and 7. Both figures display the original and compressed images and their difference, and are made from test set A.

In Fig. 6, an 8-bit quantization scheme is used with the 2.5σ-truncated Gaussian distribution and AF-normalization, such that the compression noise is lower than the image noise. The original and compressed images can not be distinguished by eye. The difference shows unstructured noise. The noise distribution of the image follows closely a Gaussian distribution.

In Fig. 7, a 2-bit quantization scheme is used with the 2.5σ-truncated Gaussian distribution and AF-normalization. As can be seen in the compressed image, the noise is significantly increased due to the low bit-rate and reasonably high S/N of the observation. The difference image again does not show any structure or bias at the position of strong sources. The noise distribution of the difference image follows again a Gaussian distribution.

From these results, it is clear that the quantization noise behaves like normal system noise in the image plane. While test sets B, C and D are short observations, test set A is a 6 h observation. Because the noise in test set A does not show systematic behaviour, one can conclude that compression noise averages down like system noise, and remains uncorrelated in longer observations.

4.3. Minimum bit-rates

Using Figs. 4 and 5, a minimum acceptable bit-rate can be derived by defining an acceptable imaging error. An acceptable imaging error might depend on the application: in a confusion-limited survey such as GLEAM (Wayth et al. 2015; Hurley-Walker et al. 2016), a compression noise addition at 10%

Table 2. Imaging error rms values with 8-bit compression.

Set	Quantization	AF rms	RF rms	Row rms
A	Gaussian	179 μJy	462 μJy	612 μJy
	Tr. Gaus. 1.5	227 μJy	92 μJy	105 μJy
	Tr. Gaus. 2.5	178 μJy	209 μJy	265 μJy
	Tr. Gaus. 3.5	178 μJy	437 μJy	572 μJy
	Uniform	299 μJy	78 μJy	82 μJy
B	Gaussian	117 μJy	174 μJy	224 μJy
	Tr. Gaus. 1.5	98 μJy	212 μJy	75 μJy
	Tr. Gaus. 2.5	91 μJy	160 μJy	113 μJy
	Tr. Gaus. 3.5	115 μJy	174 μJy	196 μJy
	Uniform	128 μJy	290 μJy	82 μJy
C	Gaussian	421 μJy	452 μJy	406 μJy
	Tr. Gaus. 1.5	616 μJy	327 μJy	360 μJy
	Tr. Gaus. 2.5	453 μJy	318 μJy	309 μJy
	Tr. Gaus. 3.5	422 μJy	459 μJy	394 μJy
	Uniform	827 μJy	411 μJy	462 μJy
D	Gaussian	81 μJy	104 μJy	93 μJy
	Tr. Gaus. 1.5	76 μJy	67 μJy	74 μJy
	Tr. Gaus. 2.5	68 μJy	70 μJy	70 μJy
	Tr. Gaus. 3.5	79 μJy	99 μJy	90 μJy
	Uniform	95 μJy	81 μJy	91 μJy
Avg	Gaussian	199 μJy	305 μJy	344 μJy
	Tr. Gaus. 1.5	225 μJy	195 μJy	153 μJy
	Tr. Gaus. 2.5	184 μJy	201 μJy	194 μJy
	Tr. Gaus. 3.5	197 μJy	296 μJy	320 μJy
	Uniform	294 μJy	243 μJy	177 μJy
Per-normalization avg		220 ± 31 μJy	248 ± 23 μJy	237 ± 39 μJy
Per-quantization avg				
	Gaussian	282 ± 62 μJy		
	Tr. Gaus. 1.5	191 ± 46 μJy		
	Tr. Gaus. 2.5	193 ± 20 μJy		
	Tr. Gaus. 3.5	271 ± 55 μJy		
	Uniform	238 ± 69 μJy		

Notes. Averages have been calculated by normalizing the results for each set.

of the system noise is likely acceptable, since this will still be well below the imaging noise level. However, in system-noise-limited spectral-line work or in Epoch of Reionization experiments, a 10% noise increase might not be acceptable.

Minimum bit-rates are now calculated for four different requirements: for requirements (i) and (ii), the dirty image error rms must be smaller than 10% and 1% of the system noise, respectively; for requirements (iii) and (iv), the deconvolved image noise level must change by less than 10% and 1%, respectively. Table 3 lists the results. With 10-bit compression, every test set can be imaged with less than 1% error. Since a 1% imaging error with a smaller imaging noise increase is probably acceptable in any project, it is likely that 10-bit compression is acceptable for all LOFAR observations, and 8-bit compression for all MWA observations.

Test sets B and D are at higher frequency and time resolution than A and C, and their S/N is thus lower. Therefore, they require fewer bits to achieve the same error. These sets can be compressed using 6-bit (LOFAR) and 5-bit (MWA) compression with less than 1% error. Test sets B and D are compressed at the correlator output resolution, and this is in most cases the resolution at which the observations are archived. Hence, assuming the results can be generalized to other observations, LOFAR and MWA archival data can be compressed with an insignificant change in image quality using 5 and 6-bit compression, respectively.

4.4. Measurement set compression

A Casacore measurement set can be decomposed into visibility data, weights and observational metadata. Here, observational metadata refers to other data inside the measurement set, such as antenna indices and timestamps. The normalization factors are considered to be part of the visibility data volume. The measurement set definition allows two ways of storing weights: a single weight per table row that applies to all visibilities of that row; and a weight per visibility. In the latter case, the weights are stored in a column named "WEIGHT_SPECTRUM". For each complex visibility consisting of two 32-bit floats, a single 32-bit float weight is stored. Therefore, the weights make up about one third of the volume of a data set. LOFAR and MWA observations require a weight per visibility, in particular because RFI can cause the weights of visibilities within a row to vary, and sensitivity will be lost if per-visibility weights are not stored and used in the processing. Another reason for per-visibility weights is to reduce the weight of the edges of a (sub)band with increased noise. Without per-visibility weights, these edges are not down-weighted and reduce the sensitivity.

When the DATA column is compressed by a factor of a few, it becomes relevant to compress the weights too. Fortunately, this is much less complicated. Normally, four linearly-polarized correlations are stored inside LOFAR measurement sets. The weights of visibilities for different polarizations are always the same. The measurement set format does not make use of this fact, and stores four equal weights in the case of LOFAR. By simply writing only one weight for the four polarized visibilities, the weight volume is reduced by a factor of 4. To increase the weight compression further, one can use the fact that quantizing the weights is much less problematic than quantizing visibilities: while changing the weights might reduce the sensitivity, even if the weights are always truncated, they will not cause a bias. Dithering is therefore not required. Weights are also not noisy, and can not have extreme outliers, making them suitable for linear quantization. I found that linearly quantizing the weights to 12 bits with row normalization does not influence the sensitivity noticeably ($\ll$1%) in the four test sets. The weights can probably be quantized with fewer bits, but extensive testing is limited in this work to compression of visibility data, which is the most pressing issue. Tests with different bit-rates for the weights were not performed. With 12-bit weights and one weight for every four differently-polarized visibilities, the weights are compressed by a factor of 10, and are no longer the dominating contribution to the size of compressed measurement sets.

Figure 8 shows the size of the data, weights and metadata of a measurement set that contains the data for test set B. The sizes before and after compression relative to the uncompressed size of the full measurement set are shown. The size of the measurement set after compression with the BZIP2 algorithm (Burrows & Wheeler 1994) is also shown for comparison. With
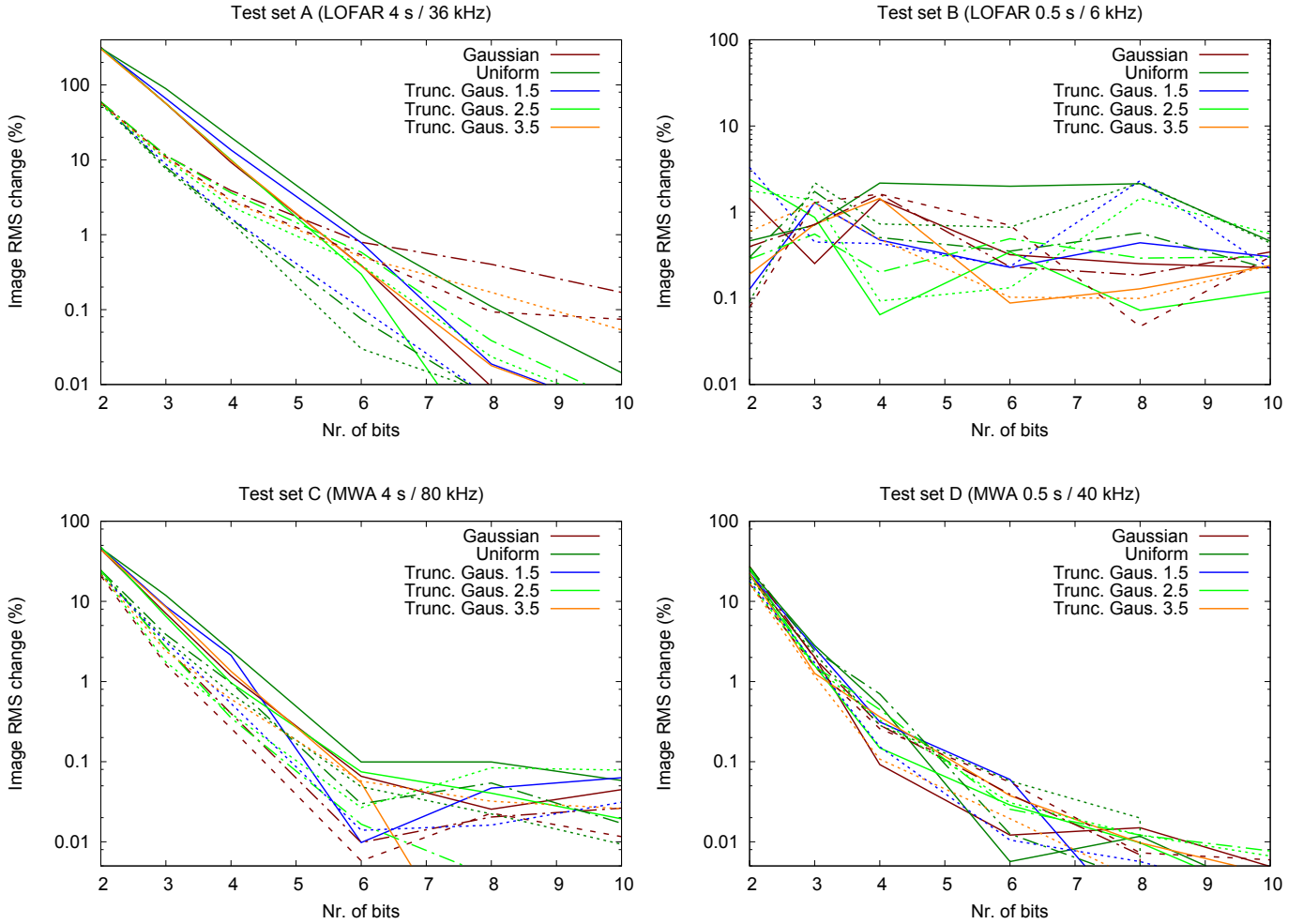
Fig. 5. Change in imaging noise as an effect of compression, measured relatively to the uncompressed image noise in an empty region of the images. The measurements were performed for different quantization schemes and test sets.

BZIP2, the visibilities are hardly compressed. However, because weights do not contain noise and are repetitive, the weights can be compressed to less than 1% of its original size. When 2 or 3 bit compression is used for the visibilities, the weights and metadata dominate the measurement set size.

4.5. Computational performance

Table 4 lists the wall-clock time of several operations. Each entry is the average of five runs. For the measurements, a high-end server with a 40-core Intel Xeon E5-2660 running at 2.20 GHz was used. For storage, a RAID disk with 4×3TB spinning disks was used. Since the ratio between compression and writing will vary strongly depending on the speed of the system, these results only provide an indicative impression of the speed of the compression.

Compressing the visibilities consists of reading the old data column and writing a new table column into the measurement set with the compressed visibilities. To replace an existing column with a compressed column, it is normally also required to replace flagged visibilities with NaN values. When performed as a separate step, this takes approximately as much time as compressing the visibilities. This is not included in the measurements. In a pipeline that writes out compressed visibilities, this step can be done on the fly. Once the compressed column has been written, the old uncompressed data column can be removed from the

data set. However, the CASACORE storage managers do not release the space of a removed data column, and the measurement set will therefore not decrease in size. In order for the space to be released, the whole measurement set needs to be rewritten. Performing the full reordering is not included in these measurements. The reordering can be avoided by writing out compressed visibilities in a pipeline or by the correlator instead of replacing the data column of an existing measurement set.

The results show that compression speeds up processing of imaging. A considerable cost in compressing the visibilities is reading the uncompressed data, which can be avoided by integrating compression in a pipeline. Using a lower bit-rate decreases the wall-clock time of the operations, because it requires less reading or writing.

5. Discussion

It was shown that Dysco works best in cases with low S/N, where it can achieve compression factors of 5 to 6, or approximately twice as much if it is acceptable to increase the system temperature by 10%. Since all four test sets are observations of bright targets, these test sets are approximately worst case situations. Most target fields will be more quiet, and will therefore compress with larger accuracy. While a few brighter targets exist (Cygnus A and Cassiopeia A in particular), these targets are often limited by dynamic range issues during deconvolution. It is
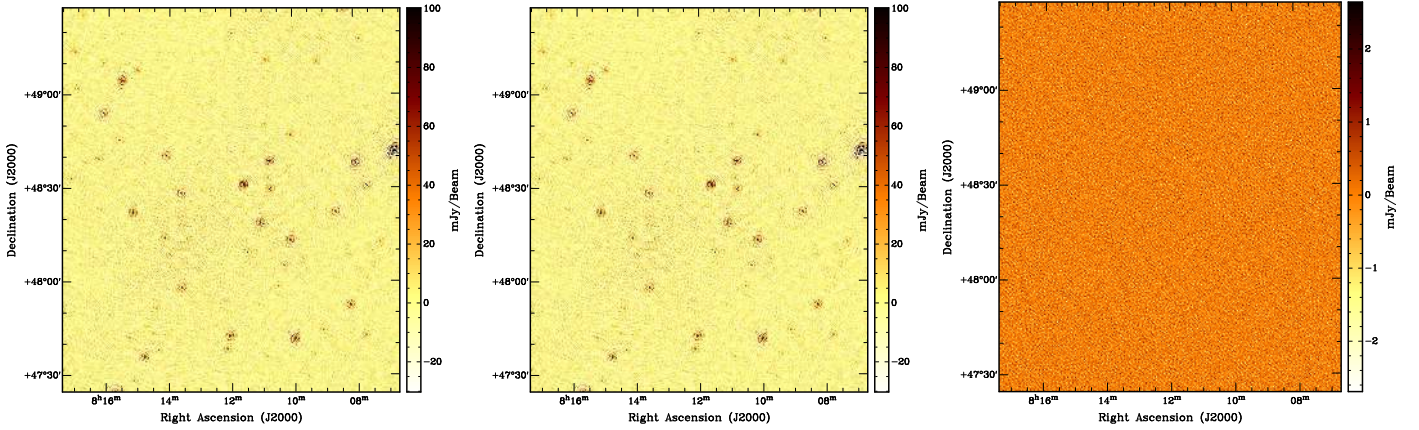
Fig. 6. Noise added by 8-bit compression using LOFAR test set A. *Left image*: result of calibration, 3C 196 subtraction and imaging without compression. *Centre image*: same, but before processing the visibilities were compressed using the 8-bit quantization scheme (4× compression) with the 2.5σ-truncated Gaussian distribution. *Right image*: difference between left and centre images – note the different colour scheme. Its rms is 430 μJy. The added noise is unstructured and well below the system noise, and the left and centre images are visually indistinguishable.
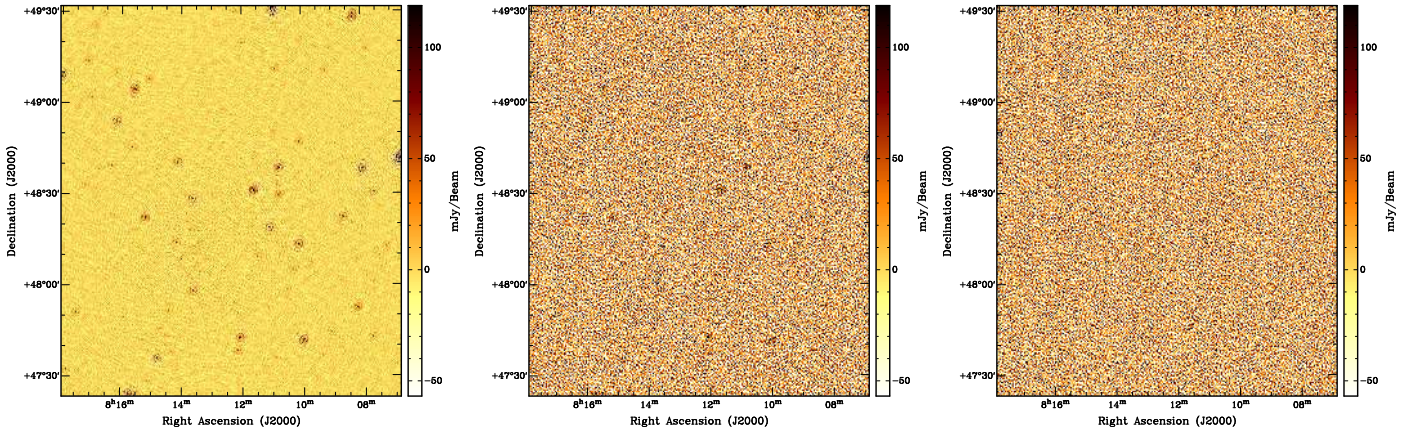

Fig. 7. Noise added by 2-bit compression using LOFAR test set A. *Left image*: results of calibration, 3C 196 subtraction and imaging without compression. *Centre image*: same, but before processing the visibilities were compressed using the 2-bit quantization scheme (16× compression) with the 2.5σ-truncated Gaussian distribution. *Right image*: difference between left and centre images. While the added compression noise dominates the noise in the image, the compression has not affected the sources systematically and the added noise is unstructured.

Table 3. Minimum bit-rate per data set, given four different requirements.

Set	$\Delta\sigma < 0.1\sigma_V$	$\Delta\sigma < 0.01\sigma_V$	$\sigma_c < 1.1\sigma_I$	$\sigma_c < 1.01\sigma_I$
A	7	10	3	5
B	3	5	2	–
C	5	8	3	4
D	3	6	2	3

Notes. The second and third columns provide the minimum bit-rate for a differential error rms < 10% and 1% of Stokes V rms, respectively. Columns four and five provide the minimum bit-rate for an absolute noise change <10% and 1%, respectively.

therefore likely that an increase in system noise will not degrade the imaging results. In the exceptional situation of very bright targets and an extremely high dynamic range, it is advisable to use no compression or compression with larger bit-rates.

Only LOFAR HBA and MWA sets were tested. Both tested telescopes operate at low frequency. It is likely that Dysco is effective for other telescopes, but further experiments are required to determine acceptable bit-rates. The LOFAR LBA measurements generally have lower S/N compared to LOFAR HBA,

hence using the same compression settings for LBA as for HBA will be at least as accurate. In the current SKA design, the station size of SKA-low stations will be similar to the size of LOFAR stations and correlator resolutions will be at least as high as for LOFAR. It can therefore be assumed that the compression of SKA-low observations is equally affective as for LOFAR. Very long baseline interferometry (VLBI) observations are typically correlated at high time and frequency resolution, and produces noise-dominated data. Compression is therefore likely to work well on VLBI observations.

Averaging in time or frequency affects the field of view because of time and frequency smearing. It also removes information that might be useful for high-resolution spectral work or calibration. The compression methods introduced here do not remove the high-resolution information. Due to the dependence on the S/N, Dysco will in particular be useful for compressing raw correlation data. Because observations are often archived at high resolution, this can significantly lower the demands for archive space. During data processing, data from the archive is often averaged before calibration and imaging to decrease the computational requirements (e.g. van Weeren et al. 2016; Hurley-Walker et al. 2016). During this step, the averaged data can be written without compression. With the implementation resulting from this work, this can be completely transparent
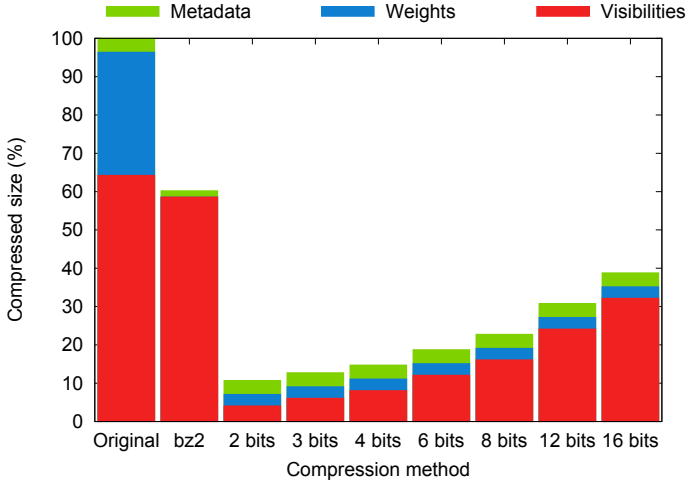
Fig. 8. Comparison of data volume using test set B. The "original" case is not compressed, and "bz2" is compressed using BZIP2 compression. For the other cases, the weights are compressed to 12 bits and the visibilities are compressed to the value on the *x*-axis with one of the quantization methods described in this paper.

Table 4. Wall-clock timings of several operations on test set D (65 GB) on a server with a 40-core Intel Xeon CPU E5-2660 @ 2.20 GHz and 4 × 3TB RAID disks.

Reading the uncompressed set	277 s
Copying the uncompressed set	679 s
Compress visibilities with 8-bit / AF-normalization	387 s
Compress visibilities with 2-bit / AF-normalization	356 s
Compress visibilities with 8-bit / RF-normalization	374 s
Compress visibilities with 2-bit / RF-normalization	351 s
Compress visibilities with 8-bit / row normalization	378 s
Compress visibilities with 2-bit / row normalization	332 s
Compress weights to 12 bits	285 s
Imaging of uncompressed set	339 s
Imaging of 8-bit compressed set	236 s
Imaging of 2-bit compressed set	198 s

Notes. Values are averages of five runs.

to the astronomer, because the compressed measurement set can be handled as a normal measurement set.

In some cases it might be helpful to compress averaged and/or calibrated data, for example to work with a limited amount of disk space or reduce IO overhead or network transfer. Because of the higher S/N, this requires more quantization bits, but compression factors of 3 and 4 can still be achieved for LOFAR and MWA, respectively.

The result that AF-normalization is on average more accurate than RF-normalization is somewhat counter-intuitive, since RF-normalization decomposes the data with more factors and has therefore more freedom to optimize the normalization. In fact, given the factors for AF-normalization, an RF-normalization can be found that performs equally well by setting each row factor to the multiplication of the two relevant antenna factors. Hence, RF-normalization never has to perform worse. The reason that this does occur is that, as discussed in Sect. 3, the AF-normalization algorithm decomposes the data in a different order: first it normalizes the channel standard deviations, then it finds the antenna factors, and finally it iteratively maximizes the channel factor or antenna factor that increases the sum of absolute value the most, until convergence. In the RF-normalization

algorithm, the standard deviations of the channels are also first normalized, but then each row is maximized. Afterwards, each channel is maximized. The latter algorithm is easier and computationally faster than the AF-normalization algorithm, but results evidently in less optimal factors. An algorithm that produces RF-normalization factors that is at least as accurate as AF-normalization, is to calculate the factors by starting from the AF-normalization factors, and then maximize each row. The computational cost of this is relatively low compared to astronomical operations such as calibration or imaging, so the extra computational cost is likely not an issue. Given that the noise variance of observations can physically be described by the AF-normalization factors, on noise-dominated sets it is likely that the extra normalization freedom of RF-normalization does not contribute much.

This work uses for the first time a non-linear quantization scheme on radio astronomical data, in which the quantization is matched to the expected distribution of the input data. In AF-normalization and RF-normalization it reduces the quantization error compared to linear quantization by 1.60 and 1.25 on average, respectively. Non-linear quantization is therefore indeed a small improvement over linear quantization for the encoding of radio data.

Our current implementation consists of a storage manager used by CASACORE and a stand-alone tool that can replace the column of an existing measurement set with a compressed column. As shown, compression works best on RFI flagged data, which is often performed by a preprocessing pipeline. For example, LOFAR uses NDPPP and MWA uses COTTER, two preprocessing pipelines that flag the data using the AOFLAGGER, convert the format of the data to a standard measurement set, and can do several other tasks at the same time. Because raw data from these telescopes have large volumes, every extra read or write of the raw data is costly. Hence, the ideal place for compression is inside these pipelines, just before writing to disk. Pipelines can do so by requesting CASACORE to use the compressing storage manager for the relevant columns.

As was shown in Sect. 4.5, writing data with compression is faster than writing without compression. In certain processing steps, this might be an important consideration. In the Factor pipeline (van Weeren et al. 2016), disk input-output is a considerable cost, because the original data needs to be read, phase changed, averaged and rewritten to a measurement set for each facet direction. One possible way to improve this, is to compress the high-resolution data at the original phase direction. This will have an insignificant loss in accuracy due to the low S/N, yet decrease the IO-overhead considerably.

5.1. Comparison with other implementations

In this section, I will briefly compare the features of Dysco to the AIPS 16-bit compression technique, the FITS file compression technique and the BITSHUFFLE compression technique.

AIPS uses row-normalization with uniform quantization, which I showed to have a high accuracy on average. As was shown, it performs not well in the case of RFI or high dynamic range. The bit-rate of 16 bits per input value allows about 4 orders of magnitude of dynamic range before catastrophic quantization occurs. Flagging the data before compression will mostly mitigate this problem. The RF and AF-normalization schemes introduced in this work also solve this issue. The lack of dithering in AIPS compression might lead to a systematic bias (Pence et al. 2010). This is only an issue in the case of high S/N. In the case of low S/N, the intrinsic noise in the visibilities will

mimic dithering. Of course, with AIPS the compression rate is limited to a factor of 2.

The BITSHUFFLE compression technique also uses uniform quantization. Before quantization, values are rounded relative to the expected noise in a sample. The effect of this is somewhat comparable to the combination of normalization and quantization used in this work, which also throws away significance that is smaller than the noise.

The BITSHUFFLE and FITS compression technique use more bits when the rounded values have higher entropy. Therefore, it compresses effectively with a variable bit-rate. This is effective for signal-dominated data with repeating values, but has no benefit for the compression of noise-dominated data with constant entropy.

The FITS compression technique uses exponential quantization. Thereby, similar to Gaussian quantization, the quantization error is smaller for values near zero. In exponential quantization, small values will be preserved with exponentially more precision than large values. This is useful for image data, where the relative error on each pixel is relevant, but is not optimal for visibilities. For those, the average error is a more relevant measure of the effects on the final science products, and since visibilities follow a Gaussian distribution – or even a more uniform distribution after normalization – quantizing with an exponential distribution is not optimal.

5.2. Guideline for method selection

While it is possible to measure the compression loss of observations and determine the acceptable compression method, this is not practical for regular observing of different targets with different correlation settings. Instead, in this section I will weigh the characteristics of the various compression configurations and suggest a rule for selecting the compression method that will not affect data.

Based on the accuracy, several combinations of quantization and normalization have been shown to perform well on all test sets. Per-row normalization with uniform quantization, similar to the method used by AIPS but including dithering and different bit-rates, is on average one of the most accurately performing methods. Moreover, it is the simplest method to implement. However, as was shown, the accuracy of compression with row normalization is sensitive to large dynamic ranges in the visibility, such as in the case of RFI (Fig. 1), and makes the compression of the different polarizations dependent on each other (Fig. 2). The RF and AF normalizations are robust to these issues. On average, the AF normalization performs slightly better than the RF-normalization method, and the most accurate combination with AF-normalization is more accurate than the most accurate combination with RF-normalization. The difference between accuracy of the normalization methods with best quantization are however not significant.

AF-normalization is theoretically slightly more computational intensive during compression compared to RF-normalization and row-normalization, but this difference is small in practice. The methods require the same time to decompress. The quantization method does not change the computational performance of the methods. AF-normalization requires special treatment of auto-correlations, but requires less metadata to be stored ($N_{pol}(N_{ch} + N_{ant})$ vs. $N_{pol}(N_{ch} + N_{ant}^2)$ per timeblock). In large measurement sets with many channels, the metadata is insignificant. All in all, there is little reason to select one method over another.

Noise normalization performed by AF normalization will equalize the statistics of all visibilities. Therefore, the extra factors that RF normalization adds do not match the physical process by which the noise is formed. When the signal adds a dominant contribution to the visibilities, the RF-normalization can have a benefit, because with RF-normalization the visibilities of different baselines can be scaled independently. The results show that AF normalization combines better with centrally-dense quantization distributions, while RF normalization favours more uniform distributions. For sets with larger number of channels or antennas, the statistics of sets after normalization will be more accurately quantized by centrally-dense distributions.

For these reasons, I suggest to use the following guideline for compression of MWA and LOFAR observations, as well as future high-resolution SKA observations:

- For the compression of high-resolution observations ($\Delta t \times \Delta \nu \leq 0.5$ s $\times$ 6 kHz for LOFAR and SKA, and $\Delta t \times \Delta \nu \leq 0.5$ s $\times$ 40 kHz for the MWA), use AF normalization with 2.5σ-truncated Gaussian quantization. As described, on average it is the most accurate method with the exception of the less stable row-normalization, it is robust and it matches the underlying physics. High-resolution LOFAR sets can be compressed with 5-bit quantization and MWA sets can be compressed with 6-bit quantization, respectively.
- Observations with a lower resolution – and therefore higher S/N – should be compressed using higher bit-rates. Up to a resolution of $\Delta t \times \Delta \nu \leq 4$ s $\times$ 36 kHz for LOFAR and SKA-low, and $\Delta t \times \Delta \nu \leq 4$ s $\times$ 80 kHz for the MWA, RF-normalization with 1.5σ-truncated Gaussian quantization should be used, with 10-bit compression for LOFAR and 8-bit compression for the MWA. On the lower-resolution sets A and C, this is on average the most accurate compression method. As was shown, it is also robust.

Compression with these parameters increases the image noise with less than 1%.

6. Possible improvements and future work

After the quantization of randomly distributed Gaussian noise with a matched quantizer, the compressed data will be independently-distributed noise with a uniform distribution that can not be compressed any further. Hence, in noise-dominated situations, using the distribution of the data and normalizing the data with AF-normalization, will lead in theory to the best compression rate. In the case of signal-dominated data, compressed data will be correlated and can be compressed further. To make use of such correlations, an encoding scheme such as Rice coding (Rice & Plaunt 1971) or LZ77 compression (Ziv & Lempel 1977) can be used to increase the compression factor. This makes the input-output of data somewhat more complex, because different timeblocks can have different sizes after Rice coding, and this requires extra administration to keep track of the place and size of each timeblock. The algorithm will also have to look for correlations in the time direction, while time is generally the slowest changing dimension. It has been shown to provide good results thought (Masui et al. 2015), and a combination of these methods might provide a generic algorithm that performs near-optimal in noise-dominated as well as signal-dominated situations.

As was shown, metadata and weights are easily compressible. At low visibility bit-rates, these can have a significant size compared to the data. While it was shown that the weights can be compressed by a factor of 10, compression with BZIP2 can compress the weights by more than a factor of 100. Hence, a similar

algorithm but one that can randomly and transparently access the weights inside a measurement set will increase the compression of the weights, compared to the method used in this work.

Baseline-dependent averaging can also reduce the size of observations. This can in theory be performed together with the compression method described in this paper. In sets that are not baseline-dependently averaged, the visibilities of long baselines receive more weight and will therefore dominate the compression error in image space. Smaller baselines can be stored with fewer bits before their inaccuracies becomes noticeable in image plane. With baseline-dependent averaging, the weight of each visibility will be more constant when using the uniform weighting scheme, and every visibility will contribute similarly to the compression error (in image plane). However, if averaging of the smaller baselines increases the S/N in those baselines too much, compression will be less accurate. In that case, the smaller baselines might dominate the compression error. The application of baseline-dependent averaging for compressing archival data is limited, because accurate RFI flagging require high time and frequency resolutions (Offringa et al. 2013) and baselines can only be averaged up to the calibration solution interval. For LOFAR, this interval can be as short as a few seconds to deal with the ionosphere in high-resolution maps (van Weeren et al. 2016).

Currently, the distribution of a particular visibility is parametrized by a single scaling factor. It is evident from the test results that this single parametrization is not always optimal; signal-dominated data sets are more accurately compressed by using a uniform distribution in the quantizer, while noise-dominated data sets are more accurately compressed by a truncated Gaussian distribution. Therefore, a possible increase in accuracy could be achieved by increasing the number of parameters of the distribution within each timeblock, for example by searching the optimal truncation value for each timeblock and storing these along with the scaling factors.

7. Conclusions

The Dysco technique for compressing visibilities is suitable for radio observations. The noise added by this compression technique acts like normal system noise. The accuracy of the compression is depending on the signal-to-noise ratio of the data: noisy data can be compressed with a smaller loss of image quality. Data with typical correlator time and frequency resolutions can be compressed by a factor of 6.4 for LOFAR and 5.3 for MWA observations with less than 1% added system noise in

image plane. After averaging observations in time and frequency to the typical resolutions used in processing, a compression factor of 3.2 to 4 can be reached with less than 1% added system noise in image plane. The technique is in particular well suited to reduce the archival space requirements.

So far, testing was performed only on low-frequency data from the MWA and LOFAR telescopes. The implementation is generic and can be applied to other telescopes. However, further experiments are required to determine acceptable bit-rates.

References

Burrows, M., & Wheeler, D. J. 1994, Technical report 124, Digital Equipment Corporation, http://www.hpl.hp.com/techreports/Compaq-DEC/SRC-RR-124.html
Favre, C., Wootten, H. A., Remijan, A. J., et al. 2011, ApJ, 739, L12
Greisen, E. W. 2016, Tech. Rep., AIPS Memo 117, http://www.aips.nrao.edu/aipsmemo.html
Heywood, I., Bannister, K. W., Marvil, J., et al. 2016, MNRAS, 457, 4160
Hurley-Walker, N., Callingham, J. R., Hancock, P. J., et al. 2016, MNRAS, 464, 1146
Jiwani, A., Colegate, T., Razavi-Ghods, N., et al. 2013, TASA, 30, 23
Kazemi, S., Yatawatta, S., & Zaroubi, S. 2013, MNRAS, 430, 1457
Masui, K., Amiri, M., Connor, L., et al. 2015, Astron. Comput., 12, 181
Morabito, L. K., Oonk, J. B. R., Salgado, F., et al. 2014, ApJ, 795, L33
Offringa, A. R., de Bruyn, A. G., Zaroubi, S., et al. 2013, A&A, 549, A11
Offringa, A. R., McKinley, B., Hurley-Walker, N., et al. 2014, MNRAS, 444, 606
Offringa, A. R., Trott, C. M., Hurley-Walker, N., et al. 2016, MNRAS, 458, 1057
Pence, W. D., White, R. L., & Seaman, R. 2010, PASP, 122, 1065
Perley, R. A., Chandler, C. J., Butler, B. J., & Wrobel, J. M. 2011, ApJ, 739, L1
Rice, R., & Plaunt, J. 1971, IEEE Trans. Commun. Technol., 19, 889
Shannon, C. E., & Weaver, W. 1949, The Mathematical Theory of Communication (Univ. of Illinois Press)
Smirnov, O. M. 2011, A&A, 527, A107
Stappers, B. W., Hessels, J. W. T., Alexov, A., et al. 2011, A&A, 530, A80
Thompson, A. R., Emerson, D. T., & Schwab, F. R. 2007, Radio Science, 42, RS3022
Thompson, A. R., Moran, J. M., & Swenson, G. W. 2001, Interferometry and Synthesis in Radio Astronomy, 2nd edn. (Wiley-Interscience)
Tingay, S. J., Goeke, R., Bowman, J. D., et al. 2013, PASA, 30, e007
van Cappellen, W. A., & Bakker, L. 2010, in Phased Array Systems and Technology (ARRAY), 2010 IEEE Int. Symp., 640
van Diepen, G. N. J. 2015, Astronomy and Computing, 12, 174
van Haarlem, M. P., Wise, M. W., Gunst, A. W., et al. 2013, A&A, 556, A2
Wayth, R. B., Lenc, E., Bell, M. E., et al. 2015, PASA, 32, e025
van Weeren, R. J., Williams, W. L., Hardcastle, M. J., et al. 2016, ApJS, 223, 2
Wells, D. C., Greisen, E. W., & Harten, R. H. 1981, A&AS, 44, 363
Wilson, W. E., Ferris, R. H., Axtens, P., et al. 2011, MNRAS, 416, 832
Ziv, J., & Lempel, A. 1977, IEEE Trans. Information Theory, 23, 337