Research Paper

MWA tied-array processing III: Microsecond time resolution via a polyphase synthesis filter

S. J. McSweeney1, S. M. Ord1, D. Kaur2, N. D. R. Bhat1, B. W. Meyers1, S. E. Tremblay1, J. Jones1, B. Crosse1 and K. R. Smith1,2,3

1International Centre for Radio Astronomy Research (ICRAR), Curtin University, 1 Turner Avenue, Technology Park, Bentley, WA 6102, Australia, 2CSIRO Astronomy and Space Science, PO Box 76, Epping, NSW 1710, Australia and 3Department of Physics and Astronomy, University of British Columbia, 6224 Agricultural Road, Vancouver, BC V6T 1Z1, Canada

Abstract

A new high time resolution observing mode for the Murchison Widefield Array (MWA) is described, enabling full polarimetric observations with up to 30.72 MHz of bandwidth and a time resolution of $\sim 0.8\,$µs. This mode makes use of a polyphase synthesis filter to ‘undo’ the polyphase analysis filter stage of the standard MWA’s Voltage Capture System observing mode. Sources of potential error in the reconstruction of the high time resolution data are identified and quantified, with the ‘undo’ the polyphase analysis filter stage of the standard MWA’s Voltage Capture System observing mode. Sources of potential error in the reconstruction of the high time resolution data are identified and quantified, with the system is further verified by observing three pulsars with known structure on microsecond timescales.

Keywords: instrumentation: interferometers – pulsars: general – techniques: interferometric

(Received 5 March 2020; revised 11 June 2020; accepted 7 July 2020)

1. Introduction

Some of the most exciting advances in time-domain astronomy have only been made possible by pushing the capabilities of latest generation telescopes to be sensitive to signals of shorter and shorter duration. The serendipitous discovery of pulsars in the late 1960s is perhaps the prototypical example (Hewish et al. 1968). In more recent times, the ongoing effort to detect nanohertz gravitational waves by means of pulsar timing arrays (PTAs) requires the continual monitoring of the times of arrival of millisecond pulsars (MSPs) with microsecond accuracy (e.g. Hobbs & Dai 2017). Pulsars are also known to exhibit temporal structures on microsecond and even nanosecond time scales (e.g. Craft, Comella, & Drake 1968; Hankins et al. 2003), providing major clues for the underlying radio emission mechanism (e.g. Cordes 1981; Popov et al. 2002). Similarly, fast radio bursts have been shown to exhibit temporal sub-millisecond structures that either point to the intrinsic emission mechanism or to interesting propagation effects occurring in the intergalactic medium (Farah et al. 2018; Hessels et al. 2019). All of these examples serve to illustrate the scientifically important and still largely untapped parameter space that is only accessible to telescopes equipped with a sufficiently high time resolution observing mode.

The Murchison Widefield Array (MWA; Tingay et al. 2013) is a low-frequency ($\sim 80$ to 300 MHz) aperture array telescope located at the Murchison Radio Observatory in Western Australia. Now in its second phase of development (Phase II; Wayth et al. 2018), it consists of 256 ‘tiles’ (sets of $4 \times 4$ cross-dipole antennas) distributed over an area approximately 5.3 km in diameter, 128 of which can be used at a single time to form an interferometer. Originally conceived as an imaging telescope (which requires only the time-averaged cross-correlation products of the tiles, or ‘visibilities’, to be retained on disk), it was subsequently augmented with the functionality to capture the raw complex voltages of each tile, known as the Voltage Capture System (VCS; Tremblay et al. 2015). This system has enabled the MWA to be used as a premier instrument for high time resolution studies of transient signals, especially pulsars (e.g. Bhat et al. 2016; McSweeney et al. 2017; Meyers et al. 2018; Kirsten et al. 2019).

Although the tile voltages are sampled at a (Nyquist) rate of 655.36 MHz, these data undergo several stages of processing before finally being written to disk. After preliminary filtering and digitisation, the raw voltages are subjected to a two-stage frequency analysis filter, which trades time resolution for increased frequency resolution. In the MWA’s case, both stages of the analysis filter were implemented as polyphase filterbanks (PFBs; Harris & Haines 2011; Prabu et al. 2015). The first-stage (‘coarse’) PFB reduces the effective sampling rate by a factor of 512, resulting in an array of complex-valued samples with 1.28-MHz resolution in frequency (‘coarse’ channels) and $\sim 0.8\,$µs in time. In the second-stage (‘fine’) PFB, each coarse channel is further split into $128 \times 10$ kHz ‘fine’ channels at the cost of decreasing the time resolution to 100 µs.

In the current MWA system design, only the latter time resolution data product (i.e. 100 µs) is made available to the user. While this is sufficient for many pulsar studies (e.g. Oronsaye et al. 2015; McSweeney et al. 2017; Bhat et al. 2018; Meyers et al. 2018), it is nevertheless too coarse for many science applications involving MSPs. In principle, the original higher time resolution...
can be recovered from the channelised output (either approximately or exactly) by means of a synthesis filter which acts as an ‘inverse’ operation to the analysis filter. The conditions under which the original time series can be exactly reproduced depend on the choice of analysis and synthesis filters.

Here, we describe the synthesis filter that is implemented as the (optional) final stage of the tied-array beamforming pipeline, the former stages of which are described in detail in Ord et al. (2019, hereafter Paper I) and Xue et al. (2019, hereafter Paper II). The synthesis filter is applied to the fine-channel output of the beamformer and recovers the coarse channel time series. That is, it effectively ‘undoes’ the fine PFB, increasing the available time resolution to \( \sim 0.8 \mu s \). A brief review of PFBs in general, and their particular implementation in the case of the MWA, is given in Section 2. The design of the synthesis filter is described in Section 3, including a discussion of its fidelity, that is, the appearance of any temporal and spectral artefacts introduced by the synthesis filter itself. Finally, the practical use of this functionality is demonstrated in Section 4.1 through three examples: MWA observations of the pulsars (PSRs) J2241–5236, J0437–4715, and B0950+08 (J0953+0755).

2. Polyphase filterbanks

PFBs are a type of analysis filter, designed to extract spectral information out of discrete time series data. They can be considered a generalisation of the more familiar discrete Fourier transform (DFT) and are designed to overcome the undesirably uneven frequency response (i.e. spectral leakage) inherent in the application of DFTs to discretely sampled time series of finite length. PFBs are well described in standard texts (Crochiere & Rabiner 1983; Harris 2004; Oppenheim & Schafer 2009); a clear and concise review of PFBs in the context of radio astronomy is given in Harris & Haines (2011). Thus, only a brief review of the salient features is given, in order to prepare the reader for the description of the synthesis filter in the following sections.

2.1. General review and mathematical notation

The PFB is a transformation from the time domain, \( x[n] \), to the frequency domain \( X_k[m] \), where the \([\cdot]\) notation denotes a discretised index, \( k \) is the channel number, and \( n, m \in \mathbb{Z} \) are the time indices for the pre- and post-channelised data, respectively. Let \( K \) be the number of (equally spaced) frequency channels required for some desired spectral resolution. Although applying a DFT to \( N = K \) adjacent time samples in \( x[n] \) will produce the desired resolution, the result will be an imperfect representation of the true spectrum because of spectral leakage, which is when power that properly ‘belongs’ to some particular frequency bin appears in (or, is aliased to) other, nearby bins. The impossibility of perfectly eliminating spectral leakage can be seen by realising that operating on a finite-length time series is equivalent to multiplying an arbitrarily long time series with a rectangular window function,

\[
w_k[n] = \begin{cases} 
1, & 0 \leq n < N, \\
0, & \text{elsewhere},
\end{cases}
\]

whose effect in the frequency domain is to convolve the ‘true’ spectrum of the signal with the Fourier transform of the window function. In the case of a rectangular window, this is the sinc function.

A common strategy for mitigating spectral leakage is to choose an alternative window function whose Fourier pair is localised in the frequency domain, and which therefore produces a tolerable level of spectral leakage when convolved with the signal’s spectrum. Many possible windowing functions have been identified, which in general trade the ‘amount’ of leakage with the ‘location’ of the leaked components. For our purposes, it is sufficient to note that the inevitable presence of a windowing function motivates the definition of the analysis filter, \( h[n] \), and the generalised windowed DFT

\[
X_k[m] = \sum_{n=0}^{N-1} h[mM - n]x[n]e^{-2\pi jkn/K},
\]

where \( j = \sqrt{-1} \) denotes the imaginary number, \( m \) is the time index of the channelised (output) data, and \( 0 \leq k < K \) denotes the channel number. Equation (2) describes the action of performing DFTs on short, windowed segments of the input time series of length \( K \). \( M \) is the number of samples that the window is translated along \( x[n] \) between successive DFT operations; thus, the index of \( h[mM - n] \) represents the shift required in order to produce the spectrum at time \( m \). If \( M < K \), then the windows overlap and the resulting channelisation is oversampled; if \( M = K \), then it is critically sampled. The choice of \( h[n] \) is motivated by the shape of its frequency response (i.e. its Fourier pair), whose characteristics (e.g. width and location of sidelobes) are chosen according to the advantages they carry in particular contexts. Leaving \( x[n] \) ‘unweighted’ is equivalent to choosing \( h[n] = w_k[n] \), in which case Equation (2) merely describes a DFT performed on each successive window, which in this context is also called a short-time Fourier transform. On the other hand, choosing \( h[n] \) to be the sinc function will result in a frequency response that approximates a rectangular window.

It is well known that scaling a function in the time domain produces the inverse scaling in the Fourier domain. This fact motivates an alternative strategy for mitigating spectral leakage. Choosing a larger window size, \( N = KP \) (for integer \( P > 1 \)), and a corresponding wider analysis filter, will result in a frequency response that is similar in shape, but \( P \) times narrower than the frequency response of the original analysis filter. A DFT applied to the larger number of samples will naturally produce a correspondingly larger number of (more closely spaced) frequency channels, but choosing only every \( P \)th channel and discarding the rest (known as decimation) ensures that the desired spectral resolution with \( K \) channels is retained. In this way, spectral leakage can be contained arbitrarily close to the ‘correct’ channel by choosing a sufficiently high value of \( P \).

The two-step algorithm described above (performing a windowed DFT on \( N = KP \) samples and decimating the resulting spectrum) defines the PFB. Formally, it is equivalent to Equation (2); however, in this context, the term critically sampled (i.e. \( M = K \)) implies that the \( N \)-length windows will now overlap. The term ‘polyphase’ derives from the fact that each block of \( K = N/P \) samples (known as taps) in \( x[n] \) is included in multiple applications of the DFT, but appearing at a different relative phase in each case.

One of the great advantages of the critically sampled PFB is the existence of a mathematically equivalent but computationally efficient implementation. It can be shown that Equation (2) is equivalent to first segmenting the windowed time series into taps, summing their respective samples elementwise, and performing a single DFT on the resulting array (now also of size \( K \)), that is,
Figure 1. A diagrammatic representation of the weighted overlap-add algorithm, as defined in Equation (3). Panel (a) shows the filter window being translated along a discretely sampled signal (in this case, containing a sinusoid and noise) with a step size of one tap. At each step, panel (b) shows how the signal (first row) is multiplied by a filter (second and third rows), and each tap is summed (bottom left) and Fourier transformed to produce the final spectrum (bottom right).

\[ X_k[m] = \sum_{n=0}^{K-1} b_m[n] e^{-2\pi j n K / K}, \]  \hspace{1cm} (3)

where

\[ b_m[n] = \sum_{\rho=0}^{P-1} h[K\rho - n] x[n + m M - \rho]. \]

A short proof of this equivalence is given in Harris & Haines (2011). The procedure described by Equation (3) is called the weighted overlap-add algorithm and is illustrated in Figure 1.

Figure 2. Top: The coefficients of the MWA’s fine PFB analysis filter, defined in Equation (4), which is composed of a Hanning window multiplied to a sinc function. Bottom: The frequency response of the analysis filter (black, solid), showing negligible attenuation across approximately 10 kHz (the bandwidth of a fine channel) and strong attenuation elsewhere. The frequency response is repeated for adjacent channels on either side (grey, dashed) showing crossover points on the channel edges at −3 dB.

2.2. MWA implementation of the fine PFB

The first-stage (coarse) PFB is described in Prabu et al. (2015), and here we only document a few details pertaining to the second-stage (fine) PFB. A PFB is specified by (1) the number of output channels, \( K \), (2) the number of taps, \( P \), and (3) the analysis filter, \( h[n] \). For the MWA, \( K = 128 \) (giving fine channels 10 kHz wide), \( P = 12 \), and

\[ h[n] = \begin{cases} w_H[n]w_s[n], & 0 \leq n < N, \\ 0, & \text{otherwise} \end{cases} \]  \hspace{1cm} (4)

where

\[ w_H[n] = \sin^2\left(\frac{\pi (n + 1)}{N + 1}\right) \]

is the Hanning window, and

\[ w_s[n] = \text{sinc}\left(\frac{\pi (n + 1 - N/2)}{K}\right) \]

is the scaled \( \text{sinc}(x) = \sin(x)/x \) function. It can be easily checked that \( h[n] \) is defined to be symmetric around sample \( n = 767 = N/2 - 1 \). The analysis filter is shown in Figure 2. The MWA’s fine PFB is critically sampled, with \( M = K = 128 \). The rationale behind the particular design choices for the MWA’s fine PFB is beyond the
scope of this paper—for the purposes of creating a synthesis filter, it is sufficient to know merely how the analysis filter is defined, and the fact that the PFB is critically sampled.

The fine PFB is implemented on field-programmable gate arrays (FPGAs) and was designed in such a way to accommodate the data rate and bit depth constraints set by the surrounding hardware. The input signal values are signed (5+5)-bit complex integers, and the final outputs are (4+4)-bit complex integers. The loss of precision associated with the demotion of 1 bit naturally places limits on the ability for any synthesis filter to perfectly reconstruct the original coarse channel time series, which is analysed for our system below. The full details of the FPGA implementation are given in the appendix.

3. Synthesis filters

In order to regain the time resolution lost during analysis, the spectral output of the PFB must be transformed back into the time domain. The inverse to the analysis operation defined in Equation (5) is analysed for our system below. The full details of the FPGA implementation are given in the appendix.

In order to regain the time resolution lost during analysis, the spectral output of the PFB must be transformed back into the time domain. The inverse to the analysis operation defined in Equation (5) for all values of time domain. The inverse to the analysis operation defined in Equation (5) is analysed for our system below. The full details of the FPGA implementation are given in the appendix.

3.1. Reconstruction error

In the context of an astrophysical signal, it is desirable to quantify the reconstruction error in terms of the signal-to-noise ratio (S/N). If we assume that individual samples follow, and are dominated by the same Gaussian noise statistics, then the reduction in S/N of a reconstructed sample can be estimated by considering the fraction of the power of the reconstructed sample that came from the original sample:

$$\left(\frac{S}{N}\right)_{\text{recon}} = \left( \frac{\sum_{m=-\infty}^{\infty} f_m[n] h_m[-n]}{\sum_{m=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} f_m[n] h_m[-n]} \right)^2. \tag{8}$$

This expression is invariant under the substitution $n \to n + \lambda M$, $\lambda \in \mathbb{Z}$, which indicates that the (average) reconstruction error is only a function of where the sample in question falls within a tap. Consequently, any synthesis filter (except the exact inverse of the analysis filter, if it exists) will introduce a ‘ringing’ effect into the reconstructed time series that has a period equal to the size of the PFB tap.

The ‘leakage’ of power into other samples implied by Equation (8) can also manifest itself as spurious imaging of a ‘true’ signal at intervals of one tap. This can be seen by considering the effect of a single-sample impulse, with a power much greater than the ambient noise, on the reconstructed time series. The sample itself, say, $x[n]$, will be reconstructed with high fidelity, as the reconstructed error is only comprised of contributions from the (relatively) low-level noise. The reconstruction error of the sample $x[n + K]$, however, will be dominated by the term in Equation (6) involving $x[n]$, according to the relative weighting introduced by the analysis and synthesis filters. Thus, the impulse will reappear at intervals of one tap, but where the relative power of each appearance is given by

$$p(s) = \left( \sum_{m=-\infty}^{\infty} f_m[n] h_m[-n] \right)^2. \tag{9}$$

This effect, termed temporal imaging, is discussed further in the context of specific filters.

3.2. Optimal and sub-optimal filter designs

Minimising the loss of $S/N$ is equivalent to solving Equation (7) for $f[n]$. However, exact solutions do not always exist, and in general, numerical methods must be employed to find a synthesis filter that minimises the reconstruction error.

Minimising the loss of $S/N$ is equivalent to solving Equation (7) for $f[n]$. However, exact solutions do not always exist, and in general, numerical methods must be employed to find a synthesis filter that minimises the reconstruction error.

*Equation (4) is defined on $\mathbb{R}$, but the actual implementation on the MWA’s FPGAs defines the analysis filter coefficients on $\mathbb{Z}$. The exact implemented values, $h^s[n]$, can be obtained from $h[n]$ via $h^s[n] \equiv \lfloor h[n] \rfloor$, where $\lfloor . \rfloor$ denotes rounding to the nearest integer and $\alpha = 117963.875$.

Xilinx Virtex4 XC4VSX35

Finding the synthesis filter that perfectly inverts a given analysis filter is tantamount to solving Equation (7) for $f[n]$. However, exact solutions do not always exist, and in general, numerical methods must be employed to find a synthesis filter that minimises the reconstruction error.
where $H^{(n)}$, $F^{(n)}$, and $D$ are matrices whose elements are given by

$$
H^{(n)}_{ij} = h_{P-1+j-i}[-n],
$$

$$
F^{(n)}_{i} = f_{j}[n],
$$

$$
D_j = 2^{j(P+j)/2},
$$

where $P$ is the number of taps in the analysis filter, and the size of $F^{(n)}$ is set to the desired number of (non-zero) taps in the synthesis filter, $P$. The indices are chosen such that, owing to the finite size of the analysis filter, the smallest $H^{(n)}$ that captures every non-trivial term in Equation (7) is the $(P + P - 1) \times P'$ matrix

$$
H^{(n)} = \begin{bmatrix}
    h_{P-1}[-n] & 0 & \cdots & 0 \\
    h_{P-2}[-n] & h_{P-1}[-n] & \cdots & 0 \\
    \vdots & \ddots & \ddots & \vdots \\
    0 & \cdots & h_0[-n] & h_1[-n] \\
    0 & \cdots & 0 & h_0[-n]
\end{bmatrix}
$$

with row number $i = (P + P)/2$ (in $H^{(n)}$ and $D$) corresponding to the $s = 0$ term.\(^2\)

Once the matrices $H^{(n)}$, $F^{(n)}$, and $D$ have been defined, solution by least-squares regression can proceed in the usual way, yielding the best-fit filter coefficients

$$
\hat{\theta}^{(n)} = \left( H^{(n)} \right)^{-1} H^{(n)} \hat{\theta}^{(n)} D,
$$

where $H^{(n)}$ is the transpose of $H^{(n)}$. With this notation, the reconstruction error can be more concisely expressed

$$
[S/N]_{\text{rec}} = \frac{(\hat{D}(P + P)/2)}{\hat{D}^T \hat{D}},
$$

where $\hat{D} = H^{(n)} \hat{\theta}^{(n)}$.

Figure 3 shows the solutions found for the MWA’s analysis filter defined in Equation (4), for 12-, 18-, and 24-tap synthesis filter sizes. Despite the complexity of the synthesis filters, they all share the same basic form as the analysis filter. This resemblance suggests that choosing a synthesis filter that is the mirror image of the analysis filter (i.e. $f_{j}[n] = h_{j}[-n]$) might also yield a reasonably good reconstruction, without having to calculate the optimal solutions numerically.

The performance of both the (sub-optimal) mirror filter and the (optimal) set of least-squares solutions can be evaluated straightforwardly using Equation (14). Figure 4 compares the loss of $S/N$ due to each filter (under the assumption of noise-dominated samples), revealing that, as expected, the filters with the larger number of taps perform better, and the mirror filter performs the least well.

Figure 4 suggests that one can achieve arbitrarily good performance by choosing a sufficiently long filter. However, the better performance of the longer filters is offset by the increased computational resources and/or time required to apply them, which may exceed the system’s design specifications. Note that the mirror filter, being the same size as the 12-tap least-squares filter, offers no advantage in terms of minimising the reconstruction error. On the other hand, the way that the signal power is distributed across multiple taps is markedly different for the two filter types, as shown in the bottom panel of Figure 4. The mirror filter performs slightly worse in nearby taps but drops off more quickly further

\(^{Note that since P = 12 for the PFB considered here, only the case where P is even has been considered.\)
out. For applications where the samples are signal-dominated, and
minimising temporal imaging is more important than measuring the
$S/N$ of the signal precisely, the mirror filter may therefore
offer some advantage over the least-squares solution. However,
observations of pulsars only rarely fall into this regime, even for
observations of single pulses, where typically many samples must
be averaged before the signal becomes more significant than the
noise. A more detailed analysis of these effects is therefore outside
the scope of the present work.

3.3. MWA implementation

We initially implemented the mirror filter for the purpose of pro-
totyping the synthesis filter, due to the ease of generating the
coefficients, and the 12-tap least-squares filter was implemented
once proof of concept had been demonstrated. Both of these
synthesis filters are available as optional components of the MWA-
VCS beamforming software VCSTools, described in Paper I. Even
though our system places no stringent limits on computational
resources, we have not implemented the larger filters due to the
identification of other sources of error that dominate over the
reconstruction error defined above, rendering the advantages of
the longer filters relatively minor in comparison. An analysis of
these errors is presented in the following sections.

Viewed as a linear operation, Equation (5) is ideally suited for
graphics processing units (GPUs), and we have implemented it in
VCSTools for NVIDIA’s Compute Unified Device Architecture
(CUDA) architecture. Rather than use existing implementations
of the implied inverse DFT, the $N \times K$ exponential terms (the
so-called ‘twiddle factors’) are pre-calculated and stored in a 2D
array of double-precision floating point numbers on the GPUs.
These are then accessed as required for the calculation of a given
sample $\hat{x}[n]$. The combined GPU-accelerated calculations for
both beamforming and coarse channel reconstruction run faster
than real time.\(^e\)

Once the synthesis filter has been applied to each of the MWA’s
polarisation streams for each coarse channel, the resulting high
time resolution time series is written out as complex voltages
(i.e. without converting to Stokes parameters) in the Very-long-
baseline interferometry Data Interchange Format (VDIF) file
format (Whitney et al. 2009). This format was chosen because it is
supported by the coherent de-dispersion functionality of DSPSR\(^f\)
(van Straten & Bailes 2011), a software suite for processing pul-
sar time series. The VDIF format requires each sample to fit into
a signed 8-bit complex integer data type, which is performed as the
final step before writing to disk.

4. System performance

The MWA implementation of the fine PFB differs from Equation
(3) in a few important respects, causing a reduction in the recon-
structed $S/N$ that would be present even if the synthesis filter
perfectly inverted the analysis filter. The most significant differ-
ence is that both the coefficients $b[n]$ and the final sum undergo
a rounding operation, resulting in the approximate (i.e. both less
precise and biased) spectrum

$$
\hat{X}_\delta[m] = \sum_{n=0}^{K-1} b_m[n] e^{-2\pi j kn/K},
$$

where $[\cdot]$ and $[\cdot]_{\text{sym}}$ denote symmetric and asymmetric
rounding operations, respectively, described in Appendix A.

The non-linearity of the back-to-back system induced by
Equation (15) implies that there is no single impulse response
test that can adequately characterise the whole system. To wit, the
results of an impulse response test depend sensitively on at least
three factors: the magnitude of the impulse; its position within a
tap; and the arbitrary scaling factor and quantisation applied at the
end of the test required by the integer-based output formats. For
example, an impulse at tap position $n \equiv 0 \pmod K$ will be perfectly
reproduced if its magnitude is such that the amount of rounding
that takes place during analysis is minimised. On the other hand,
an impulse at $n \equiv 64 \pmod K$ can be chosen such that the response
is significantly worse than that implied by Figure 4.

For this reason, and also because the vast majority of applica-
tions of this system falls in the regime of noise-dominated samples,
we have decided to forego the traditional impulse response test in
favour of a back-to-back test involving real data collected expressly
for this purpose. This is an empirical test of system that made use
of a non-standard observing mode to record a small amount of
simultaneous coarse and fine channel data (i.e. both before and
after the fine PFB analysis filter stage). The fine channels from
one polarisation of a single tile were extracted and subjected to
both the mirror filter and the 12-tap least-squares filter to recon-
struct the 1.28 MHz coarse channel time series, which could then
be compared directly with the original data. The real and imagi-
ary parts of the time series resulting from the least-squares filter
are shown in Figure 5.

With the original and reconstructed time series in hand, we
estimated the $S/N$ loss for each tap position by comparing the vari-
ance of the original time series (the ‘signal’), $\sigma^2_s$, with the variance
of the residuals (the ‘noise’), $\sigma^2_N$. In analogy with Equation (8),

$$
[S/N]_{\text{recon}} = \frac{\sigma^2_s}{\sigma^2_s + \sigma^2_N}.
$$

Recognising that some of the noise variance is due to the synthe-
sis filter (cf. Figure 4), we have estimated the contribution due to
rounding errors (and any other implementation-specific effects)
by calculating the ‘filter-subtracted’ $S/N$ loss:

$$
[S/N]_{\text{recon}} = \frac{\sigma^2_s}{\sigma^2_s + (\sigma^2_N - \sigma^2_{\text{filter}})},
$$

where $\sigma_{\text{filter}}$ is derived from Equation (8).

Figure 6 shows the results for just under one second’s worth of data
(1278464 × 0.78 µs samples), where 1 536 samples (one
tap) were excised due to the synthesis filter being applied to
zero-padded data beyond the edge of the second. The remaining
samples were sorted into their respective tap positions, and the
variances calculated for each set (containing 12 78 464
9 988 samples). Near the edges of the tap, the rounding errors
dominate the reconstruction noise, which we estimate contributes
roughly $-0.26$ dB of signal loss at all tap positions. In more central
tap positions, however, the contributions from the rounding errors
and the filters are comparable, and significant improvements could
be made by using longer filters. Nevertheless, unless there is a way
to mitigate the rounding errors during the application of the anal-
sis filter (explored below), it is unlikely that we will be able to
achieve a total $S/N$ loss better than approximately $-0.26$ dB at any
tap position.

\(^e\)https://github.com/CIRCA-Pulsars-and-Transients-Group/vcstools.

\(^f\)Verified for CUDA compute capability $>3.5$.

\(^f\)http://dspsr.sourceforge.net/.
Mitigation of rounding errors

The original (i.e. before applying the fine PFB) high time resolution samples are (5+5)-bit complex integers, scaled such that the bit occupancy is not so low that information is lost, and not so high that individual samples are clipped. Interestingly, one may question whether or not this known quantisation can be used to recover some of the information lost by the imperfect back-to-back system. If the errors are sufficiently small, then re-quantising the reconstructed samples will correct more errors than it introduces, giving overall better performance.

In this section, we offer a short proof of the condition for which re-quantisation results in a net recovery of S/N for samples following Gaussian statistics. As before, let $\sigma_k^2$ be the variance of the residuals which are assumed to be drawn from a normal distribution with zero mean. The effect of re-quantisation on the variance is that every sample between $k - \frac{1}{2}$ and $k + \frac{1}{2}$, for any integer $k$, gets ‘counted’ as if it had the value $k$. The adjusted variance, $\hat{\sigma}_N^2$, can then be expressed in terms of $\sigma_k^2$ via the second moment,

$$\hat{\sigma}_N^2 = \sum_{k \in \mathbb{Z}} \int_{k - \frac{1}{2}}^{k + \frac{1}{2}} \frac{1}{\sigma_N \sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_N^2}} dx,$$

$$= \sum_{k \in \mathbb{Z}} k^2 \left[ \frac{1}{2} \text{erf} \left( \frac{k + \frac{1}{2}}{\sigma_N \sqrt{2}} \right) - \frac{1}{2} \text{erf} \left( \frac{k - \frac{1}{2}}{\sigma_N \sqrt{2}} \right) \right],$$

$$= \sum_{k=1}^{\infty} k^2 \left[ \text{erf} \left( \frac{k + \frac{1}{2}}{\sigma_N \sqrt{2}} \right) - \text{erf} \left( \frac{k - \frac{1}{2}}{\sigma_N \sqrt{2}} \right) \right],$$

where the last equality takes advantage of the symmetry of the error function.

Re-quantisation is beneficial precisely when $\hat{\sigma}_N^2 < \sigma_N^2$, which can be shown by numerical methods to occur when $\sigma_N^2 \lesssim 0.29$. The MWA test data set presented above has a variance of approximately 6.7, which implies that the total S/N loss of the back-to-back system would have to be smaller than $-0.05$ dB before re-quantisation would improve the signal reconstruction. As Figure 6 shows, this is not met for any tap position, so we did not pursue this possibility further. However, systems that are able to achieve better than $\hat{\sigma}_N^2 \lesssim 0.29$ may do better yet by re-quantising the suitably scaled reconstructed samples.

4.1. Verification using pulsar observations

Observations of three pulsars are presented here to validate the synthesis filter as a useful tool for undertaking high time resolution studies of pulsars with the MWA: MSPs J2241–5236 and J0437–4715, and the bright, long-period PSR B0950+08. The first two pulsars were processed with the mirror filter, and B0950+08 was processed with the 12-tap least-squares filter. As Figure 6 implies, the use of the mirror filter instead of the more optimal least-squares filter would result in a further $\sim 0.1$-dB reduction in S/N.

4.1.1. PSR J2241–5236

J2241–5236 has a rotation period of 2.18 ms and a dispersion measure (DM) of 11.41 pc cm$^{-3}$. The resulting dispersion smear across 10 kHz channels at the central observing frequency of 150.4 MHz is approximately 0.2 ms, or 45$^\circ$ of rotation phase, consistent with
the width of the average profile formed from incoherently de-dispersed, fine-channelised data recorded with the MWA-VCS system. Upon applying the synthesis filter and coherently de-dispersing the reconstructed time series sampled at the much higher rate of 1.28 MHz, the same data revealed exquisite detail in the average profile, including a pair of pre-cursor components that are only marginally visible (if at all) at higher frequencies (Figure 7). These have since been confirmed during follow-up observations (Kaur et al. in prep) using the Band 3 receiver (250–500 MHz) of the upgraded Giant Metrewave Radio Telescope. In addition, these new, low-frequency observations have allowed us to measure the DM of this pulsar with unprecedented precision ($\sim (2 \pm 6) \times 10^{-6}$ pc cm$^{-3}$), with important consequences for measuring DM chromaticity and evaluating its effect on pulsar timing experiments. These results are discussed in detail in Kaur et al. (2019).

4.1.2. PSR J0437–4715

PSR J0437–4715 is a nearby MSP (distance $\approx 157$ pc; period $P = 5.76$ ms) and an established target for PTA applications. It boasts a complex, multi-component polarimetric profile (Figure 8) that spans more than half of its rotation period (cf. Yan et al. 2011). It was used in Paper I as a verification of the beamforming method employed in the VCS processing pipeline, although there is a noticeable difference between the circular polarisation published in that work and that shown here (see the Figure caption for details).

As evident from Figure 8, the application of the synthesis filter successfully recovers some curious fine structure (e.g. notch-like features at pulse phases $\sim 0.54$ and $\sim 0.7$) characteristic to this pulsar, first reported in early high time resolution studies made at 430 MHz (Navarro et al. 1997). The pulsar detection (for Stokes I only) was originally presented in Bhat et al. (2018), where the combination of a lower time resolution (100 $\mu$s) and a non-negligible dispersive smearing ($\sim 45$ $\mu$s) obscured the detection of these fine structures. Figure 8 thus presents the highest-fidelity detection of this important pulsar at frequencies below 200 MHz.

4.1.3. PSR B0950+08

B0950+08 is a bright, long-period ($P = 0.253$ s) pulsar known to exhibit microstructure (Popov et al. 2002; Kuzmin et al. 2003). A total of 80 seconds (315 pulses) of data (128 $\times$ 10 kHz fine channels) were recorded, beamformed, and subjected to the polyphase synthesis filter. Across the MWA bandwidth, we used the rotation measure (RM) synthesis technique\(^6\) (Brentjens & de Bruyn 2005) to measure an RM of $1.43 \pm 0.15$ rad m$^{-2}$, which is consistent with previous measurements of this pulsar’s RM (e.g. Noutsos et al. 2015). The mean profile formed from the one coarse channel is shown in Figure 9. Several individual pulses were sufficiently bright to see substructure. One such pulse is showcased in Figure 10, where all 24 coarse channels were integrated to maximise the $S/N$. To highlight the pulse’s substructure, we show it at

\(^6\)https://github.com/gheald/RMtoolkit.
three different time resolutions (corresponding to the chosen bin width of each plot).

Using observations at 102.5 MHz, Kuzmin et al. (2003) report microstructure with a characteristic timescale of \( \sim 60 \mu s \), which we are able to unambiguously identify in our data. We suggest that the finest observable structures (Figure 10, bottom plot) might correspond to the \( \sim 10 \mu s \) structures reported by Popov et al. (2002) at 1.65 GHz. However, Kuzmin et al. (2003) used only a single linear polarisation feed, and Popov et al. (2002), only a left-handed circularly polarised feed. With the benefit of full Stokes polarimetry, we are able to verify that on the smallest time scales, the emission is almost 100% polarised—mostly linear, but with a small amount of circular polarisation during the brightest part of the pulse. This is evidence that the errors introduced by the analysis-synthesis filter do not contribute a significant amount of polarisation leakage into neighbouring time bins—in particular, at intervals of 50 \( \mu s \).

5. Conclusion

The MWA-VCS telescope system outputs voltage data with a frequency resolution of 10 kHz and a time resolution of 100 \( \mu s \), which is suitable for many pulsar applications, as demonstrated in several pulsar science papers published to date. In this paper, we have presented an algorithm to undo the fine channelisation stage (fine PFB) of the VCS pipeline by using a synthesis polyphase filter, implemented as part of our processing software suite, VCSTools. The result is a reconstructed, coarse channelised (pre-PFB) time series, suitable for high time resolution studies of MSPs and other rapid transient signals. Although the reconstruction is not perfect, the average error does not exceed \(-0.65\) dB for noise-dominated samples.

We have further verified our system by observing PSRs J2241–5236, J0437–4715, and B0950+08, all of which are known to exhibit structures on \( \sim 1–10 \mu s \) timescales. In each case, our results have been shown to be consistent with observations at other radio frequencies. The advent of the VCS high time resolution observing mode thus distinguishes the MWA as a premier low-frequency instrument for studying pulsars and other transients at microsecond resolution, with important implications for studies of the pulsar emission mechanism, the characterisation of the interstellar medium in the ongoing search for gravitational waves, and others.

Acknowledgements. We thank the anonymous referee for suggestions that greatly improved this work. This scientific work uses data obtained from the Murchison Radio-astronomy Observatory. We acknowledge the Wajarri Yamatji people as the traditional owners of the Observatory site. Support for the operation of the MWA is provided by the Australian Government (NCRIS), under a contract with Curtin University administered by Astronomy Australia.
Limited. SJM acknowledges the contribution of an Australian Government Research Training Program Scholarship in supporting this research. DK acknowledges support from both the Curtin International Postgraduate Research Scheme and the NANOGrav project, which receives support from NSF Physics Frontier Center award number 1430284. This work was further supported by resources provided by the Pawsey Supercomputing Centre with funding from the Australian Government and the Government of Western Australia. We would also like to thank Ian Morrison, Franz Kirsten, and Clancy James for reviewing this manuscript and offering several helpful suggestions to improve it.

References

Bhat, N. D. R., Ord, S. M., Tremblay, S. E., McSweeney, S. J. & Tingay, S. J. 2016, ApJ, 818, 86
Bhat, N. D. R., et al., 2018, ApJS, 238
Brentjens, M., & de Bruyn, A., 2005, A&A, 441, 1217
Cordes, J. 1981, in Pulsars: 13 Years of Research on Neutron Stars, 115, http://adsabs.harvard.edu/full/1981IAUS...95.115C
Craft, H. D., Comella, J. M., & Drake, F. D. 1968, Nature, 218, 1122
Croppi, R. E., & Rabiner, L. R. 1983, Multirate Digital Signal Processing. Prentice-Hall Signal Processing Series: Advanced Monographs (New Jersey: Prentice-Hall)
Farah, W., et al. 2018, MNRAS, 478, 1209
Hankins, T. H., Kern, J. S., Weatherall, J. C., & Eilek J. A. 2003, Nature, 422, 141
Harris, C., & Haines K. 2011, PASA, 28, 317
Harris, F. J. 2004, Multirate Signal Processing For Communication Systems (New Jersey: Prentice-Hall)
Hessels, J. W. T., et al. 2019, ApJ, 876, L23
Hewish, A., Bell, S. J., Pilkington, J. D. H., Scott, P. F., & Collins, R. A. 1968, Nature, 217, 709
Hobbs, G., & Dai, S. 2017, 10.1093/nat/snx126
Kaur, D., et al. 2019, ApJ, 882, 133
Kirsten, F., Bhat, N., Meyers, B., Macquart, J. P., Tremblay, S., & Ord, S. 2019, ApJ, 874, 179
Kuzmin, A. D., Hamilton, P. A., Shirto, Y. P., McCulloch, P. M., McConnell, D., & Pugatchev, V. D. 2003, MNRAS, 344, 1187
McSweeney, S. J., Bhat, N. D. R., Tremblay, S. E., Deshpande, A. A., & Ord, S. M. 2017, ApJ, 836, 224
Meyers, B. W., et al. 2018, ApJ, 869, 134
Navarro, J., Manchester, R. N., Sandhu, J. S., Kulkarni, S. R., & Bailes, M. 1997, ApJ, 486, 1019
Noutsos, A., et al. 2015, A&A, 576, A62
Oppenheim, A. V., & Schafer, R. W. 2009, Discrete-Time Signal Processing (3rd edn.; New Jersey: Prentice-Hall)
Ord, S. M., Tremblay, S. E., McSweeney, S. J., Bhat, N. D. R., Sobey, C., Mitchell, D. A., Hancock, P. J., & Kirsten, F. 2019, PASA, 36, e030
Oronsay, S. I., et al. 2015, ApJ, 809, 51
Popov, M. V., Bartel, N., Cannon, W. H., Novikov, A. Y., Kondratiev, V. I., & Altmun, V. I. 2002, A&A, 396, 171
Prabu, T., et al. 2015, Experimental Astronomy, 39, 73
Tingay, S. J., et al. 2013, PASA, 30, 7
Tremblay, S. E., et al. 2015, PASA, 32, e005
Wayth, R. B., et al. 2018, PASA, 35, 33
Whitney, A., Kettenis, M., Phillips, C., & Sekido, M. 2009, in 8th International e-VLBIWorkshop. http://adsabs.harvard.edu/abs/2009evbl.confE.42W
Xue, M., Ord, S. M., Tremblay, S. E., Bhat, N. D. R., Sobey, C., Meyers, B. W., McSweeney, S. J., & Swainston, N. A., 2019, PASA, 36, e025
Yan, W. M., et al. 2011, Ap&SS, 335, 485
van Straten, W., & Bailes M. 2011, PASA, 28, 1

A. The FPGA implementation of the fine PFB

Each coarse channel is independently processed by dedicated FPGAs to convert (5-5)-bit complex integers sampled at 1.28 MHz into a series of spectra composed of 128×(4+4)-bit complex integers sampled at 10 kHz. The PFB is implemented in two stages: (1) a ‘front-end’ stage which prepares the array b[n] (see Equation (3)), and (2) the fast Fourier transform stage, which calculates the 128-point spectrum, X[n][m].

The first stage involves a multiplication of a window of 1 536 consecutive coarse channel samples with the analysis filter shown in Figure 2, and then a summation of the 12 taps together to produce a single array of 128 complex integers. The combination of the allowed input values \([-16, 15]\) and the filter values guarantees that the magnitudes of the summed (signed) numbers do not exceed 2^{21} = 20 97 152. At this stage of the processing, they are stored as 48-bit signed integers, of which only the bottom 22 bits are therefore significant. Each integer n (either real or imaginary) is then reduced from 48 bits to 8 bits in the following manner. If n is positive, then bits 14 through 21 (counting from the least-significant bit) are selected to form the 8-bit integer, and 1 is subsequently added if bit 13 is 1. This is equivalent to rounding the number n/2^{14} to the nearest integer, where fractional values of 0.5 are always rounded up. If n is negative, then bits 14 through 21 are selected, but no rounding occurs. This is equivalent to applying the floor operation to n/2^{14}. It should be noted that this rounding scheme introduces a bias into the distribution of values. In particular, the distribution of 8-bit values has a different mean than the distribution of the original 5-bit values, and it results in an

\[
\begin{align*}
\text{Symmetric rounding} & \quad \text{FPGA rounding} \\
\end{align*}
\]

\[
\begin{align*}
\text{Normalized frequency} & \quad \text{8-bit pre-FFT value} \\
\end{align*}
\]

Figure A1. The first-stage (i.e. pre-FFT) rounding scheme implemented in the MWA FPGAs compared with a symmetric scheme. The top panel illustrates the two rounding schemes (each has been given a slightly different y-offset for visual clarity). A population of 5-bit samples was drawn from a rounded Gaussian distribution with mean \(\mu = 0\) and standard deviation \(\sigma = 2\). The vertical dashed lines show the means of the distributions after the population has been subjected to the first stage of the PFB (i.e. converted to 8-bit integers) and rounded according to the two different schemes. The bottom panel compares the distributions of the resulting population of 8-bit samples.
artificial deficit in the number of values that get rounded to zero. The rounding scheme and its effect on the distribution are illustrated in Figure A1, including a displaced mean which artificially adds power to the DC bin of the spectrum.

The second stage calculates the spectrum of $b[n]$ using the standard fixed point FFT algorithm that is implemented on (Xilinx Virtex4 XC4VSX35) FPGAs. The output values are scaled and (symmetrically) rounded, producing (4+4)-bit output values.