

# A framework for assessing the performance of pulsar search pipelines

E. van Heerden,<sup>1</sup>★ A. Karastergiou<sup>2,3,4</sup> and S. J. Roberts<sup>1</sup>

<sup>1</sup>Information Engineering, University of Oxford, Parks Road, Oxford OX1 3PJ, UK

<sup>2</sup>Astrophysics, University of Oxford, Denys Wilkinson Building, Keble Road, Oxford OX1 3RH, UK

<sup>3</sup>Physics Department, University of the Western Cape, Cape Town 7535, South Africa

<sup>4</sup>Department of Physics and Electronics, Rhodes University, PO Box 94, Grahamstown 6140, South Africa

Accepted 2016 November 22. Received 2016 November 22; in original form 2016 June 14

## ABSTRACT

In this paper, we present a framework for assessing the effect of non-stationary Gaussian noise and radio frequency interference (RFI) on the signal-to-noise ratio, the number of false positives detected per true positive and the sensitivity of standard pulsar search pipelines. The results highlight the necessity to develop algorithms that are able to identify and remove non-stationary variations from the data before RFI excision and searching is performed in order to limit false-positive detections. The results also show that the spectrum whitening algorithms currently employed, severely affect the efficiency of pulsar search pipelines by reducing their sensitivity to long-period pulsars.

**Key words:** methods: analytical – methods: data analysis – methods: statistical – pulsars: general.

## 1 INTRODUCTION

Pulsars provide a wealth of information about neutron star physics, the interstellar medium and stellar evolution (Lorimer & Kramer 2005). Furthermore, their clock-like properties allow for sensitive measurements of their orbital dynamics that are used to constrain the equation of state of ultra-dense matter (Hessels et al. 2006; Demorest et al. 2010), probe the physics of binary evolution and test the predictions of general relativity (Antoniadis 2014). The continued discovery of new pulsars through pulsar surveys is paramount if we are to improve our understanding of the radio pulsar population as well as expand research in the aforementioned areas. Consequently, pulsar surveys remain a driving force in the field of astrophysics.

Pulsar research has in the past been driven by a number of large-scale surveys carried out with various radio telescopes. Surveys of the Galactic plane (Manchester et al. 2001; Johnston et al. 1992), supernova remnants (Seward & Harnden 1982), globular clusters (Manchester et al. 1991) and all-sky surveys (Manchester et al. 1996; Cordes et al. 2006a) have led to the discovery of more than 2200 pulsars.

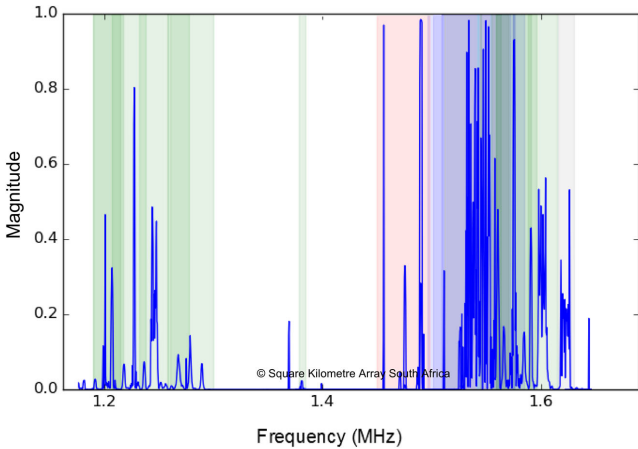
Pulsar population synthesis models (Lorimer 2011), based on pulsar surveys and the known pulsar population, are used to predict the number of pulsars expected to be discovered in future pulsar surveys (Lorimer et al. 2006; Bates et al. 2014). These techniques are also used to estimate the number of potentially detectable (i.e. those that are beaming towards us as well as being luminous enough) normal pulsars and millisecond pulsars (MSPs) in the Galaxy.

The number of pulsars actually discovered in recent surveys (Swiggum et al. 2014; Lazarus et al. 2015) has fallen well short of the number predicted by the aforementioned estimation techniques. It was predicted that the Arecibo PALFA Precursor survey (Swiggum et al. 2014) should have detected  $490^{+160}_{-115}$  normal pulsars and  $12^{+70}_{-5}$  MSPs by the beginning of 2014, but managed to detect only 283 normal pulsars and 31 MSPs. The full PALFA survey, when complete, is expected to have detected  $1000^{+330}_{-230}$  normal pulsars and  $30^{+200}_{-20}$  MSPs. However, close to completion it has only managed to detect  $\sim 443$  normal pulsars and  $\sim 40$  MSPs, respectively (Lazarus et al. 2015). It is worth noting that the largest discrepancy between predictions and detections is for normal pulsars, i.e. pulsars with long periods. Furthermore, it is estimated that there are between 82 000 and 143 000 detectable normal pulsars and 9000–100 000 detectable MSPs in the Galactic disc alone (Lorimer et al. 2006; Swiggum et al. 2014), yet to date we have only discovered some 2200 pulsars (Hobbs et al. 2004). The deficiencies in pulsar detections have been attributed to radio frequency interference (RFI) and scintillation (Lorimer 2011), both these effects are not addressed in current population synthesis models (Levin et al. 2013).

Electromagnetic radiation with frequencies between circa 10 kHz and 100 GHz is referred to as radio frequency (RF). RFI in the context of a pulsar survey is any signal or disturbance emitted from a man-made source either extra-terrestrial or terrestrial that corrupts the measurements of data obtained. The spatial and temporal variabilities of RFI make it difficult to identify and to mitigate. If RFI is not dealt with then spurious trends may occur in the data collected, thereby decreasing the signal-to-noise ratio (SNR) and making it more difficult, or impossible, to detect new pulsars.

All-sky pulsar surveys, such as the Arecibo PALFA survey, are more often than not conducted in the  $L$  band (the 1–2 GHz range

★ E-mail: [elmarie.vanheerden@eng.ox.ac.uk](mailto:elmarie.vanheerden@eng.ox.ac.uk)



**Figure 1.** Typical spectrum occupancy in the  $L$  band. *Source:* Square Kilometre Array South Africa.

of the radio spectrum), more specifically the frequency range 1.2–1.6 GHz (Burgay et al. 2006; Lazarus et al. 2015). The frequency range 1.2–1.6 GHz happens to overlap with frequencies that have been earmarked for other applications such as satellite navigation, telecommunication, aircraft surveillance, amateur radio and digital audio broadcasting (Regulations 2008). Most of the aforementioned RFI sources severely decrease the sensitivity of surveys conducted in the  $L$  band.

Spectrum occupancy in the  $L$  band, depicted in Fig. 1, is dominated by RFI mainly from satellites. The colours in Fig. 1 represent interference from different satellites: red – Afristar, yellow – Thuraya, blue – Inmarsat, cyan – Satellite Radio, grey – IRIDIUM, green – {Galileo, Beidou, GPS, GLONASS} and grey – {Fengun, Meteosat}.

In a recent study by Lazarus et al. (2015), synthetic pulsars with various periods and pulse widths were injected into actual PALFA survey data with the aim to assess the effect of RFI and red noise<sup>1</sup> on the survey sensitivity. The study found that there is a significant degradation in the sensitivity of between 10 per cent and a factor of 2 for pulsars with spin periods between 0.1 and 2 s and dispersion measure (DM)  $> 150 \text{ pc cm}^{-3}$  due to red noise induced by RFI, receiver gain fluctuations and opacity variations of the atmosphere. Additionally, a population synthesis analysis based on the empirical survey sensitivity found that  $35 \pm 3$  per cent of pulsars, with predominantly long periods, are missed compared to expectations that are based on the theoretical sensitivity curves as derived from the radiometer equation. With these results, the authors conclude that the reduced sensitivity to long-period pulsars is mainly attributed to red noise. All the results were obtained despite applying a red-noise suppression algorithm.

In this paper we show, supplementary to the Lazarus et al. (2015) study, that frequency-dependent noise such as red noise indeed reduces the SNR of long-period pulsars and increases the number of false detections (Lyon et al. 2016). Moreover, we offer an explanation as to how the red-noise suppression technique in the Lazarus et al. (2015) paper actually contributed to the reduced sensitivity of long-period pulsars by explaining what the algorithm does and quantifying the loss of SNR when the algorithm is applied.

<sup>1</sup> Red noise is a type of signal noise with a power spectral density inversely proportional to  $f^2$ , which means it has more energy at lower frequencies.

It is evident, from the number of pulsars missed, that RFI and frequency-dependent noise greatly affect the sensitivity of radio telescopes to normal pulsars, i.e. pulsars with long periods. Therefore, the aims of this study are:

- (i) to quantify the effect that non-stationary Gaussian noise and RFI have on the performance of pulsar search pipelines;
- (ii) to examine the effectiveness of the current spectrum whitening methods available in pulsar search software suites;
- (iii) to determine if detrending the data with a moving average filter before searching for pulsars is effective;
- (iv) to examine the effectiveness of the current RFI detection and mitigation methods available in pulsar search software suites;
- (v) to investigate the reduction in sensitivity as a function of both the correlation length of the non-stationary noise and the pulse period.

We start in Section 2.2 by describing the building blocks of a typical pulsar search pipeline for normal pulsars. There exist various software implementations of this pipeline. The ones used in this analysis are introduced in Section 2.3. Although these software packages differ in a number of ways, the way in which they deal with frequency-dependent noise is of particular interest for this study. Therefore, in Section 2.4, the different spectrum whitening algorithms available in the software packages considered are mathematically described. The method used for generating the synthetic filterbank files with non-stationary Gaussian noise is detailed in Section 3.1.1, and the RFI added to some of the files can be found in Section 3.1.2. In Sections 3.2.1–3.2.4, we introduce the experimental framework we used for generating and processing filterbank files containing different noise processes.

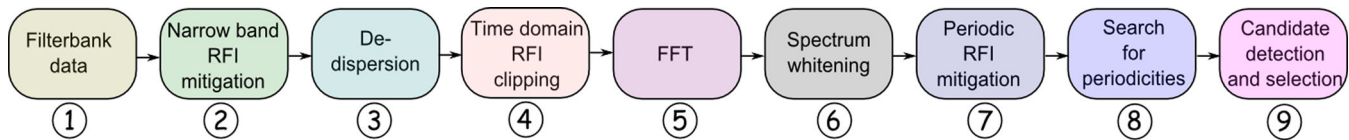
The aim of this framework is to assess the ability of different pulsar search pipelines to detect pulsars embedded in non-stationary Gaussian noise amidst RFI. It is worth noting that this study differs from the Lazarus et al. (2015) study in that the aim is to quantify the sensitivity of different pulsar search pipelines as a function of noise correlation length and pulsar spin period, whereas the latter aimed at quantifying the Arecibo PALFA survey’s sensitivity as a function of DM and pulsar spin period. Finally, the results, discussion and conclusions can be found in Sections 4–6.

## 2 SEARCH PROCESS FOR NORMAL PULSARS

In order to determine how phenomena like RFI and non-stationary noise can hinder pulsar detections, it is necessary to first understand the nature of the acquired data and the functionality of each of the building blocks that form part of a pulsar search pipeline. Therefore, a detailed description of a typical pulsar search pipeline is presented in this section. Two existing software implementations of the pulsar search pipeline and a detailed algorithmic description of the spectrum whitening techniques available in each of them are also presented here. Different configurations of these software suites are used in a subsequent section to process pulsar data where the results will be used to assess their abilities to deal with non-stationary noise and RFI.

### 2.1 Search data

The data that are searched for pulsars are time series of total power per frequency channel typically referred to as filterbank data. The number of frequency channels, the temporal resolution and the dynamic range (i.e. 1, 8 or 16 bit) of the data are unique to each survey.



**Figure 2.** Schematic illustration of a typical pulsar search pipeline, see the text for details.

Files that contain filterbank data are currently processed offline; however, with the increase in scope and sensitivity of future surveys it will become infeasible to store the raw data for offline processing due to capacity and input/output constraints. Hence, there is a need for a paradigm shift from offline to real-time processing of survey data.

Real-time processing entails block-wise dedispersion for the purposes of rapid reporting of fast radio burst (FRB) detections, which constitute a byproduct of pulsar searches. The time duration of each block depends on the DM search and the observing frequency, and is likely to be of the order of a few seconds for typical searches. RFI cleaning must happen prior to dedispersion. Thereafter, all frequency information is lost and likewise the opportunity to detect and mitigate RFI in the time–frequency plane. The relevant time-scale on which any RFI excision technique needs to operate is therefore more likely to be related to the FRB detection buffer size, rather than the full integration required for periodicity searches.

## 2.2 Pipeline for a standard pulsar search

Detecting radio pulses produced by pulsars is an intrinsically difficult task due to their narrow duty cycles, low signal strengths, dispersion effects and the presence of non-Gaussian noise.

Numerous techniques have been developed to overcome some of the difficulties highlighted above. These techniques are combined to form the standard pulsar search pipeline.

The typical pipeline used for the processing of filterbank files consists of nine stages as depicted in Fig. 2. The pipeline starts with noisy filterbank files (Fig. 2.1) that may or may not contain a pulsar.

Each filterbank file is examined for narrow-band RFI signals, which are excised by replacing the affected samples with constant values chosen to match the median bandpass (Fig. 2.2; Lazarus et al. 2015).

The corrected filterbank files are then dedispersed for a number of trial DM values to compensate for the dispersion induced by the interstellar medium (Fig. 2.3; Lorimer & Kramer 2005). The zero-DM time series is used to identify and mitigate broad-band RFI (Fig. 2.4) that went undetected by the narrow-band RFI excision process. After mitigating broad-band RFI, the Fourier transform of each dedispersed time series is computed (Fig. 2.5).

The power spectrum is whitened (Fig. 2.6) so that the response is as uniform as possible, i.e. mitigating frequency-dependent noise, subtracting a running median and normalizing the local root mean square (rms) of the power spectrum such that it has a zero mean and unit rms. A whitened power spectrum is preferred, because estimating the significance level of any signal present is relatively easy. Different techniques have been implemented to whiten the spectrum, and these will be described in more detail in Section 2.4.

The next stage of the pipeline concerns identification of periodic RFI. Known periodic signals that are present all or most of the time, such as power lines carrying alternating current and communication systems such as airport radar systems are flagged with their harmonics and their bandwidths determined. These interferences are mitigated by creating a spectral mask (see Fig. 2.7). This mask

consists of a list of all the Fourier bins affected and which should be ignored in all subsequent processing.

Radio pulses from pulsars generally have narrow duty cycles that, in the Fourier domain, results in the power to be distributed between the fundamental frequency and a number of harmonics (van Heerden et al. 2014). Therefore, to take full advantage of the power contained in the harmonics the whitened spectrum is harmonically summed by adding the higher harmonics to the fundamentals. The original power spectra as well as the composite spectra formed by summing 2, 4, 8 and 16 harmonics (Lyne & Graham-Smith 2012) are each searched for periodicities (Fig. 2.8; Cordes et al. 2006a). The best candidates from each trial DM are saved.

After all the time series have been processed, a list of pulsar candidates is compiled. This list is pruned by post-processing procedures (Fig. 2.9) ranging from sifting and folding to sophisticated machine learning candidate selection (Lazarus et al. 2015). The most promising pulsar candidates are saved for future observation and follow-up (Cordes et al. 2006a).

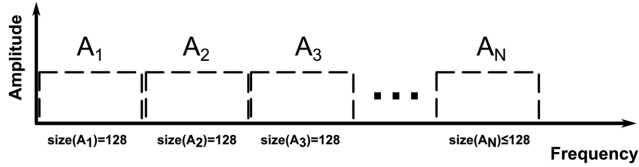
## 2.3 Pulsar search software

The pulsar search pipeline described above is available in a number of pulsar search software packages: SIGPROC developed by Lorimer (2001), PRESTO developed by Ransom (Ransom 2011), PEASOUP developed by Barr (2013) and PULSARHUNTER developed by Keith (2007). The two most frequently used packages are SIGPROC and PRESTO, both of which are freely available and well tested. Together they have been responsible for the discovery of most of the pulsars known today. We refer the interested reader to Cordes et al. (2006b), Rane et al. (2016), Stovall et al. (2014) and Lazarus et al. (2015) for a comprehensive discussion of how these two pipelines are typically used in real pulsar surveys.

There are three main differences between SIGPROC and PRESTO. First, the manner in which they search for pulsars that orbit a companion, namely binary pulsars. Radio pulses from binary pulsars typically exhibit Doppler shifts in their rotational period, caused by the acceleration of the pulsar around its companion. To be efficient, these shifts need to be accounted for in the so-called acceleration searches. PRESTO performs acceleration searches in the Fourier domain (Ransom 2001). SIGPROC on the other hand does time-domain re-sampling to carry out acceleration searches. Secondly, SIGPROC looks for harmonically related signals in the amplitude spectrum, whereas PRESTO uses the power spectrum to identify possible pulsar candidates (Lorimer & Kramer 2005). Lastly, SIGPROC uses SNR as a metric to identify peaks in the normalized power spectrum whereas PRESTO uses the Gaussian significance (adjusted for the number of trials searched) of the peaks as a metric under a white noise assumption. Hereinafter, the terms SNR and Gaussian significance shall be collectively referred to as *detection significance*.

## 2.4 Spectrum whitening

A stochastic process is considered white if and only if it is stationary and independent at all points. As a consequence, the power



**Figure 3.** Amplitude spectrum partitioning for the whitening algorithm implemented in SIGPROC, see the text for details.

spectral density of a white process is uniformly distributed across the whole available frequency range. A non-white process is instead characterized by a given distribution of the power per unit frequency along the available frequency bandwidth (Lazarus et al. 2015). A whitening operation on any non-white process entails forcing the said process to adhere to the conditions described above for a white process.

In the case of pulsar searching, well-behaved white noise is sought after, because it simplifies any attempt at estimating the significance levels of any signal present in the data and consequently makes detection easier. Hence, it is standard practice to whiten the power spectral density by suppressing frequency-dependant noise, in particular red noise, so that the response to noise is as uniform as possible.

The spectrum whitening techniques implemented in SIGPROC and PRESTO are similar in that they aim to normalize the spectrum. However, the way in which these techniques algorithmically operate in normalizing the spectrum is quite different. The different whitening options available in SIGPROC and PRESTO are mathematically described in the two subsequent sections.

#### 2.4.1 Spectrum whitening in SIGPROC

In SIGPROC, there are three spectrum whitening options for the SEEK function. In all three approaches, the spectrum is divided into blocks of  $\max\{128, (\text{numberofspectraldatapoints}/400\,000)\}$  Fourier bins. The simulation parameters of this analysis resulted in the number of Fourier bins per block to be consistently 128 and thus for all subsequent explanations we assume that the spectrum is divided into blocks of 128 Fourier bins as depicted in Fig. 3.

The algorithms implemented for the three options in SIGPROC are as follows:

##### (i) Option 1:

The default whitening algorithm executed when the function SEEK is called computes the mean and corrected sample standard deviation, for each of the blocks  $A_1, \dots, A_N$  in Fig. 3. The mean is subtracted from each Fourier bin in the block, whereafter the bins are scaled by the corrected sample standard deviation of that particular block. The algorithmic steps are detailed in Algorithm 1.

---

#### Algorithm 1 SIGPROC: default

---

```

1: for  $i = 1, \dots, N$  do
2:    $\mu_i = \text{mean}(A_i)$ 
3:    $s_i = \text{corrected standard deviation}(A_i)$ 
4:    $A_i^{\text{new}} = (A_i^{\text{old}} - \mu_i)/s_i$ 
5: end for

```

---

##### (ii) Option 2:

The whitening algorithm executed when the function SEEK is called with the flag `-submn` is identical to Algorithm 1 except for one difference. The blocks  $A_1, \dots, A_N$  in Fig. 3 are scaled with the rms

of that particular block instead of the corrected sample standard deviation.

##### (iii) Option 3:

The whitening algorithm executed when the function SEEK is called with the flag `-submjk` computes the mean and corrected sample standard deviation for each of the blocks  $A_1, \dots, A_N$  in Fig. 3. Thereafter, the gradients of the mean and corrected sample standard deviation between adjacent blocks of 128 Fourier bins are computed. For each Fourier bin  $j = 1, \dots, 128$  in a block the mean is subtracted and the result scaled with the corrected sample standard deviation, whereafter the mean and corrected sample standard deviation is updated with the calculated gradients for that particular block. The algorithmic steps are detailed in Algorithm 2.

---

#### Algorithm 2 SIGPROC: submjk

---

```

1: for  $i = 1, \dots, N$  do
2:    $\mu_i = \text{mean}(A_i)$ 
3:    $\mu_{i+1} = \text{mean}(A_{i+1})$ 
4:    $s_i = \text{corrected standard deviation}(A_i)$ 
5:    $s_{i+1} = \text{corrected standard deviation}(A_{i+1})$ 
6:    $\text{slope}_{\text{mean}_i} = (\mu_{i+1} - \mu_i)/128$ 
7:    $\text{slope}_{s_i} = (s_{i+1} - s_i)/128$ 
8:   for  $j = 1, \dots, 128$  in  $A_i$  do
9:      $A_i^{\text{new}}[j] = (A_i^{\text{old}}[j] - \mu_i)/s_i$ 
10:    Update:  $\mu_i = \mu_i + \text{slope}_{\text{mean}_i}$ 
11:    Update:  $s_i = s_i + \text{slope}_{s_i}$ 
12:   end for
13: end for

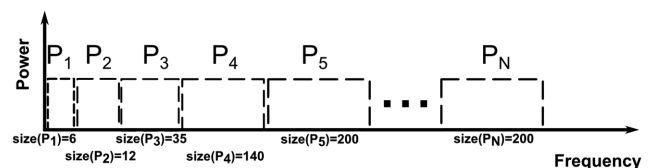
```

---

#### 2.4.2 Spectrum whitening in PRESTO

In PRESTO, there is only one spectrum whitening technique implemented to suppress frequency-dependent noise (Ransom, Eikenberry & Middleditch 2002). The median power level is measured in blocks across Fourier bins and then multiplied by  $\log 2$  to convert the median value to an equivalent mean level assuming that the powers are distributed exponentially. Thereafter, the measured mean power values (variable  $P$  in Algorithm 3) are used to compute the slope between two adjacent Fourier bins that in turn is used to normalize the complex Fourier amplitudes (variable  $A$  in Algorithm 3).

The number of Fourier frequency bins per block starts with 6 and increases logarithmically to 200 (see Fig. 4). Thus, for frequencies where there is little coloured noise, the number of bins used per block is 200. The algorithmic steps for the spectrum whitening technique implemented in PRESTO are detailed in Algorithm 3. Irrespective of the red-noise suppression algorithm being applied, PRESTO by default normalizes the power spectrum using median blocks before performing harmonic summing and searching.



**Figure 4.** Power spectrum partitioning for the whitening algorithm implemented in PRESTO, see text for details.

**Algorithm 3** PRESTO: default

---

```

1: for  $i = 1, \dots, N$  do
2:    $\mu_i = \text{median}(P_i)/\log 2$ 
3:    $\mu_{i+1} = \text{median}(P_{i+1})/\log 2$ 
4:    $\text{slope}_i = (\mu_{i+1} - \mu_i)/(\text{size}(P_i) + \text{size}(P_{i+1}))$ 
5:    $\text{lineoffset} = \frac{1}{2}(\text{size}(P_i) + \text{size}(P_{i+1}))$ 
6:   for  $j = 1, \dots, \text{size}(P_i)$  do
7:     Update:  $\text{lineval} = \mu_i + \text{slope}_i \times (\text{lineoffset} - j)$ 
8:     Update:  $\text{scaleval} = 1/\sqrt{\text{lineval}}$ 
9:     Update:  $\mathbf{Re}(A)_i^{\text{new}}[j] = \mathbf{Re}(A)_i^{\text{old}}[j] \times \text{scaleval}$ 
10:    Update:  $\mathbf{Im}(A)_i^{\text{new}}[j] = \mathbf{Im}(A)_i^{\text{old}}[j] \times \text{scaleval}$ 
11:   end for
12: end for
    
```

---

### 3 METHODOLOGY

Pulsar software currently available for generating synthetic pulsar data as well as the detection and timing thereof assumes that the noise present in the signals acquired by radio telescopes is additive white Gaussian noise. This assumption ignores the fluctuating nature of the sky temperature (Nice, Fruchter & Taylor 1995; Lorimer & Kramer 2005) as well as the effects that RFI have on the noise baselines of the data. Consequently, it is poorly understood how the aforementioned phenomena, which are clearly non-stationary, affect the ability of pulsar search pipelines to detect pulsars. Hence, the need for software to emulate these phenomena (see Section 3.1) and a framework whereby their effect on the ability of pulsar search pipelines to detect pulsars can be investigated and quantified (see Section 3.2).

#### 3.1 Synthetic file generation

In Section 3.1.1, we present the low-pass filter that was inspired by a Gaussian Process (Rasmussen & Williams 2006) to generate synthetic filterbank files with non-stationary noise baselines. Additionally, in Section 3.1.2, we describe the choice of RFI that we injected into a subset of the synthetic filterbank files.

##### 3.1.1 Non-stationary Gaussian noise

Filterbank files contain quantized power values computed by superimposing tens or even hundreds of single Nyquist power measurements (see equation 1). The power measurement of a single Nyquist sample is computed from the real and imaginary parts of the raw voltages associated with either linear or circular polarized electromagnetic waves acquired by radio telescopes. The power measurement of a single Nyquist sample is given as

$$\text{Power} = X_{\text{real}}^2 + X_{\text{imag}}^2 + Y_{\text{real}}^2 + Y_{\text{imag}}^2, \quad (1)$$

where  $X$  and  $Y$  are either the horizontal and vertical components of linear polarization or the left- and right-handed components of circular polarization.

The power values found in filterbank files comprise both signal and noise. The noise levels in the filterbank files are proportional to the overall system temperature that is affected by RFI, the sky temperature and the receiver temperature. During an observation, various objects and RFI with different brightness levels are encountered so the duration and magnitude of the non-stationarity associated with each of these phenomena differ greatly.

In order to generate a time series with correlated samples, i.e. a varying noise baseline, we constructed a low-pass filter (see

equation 2) and convolved it with random samples drawn from a Gaussian distribution with zero mean and unit variance ( $\mathcal{N}(0, 1)$ ) (see equation 3), where  $\epsilon = 1 \times 10^{-5}$ ,  $t$  is the sampling interval and  $N$  is the number of samples in the observation. Consequently, the convolution yields a vector,  $\mathbf{w}$  with samples correlated over length-scales defined by  $\lambda$  and magnitudes proportional to  $h$  (see equation 4).

$$\mathbf{u} := h^2 \exp \left[ -\left( \frac{t}{\lambda} \right)^2 \right] \forall t \in \mathbb{R} \quad \text{such that} \quad \mathbf{u} > \epsilon \quad (2)$$

$$\mathbf{v} := z_1, z_2, \dots, z_N \sim \mathcal{N}(0, 1) \quad (3)$$

$$\mathbf{w} = \mathbf{u} * \mathbf{v} \quad (4)$$

In order to generate  $N$  data points that are correlated over long length-scales (i.e.  $\lambda \gg$ ) requires  $N$  random samples drawn from a standard normal distribution ( $\mathcal{N}(0, 1)$ ) to be convolved with a finely sampled low-pass filter that is compact on a large interval. Consequently, convolving two large vectors is computationally very expensive. To circumvent this challenge, we generate data points with the required correlation length by convolving a fraction of the random samples drawn from a standard normal distribution with a coarsely sampled low-pass filter and then interpolating between the resultant points to produce a time series with the desired number of points.

The vector  $\mathbf{w}$  generated by convolving the low-pass filter with samples drawn from a standard normal distribution is not always positive. However, for the purpose of these experiments non-negative samples are desired because the standard deviation of the baseline drift needs to be proportional to the square root of the mean. Therefore, a new vector  $\mathbf{g}$  is defined:

$$\mathbf{g} = \mathbf{w} - \min(\mathbf{w}), \quad (5)$$

such that the offset is equal to zero and all the values are non-negative.

The mean vector  $\mathbf{g}$  is used to generate samples for the vectors  $X_{\text{real}}, X_{\text{imag}}, Y_{\text{real}}$  and  $Y_{\text{imag}}$ :

$$X_{\text{real}}, X_{\text{imag}}, Y_{\text{real}}, Y_{\text{imag}} \stackrel{i.i.d.}{\sim} \mathcal{N}(\mathbf{g}, \sqrt{\mathbf{g}}), \quad (6)$$

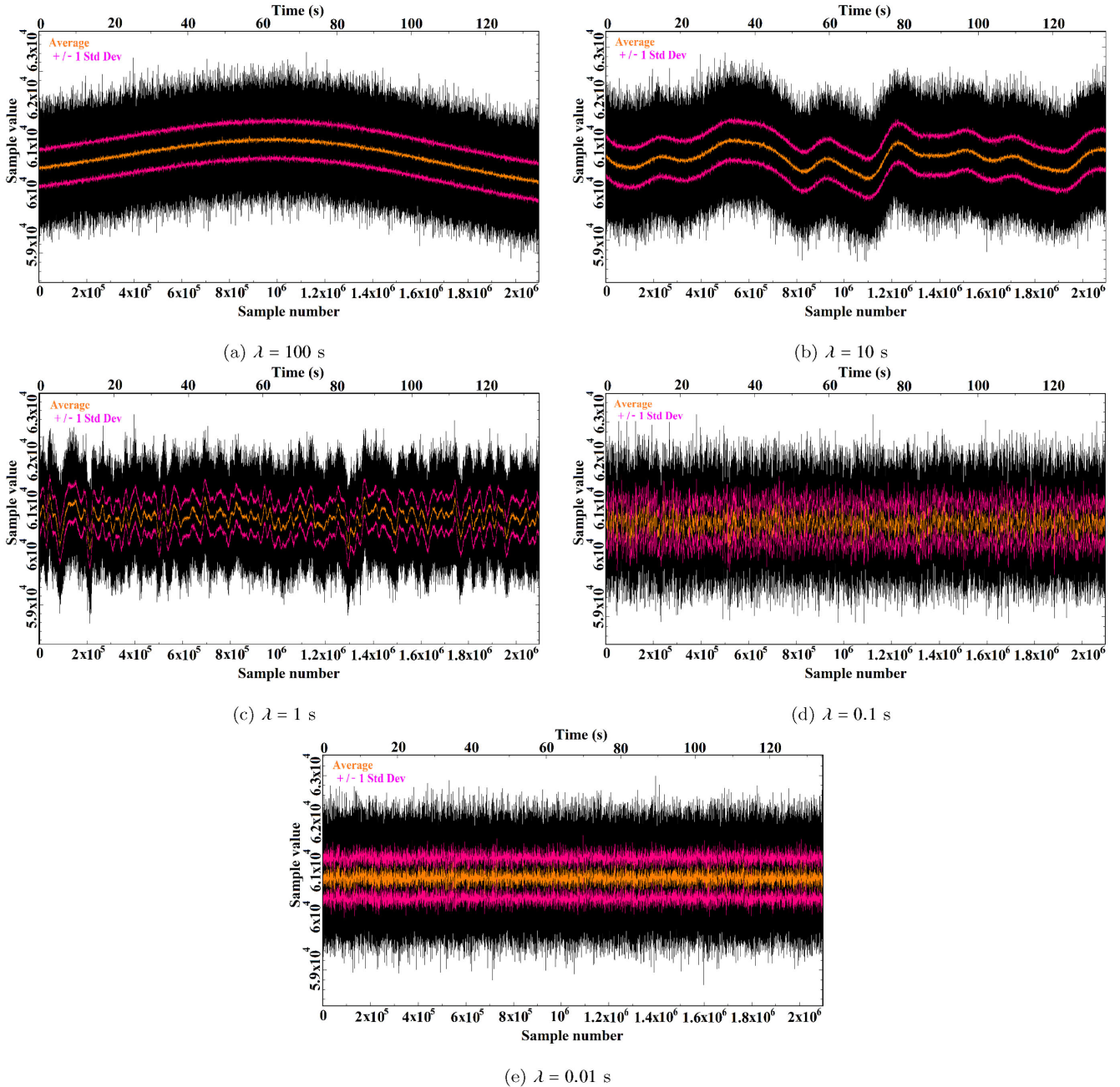
such that the power for each sample in the synthetic filterbank file can be computed using equation (1).

A large value for the length-scale variable  $\lambda$  of the low-pass filter results in a slow drifting baseline as depicted in Fig. 5(a). As the value of  $\lambda$  decreases the baseline drifts becomes more capricious as depicted in Figs 5(b)–(e).

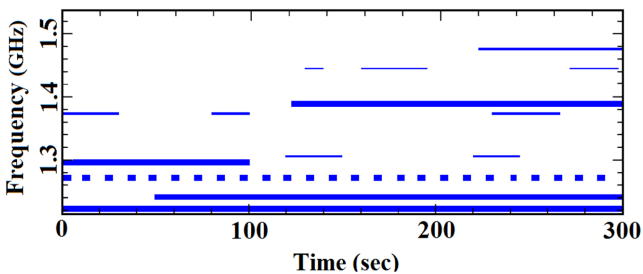
##### 3.1.2 RFI injected

For the experiments aimed at investigating the effects of RFI on the performance of pulsar search pipelines, we injected the same RFI into all the filterbank files (see Fig. 6). The choice of injected RFI was obtained by studying spectrum occupancy data from the KAT-7 radio telescope over several months (see Fig. 1) and cross-correlating that with the  $L$ -band spectrum allocation as determined by the International Telecommunication Union (Regulations 2008). The RFI injected include:

- (i) broad-band periodic RFI with a period of 0.02 s and a duty cycle of 50 per cent,
- (ii) narrow-band periodic RFI with a period of 12 s and a duty cycle of 25 per cent affecting the frequency channels 1.266–1.276 GHz,



**Figure 5.** Examples of dedispersed time series that correspond to the five correlation lengths used to simulate non-stationary noise processes. Black represents the actual signal in the filterbank file, orange the mean and pink the standard deviation ( $1\sigma$ ) of the dedispersed time series. The correlation length decreases from (a) 100 s to (e) 0.01 s.



**Figure 6.** The RFI injected into each filterbank file.

(iii) several instances of narrow-band RFI with random durations affecting the frequency channels identified from the RFI characterization plot in Fig. 1.

Various routines exist in current pulsar search pipelines for excising bright RFI, but the effect of weak and unknown sources of RFI is unbeknownst to us. Therefore, the magnitudes of the various instances of injected RFI were deliberately chosen to be within  $1\sigma$  of the baseline such that the effect of weak RFI on pulsar search pipelines could be investigated. The percentage of samples affected by RFI is 12 per cent for each filterbank file.

**Table 1.** Simulated observation parameters.

Parameter	Value
$t_{\text{obs}}$	300s
$t_{\text{samp}}$	64 $\mu\text{s}$
$n_{\text{bits}}$	8
$n_{\text{chans}}$	512
$f_{\text{low}}$	1214 MHz
$f_{\text{high}}$	1536 MHz
Bandwidth, $\Delta f$	322 MHz
Channel bandwidth, $\Delta f_{\text{chan}}$	628.91 kHz

### 3.2 Framework for pulsar search pipeline analysis

In this section, a framework to generate and process non-stationary noise files with RFI is introduced. This framework allows for the understanding of how non-stationary noise processes with different correlation lengths can impede the detection of pulsars with specific periods. Additionally, it contributes to the understanding of how RFI can pass undetected through current pulsar search pipelines and the consequences of not mitigating these spurious sources of interference.

The generation part of this framework includes the synthetic observation parameters (see Section 3.2.1), the experimental design specifics for each experiment (see Section 3.2.2) and the periods of the pulsars injected for this analysis (see Section 3.2.3).

The processing part of this framework includes the different configurations of SIGPROC and PRESTO that were analysed (see Section 3.2.4).

#### 3.2.1 Simulated observation parameters

The observation parameters that were used for generating the synthetic filterbank files were chosen to match the Arecibo PALFA survey (Lazarus et al. 2015) parameters and are summarized in Table 1.

#### 3.2.2 Experiments

The experiments that we designed for analysing various configurations of SIGPROC and PRESTO are summarized in Table 2.

The design of experiments 1–4 is such that each one emulates a blind pulsar survey. Each experiment comprises 100 simulated pointings with a subset of these containing a pulsar. The differences between the experiments are the type of noise processes simulated and whether RFI is present or not.

Experiment 1 comprises 100 files with stationary Gaussian noise. Fifteen of the 100 files contain an injected pulsar (see Table 3) and the remainder are without it. The results from experiments 2–4 will be benchmarked against the results of experiment 1 because of its idealized noise process and lack of RFI.

Experiment 2 comprises 100 files with non-stationary Gaussian noise. Fifteen of the aforementioned files contain an injected pulsar that is unique (see Table 3) and the remainder of the files are without a pulsar. The non-stationary Gaussian noise processes were generated according to the procedure described in Section 3.1.1. Note that every noise process is unique in that each one is defined by a different non-stationary vector,  $\mathbf{g}$ , and the additive Gaussian noise has zero mean and variance proportional to the square root of the non-stationary vector (see equation 6). The length-scales,  $\lambda$ , for the non-stationary variability of the noise baselines range from  $10^{-2}$  to  $10^2$  s in factors of 10, i.e. 20 files were generated per  $\lambda$ . Consequently, each file exhibits a unique variation because of the stochastic nature of the generation process despite having the same correlation length.

The correlation lengths can be chosen to represent any time-scale that we could consider to have an effect on the survey sensitivity to periodic pulsars, and could result from instrumental variability, to environmental effects and RFI. The power spectrum of a non-stationary noise baseline with a given correlation length will contain more power in the frequencies that correspond to that length. For this reason, we choose our length-scales to sample a broad range that is relevant to the pulse periods searched, as mentioned above. Comparing the results from experiment 2 with the results of experiment 1 enables the quantification of the effect that non-stationary Gaussian noise has on the performance of pulsar search pipelines. Moreover, experiment 2 allows for the determination of the effectiveness of the spectrum whitening techniques described in Section 2.4 and whether or not detrending the data with a moving average filter before searching for pulsars is effective.

Experiment 3 is identical to experiment 1 except for the addition of RFI (see Section 3.1.2). The experimental design of experiment 3 serves to investigate the ramifications when weak RFI (see Section 4.4) passes undetected through a pulsar search pipeline. Furthermore, it serves to investigate the efficacy of RFI detection and mitigation algorithms currently employed.

Experiment 4 is identical to experiment 2 except for the addition of RFI (see Section 3.1.2). Comparing the results from experiment 4 to the results of experiments 1–3, respectively, serves to quantify the combined effect that non-stationary Gaussian noise and RFI have on the performance of pulsar search pipelines and to deduce which phenomenon has the greatest impact on the said performance.

Experiment 5 comprises 75 files in total, 15 files per correlation length  $\lambda$  (see Table 2). Each one of the pulsars in Table 3 was separately injected into the 15 files with the same correlation length and this was repeated for all five values of  $\lambda$ .

Experiment 6 is identical to experiment 5 except for the addition of RFI (see Section 3.1.2). The experimental design of experiments 5 and 6 serves to investigate the reduction in sensitivity of pulsar search pipelines as a function of both the correlation length of the non-stationary noise and the pulse period of a pulsar.

**Table 2.** Summary of the experiments conducted.

Experiment	No. of Files	No. of pulsars	Noise	$h$	$\lambda$ (s)
1. Stationary	100	15	Stationary Gaussian	–	–
2. Non-stationary	100	15	Non-stationary Gaussian	0.1, 0.2, 0.3, 0.4	0.01, 0.1, 1.0, 10.0, 100.0
3. Stationary + RFI	100	15	Stationary Gaussian	–	–
4. Non-stationary + RFI	100	15	Non-stationary Gaussian	0.1, 0.2, 0.3, 0.4	0.01, 0.1, 1.0, 10.0, 100.0
5. All pulse periods per $\lambda$	75	75	Non-stationary Gaussian	0.1, 0.2, 0.3, 0.4	0.01, 0.1, 1.0, 10.0, 100.0
6. All pulse periods per $\lambda$ + RFI	75	75	Non-stationary Gaussian	0.1, 0.2, 0.3, 0.4	0.01, 0.1, 1.0, 10.0, 100.0

**Table 3.** Synthetic pulsar properties.

Parameter	Value			
Period(ms)	1.102	2.218	5.218	10.870
	18.505	61.965	126.175	286.555
	533.320	850.158	1657.496	2643.410
	3927.013	5580.899	9964.532	–
Amplitude	All pulsars are detectable with a detection significance of $\sim 12$ in SIGPROC in the presence of stationary Gaussian noise when processed with pipeline H in SIGPROC.			
Duty cycle	12 per cent (fixed)			
Dispersion measure	68			

**Table 4.** The 12 SIGPROC pipeline configurations used to process all the files in this analysis.

Pipeline	Baseline	Red-noise mitigation
A	Removed	default
B	Removed	submn
C	Removed	submjk
D	Removed	–
E	Intact	default
F	Intact	submn
G	Intact	submjk
H	Intact	–
I	Moving average filter	default
J	Moving average filter	submn
K	Moving average filter	submjk
L	Moving average filter	–

Lastly, experiments 2–6 are repeated for four different values of the magnitude parameter  $h$  defined in equation (2), namely  $h = 0.1, 0.2, 0.3, 0.4$ .

### 3.2.3 Pulsar properties

The properties of the 15 pulsars that were randomly injected into the synthetic filterbank files were taken in part from the Arecibo sensitivity study (Lazarus et al. 2015) and are summarized in Table 3.

A pulse with the profile of pulsar PSR B0833-45 at 1.4 GHz obtained from the EPN data base (Lorimer et al. 1998) with a duty cycle of 12 per cent was injected into all the files.

### 3.2.4 Pulsar search pipeline configurations

All of the files generated for the experiments described in Table 2 were processed by both SIGPROC and PRESTO, i.e. 12 different configurations of SIGPROC (see Table 4) and eight different configurations of PRESTO (see Table 5) were used to search every single synthetic file. Because the aim with this analysis is not to investigate sensitivity as a function of DM, all the files were dedispersed at the correct DM.

SIGPROC by default removes the baseline of a dedispersed time series by linearly detrending the time series unless the flag `-nobaseline` is set when the `dedisperse` function is called. In addition to the option available in SIGPROC for detrending the baseline, a 10 s moving average filter was implemented as a second option for normalizing the baseline of a dedispersed time series. The red-noise mitigation techniques applied in the different SIGPROC pipelines are described in Section 2.4.

To process all the files with SIGPROC, the following functions and their associated flags were called:

**Table 5.** The eight PRESTO pipeline configurations used to process all the files in this analysis.

Pipeline	RFI mask	Moving average filter	Red-noise mitigation
A	X	X	X
B	X	X	✓
C	X	✓	X
D	X	✓	✓
E	✓	X	X
F	✓	X	✓
G	✓	✓	X
H	✓	✓	✓

(i) function `dedisperse` with the flags `-d`, `-o` and `with/without -nobaseline`,

(ii) function `seek` (number of summed harmonics is 16) with the flags `-z` and `-submn/-submn/-submn`,

(iii) function `best` with flag `-s8`.

The function `best` in SIGPROC produces a ‘.lis’ file that was searched for possible candidates based on the SNR of the peaks.

The RFI mask configuration option in Table 5 refers to the RFI mask computed in PRESTO when the `rfifind` function is called. In this analysis, an RFI mask was computed and applied to each synthetic filterbank file at integration intervals of 8 s. An integration time of 8 s was chosen to resemble typical real-time processing intervals. The default values for the time and frequency rejection thresholds in the `rfifind` function were selected. A moving average filter of 10 s was also implemented as a processing step in the PRESTO pipelines. Lastly, details of the red-noise mitigation technique in PRESTO can be found in Section 2.4.

To process all the files with PRESTO, the following functions and their associated flags were called:

(i) function `prepdata` with the flags `-dm`, `-o` and `with/without the flag -mask`,

(ii) function `realfft`,

(iii) function `zapbirds` with the flags `-zap` and `-zapfile`,

(iv) function `accelsearch` with the flags `-sigma 1.0`, `-flo 0.1`, `-zmax 0` (acceleration searching was turned off by setting the flag `-zmax 0`) and `-numharm 16` (i.e. the number of summed harmonics is 16).

The `accelsearch` function in PRESTO produces an ACCEL file that was searched for possible candidates based on the Gaussian significance of the peaks under the assumption of pure white noise.

## 4 RESULTS

The results are organized according to the aims set forth in Section 1 of this paper.

The heuristics used for quantifying the results are as follows:

(i) A signal is considered a candidate if its detection significance is greater than the default detection threshold in SIGPROC or PRESTO.

(ii) Injected pulsars are considered discovered when (a) holds true AND if the difference between the periods of the discovered and injected signals are less than the allowed *error*, i.e.:

$$|\text{Period}_{\text{discovered}} - \text{Period}_{\text{injected}}| \leq \text{error}, \text{ where}$$

(a)  $\text{error} = 0.02$  ms if  $\text{period} \leq 10$  ms

(b)  $\text{error} = 0.20$  ms if  $10 \text{ ms} < \text{period} \leq 100$  ms

- (c)  $error = 2.00$  ms if  $100 \text{ ms} < \text{period} \leq 1000$  ms
- (d)  $error = 20.0$  ms if  $1000 \text{ ms} < \text{period} \leq 10\,000$  ms.

(iii) The discoveries from (b) are validated by visual inspection of their folded profiles produced by folding the inverse Fourier transform of their whitened spectra at the detected periods. At this stage, a detection is rejected if the folded profile does not resemble a real pulsar.

(iv) The non-fundamental harmonics were not considered when determining whether a pulsar was detected or not.

(v) Harmonically related candidates are removed.

(vi) Only candidates with  $1 \text{ ms} \leq \text{period} \leq 10\text{s}$  are considered.

For the direct SIGPROC and PRESTO comparisons, the exact same files were searched by both routines.

The sensitivity of a pipeline refers to the ability of the pipeline to detect the 15 randomly injected pulsars expressed as a percentage. The number of false positives detected per true positive is the total number of false-positive candidates detected across all 100 files divided by the number of true positives detected.

Note that all the results presented here should be interpreted as per DM.

#### 4.1 Non-stationary Gaussian noise and RFI

The results of processing the synthetic files from the emulated blind surveys (see experiments 1–4 in Table 2) with the default pulsar search pipelines of SIGPROC and PRESTO are shown in Fig. 7. Note that the metric used in this section to express the performance of each pipeline is the number of false positives detected for every true positive detected across all 100 files for each experiment.

The number of false positives per true positive detected by SIGPROC increases approximately proportionally with a linear increase in the amplitude of the non-stationary noise (see Fig. 7a). This trend is

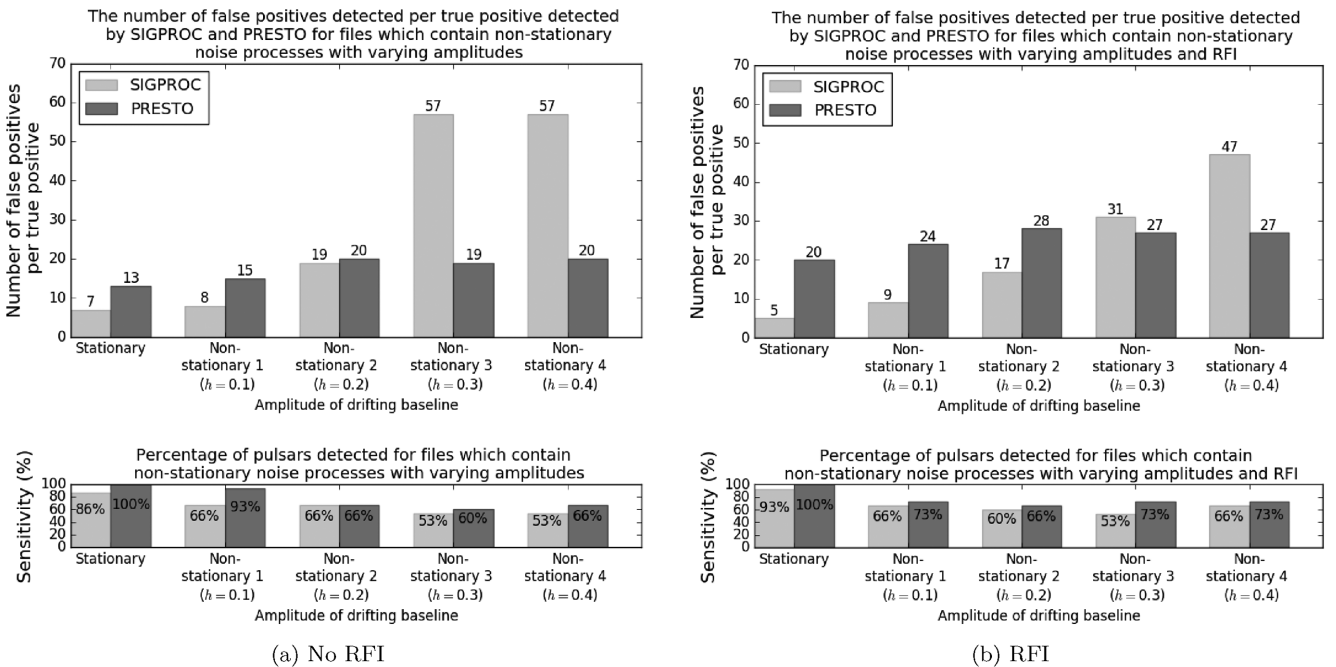
also visible in Fig. 7(b) when RFI is injected. Hence, the default SIGPROC pipeline is very sensitive to non-stationary noise.

The number of false positives detected per true positive by PRESTO is unaffected by the type and amplitude of the noise process present in the files, i.e. the number of false positives detected per true positive is almost constant irrespective of the amplitude of the non-stationary noise (see Fig. 7a). However, the number of false positives detected per true positive by PRESTO is slightly higher when weak RFI is present (see Section 4.4) compared to when no RFI is injected.

The sensitivity of SIGPROC and PRESTO can be seen in Fig. 7(a) to decrease by at least 20 and 7 per cent, respectively, in the presence of non-stationary noise compared to the stationary noise case. The 20 and 7 per cent losses recorded in sensitivity were averaged over all the pulse periods; however, the long-period pulsars were much more affected. Note that the amplitudes of the injected pulsars were chosen such that they are detectable at an SNR of  $\sim 12$  in the presence of white noise when processed with pipeline H in SIGPROC (see Table 3); however, the addition of any (significant) amount of non-stationary noise rendered the pulsars undetectable. Consequently, there is no correlation visible between sensitivity loss and the non-stationary noise amplitude.

There is no correlation between the loss in sensitivity of SIGPROC and the amplitude of the non-stationary noise when weak RFI is present. Interestingly, the presence of weak RFI leads to an increase in the sensitivity of SIGPROC for the stationary noise case with RFI compared to the stationary noise files without RFI. Similarly, there is an increase in the sensitivity for the non-stationary 4 case when RFI is present compared to the no RFI case.

Comparing the sensitivity of PRESTO in Fig. 7(a) to the sensitivity in Fig. 7(b) it becomes apparent that PRESTO is not sensitive to weak RFI. The highest sensitivity attainable with PRESTO for files containing weak RFI and non-stationary noise is 73 per cent; furthermore,



**Figure 7.** The performance of SIGPROC and PRESTO for processing files that contain either stationary noise or non-stationary noise with varying amplitudes (i.e. different values for  $h$ ): (a) without RFI (see experiments 1 and 2 in Table 2); (b) with RFI (see experiments 3 and 4 in Table 2). (SIGPROC pipeline: default baseline subtraction  $\rightarrow$  default red-noise removal. PRESTO pipeline: RFI mask  $\rightarrow$  baseline not subtracted  $\rightarrow$  default red-noise removal.)

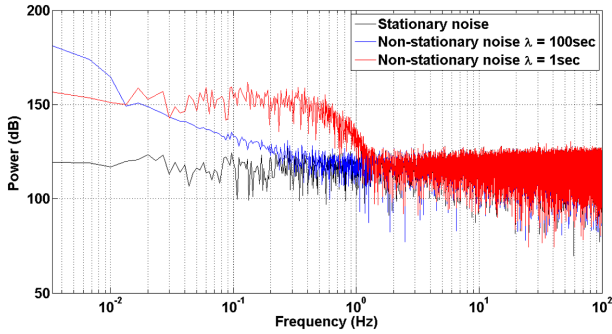


Figure 8. The power spectrum density for different noise processes.

a direct comparison reveals that PRESTO's sensitivity is on average 11 per cent better than SIGPROC's sensitivity.

## 4.2 Spectrum whitening methods

The power versus log-frequency plot in Fig. 8 shows the power spectrum density of two non-stationary Gaussian noise processes with correlation lengths  $\lambda = 1$  s (red) and  $\lambda = 100$  s (blue) as well as for a stationary Gaussian noise process (black). It is evident from Fig. 8 that the power spectrum density of a non-stationary process diverges from the desired flat power spectrum density of a stationary white noise process as the correlation length of the non-stationary process shortens.

With Fig. 8 in mind, four spectrum whitening methods (see Section 2.4) were assessed and the results can be seen in Fig. 9. The spectrum whitening techniques in both SIGPROC and PRESTO reduce the number of false positives detected per true positive significantly compared to the case when no spectrum whitening is applied. In the presence of non-stationary noise, the sensitivity of SIGPROC improved slightly from 53 to 60 per cent when the default spectrum whitening method was applied but the other methods had no effect on sensitivity (see Fig. 9a). The sensitivity of PRESTO is unchanged

when the spectrum whitening method is applied compared to no spectrum whitening.

## 4.3 Detrending the data before processing

Detrending the baseline in SIGPROC with either a 10 s moving average filter or the built-in detrending method led to an increase in the number of false positives detected per true positive compared to when the baseline was left intact (see Fig. 10a). However, the 10 s moving average filter did improve the sensitivity by 6 per cent.

Detrending the baseline in PRESTO with a 10 s moving average filter reduced the number of false positives detected per true positive and increased PRESTO's sensitivity with 7 per cent (see Fig. 10b).

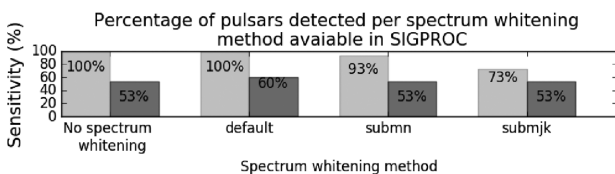
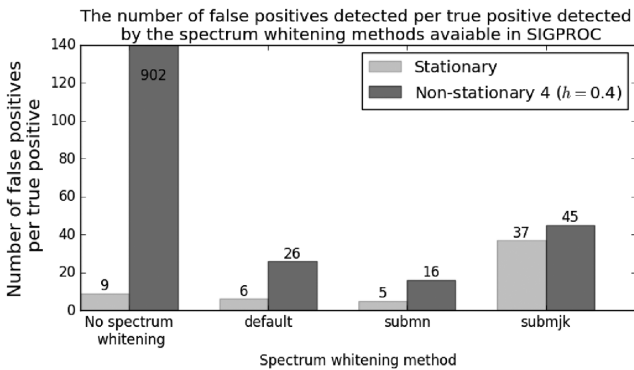
These results hint at the improved sensitivity attainable when the file contains both a baseline with long correlations and a slowly pulsating pulsar, i.e. removing the baseline significantly improves the sensitivity of detecting slow pulsars. This fact is highlighted with the postcard plots in Sections 4.6 and 4.7, described later in the paper.

## 4.4 RFI detection and mitigation methods

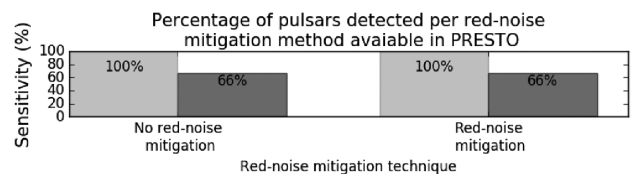
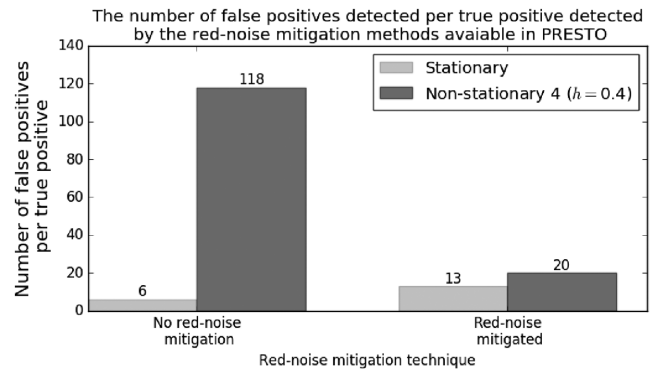
RFI masks created with PRESTO's `rfifind` function, for the same file, are depicted in Fig. 11. The masks differ with respect to the integration times used to create them. With the plots in Fig. 11, we show that the RFI we injected, although visible, is weak compared to the amplitude of the non-stationary baseline.

The default integration length of 30 s is most successful at detecting the actual injected RFI (see Fig. 11c). The two masks created with shorter integration times mostly flagged the maxima of the non-stationary baseline as opposed to the actual injected RFI (see Figs 11a and b).

RFI was injected such that 12 per cent of all the samples in the data are affected. The 2 s mask in Fig. 11(a) found 6.868 per cent of the 2 s intervals to be affected by RFI, the 8 s mask in Fig. 11(b) found 6.887 per cent of the 8 s intervals to be affected by RFI and

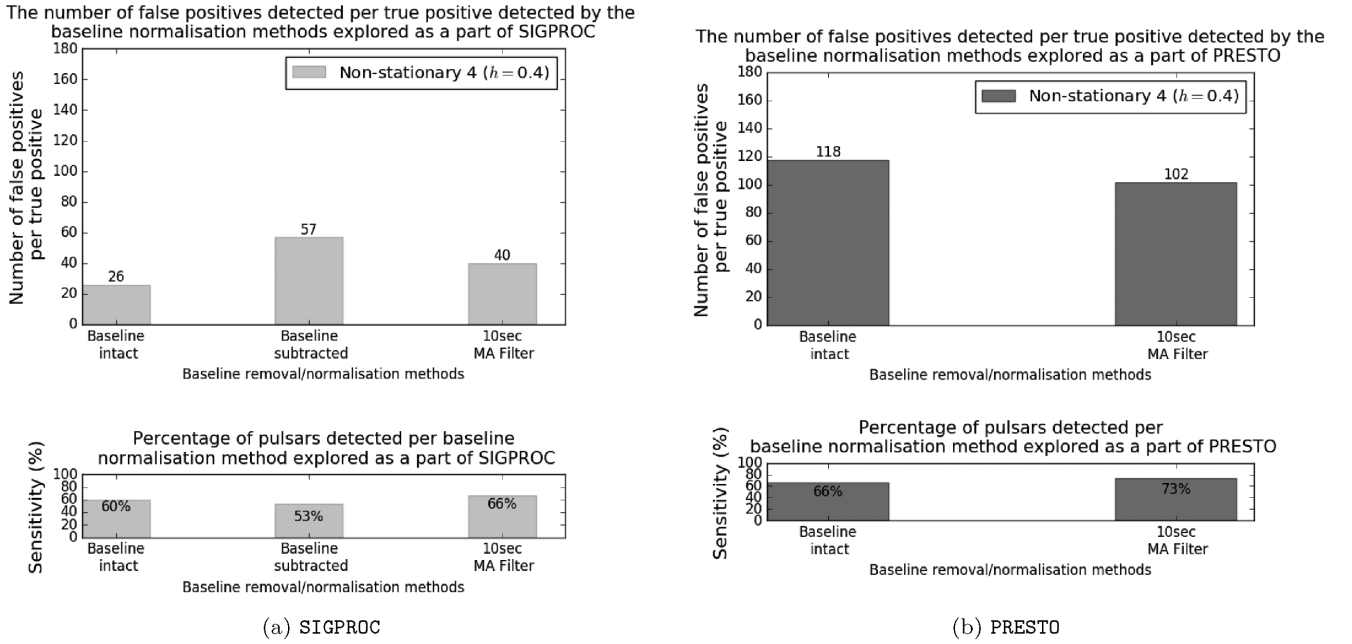


(a) SIGPROC

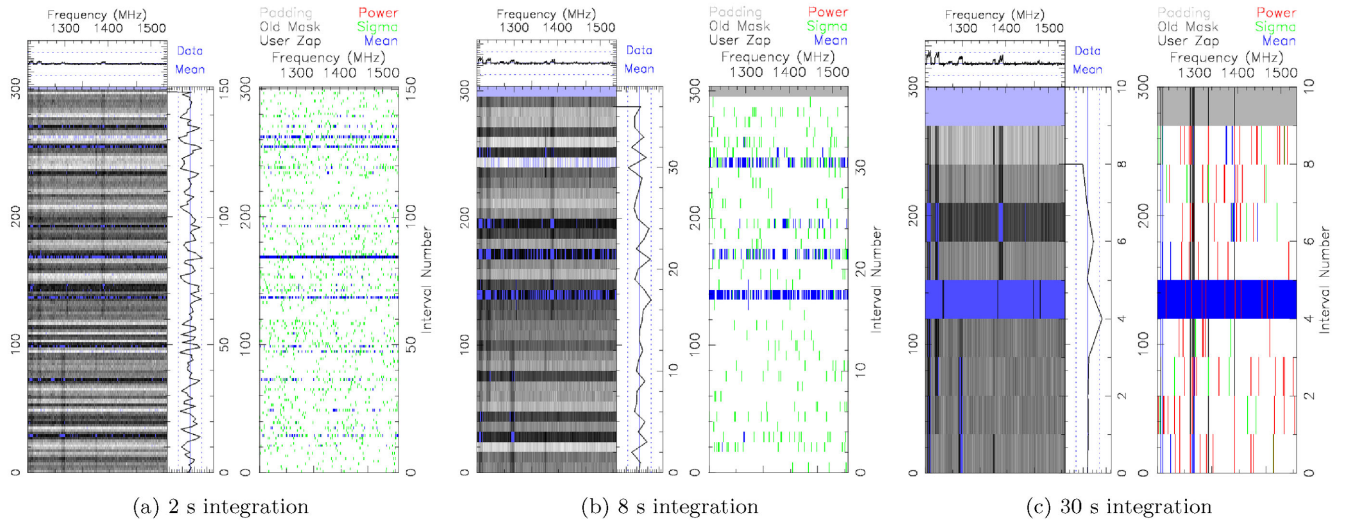


(b) PRESTO

Figure 9. The performance of the red-noise mitigation methods available in (a) SIGPROC and (b) PRESTO for processing files that contain either stationary noise (see experiment 1 in Table 2) or non-stationary noise (see experiment 2 in Table 2). No RFI was present in the files analysed. (SIGPROC pipeline: baseline intact → red-noise removal methods. PRESTO pipeline: RFI mask → baseline not subtracted → red-noise removal method.)



**Figure 10.** The performance of the time-domain baseline normalization methods available in (a) SIGPROC and (b) PRESTO for processing files that contain non-stationary noise (see experiment 2 in Table 2). No RFI was present in the files analysed. (SIGPROC pipeline: three time-domain baseline normalization methods  $\rightarrow$  default red-noise removal. PRESTO pipeline: No RFI mask  $\rightarrow$  baseline normalization method  $\rightarrow$  no red-noise removal.)



**Figure 11.** RFI masks created with PRESTO's `rfifind` function. The plots are for the same file but created with different integration times. The `-timesig` threshold was set to three and the `-freqsig` threshold was set to eight for the `rfifind` function.

the 30 s mask found that 16.895 per cent of the 30 s intervals are affected by RFI.

The focus of this section is on the real-time detection and mitigation of RFI; hence, the decision to investigate the effectiveness of an RFI mask integrated over a few seconds when applied to the synthetic filterbank files. The results of which can be seen in Fig. 12.

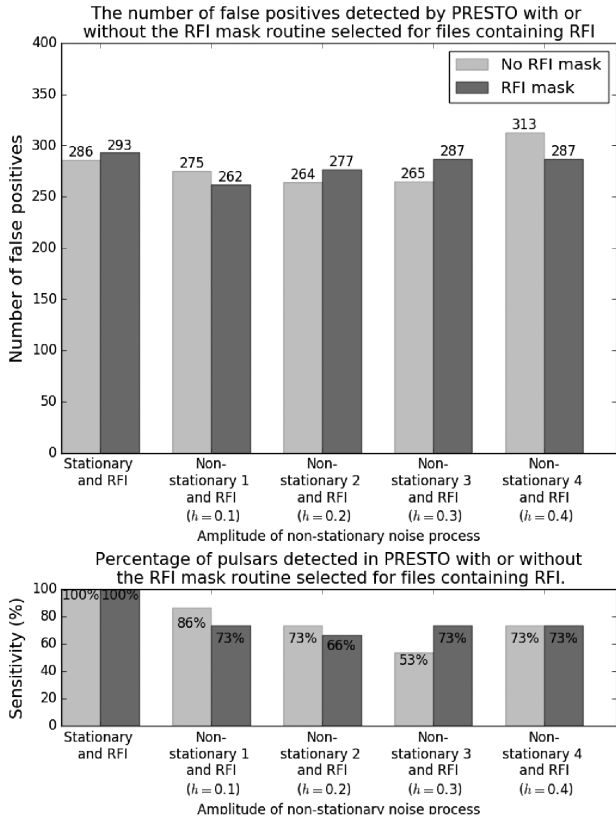
The RFI detection and masking routine available in PRESTO had on average little to no effect on both the sensitivity and the number of false positives detected per true positive except for the non-stationary 3 with RFI case where the sensitivity was increased from 53 to 73 per cent when the mask was applied. Greater insight on the effect of applying the RFI detection and mitigation routine in

PRESTO to files containing weak RFI can be gained by comparing the results of pipelines A–D to those of pipelines E–H in Fig. 14.

#### 4.5 Variation of detection with signal significance

Pulsars, embedded in different noise processes, were searched for by the default search pipelines of SIGPROC and PRESTO. The detection significance for each detected pulsar is shown in Fig. 13; where the colours indicate:

(i) Green square: an injected pulsar was detected (the detection significance is printed in the square).



**Figure 12.** The efficacy of the RFI detection and masking routine in PRESTO when processing files that contain either non-stationary noise (see experiment 2 in Table 2) or non-stationary noise with weak RFI (see experiment 4 in Table 2). (PRESTO pipeline: RFI mask yes/no  $\rightarrow$  baseline intact  $\rightarrow$  default red-noise removal.)

(ii) Orange square: an injected pulsar is amongst the detected signals but is not considered a candidate because its detection significance is not above the default threshold value.

(iii) Red square: signifies that an injected pulsar was missed.

(iv) Grey square: signifies that an injected pulsar was detected but is not considered a candidate because it has an abnormally high detection significance.

Each box in Fig. 13 shows a single instantiation of a pulsar/non-stationary baseline pair that was searched by both SIGPROC and PRESTO.

The left-hand sides of both matrices in Fig. 13 are populated with detections, whereas the right-hand sides are predominantly populated with misses. Hence, long-period pulsars embedded in non-stationary noise processes across a range of correlation lengths are missed by both the default SIGPROC and PRESTO pulsar search pipelines.

The detection significance at which pulsars with periods greater than 50 ms are detected decreases as the correlation length of the non-stationary noise shortens, whereas the detection significance of fast-period pulsars are unaffected by the correlation length of the non-stationary noise.

The results in Fig. 13 portray single trials for near-threshold signals that are very sensitive to the noise realization used. To help understand the average and variance associated with the detection significance of these signals, we injected a pulsar with a period of 0.126 s in an ensemble of 20 noise realizations each with the same correlation length of 1 s and amplitude  $h = 0.4$  (see equation 2).

This additional experiment showed that the average SNR at which the pulsar was detected in SIGPROC is 9 with a standard deviation of 1.45 compared to the SNR of 12.1 at which the pulsar is detected when embedded in stationary Gaussian noise. Similarly, the average Gaussian significance of the detected pulsar in PRESTO is 6.621 with a standard deviation of 1.14 compared to the stationary Gaussian noise case of 7.7. Consequently, the results for this particular combination of period and correlation length that are plotted in Figs 13–16 would show very little variability had there been more realizations of the same experiments. Multiple repetitions of this experiment over all combinations are extremely time costly and have not been attempted here. We have however computed similar standard deviations for other combinations of period and length-scale (e.g. periods of 5, 0.01 and 0.002 s, and  $\lambda$ s of 1, 0.01 and 100 s), using small numbers of realizations (5–20). We find that the standard deviation in the experiments with no injected RFI remains similar to the measurements above, whereas the cases with RFI show increased variance, with measured standard deviations between 2 and 3. Although this will have an effect on a case-by-case basis, the overall statistical picture can be interpreted.

It is evident from Figs 13(a) and (b) that the default search pipeline in PRESTO is better at finding pulsars of various periods embedded in different non-stationary noise processes compared to SIGPROC.

#### 4.6 Sensitivity postcard plots of all the pipelines used to process files with PRESTO

The sensitivity plots for all the search pipelines explored in PRESTO (see Table 5) are depicted in Fig. 14.

None of the pipelines in PRESTO are able to detect all the pulsars embedded in the different noise processes. Moreover, most of the pipelines miss the long-period pulsars. The addition of weak RFI, in general, does not alter PRESTO’s ability to find pulsars.

Pipelines A, C, E and G in Fig. 14 contain detections with Gaussian significances well in excess of the expected maximum Gaussian significance. We do not consider these outliers as true detections. However, do note that these pipelines all have one thing in common and that is they do not whiten the spectrum.

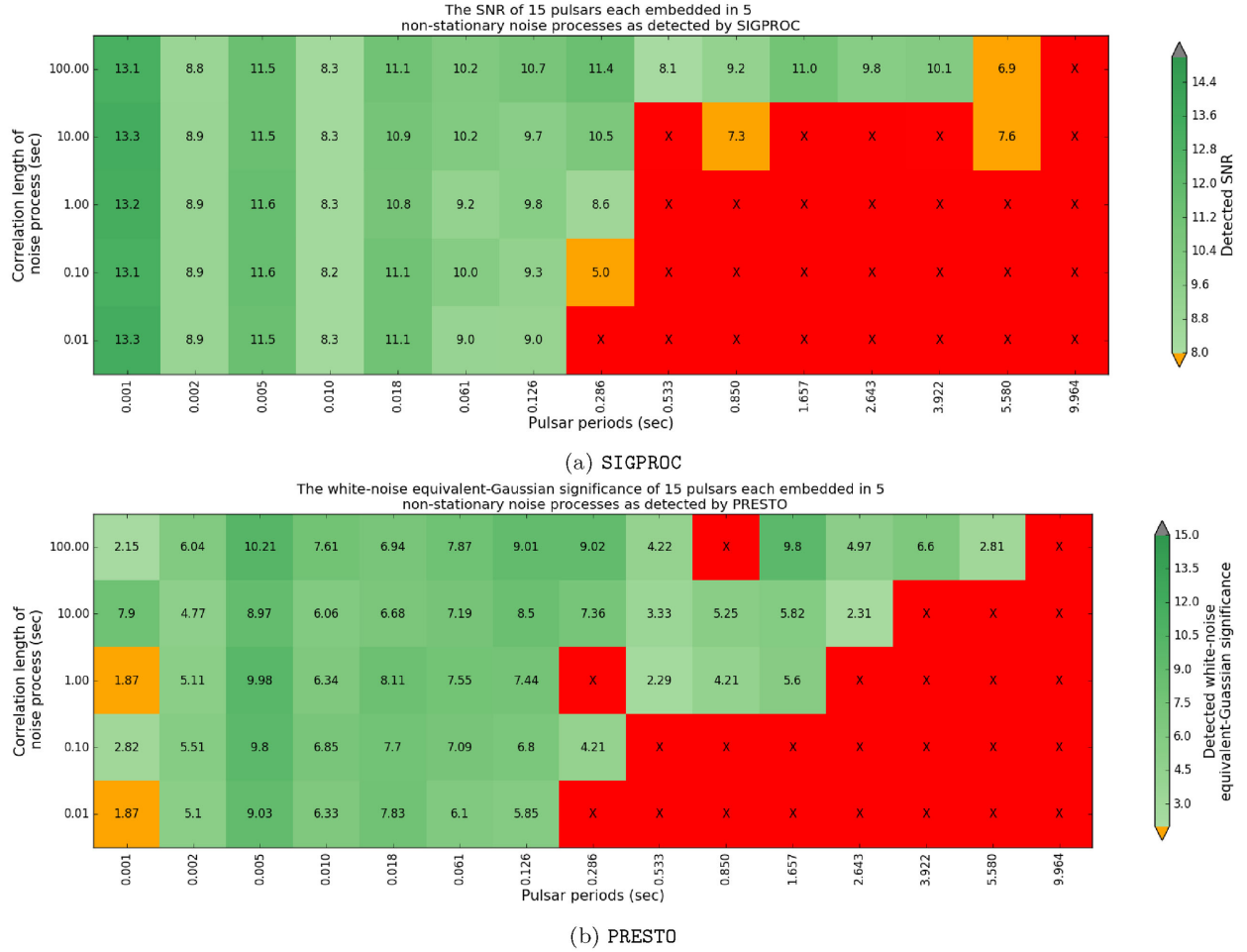
The only difference between pipelines A–D and E–H is the application of the RFI masking routine in PRESTO. From the results it appears that the RFI routine attenuates the Gaussian significance of short-period pulsars below the detection threshold both in the presence and absence of RFI.

From these plots, it is evident that running a moving average filter to normalize the time-domain data results in improved sensitivity, for example compare pipeline E with G and pipeline F with H. Note that when the moving average filter is applied in conjunction with the red-noise suppression method (see pipeline H in Fig. 14) then more long-period pulsars embedded in non-stationary noise processes with long correlation lengths are detected compared to when only the moving average filter is applied (see pipeline F in Fig. 14).

The pulsar search pipeline D in PRESTO (no RFI mask  $\rightarrow$  MA filter  $\rightarrow$  red noise mitigation) yields the best results amongst all the set-ups both in the presence and absence of RFI.

#### 4.7 Sensitivity postcard plots of all the pipelines used to process files with SIGPROC

The sensitivity plots for all the search pipelines explored in SIGPROC (see Table 4) are depicted in Figs 15 and 16.



**Figure 13.** The detection significance (values in black) at which 15 pulsars with different periods were detected (green squares) in files containing non-stationary noise processes with a relative amplitude of  $h = 0.4$  and five different correlation lengths (see experiment 5 in Table 2). Each box represents a single instantiation of a pulsar/non-stationary baseline pair that was searched by both SIGPROC and PRESTO. The red squares represent missed pulsars and the orange squares represent detected pulsars with detection significances below the default threshold levels. (a) The results for files processed with the default pipeline in SIGPROC and (b) the results for files processed with the default pipeline in PRESTO.

Note that the pulsar with period 0.002218 s is detected below the detection threshold (see orange squares in Figs 15 and 16) by almost all of the pipeline configurations in SIGPROC when RFI is present despite the other MSPs being detected. This pulsar is missed due to the increased variance of the SNR associated with the presence of RFI as explored and explained in Section 4.5.

It is apparent from pipelines D and H in Fig. 15 and pipeline L in Fig. 16 that not normalizing the spectrum results in a lot of pulsars being missed. Furthermore, mostly the long-period pulsars are regularly missed irrespective of the pipeline used in SIGPROC.

Overall PRESTO’s performance across all the pipelines is more consistent when compared the pipelines in SIGPROC.

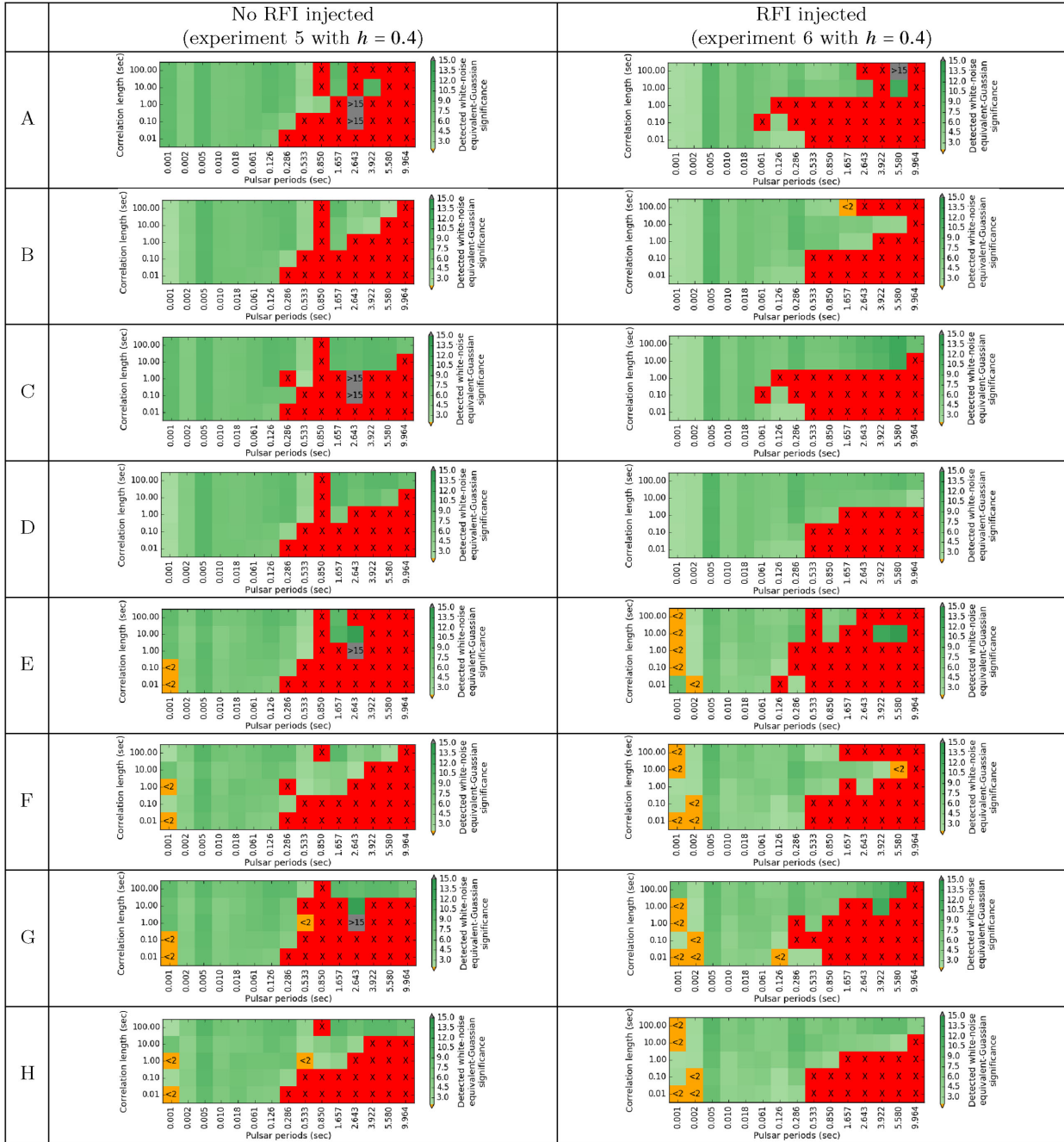
## 5 DISCUSSION

With the advent of instruments like the Square Kilometre Array, real-time processing will become essential. Therefore, it is crucial that the pipeline employed to do this processing is optimal from the start. The purpose of the analysis in this paper was to investigate what improvements to current pulsar search pipelines are necessary before embarking on the development of a new real-time processing

pipeline that is adept at dealing with the demands posed by this new era of pulsar astronomy.

This analysis demonstrated that non-stationary Gaussian noise processes with different correlation lengths lead to an increase in the number of false detections per true pulsar detection because of the static threshold applied in the power spectrum to distinguish between possible pulsar candidates and noise, i.e. non-stationary Gaussian noise is partly to blame for the so-called crisis in candidate selection (Lyon et al. 2016). In order to reduce the high number of false positives, SIGPROC as well as PRESTO employ spectrum whitening methods. Our analysis has revealed how these methods decrease the number of false positives per true positive at the cost of a loss in sensitivity and detection significance to long-period pulsars.

The spectrum whitening techniques assessed in this analysis suppress the power in the lower frequencies to conform to the power levels of the higher frequencies. Consequently, the spectral power of real signals from slowly rotating pulsars is attenuated along with the noise. This analysis serves as evidence that there is room for improvement in the effectiveness of the current spectrum whitening methods. Instead of forcing the spectrum to be uniform in the lower frequencies, the solution should rather be to accurately model the noise both the spectral and the time domain. In fact, we have



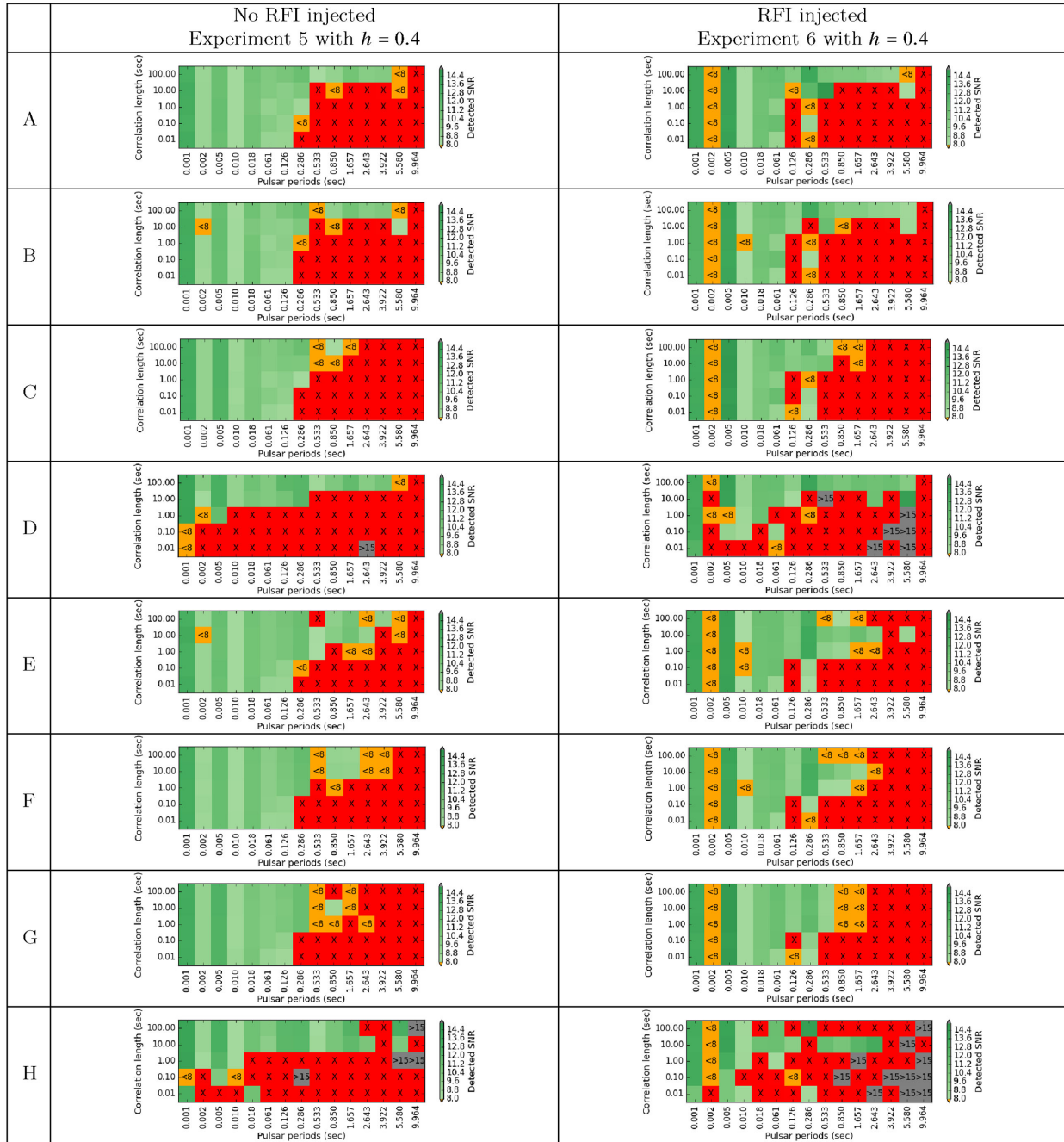
**Figure 14.** The Gaussian significance at which pulsars were detected (green squares) after files containing them (see experiments 5 and 6 in Table 2) were processed by eight different pipelines in PRESTO. The red squares represent missed pulsars, the orange squares represent detected pulsars with Gaussian significances below the default threshold level of 2 and the grey squares represent detected pulsars with Gaussian significances above the average maximum Gaussian significance of 15.

shown that applying a 10 s moving average filter in the time domain resulted in a greater number of detections of long-period pulsars. Consequently, leveraging moving averages on streaming data can help meet future real-time processing requirements whilst increasing surveys’ sensitivity to long-period pulsars.

As we discussed earlier, the results presented in Figs 13–16 are based on single realizations of period–length-scale combinations. However, we have sampled the standard deviation of the significance of these detections for several cases, and we conclude that although the picture may change for different realizations and different initial

SNR values of the injected pulsars, the areas in the plots that are most affected remain the same.

In this analysis, we dedispersed all the files at the same DM as what we injected the pulsars at ( $DM = 68 \text{ pc cm}^{-3}$ ). However, a subset of the files we dedispersed at four additional DM values, namely 0, 20, 150 and  $300 \text{ pc cm}^{-3}$ . Dedispersing the files at these four additional DMs allowed us to confirm that the number of false positives detected by the pulsar search pipelines for the files containing both non-stationary noise and RFI is greatest when the filterbank files are not dedispersed and decreases as one moves



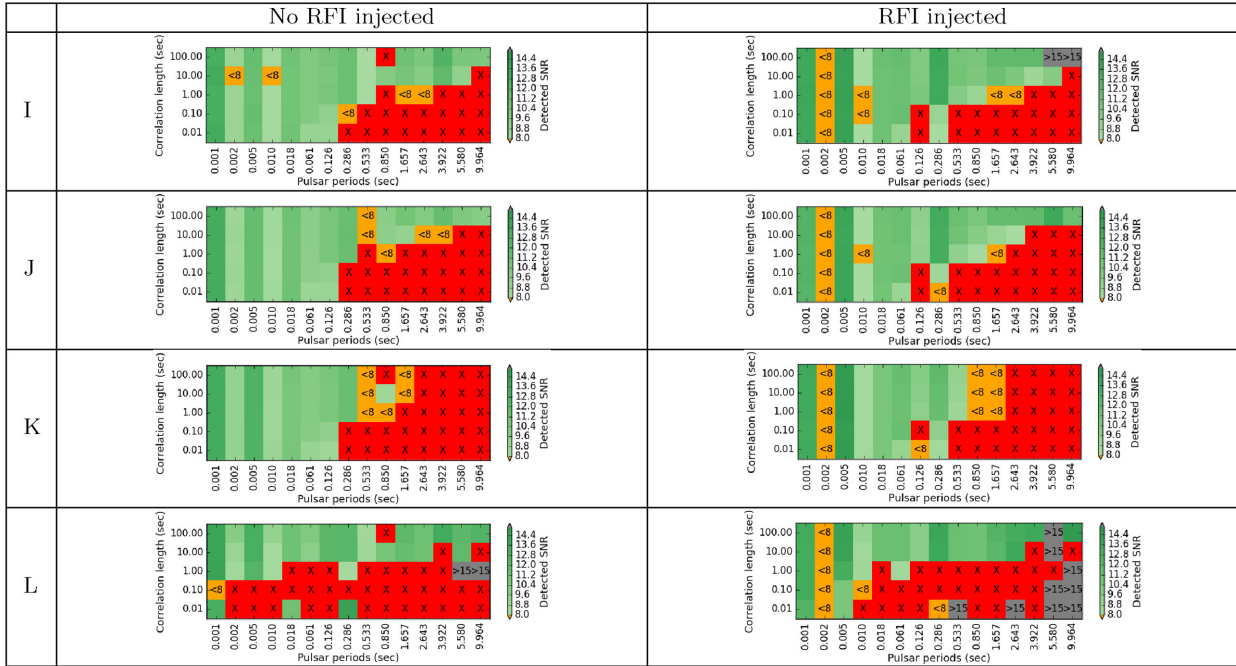
**Figure 15.** The SNR at which pulsars were detected after files containing them (see experiments 5 and 6 in Table 2) were processed by eight different pipelines in SIGPROC. The red squares represent missed pulsars, the orange squares represent detected pulsars with SNRs below the default threshold level of 8 and the grey squares represent detected pulsars with SNRs above the average maximum SNR of 15.

away from 0 DM. However, the number of false positives detected by the pulsar search pipelines is very similar to the five DMs used to dedisperse the data. Consequently, the number of false positives detected for files containing only non-stationary noise is similar irrespective of the DM used to dedisperse the data.

In this analysis, it was demonstrated that the RFI detection algorithm in PRESTO is very sensitive to the interplay between integration length over which the statistics of the filterbank files are computed and the rejection thresholds both in time and frequency of said statistics. For offline processing, this interplay can be fine-tuned so that most RFIs at different brightness levels can be detected and

masked. However, for the real-time detection of RFI this exploration of parameter space is not always possible because of the time constraint as well as the dynamic nature of the RFI environment.

There are a multitude of modules each placed strategically throughout current pulsar search pipelines for detecting different sources of RFI. Most of these RFI detection algorithms are largely amplitude based and are therefore very sensitive to non-stationary baselines. Consequently, data that contain no RFI but have higher than average mean and standard deviation are flagged as RFI. This analysis demonstrated that by flagging and replacing blocks of non-stationary data that contain no RFI or weak RFI may result in



**Figure 16.** The SNR at which pulsars were detected after files containing them (see experiments 5 and 6 in Table 2) were processed by four additional pipelines in SIGPROC. The red squares represent missed pulsars, the orange squares represent detected pulsars with SNRs below the default threshold level of 8 and the grey squares represent detected pulsars with SNRs above the average maximum SNR of 15.

short-period pulsars being attenuated below the detection threshold. Hence, there is a need for algorithms that can simultaneously normalize a non-stationary baseline and excise RFI signals superposed on said baseline without compromising the data that are not affected.

## 6 CONCLUSION

This paper gives a unified view of a typical pulsar search system. Moreover, it delves into the particulars of the algorithms available in the pulsar search software packages SIGPROC and PRESTO for spectrum whitening.

This analysis accords with the Lazarus et al. (2015) PALFA sensitivity analysis that non-stationary noise and weak RFI lead to an increase in the number of false positives and lower sensitivity for long-period pulsars. These two effects have resulted in overestimates of survey production.

The severe degradation of the detection significance is partly due to frequency-dependent noise and the attenuating nature of the spectrum whitening algorithms implemented in pulsar search software. Both these effects serve as explanation for why so many detectable long-period normal pulsars are missed by pulsar search pipelines.

The analysis revealed that an increase in sensitivity was achieved when the data were detrended with a moving average filter with a window size larger than the slowest pulsar. However, it should be noted that the efficacy of the filter is dependent on the filter window size relative to the correlation length of the non-stationary noise process.

Following from the results of this paper, it is now feasible to investigate methods for normalizing a varying baseline as well as addressing the question of how to decouple the red noise from the signal without attenuating the detection significance. The effective-

ness of these methods can be determined by applying them to the files created for this sensitivity analysis and then re-processing the normalized and modified files with the same pulsar search pipelines.

## ACKNOWLEDGEMENTS

The authors wish to thank Paul Brook, Marisa Geyer, Jayanth Chennamangalam and Chris Williams for useful discussions. Additionally, the authors would like to thank Sean Passmoor, Lindsay Magnus and Justin Jonas at SKA South Africa for supplying the necessary RFI information required for this analysis. E. van Heerden would like to thank Bernard van Heerden for editorial revisions and thanks with gratitude the Commonwealth Scholarship Commission in the UK for providing financial support for this work. Finally, the authors would like to thank Scott Ransom for useful comments that have helped improve the manuscript significantly. Part of this research has made use of the database of published pulse profiles maintained by the European Pulsar Network, available at: <http://www.jb.man.ac.uk/research/pulsar/Resources/e/pn/>

## REFERENCES

- Antoniadis J., 2014, *Multi-wavelength Studies of Pulsars and their Companions*. Cham, Switzerland
- Barr E., 2013, *Peasoup*, <https://github.com/ewanbarr/peasoup>
- Bates S., Lorimer D., Rane A., Swiggum J., 2014, *MNRAS*, 439, 2893
- Burgay M. et al., 2006, *MNRAS*, 368, 283
- Cordes J. M. et al., 2006a, *ApJ*, 637, 446
- Cordes J. M. et al., 2006b, *ApJ*, 637, 446
- Demorest P., Pennucci T., Ransom S., Roberts M., Hessels J., 2010, *Nature*, 467, 1081
- Hessels J. W., Ransom S. M., Stairs I. H., Freire P. C., Kaspi V. M., Camilo F., 2006, *Science*, 311, 1901

- Hobbs G., Manchester R., Teoh A., Hobbs M., 2004, in Camilo F., Gaensler B. M., eds, IAU Symp. 218, Young Neutron Stars and Their Environments. Astron. Soc. Pac., San Francisco, CA, p. 139
- Johnston S., Lyne A., Manchester R., Kniffen D., D'amico N., Lim J., Ashworth M., 1992, MNRAS, 255, 401
- Keith M., 2007, PhD thesis, University of Manchester
- Lazarus P. et al., 2015, ApJ, 812, 81
- Levin L. et al., 2013, MNRAS, 434, 1387
- Lorimer D., 2001, Technical report, SIGPROC-v1. 0:(Pulsar) Signal Processing Programs. Arecibo Technical Memo
- Lorimer D. R., 2011, in Torres D. F., Rea N., eds, High-Energy Emission from Pulsars and their Systems. Springer, Berlin, Heidelberg, p. 21.
- Lorimer D. R., Kramer M., 2005, Handbook of Pulsar Astronomy. Vol. 4, Cambridge Univ. Press, Cambridge
- Lorimer D. et al., 1998, A&AS, 128, 541
- Lorimer D. et al., 2006, MNRAS, 372, 777
- Lyne A., Graham-Smith F., 2012, Pulsar Astronomy. No. 48, Cambridge Univ. Press, Cambridge
- Lyon R., Stappers B., Cooper S., Brooke J., Knowles J., 2016, MNRAS, 459, 1104
- Manchester R., Lyne A., Robinson C., D'Amico N., Bailes M., Lim J., 1991, Nature, 352, 219
- Manchester R. et al., 1996, MNRAS, 279, 1235
- Manchester R. et al., 2001, MNRAS, 328, 17
- Nice D., Fruchter A., Taylor J., 1995, ApJ, 449, 156
- Rane A., Lorimer D., Bates S., McMann N., McLaughlin M., Rajwade K., 2016, MNRAS, 455, 2207
- Ransom S. M., 2001, PhD thesis, Harvard University Cambridge, Massachusetts
- Ransom S., 2011, Astrophysics Source Code Library, record ascl:1107.017
- Ransom S. M., Eikenberry S. S., Middleditch J., 2002, AJ, 124, 1788
- Rasmussen C. E., Williams C. K. I., 2006, Gaussian Processes for Machine Learning, vol. 38. The MIT Press, Cambridge, MA, p. 715
- Regulations R., 2008, Radiocommunication Sector. International Telecommunication Union Radiocommunication, Geneva
- Seward F., Harnden F., Jr, 1982, ApJ, 256, L45
- Stovall K. et al., 2014, ApJ, 791, 67
- Swiggum J. et al., 2014, ApJ, 787, 137
- van Heerden E., Karastergiou A., Roberts S., Smirnov O., 2014, in General Assembly and Scientific Symposium (URSI GASS), 2014 XXXIth URSI. IEEE, p. 1

This paper has been typeset from a  $\text{\TeX}/\text{\LaTeX}$  file prepared by the author.