OPEN ACCESS
IOP Publishing
Classical and Quantum Gravity
Class. Quantum Grav. 34 (2017) 015010 (20pp)
doi:10.1088/1361-6382/34/1/015010

A targeted spectral interpolation algorithm for the detection of continuous gravitational waves

Gareth S Davies[1,2], Matthew Pitkin[1] and Graham Woan[1]

[1] SUPA, University of Glasgow, Glasgow, G12 8QQ, UK
[2] Cranfield Forensic Institute, Cranfield University, Shrivenham, SN6 8LA, UK

E-mail: g.s.davies@cranfield.ac.uk

Received 2 March 2016, revised 4 November 2016
Accepted for publication 23 November 2016
Published 14 December 2016



Abstract

We present an improved method of targeting continuous gravitational-wave signals in data from the LIGO and Virgo detectors with a higher efficiency than the time-domain Bayesian pipeline used in many previous searches. Our spectral interpolation algorithm, SplInter, removes the intrinsic phase evolution of the signal from source rotation and relative detector motion. We do this in the frequency domain and generate a time series containing only variations in the signal due to the antenna pattern. Although less flexible than the classic heterodyne approach, SplInter allows for rapid analysis of putative signals from isolated (and some binary) pulsars, and efficient follow-up searches for candidate signals generated by other search methods. The computational saving over the heterodyne approach can be many orders of magnitude, up to a factor of around fifty thousand in some cases, with a minimal impact on overall sensitivity for most targets.

Keywords: gravitational waves, continuous gravitational waves, neutron stars, pulsars

(Some figures may appear in colour only in the online journal)

1. Introduction

Rapidly rotating neutron stars are promising sources of long-lived gravitational-wave signals and one of the key science targets of the LIGO and Virgo Scientific Collaborations [1]. The full parameter space for these signals is too large for simple coherent methods to be



Original content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

employed on timescales longer than a few days, so a range of more specific methods have been developed to explore specific regions of the space to different depths [2]. Known radio and x-ray pulsars comprise an important class of potential gravitational-wave source and three analysis pipelines have been specifically developed to exploit the known rotational phase evolution of these targets [3–5]. These targeted pipelines are fully-coherent over arbitrary lengths of time, tracking the predicted gravitational signal based on electromagnetic observations. They perform the deepest gravitational wave searches in the field and all three use both time and frequency domain techniques to reduce the data at relatively low computational cost. However, as these pipelines are now being applied more widely to candidate sources identified by other searches there is significant benefit in reducing the cost still further. Two of these (the Bayesian targeted and $\mathcal{G}$ statistic [3, 5] pipelines), rely on data from a carefully implemented, but slow, heterodyne step developed by Dupuis and Woan [2, 3, 6, 7] that allows the data to be sampled at a much lower rate than is generated by the detector (from $\sim$16 kHz to usually one sample per minute). Although we believe this exact solution is still the best way to approach high-value targets, short-period binaries and targets close to spectral lines (such as the Crab pulsar), when certain approximations are valid the step can be performed more efficiently for many other targets using fast fourier transform (FFT) methods. We therefore present an efficient method for down-sampling gravitational wave data and removing the effects of detector motion with respect to the source based on FFTs. Similar spectral methods have been used widely in the field for many years, and indeed form the basis of the $\mathcal{F}$-statistic search methods [8] which are ubiquitous. Our version of spectral interpolation, *SplInter*, is designed as a more efficient replacement for the heterodyne algorithm in certain situations. As we will show, our algorithm's large computational costs savings and very small sensitivity losses (when certain signal assumptions are valid) mean that it can be quickly and easily applied to a large number of targets, e.g. if following up large numbers of potential candidate signals from blind all-sky searches.

In section 3 we show how to calculate the down-sampled data streams using this method. In section 3 we confirm the equivalence of these streams to those from the heterodyne method, and assess the improvement in computational efficiency offered by SplInter.

The continuous gravitational wave strain signal in the output of a detector depends on the source emission mechanism and the source/detector geometry, but for the purposes of this analysis we assume it to be quasi-sinusoidal, with the form

$$h(T) = A(T)\mathrm{e}^{\mathrm{i}\Phi(T)} + A^{*}(T)\mathrm{e}^{-\mathrm{i}\Phi(T)}, \tag{1}$$

where $A(T)$ contains both the antenna response pattern and source amplitude parameters. $\Phi(T)$ is the apparent phase evolution of the signal due to source rotation and T is the time in a suitable inertial frame (see below). For example in the case of a triaxial neutron star rotating about a principle axis, emitting gravitational waves at twice the rotation frequency, A has the form [3]

$$A(T) = \frac{1}{4}F_{+}(T;\psi)h_0(1+\cos^2\iota) - \frac{\mathrm{i}}{2}F_{\times}(T;\psi)h_0\cos\iota, \tag{2}$$

where ι is the inclination angle between the source rotation axis and the line of sight from the detector to the source, ψ is the gravitational wave polarisation angle and h_0 is the wave amplitude. F_{+} and $F_{\times}$ are the antenna pattern responses to plus and cross polarisations respectively. The evolution of $\Phi(T)$ depends on the intrinsic rotational evolution of the neutron star, defined by its frequency and frequency derivatives $f^{(0),(1),(2)\ldots}_{\mathrm{rotation}}$. Over short timescales, the time-dependence of $h(T)$ is dominated by this phase term, expanded as

$$\Phi(T) = \Phi_0 + 2\pi \sum_{l=0} \frac{f^{(l)}(T - T_0)^{l+1}}{(l+1)!}, \tag{3}$$

where $f^{(l)}$ is the lth time derivative of the *gravitational wave* frequency (note that these are twice the rotation values for the $l = m = 2$ harmonic of a non-precessing, triaxial neutron star), T_0 is the epoch at which $\Phi(T_0) = \Phi_0$. The $f^{(l)}$ values are derived from radio, x-ray, or γ-ray pulse times-of-arrival, preferably from data spanning the same analysis period as the gravitational wave search in question.

The rotational and orbital motions of the Earth put the detector in a non-inertial rest frame, but for a given source position on the sky we can relate the topocentric signal arrival times at the detector, t', to those in the source's frame of reference by

$$T = t' + \delta(t'), \tag{4}$$

where $\delta(t')$ comprises four time-delay terms [3]:

$$\delta(t') = \Delta_{\mathrm{R}\odot} + \Delta_{\mathrm{S}\odot} + \Delta_{\mathrm{E}\odot} + \Delta_{\mathrm{Binary}}. \tag{5}$$

The Roemer delay $\Delta_{\mathrm{R}\odot}$ is the dominant term; it is the Euclidian difference in time taken for the signal to arrive at the detector and the solar system barycentre (SSB). $\Delta_{\mathrm{S}\odot}$ is the Shapiro delay, caused by the bending of spacetime near to massive bodies, which in the case of an Earth-based detector is dominated by the Sun's contribution. The Einstein delay $\Delta_{\mathrm{E}\odot}$ combines the effects of special and general relativistic time dilation due to motion and the presence of massive bodies. All of these terms vary slowly over the course of a year, and by small amounts over the course of a day, and as such can be considered as changing only linearly over the half-hour intervals we will consider shortly. These effects can be addressed in a number of ways, including resampling [9] or heterodyning [3] the data. Δ_{Binary} however can vary more quickly, on the timescale of the period of binary motion, this is an additional all-encompassing term that combines the Roemer, Shapiro and Einstein delays caused by the source's non-inertial motion, should it be in a binary system. In terms of the topocentric time we now have

$$\Phi(t') = \phi_0 + 2\pi \sum_{l=0} \frac{f^{(l)}(t' + \delta(t) - t_0)^{l+1}}{(l+1)!}, \tag{6}$$

where t_0 is the time at which $\Phi(t_0 - \delta) = \phi_0$.

Any difference in the assumed and actual phase evolution would introduce a residual phase evolution and a reduction in final search sensitivity. Typically, known radio and x-ray pulsars are timed sufficiently well for these effects to be negligible, but they can be included straightforwardly in the subsequent parameter estimation stages of a search. Once we have corrected for this known phase variation the only remaining time-dependence in equation (1) is from $A(t)$, which evolves as the antenna pattern sweeps over the source in a sidereal day. We can therefore sample the data at a much reduced rate, limited only by the changing antenna pattern, provided that we still capture this. The original heterodyne pipeline achieves phase correction by multiplying the strain time series $s(t')$ (where we use the notation of [3] that $s(t') = h(t') + n(t')$, where $h(t')$ is the signal from equation (1) and $n(t')$ is Gaussian noise) by $\exp[-\mathrm{i}\phi(t')]$, where $\phi(t') = \Phi(t') - \phi_0$, effectively shifting the signal frequency to zero and leaving $A(t)$ as the only time-dependent term. After applying a low-pass filter (conventionally with a time constant of 1 min) we average over M data samples to obtain a down-sampled time series of the form

$$B_K = \frac{1}{M} \sum_{j=1}^{M} s(t'_j) \exp[-\mathrm{i}\phi(t'_j)], \tag{7}$$

where M is the number of raw data samples contributing to B_K (following [3] this is often chosen to give one sample per minute), and K is the time index for the resulting time series. We model this as a combination of a signal term $y_K = A(t_K)\exp(\mathrm{i}\phi_0)$ and, appropriately over the narrow bands considered here, white Gaussian noise $n_K \sim N(0, \sigma_K^2)$

$$B_K = y_K + n_K. \tag{8}$$

σ_K is modelled as constant over short timescales, and is related to the original time-series noise σ_T by

$$\sigma_K^2 = \frac{\sigma_T^2}{M} = \frac{\sigma_T^2}{r\Delta t}, \tag{9}$$

where r is the original sample rate and Δt is the down-sampled period.

2. Formulation in the frequency domain

We can consider a similar analysis in the frequency domain. The Fourier transform of a signal $h(t')$ limited in duration to Δt centred on a time $(t_k - t_0)$ is

$$H_k(f) = \int_{t_k - t_0 - \Delta t/2}^{t_k - t_0 + \Delta t/2} [A(t')\mathrm{e}^{\mathrm{i}\Phi(t')} + A^*(t')\mathrm{e}^{-\mathrm{i}\Phi(t')}] \exp\left[-2\pi \mathrm{i} f(t' - t_0 - t_k + \Delta t/2)\right] \mathrm{d}t'. \tag{10}$$

In this Fourier-based version, t_k become the time-stamps of a series similar to the t_K series defined above, with $H_k(f_{\mathrm{signal}})$ playing the role of y_K. However, we now use k rather than K as the index to highlight that the two sampling rates need not be (and indeed usually are not) the same. t_0 is (again) the reference epoch of our timing solution.

We now consider $\phi(t')$, the time-dependent part of $\Phi(t')$, and use a time coordinate t with its origin at the mid-point of the data under consideration i.e. $t' \to t = t' - t_k + t_0$. Importantly, the time-delay correction term $\delta(t)$ in (5) will vary slightly over the time Δt. We approximate these changes in the arrival time to first order in t, defining $\delta_k = \delta(t_k)$ and $\dot{\delta}_k = \frac{\mathrm{d}}{\mathrm{d}t}\delta(t_k)$, such that for the duration of the data $\delta(t) \approx \delta_k + \dot{\delta}_k t$. Equation (6) now becomes

$$\phi(t) \approx 2\pi \sum_{l=0} \frac{f^{(l)}[t(1+\dot{\delta}_k) + t_k - t_0 + \delta_k]^{l+1}}{(l+1)!}. \tag{11}$$

To second order in t

$$\phi(t) \approx \phi_k + 2\pi f_k t + \pi \dot{f}_k t^2, \tag{12}$$

where

$$\phi_k := 2\pi \sum_{l=0} \frac{f^{(l)}(t_k - t_0 + \delta_k)^{l+1}}{(l+1)!}, \tag{13}$$

and

$$f_k := (1+\dot{\delta}_k) \sum_{l=0} \frac{f^{(l)}(t_k - t_0 + \delta_k)^l}{l!}. \tag{14}$$

We also approximate $\dot{f}_k$ as

$$\dot{f}_k = \frac{f_{\mathrm{end}} - f_{\mathrm{start}}}{\Delta t}. \tag{15}$$

where

$$f_{\text{start}} = \left[1 + \frac{\mathrm{d}}{\mathrm{d}t}\delta\left(t_k - \frac{\Delta t}{2}\right)\right]\sum_{l=0} \frac{f^{(l)}\left[t_k - \frac{\Delta t}{2} - t_0 + \delta\left(t_k - \frac{\Delta t}{2}\right)\right]^l}{l!}. \tag{16}$$

and

$$f_{\text{end}} = \left[1 + \frac{\mathrm{d}}{\mathrm{d}t}\delta\left(t_k + \frac{\Delta t}{2}\right)\right]\sum_{l=0} \frac{f^{(l)}\left[t_k + \frac{\Delta t}{2} - t_0 + \delta\left(t_k + \frac{\Delta t}{2}\right)\right]^l}{l!}. \tag{17}$$

We assume the signal amplitude and antenna pattern contributions to $A(t)$ are approximately constant on timescales of a small fraction of a day, so when Δt is small we can replace $A(t)$ above with $A(t_k)$. Having defined $y_k := A(t_k)\mathrm{e}^{\mathrm{i}\phi_0}$ we can therefore write, using these approximations,

$$H_k(f) \approx \mathrm{e}^{-\mathrm{i}\pi f\Delta t}\int_{-\Delta t/2}^{\Delta t/2}\left[\, y_k\mathrm{e}^{\mathrm{i}\phi(t)} + y_k^*\mathrm{e}^{-\mathrm{i}\phi(t)}\right]\mathrm{e}^{-2\pi\mathrm{i}ft}\,\mathrm{d}t. \tag{18}$$

This is the Fourier transform which will be considered in the following models.

2.1. The form of the signal in a short transform

Our signal is quasi-sinusoidal, but with amplitude and phase varying slowly as the source moves though the antenna pattern, and with changes in delay and doppler shift as well as intrinsic variations in the source spin rate. To first order in f, using $H_k(f)$ from (18) and $\phi(t)$ from (12) we have

$$\begin{aligned} H_k(f) \approx\ & y_k\mathrm{e}^{\mathrm{i}\phi_k - \mathrm{i}\pi f\Delta t}\int_{-\Delta t/2}^{\Delta t/2}\exp\left[2\pi\mathrm{i}(f_k - f)t + \mathrm{i}\pi\dot{f}_k t^2\right]\mathrm{d}t \\ & + y_k^*\mathrm{e}^{-\mathrm{i}\phi_k - \mathrm{i}\pi f\Delta t}\int_{-\Delta t/2}^{\Delta t/2}\exp\left[-2\pi\mathrm{i}(f_k + f)t - \mathrm{i}\pi\dot{f}_k t^2\right]\mathrm{d}t. \end{aligned} \tag{19}$$

These expressions are not strictly analytic due to the t^2 phase-dependency of the exponent, but are forms of the familiar Fresnel integral. The limiting form, when $\dot{f}_k$ is small, is just the Fourier transform of a time-limited sinusoid, so we will consider this as a special case below.

2.1.1. The sinc approximation, $\dot{f}_k = 0$. The contribution of the intrinsic spin-down of the source, $f^{(1)}$, to the overall $\dot{f}$ is generally negligible over the course of a single transform (maybe lasting minutes to hours) and the change in frequency due to this component will be much smaller than the frequency resolution. For example the Crab pulsar, which has an unusually large spin-down of $f^{(1)} = 7.4\times10^{-10}\,\mathrm{Hz\,s^{-1}}$ [10, 11], will change in frequency by only $\Delta f = 1.3\times10^{-6}\,\mathrm{Hz}$ over 30 min, which is 0.1% of the width of a frequency bin. Instead, $\dot{f}_k$ is dominated by the $\dot{\delta}_k$ term caused by the orbital motions of the source and observer. The spin and orbital motion of the observer are also usually negligible over $\sim$1 h, so for a source that is *not* in a binary system we can assume $\dot{f}_k = 0$ for the duration of the integral, so that

$$\begin{aligned} H_k(f) \approx\ & y_k\mathrm{e}^{\mathrm{i}\phi_k - \mathrm{i}\pi f\Delta t}\int_{-\Delta t/2}^{\Delta t/2}\exp\left[2\pi\mathrm{i}(f_k - f)t\right]\mathrm{d}t \\ & + y_k^*\mathrm{e}^{-\mathrm{i}\phi_k - \mathrm{i}\pi f\Delta t}\int_{-\Delta t/2}^{\Delta t/2}\exp\left[-2\pi\mathrm{i}(f_k + f)t\right]\mathrm{d}t \end{aligned} \tag{20}$$

$$\approx y_k \Delta t \exp\left[\mathrm{i}\phi_k - \mathrm{i}\pi f \Delta t\right] \mathrm{sinc}\left[\pi(f_k - f)\Delta t\right] + y_k^* \Delta t \exp\left[-\mathrm{i}\phi_k - \mathrm{i}\pi f \Delta t\right] \mathrm{sinc}\left[\pi(f_k + f)\Delta t\right], \tag{21}$$

where we use the convention

$$\mathrm{sinc}x := \frac{\sin x}{x}. \tag{22}$$

Close to the signal frequency (i.e. when $f \simeq f_k$)

$$\mathrm{sinc}\left[\pi(f_k - f)\Delta t\right] \gg \mathrm{sinc}\left[\pi(f_k + f)\Delta t\right], \tag{23}$$

so we can remove the second term in (21) to give

$$H_k(f) \approx y_k \Delta t \exp\left[\mathrm{i}\phi_k - \mathrm{i}\pi f \Delta t\right] \mathrm{sinc}\left[\pi(f_k - f)\Delta t\right]. \tag{24}$$

We will refer to this below as the *sinc approximation*.

2.1.2. The Fresnel approximation. If $\dot{f}_k$ is non-negligible then we can still approximate the form of $H_k(f)$ through a numerical integration. Such circumstances would occur if the doppler-shifted frequency was evolving significantly on timescales of Δt due to the orbital motion of the source or observer. If the rate of change of signal frequency is a constant, i.e. $\ddot{f}_k = 0$, we would expect the signal to appear as a 'Fresnel' pattern in the Fourier transform, characterised by the quadratic evolution of phase with time. Fresnel integrals have been studied extensively, and there are good algorithms for fairly rapid calculation [12]. They comprise a pair of functions defined as [13]

$$\mathcal{C}[w] := \int_0^w \cos\left(\frac{\pi x^2}{2}\right) \mathrm{d}x \tag{25}$$

and

$$\mathcal{S}[w] := \int_0^w \sin\left(\frac{\pi x^2}{2}\right) \mathrm{d}x. \tag{26}$$

In terms of these integral functions, (19) becomes

$$H_k(f) \approx \frac{y_k}{2\sqrt{2|\dot{f}_k|}} \exp(\mathrm{i}\Delta\phi) \left\{ \mathcal{C}[w_{\mathrm{end}}] + \mathrm{i}\frac{\dot{f}_k}{|\dot{f}_k|} \mathcal{S}[w_{\mathrm{end}}] - \mathcal{C}[w_{\mathrm{start}}] - \mathrm{i}\frac{\dot{f}_k}{|\dot{f}_k|} \mathcal{S}[w_{\mathrm{start}}] \right\}, \tag{27}$$

where

$$\Delta\phi = \phi_k - \pi f \Delta t + \pi \frac{\dot{f}_k \Delta t^2}{2} - \frac{\pi}{\dot{f}_k} \left(f_k + \frac{\dot{f}_k \Delta t}{2} - f \right)^2, \tag{28}$$

$$w_{\mathrm{end}} = \frac{\sqrt{2|\dot{f}_k|}}{\dot{f}_k} \left(f_k - f + \frac{\dot{f}_k \Delta t}{2} \right) + \sqrt{\frac{|\dot{f}_k|}{2}} \Delta t \tag{29}$$

and

$$w_{\mathrm{start}} = \frac{\sqrt{2|\dot{f}_k|}}{\dot{f}_k} \left(f_k - f + \frac{\dot{f}_k \Delta t}{2} \right) - \sqrt{\frac{|\dot{f}_k|}{2}} \Delta t. \tag{30}$$

Here, we have again ignored the $(f + f_k)$ term in (19), as again it is negligible in the interpolation region where $f \simeq f_k$.

We will refer to (27) as the *Fresnel approximation* to the signal Fourier transform, and we calculate the Fresnel integral terms with sufficient numerical precision using the algorithm in [12]. For small $\dot{f}_k$ the Fresnel approximation reduces to the sinc approximation described above. Computationally it is more expensive than the sinc approximation, however it need only be implemented during periods of time corresponding to large values of $\dot{f}_k$, i.e. $|\dot{f}_k|\Delta t^2 > 0.1$.

3. The spectral interpolation algorithm

The spectral interpolation algorithm (SplInter) is an alternative to the time-domain heterodyne algorithm of Dupuis and Woan [3] originally developed for gravitational-wave searches targeting known pulsars. In contrast to this heterodyne algorithm, SplInter uses fast Fourier transform methods to generate a similarly narrow-band time series but can process multiple targets very much more efficiently and usually with an acceptable impact on overall sensitivity. Within the LIGO Scientific Collaboration a Fourier transform data format, known as 'short-time Fourier transforms' (SFTs), has been defined [14, 15] for use in a variety of continuous gravitational wave searches. These SFTs contain discrete Fourier transforms of windowed data segments that are much shorter than the duration of the experiment (usually around half an hour). Of course there is an associated computational load in creating these SFTs, but this is offset by the efficiency of the SplInter algorithm. In addition, SFTs for several types of continuous-wave search (such as [16]) can be shared with SplInter.

In the first stage of the SplInter algorithm we take a discretely-sampled Fourier transform, in the form of an SFT, and compute a value of H_k at the instantaneous topocentric signal frequency using one of the interpolation methods described above over a small number of spectral points either side of the central topocentric frequency bin. We denote the result of this spectral interpolation B_k. In addition, we wish to calculate σ_k, the standard deviation of the noise on our estimate of B_k.

3.1. B_k and σ_k calculation

An SFT is of course a discrete Fourier transform, so we must interpolate between frequency bins to recover an unbiased estimate, B_k, of the signal, y_k. The interpolation is best understood in Bayesian terms: we compute the most probable value of y_k by choosing the value that maximises its posterior probability, given the data and a model for the signal. We choose to estimate the signal and noise separately, so for the signal estimate we will marginalise over the (unknown but assumed constant) noise variance.

We start by noticing that we can express the signal Fourier transform $H_k(f)$ (using either the sinc or Fresnel approximation for this expression) as the product of our unknown signal amplitude, y_k and a known signal shape function, μ_k, defined as

$$\mu_k(f) := \frac{H_k(f)}{y_k}. \tag{31}$$

If the Fourier transform of the data is $S_k(f)$ then, writing $S_{ki} \equiv S_k(f_i)$ and $\mu_{ki} \equiv \mu_k(f_i)$ the likelihood of the set of data $\{S_k(f)\}$ given B_k, spectral noise σ_F and signal shape $\mu_k(f)$ is

$$\mathrm{p}(\{S_k(f)\}|B_k, \sigma_F, \{\mu_k(f)\}) = \frac{1}{(2\pi\sigma_F^2)^N} \exp\left(-\frac{1}{2\sigma_F^2}\sum_{i=1}^{N}|S_{ki} - B_k\mu_{ki}|^2\right). \tag{32}$$

where the sum in i is over a window of N frequency bins around the signal frequency for which $|\mu|$ is significantly greater than zero. Here we have assumed that the noise is uncorrelated between frequency bins and has a constant standard deviation σ_F. We can consider σ_F as a nuisance parameter, and marginalise over it. Choosing a Jeffreys prior for σ_F of $\mathrm{p}(\sigma_F) \propto 1/\sigma_F$, $\sigma_F > 0$ and a uniform prior on B_k, for $B_k = -\infty$ to ∞ we obtain, after marginalisation, a log posterior for B_k of

$$\log\left[\,\mathrm{p}(B_k|\{S_{ki}\},\{\mu_{ki}\})\right] \propto -N\log\left(\sum_i |S_{ki} - B_k\mu_{ki}|^2\right). \tag{33}$$

The maximum of this log posterior occurs when $\sum_i |S_{ki} - B_k\mu_{ki}|^2$ is minimised. If we differentiate with respect to B_k^*[3] and set this to be zero we get

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}B_k^*}\left(\sum_i |S_{ki} - B_k\mu_{ki}|^2\right) &= \frac{\mathrm{d}}{\mathrm{d}B_k^*}\sum_i\left(S_{ki}S_{ki}^* - B_k\mu_{ki}S_{ki}^* - S_{ki}B_k^*\mu_{ki}^* + B_kB_k^*\mu_{ki}^*\mu_{ki}\right)\\ &= \sum_i\left(-S_{ki}\mu_{ki}^* + B_k\mu_{ki}^*\mu_{ki}\right) = 0.\end{aligned} \tag{34}$$

The most probable value of B_k is therefore

$$B_k = \frac{\sum_i[S_{ki}\mu_{ki}^*]}{\sum_i[\mu_{ki}^*\mu_{ki}]}, \tag{35}$$

a result that is familiar from least-squares analysis.

To estimate the variance of the noise, σ_k^2, we would ideally follow a similar route, marginalising over B_k in (32) and maximising the posterior for σ_k

$$\mathrm{p}(\sigma_k|\{S_{ki}\},\{\mu_{ki}\})_{\max} = \left[\int_{-\infty}^{\infty}\mathrm{p}(\sigma_k|\{S_{ki}\},\{\mu_{ki}\},B_k)\,\mathrm{p}(B_k)\,\mathrm{d}B_k\right]_{\max}. \tag{36}$$

However, this integral is not analytic. We therefore choose to use our calculated value of B_k from (35) to obtain the best estimate of σ_k, equivalent to using the Dirac delta function as the prior on B_k in (36)

$$\mathrm{p}(B_k) = \delta_D\left(B_k - \frac{\sum_i[S_{ki}\mu_{ki}^*]}{\sum_i[\mu_{ki}^*\mu_{ki}]}\right). \tag{37}$$

The application of this is straightforward: we use the most probable B_k calculated above to return best estimate of $H_k(f_i)$, $H_{\mathrm{best}}(f_i) = B_k\mu(f_i)$. The difference between $H_{\mathrm{best}}(f_i)$ and $S(f_i)$ is our best estimate of the noise $N_{\mathrm{best}}(f_i)$. We take these noise residuals around the signal frequency and then calculate their variance to give us σ_F^2.

The spectral noise variance, σ_F^2, is related to the time domain noise through Parseval's theorem ($\sigma_F^2 = \sigma_T^2\frac{2}{\Delta t^2 r}$). Using this and (9) we get

$$\sigma_k^2 = \sigma_F^2\frac{\Delta t}{2}. \tag{38}$$

[3] B_k^* has the simultaneous properties of

(a) $B_k \to B_k^*$ is conjugate conformal, leading to $\frac{\mathrm{d}B_k^*}{\mathrm{d}B_k} = 0$ and $\frac{\mathrm{d}B_k}{\mathrm{d}B_k^*} = 0$

(b) B_k^* and B_k are mutually defined; the most likely value of B_k^* defines the most likely value of B_k.

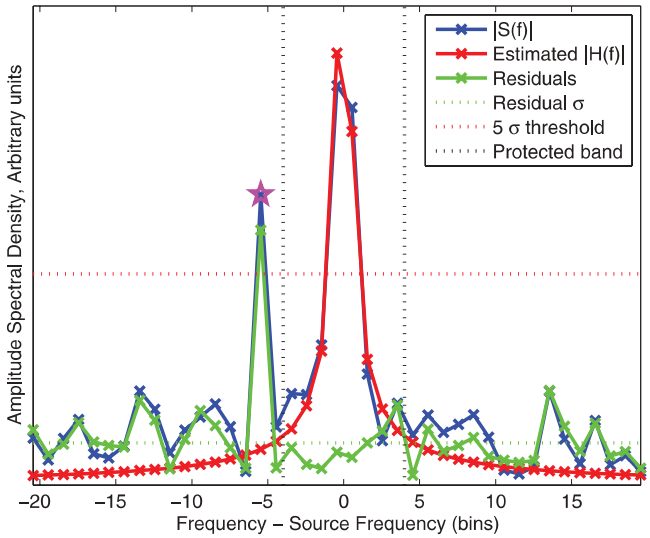


Figure 1. An illustration of the type of outlier removed by the second outlier removal step. Shown are the best fit of the noise, the standard deviation of the residuals, the threshold for removal, and the protected band around the source frequency. The removed data point is indicated by the magenta star.

We now have our calculated B_k and an estimate of σ_k. The parameter estimation stage of the pipeline used by Dupuis and Woan [3] treated the noise variance as a nuisance parameter to be marginalised over 30 min segments to give a Student's t-likelihood for the signal. However here we use the direct estimates of σ_k described above, giving a Gaussian likelihood for use in parameter estimation.

3.2. Outlier removal

The noise in gravitational-wave data contains many line features that may affect our estimates of B_k if they are close to the source frequency. We minimise this contamination by performing three outlier removal steps. The first outlier removal routine uses the standard deviation of $S(f)$ as an initial estimate of the noise. We then multiply this standard deviation by a number (typically around ten) decided by the user and remove any $S(f)$ data points with an absolute value above this threshold. This threshold is set to be large, to remove only the strongest lines, and the five bins either side of the signal frequency are excluded from this first step.

The second outlier removal step takes place after initial estimates of B_k and σ_k have been made and is shown in figure 1. By this stage we have an estimate of the noise in the frequency domain, σ_F, so we identify $S(f)$ data points with residual values above a threshold factor of this standard deviation. We use a factor of five in the illustrations given here. This threshold is lower than that employed in the first step, and now all but the closest ± 4 data points to the signal are involved. If any data points are removed by this process B_k and σ_F are recalculated and this outlier removal step repeated to convergence.

The third outlier removal step takes place after all B_k and σ_k have been calculated, and uses the noise estimates over the entire data set. We calculate the mean value of $\{\sigma_k\}$, $\langle\sigma\rangle$, and

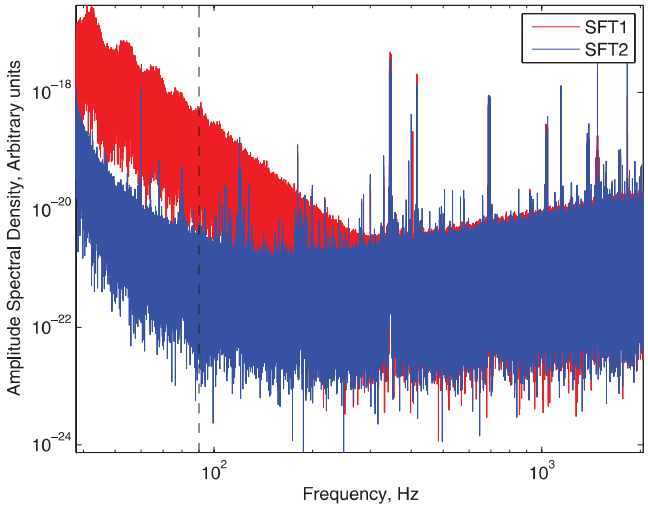


Figure 2. An illustration of the type of outlier removed by the third outlier removal step, showing the power spectra of two example SFTs. SFT1 has an unusually high low-frequency noise contribution, bleeding power into frequency channels up to around 300 Hz. For sources with signals in this lower range (such as a 90 Hz signal shown by the black line) the noise estimation for SFT1 would be large compared to that in SFT2 (a normal SFT). Sources at frequencies above ∼300 Hz would be unaffected by this outlier removal.

remove any data for which $|\mathfrak{Re}[B_k]|$, $|\mathfrak{Im}[B_k]|$ or σ_k is above a threshold factor of this value. This step removes B_k data points which are unusually noisy, but for which the noise is broad-band and was not detected by the first two outlier removal steps, as shown in figure 2.

We consider it rather unlikely that our methods would accidentally veto astrophysical signals that are slightly offset from our expected frequency, as a real signal would have to be exceptionally strong to show up as significantly as the lines we are vetoing in a single SFT.

Figure 3 shows the full SplInter algorithm for a single SFT. The detector data includes data quality flags and we restrict our analysis to segments of data in 'science mode'. The input files are therefore a science segment list and pointers to the corresponding Fourier data and a set of files defining the targets. The SplInter algorithm loops through segments, and in each segment processes each SFT according to figure 3.

4. Testing the SplInter algorithm

We tested the SplInter algorithm against the standard heterodyne method currently employed for both accuracy and performance. The first accuracy test is described in section 4.1.1 and checked that the $B_{k/K}$ outputs from the two routines are consistent in the noiseless case. In section 4.1.3 we check that the noise estimation σ_k is also accurate, and that this estimate is consistent with that estimated from the heterodyned B_K. In section 4.1.4 we perform a black-box replacement test, comparing the performance of our routines end-to-end for the analysis of hardware signal injections in LIGO S6 data [17]. Finally we test the algorithm performance in section 4.2, particularly the speed increase of SplInter compared to the heterodyne routine.

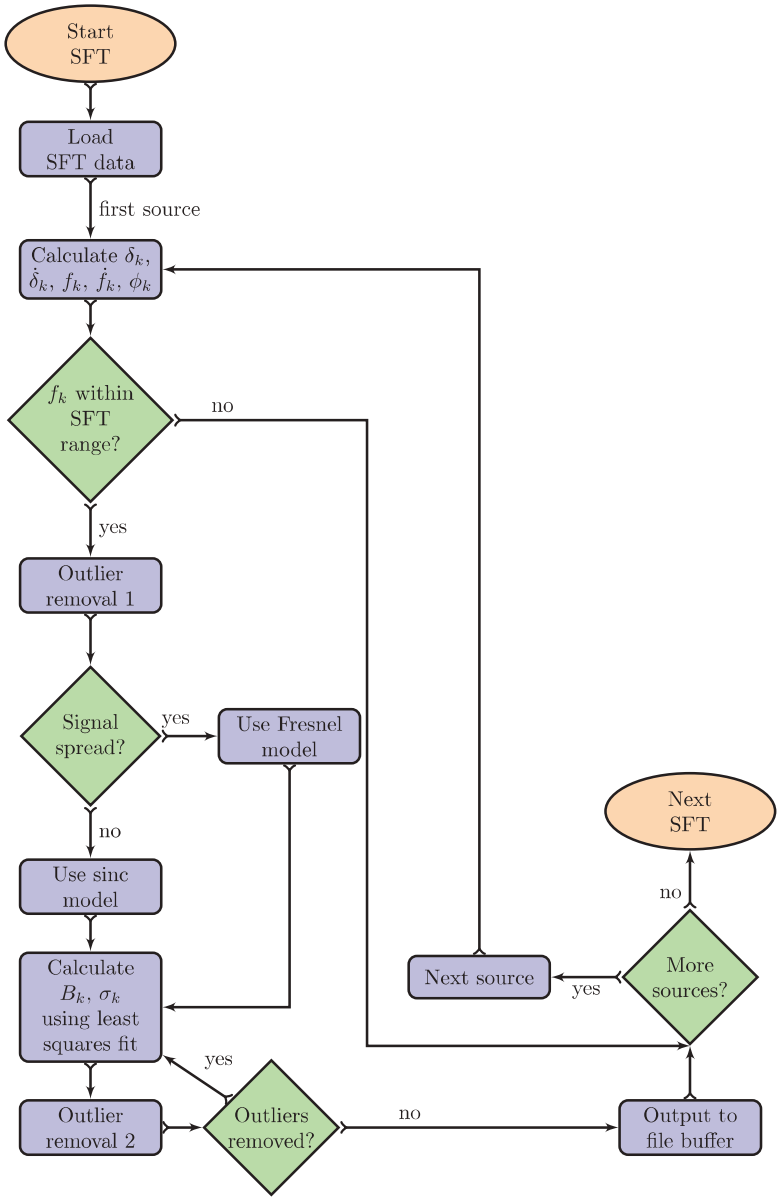


Figure 3. Flowchart showing the spectral interpolation algorithm during each SFT. This flowchart runs for each SFT, running on a loop within each science segment, which itself is looped over. The third outlier removal is not shown, as it does not take place within this loop.

4.1. Accuracy and testing

We now compare the SplInter output, B_k, with the standard heterodyne output, B_K (which we assume to be exact for this comparison), using the *mismatch*, m, between the two, defined as

$$m = \left| 1 - \frac{\sum_k B_{k,\mathrm{Spl}} \cdot B_{k,\mathrm{het}}}{\sum_k B_{k,\mathrm{het}} \cdot B_{k,\mathrm{het}}} \right|. \tag{39}$$

The mismatch is a useful indicator of how well our approximation matches the exact solution, and gives an approximate figure for the drop in SNR caused by these approximations. We define $B_{k,\mathrm{het}}$ as the average B_K value over the duration of the corresponding SFT, which is equivalent to performing the heterodyne with a Δt value of 30 min.

4.1.1. Recovery of noiseless signals from isolated pulsars. In the case of a noiseless signal, the heterodyne and spectral interpolation routines should, ignoring approximations, agree exactly, as $B_k = y_k$. Figures 4 and 5 show the result of applying the SplInter and heterodyne routines to noise-free data. The frames and SFTs were made using `lalapps_create_pulsar_signal_frame` and `lalapps_MakeSFTs` respectively[4]. We see that the SplInter and heterodyne B-estimates agree well, and always to better than 1%. We apply a hybrid interpolation scheme here, using the sinc approximation when $\dot{f}_k$ is small, and the Fresnel approximation otherwise, with a changeover point at $|\dot{f}_k|\Delta t^2 = 0.1$ [18].

Figure 5 shows the importance of using the Fresnel rather than the sinc approximation for PULSAR4. This source has both a high frequency and a relatively low declination, leading to a large second order change in phase over the duration of the SFT.

4.1.2. Recovery of noiseless signals from binary pulsars. The signal delay for a binary pulsar contains an extra term, Δ_{Binary}, in (5) from the Roemer, Shapiro and Einstein delays in the binary system itself, and this delay can introduce rapid variations in apparent frequency. Figure 6 shows which of the 97 known binary pulsars have a mismatch below 0.1 (circled) when comparing the B_k values analysed with SplInter and heterodyne respectively. We compute the mismatch over one day if the binary period P_b is <1 d, over the binary period if $1 < P_\mathrm{b} < 5$ d and over 5 d if $P_\mathrm{b} > 5$ d.

50 of these sources are in systems that show mismatches above 0.1 for SplInter, and we are therefore unable to use this method for these and maintain accuracy and sensitivity. The mismatch comes from significant high-order frequency derivatives in these pulsars over the 30 min period of the SFT. The second-order frequency derivative $\ddot{f}_k$ is proportional to $f^{(0)} a_1 / P_\mathrm{b}^3$, where a_1 is the projected semi-major axis of the binary system. By considering which binary pulsars we are unable to analyse using SplInter (those not circled in figure 6), we can set an empirical upper-limit on $\ddot{f}_k$. When using 30 min SFTs, this limit is

$$\frac{f^{(0)}}{1\ \mathrm{Hz}} \frac{a_1}{1\ \mathrm{lightsecond}} \left(\frac{P_\mathrm{b}}{1\ \mathrm{day}} \right)^{-3} \lesssim 10, \tag{40}$$

and this limit is delineated in figure 6 for pulsars with gravitational wave frequencies of 10, 100 and 1000 Hz.

[4] These routines are within the LALsuite software repository www.lsc-group.phys.uwm.edu/daswg/projects/lalsuite.html

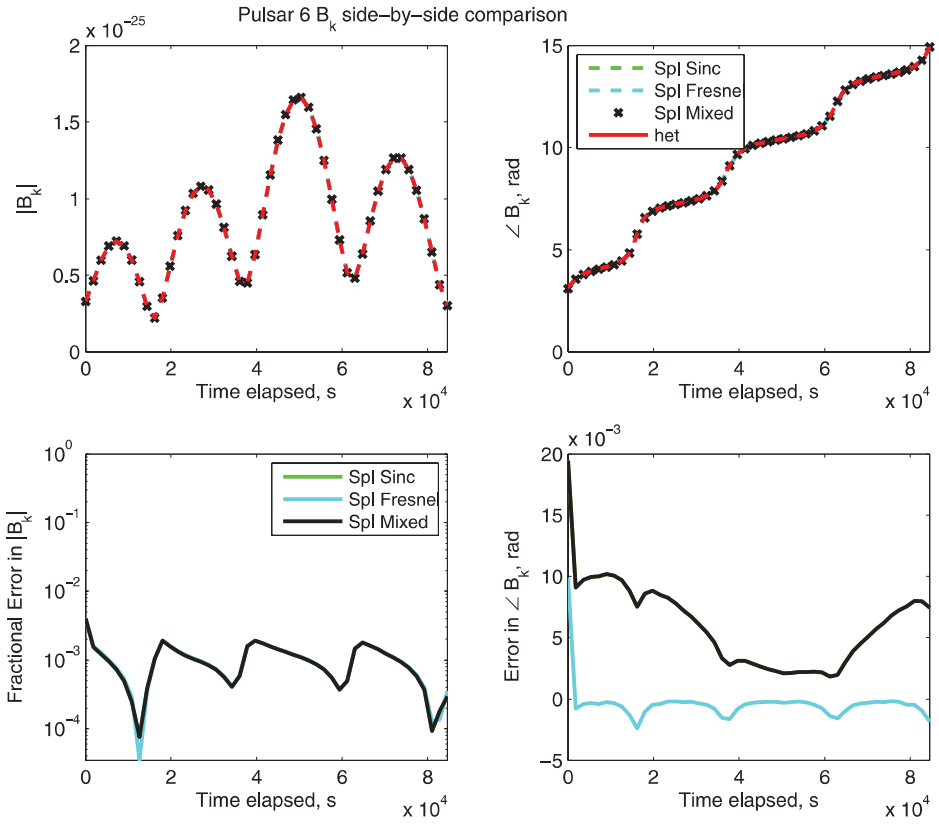


Figure 4. SplInter B_k (green/cyan/black) and fine heterodyne B_K (red) values over one day, with amplitude on the left and phase on the right, for a noiseless signal corresponding to hardware injection PULSAR6. Below are the fractional difference between the two. SplInter values are shown using the sinc approximation (green), the Fresnel approximation (cyan) and the mixed interpolator (black), which uses the sinc approximation when $|\dot{f}_k|\Delta t^2 < 0.1$ and the Fresnel approximation otherwise. $m = 0.0029$ for the sinc approximation, $m = 0.0029$ for the mixed interpolation and $m = 0.0028$ for the Fresnel approximation. The two methods are equally precise in this case, as the frequency does not significantly change during the SFT length.

4.1.3. Noise estimation tests. We tested noise estimation using SFTs and frames with known levels of white noise but no signal. After running the SplInter and heterodyne algorithms, we checked that the $B_{k/K}$ noise estimates were consistent with the injected value and with each other. We compared noise estimates from the SplInter routine to noise estimates from the heterodyne routine for noise data with a time-domain variance of $\sigma_T^2 = 1$. The estimate of the noise from the heterodyne, σ_H, was obtained using an average of the variance of the real and imaginary heterodyne parts over the duration of an SFT, converted into the equivalent noise value for the 30 min cadence of B_k.

We see in figure 7 that the heterodyne and SplInter noise estimates are consistent, and that both agree with the injected value of the noise and the expected distribution of these estimates. The expected distribution is a χ^2 distribution with $n - 1$ degrees of freedom, where n is the number of data points used to calculate the noise estimate:

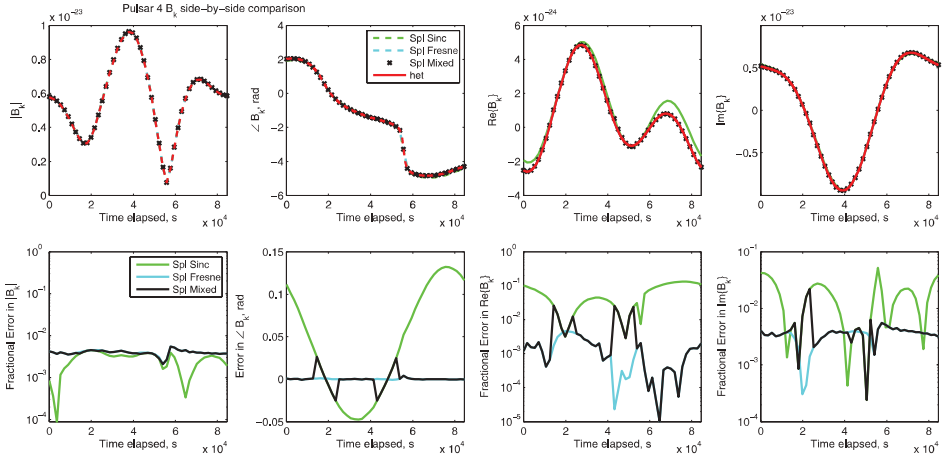


Figure 5. SplInter B_k (green/cyan/black) and fine heterodyne B_K (red) values over one day, showing amplitude, phase and real/imaginary parts for a noiseless signal corresponding to hardware injection PULSAR4 (colours as figure 4. $|\dot{f}_k|$ is high for this pulsar, and the Fresnel approximation is used by the mixed interpolator to maintain accuracy. There is a significant discrepency in the real part of B_k when using just the sinc approximation. ($m = 0.0101$ for sinc, 0.0040 for mixed and 0.0039 for the Fresnel approximation.

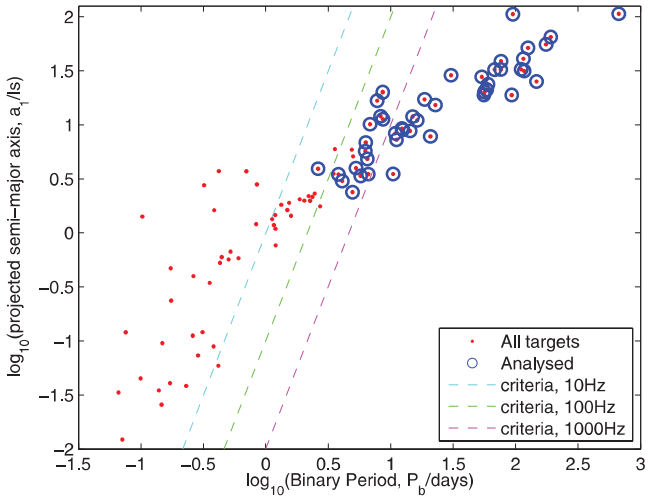


Figure 6. Binary period versus projected semi-major axis for targeted binary pulsars, indicating which binary pulsars have a small enough mismatch to be analysed using SplInter and which cannot. We include an indication of the empirical criteria we set for analysis of a target in a binary system, given in (40), for pulsars with source frequency of 10, 100 and 1000 Hz.

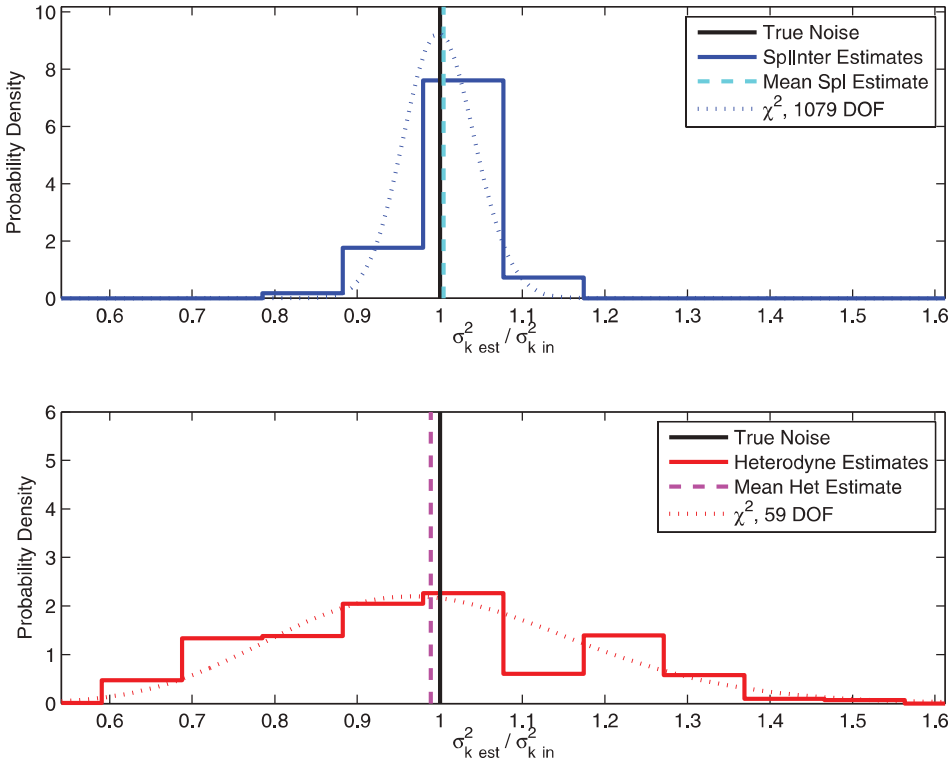


Figure 7. A histogram of standard deviation estimates of white noise from SplInter (top) and heterodyne (bottom), with the mean estimated values shown as dashed lines. Also shown is the true value of the noise (black dashed line) and the expected distributions of these values (dotted lines).

$$\frac{\sigma_k^2}{\sigma_{\text{true}}^2} \sim \frac{\chi^2(n-1)}{n-1}, \tag{41}$$

which for large n approximates a normal distribution with unit mean and variance $2/(n-1)$. The heterodyne noise estimate used 30 B_K data points from each of the real and imaginary parts of the data, leading to an expected distribution of a χ^2 with 59 degrees of freedom, shown in the figure by the red dotted line. Here we used the spectral interpolation algorithm with a bandwidth of 0.3 Hz around the signal frequency, leading to a χ^2 distribution on σ_k^2 with 1079 degrees of freedom (shown on the figure as a blue dotted line). One might get a marginally better noise estimate using a wider bandwidth, however the frequency dependence of the noise limits this. Additionally, there are diminishing returns in computational efficiency, and repeated use of the algorithm has found that a bandwidth of 0.3 Hz is a good compromise between these considerations.

4.1.4. Full testing with hardware injections. The LSC and Virgo collaborations inject artificial signals into the detector hardware control loops to test analysis pipelines. Here we show the results of running both the heterodyne and SplInter pipelines on two hardware-injected pulsars ('PULSAR4' and 'PULSAR6', the parameters of which are given in table 1) in just under

Table 1. Parameters of hardware injection pulsars 4 and 6.

	$f^{(0)}$ Hz	$f^{(1)}$ Hz s^{-1}	RA	Dec
PULSAR4	1403.16	-2.54×10^{-8}	$18^{\mathrm{h}}\,39^{\mathrm{min}}\,57.04^{\mathrm{s}}$	$-12^{\circ}\,27'\,59.85''$
PULSAR6	148.72	-6.73×10^{-9}	$23^{\mathrm{h}}\,55^{\mathrm{min}}\,0.23^{\mathrm{s}}$	$-65^{\circ}\,25'\,21.45''$

Table 2. SNRs of hardware injections 4 and 6 recovered by the two pipelines from four months of LIGO Hanford S6 strain data, calculated using the nested sampling algorithm `lalapps_pulsar_parameter_estimation_nested` [19].

$B_{k/K}$ algorithm	SplInter	Heterodyne	
Likelihood distribution	Gaussian		Student's t
PULSAR4	234.67	251.99	235.23
PULSAR6	16.02	17.22	17.18

four months of LIGO S6 data from the Hanford detector (LHO). During this interval the LHO duty cycle was 47%, giving $\sim 5 \times 10^6$ s of science data. With 4 months of data the injections can be recovered with a high signal-to-noise ratio, but with the posteriors retaining sufficient width to usefully assess our noise estimates. Table 2 lists the returned signal-to-noise ratios for the two pulsars using both the SplInter and heterodyne pipelines, running the latter with both a Gaussian and Student's t likelihood.

We see that the SplInter results are consistent with those from the heterodyne pipeline, with SNR values around 7% below those from heterodyne. This is to be expected in real data containing segments and lines as the filtering is different in the two pipelines. Most of this drop in SNR (around 5%) is due to SplInter's need for contiguous 30 min stretches of science data, rather than the 60 s stretches used by the heterodyne pipeline.

Figures 8 and 9 show the posterior distributions of the four source parameters determined in targeted searches, h_0, ϕ_0, $\cos\iota$ and ψ.

Again, in this example the posteriors generated by the two pipelines show sufficiently good agreement to allow the SplInter pipeline to replace the heterodyne pipeline without a systematically significant impact on overall performance. In figure 8 the discrepancies between the injected values and the recovered posteriors are within the range expected due to calibration uncertainties, and the discrepancies between the heterodyne and SplInter data are small enough to demonstrate that SplInter is a viable replacement for most targets, but that the heterodyne method should be retained for more accurate analysis.

4.2. Speed testing

The purpose of SplInter is to decrease the computational cost of targeted searches without significantly affecting sensitivity. We now consider the speed of SplInter in comparison with the heterodyne algorithm. The fundamental speed increase comes from the fact that we do not require the entire data bandwidth for our estimate of B_k. SplInter only requires a small frequency band of less than 1 Hz, whereas the heterodyne algorithm initially starts with a dataset containing the equivalent of 16 384 Hz. In addition to this, we can analyse the sources in parallel for each SFT, reducing overall file input/output time when alaysing multiple sources.

It is simplest to compare the total algorithm time taken per source, as the heterodyne algorithm simply takes the sources one at a time. However the SplInter execution time is not linear in the number of sources, so we also compare the total time per source, for one, ten,

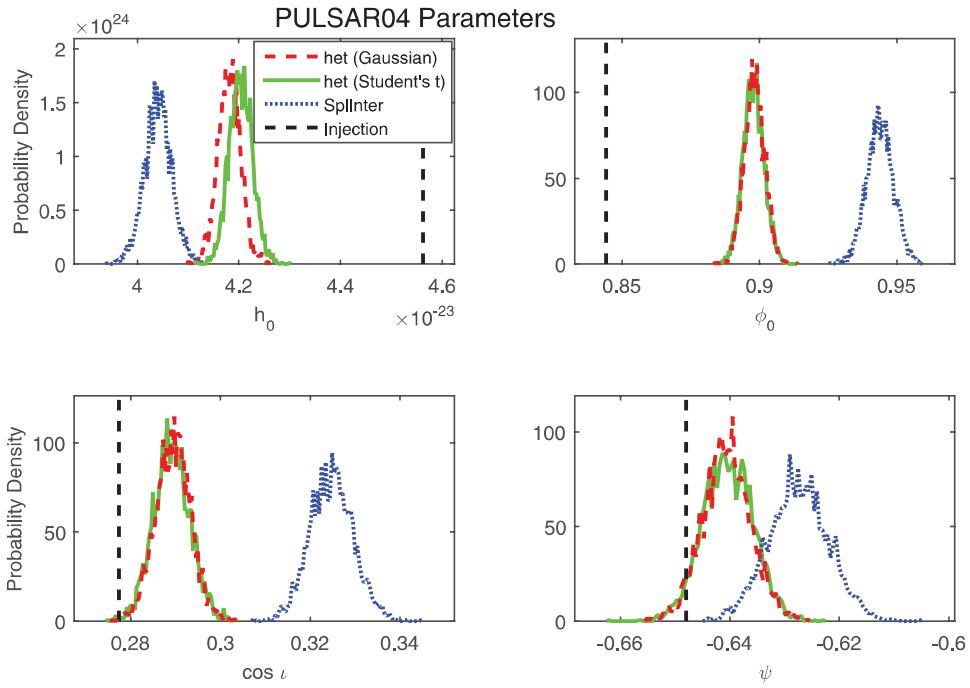


Figure 8. Posterior distributions on h_0, ϕ_0, $\cos\iota$ and ψ for hardware injection PULSAR4 using four months of LHO S6 data. The blue dotted line shows posteriors made using SplInter for the calculation of $B_{k/K}$, and the red dashed and green solid lines show the heterodyne algorithm with Gaussian and Student's t-likelihood distribution respectively. The vertical black line shows the injection value, which is slightly offset from the recovered signal due to calibration uncertainties.

one hundred and one thousand sources at a time. These tests use the mixed interpolation scheme, so that we gain accurate timing results, including the occasional use of the Fresnel approximation.

Table 3 and figure 10 show the amount of time taken to analyse the sources for a day of continuous data using both the heterodyne and SplInter routines. Table 3 also shows the computational improvement in CPU hours per source per hour of data. We see that the SplInter routine can improve the computational efficiency of the $B_{k/K}$ calculation by up to two orders of magnitude for single source input, and four orders of magnitude if we use multiple source input. These analyses were performed on the `atlas` computing cluster at the Albert Einstein Institute, Hannover.

The improvement in computational efficiency is not just limited to the B_k calculation stage. The lower cadence of the SplInter B_k data results in fewer data points containing a similar amount of information and hence faster calculations at the parameter estimation stage. This does however come with a cost: SplInter requires contiguous 30 min periods and lacks the flexibility of the heterodyne code in that respect.

5. Discussion

We have introduced *SplInter*, a new spectral interpolation method of calculating the down-sampled complex amplitude of a continuous wave signal with relative motion and source

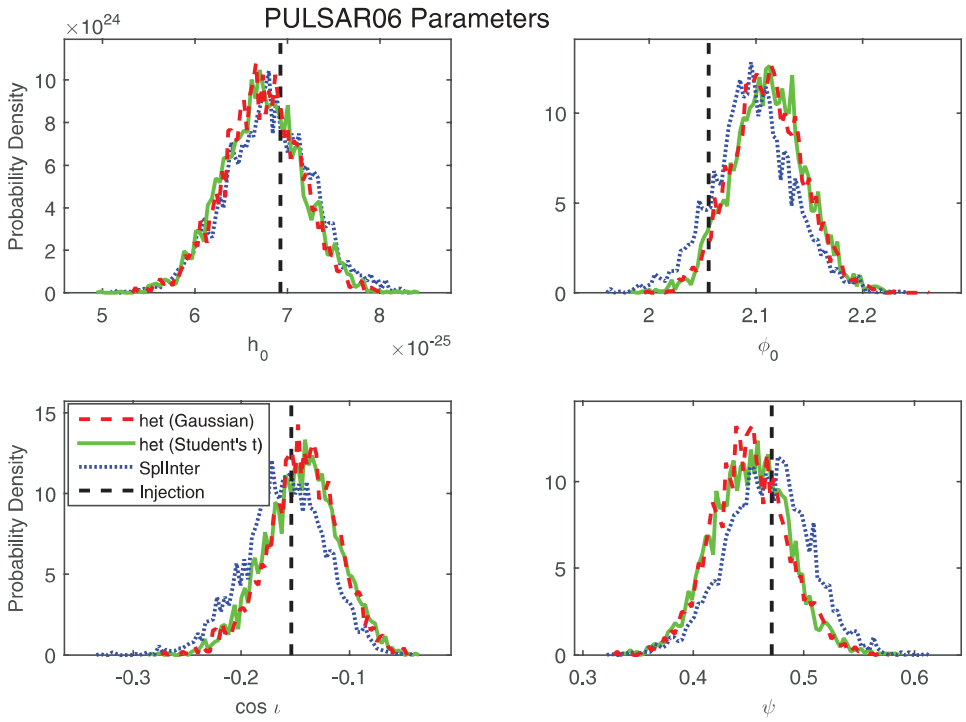


Figure 9. Posterior distributions on h_0, ϕ_0, $\cos\iota$ and ψ for hardware injection PULSAR6 using four months of LHO S6 data. Colour scheme as in figure 8.

Table 3. Median time taken to analyse sources for a day of data using heterodyne and Spectral Interpolation in seconds and CPU core hours per number of sources per hour of data for each interferometer.

	Heterodyne	SplInter			
Sources	1	1	10	100	1000
Time/day (s)	5.0×10^2	1.0	1.7	2.5	10.9
CPUh/N/h	5.8×10^{-3}	1.1×10^{-5}	1.9×10^{-6}	2.9×10^{-7}	1.2×10^{-7}

rotation effects removed. We have shown that this algorithm improves the computational efficiency of this part of the Bayesian targeted and $\mathcal{G}$-statistic pipelines by up to four orders of magnitude, and have explained how using longer time steps for $B_{k/K}$ with an estimate of the noise has a knock-on effect of improving computational speed in the parameter estimation stage. We have also shown that the SplInter routine performs well in comparison to the heterodyne routine in most cases, and that there is no significant drop in the recovered SNR. The increase in computational efficiency means that the search is a viable rapid follow-up pipeline for all-sky or directed search candidates, and preliminary results for this secondary use are in [18]. This method has been used in dual-harmonic searches for gravitational waves from spinning neutron stars [20], in which the spectral interpolation algorithm was found to improve the upper limit on J1748 − 2446ac by a factor of 1.7 compared to [2] by the use of the line removal routine.

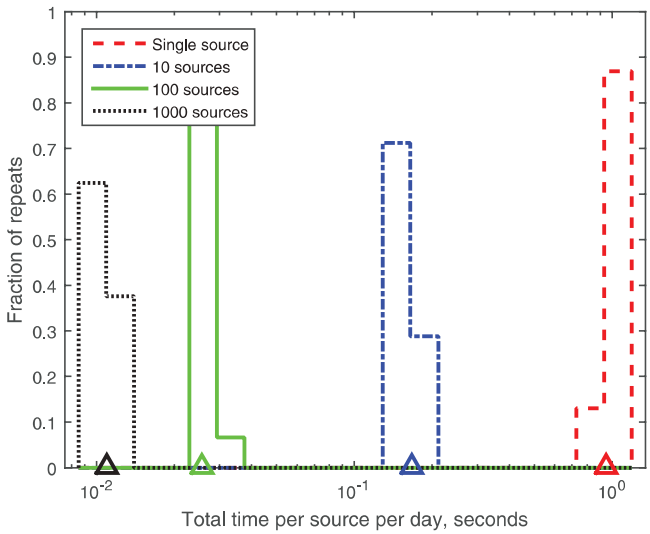


Figure 10. Histograms of the average time taken to analyse sources for a day of continuous data for the SplInter algorithm. The time taken by the heterodyne algorithm is 5.0×10^2 seconds per day per source, which would be over two orders of magnitude past the right hand limit of this figure. The different coloured histograms show the different number of sources used in each analysis. The horizontal axis markers denote the mean values for these times for the different numbers of sources.

Tests on the B_k output of the SplInter and heterodyne output have shown that the SplInter algorithm is not suitable for sources in binary systems with relatively short binary periods compared to the length of the SFT, this is as the frequency of the signal will alter significantly and non-linearly over the course of the SFT. Work is planned to provide a solution to this problem, which could include switching between the time and frequency domain to make shorter Fourier transforms in the cases of high frequency variability during the 30 min duration of the SFTs. The inverse FFT and FFT required to do this would not be computationally expensive, due to the efficiency of the FFT and inverse FFT algorithms. This method could be able to be used in a flexible way, calculating the required timestep for the new transforms for each SFT, meaning that the number of returned data points is optimal.

Acknowledgments

The authors would like to thank members of the Institute for Gravitational Research at the University of Glasgow and the LIGO Scientific Collaboration continuous waves group for constructive comments and discussion. In particular we would like to thank Damir Buskulic for useful comments in preparation of this manuscript. GSD is funded by a studentship from the Science and Technologies Facilities Council. MP and GW are funded by the STFC through grant number ST/L000946/1. This document has been assigned the LIGO Document Control Center number LIGO-P1500258.

References

[1] Aasi J *et al* 2014 Gravitational-waves from known pulsars: results from the initial detector era *Astrophys. J.* **785** 119

[2] Abbott B P *et al* 2010 Searches for gravitational waves from known pulsars with science run 5 LIGO data *Astrophys. J.* **713** 671–85
[3] Dupuis R and Woan G 2005 Bayesian estimation of pulsar parameters from gravitational wave data *Phys. Rev.* D **72** 102002
[4] Astone P, Colla A, D'Antonio S, Frasca S and Palomba C 2012 Coherent search of continuous gravitational wave signals: extension of the 5-vectors method to a network of detectors *J. Phys.: Conf. Ser.* **363** 012038
[5] Jaranowski P and Królak A 2010 Searching for gravitational waves from known pulsars using the $\mathcal{F}$ and $\mathcal{G}$ statistics *Class. Quantum Grav.* **27** 194015
[6] Dupuis R 2004 Bayesian searches for gravitational waves from pulsars *PhD Thesis* University of Glasgow
[7] Abbott B *et al* 2007 Upper limits on gravitational wave emission from 78 radio pulsars *Phys. Rev.* D **76** 042001
[8] Jaranowski P, Królak A A 2009 *Analysis of Gravitational-Wave Data* (*Cambridge Monographs on Particle Physics, Nuclear Physics, Cosmology* vol 29) (Cambridge: Cambridge University Press)
[9] Patel P, Siemens X, Dupuis R and Betzwieser J 2010 Implementation of barycentric resampling for continuous wave searches in gravitational wave data *Phys. Rev.* D **81** 084032
[10] Australia Telescope National Facility The ATNF pulsar database www.atnf.csiro.au/people/pulsar/psrcat/
[11] Manchester R N, Hobbs G B, Teoh A and Hobbs M 2005 The Australia telescope national facility pulsar catalogue *Astron. J.* **129** 1993–2006
[12] Press W H, Teukolsky S A, Vetterling W T and Flannery B P 1992 *Numerical Recipes: the Art of Scientific Computing* 2nd edn (Cambridge: Cambridge University Press) (available online http://apps.nrbook.com/c/index.html)
[13] Woan G 2000 *The Cambridge Handbook of Physics Formulas* (Cambridge: Cambridge University Press)
[14] Mendell G 2002 Short-time fourier transform specification *Technical Report* LIGO-T020043-00-W LIGO Hanford Observatory
[15] Abbott B *et al* 2004 Setting upper limits on the strength of periodic gravitational waves from PSR J1939 + 2134 using the first science data from the GEO 600 and LIGO detectors *Phys. Rev.* D **69** 082004
[16] Mendell G and Wette K 2008 Using generalized powerflux methods to estimate the parameters of periodic gravitational waves *Class. Quantum Grav.* **35** 114044
[17] Aasi J *et al* 2015 Characterization of the ligo detectors during their sixth science run *Class. Quantum Grav.* **32** 115012
[18] Davies G S 2015 Bayesian searches for continuous gravitational waves in the frequency domain *PhD Thesis* University of Glasgow
[19] Pitkin M, Gill C, Veitch J, Macdonald E and Woan G 2012 A new code for parameter estimation in searches for gravitational waves from known pulsars *J. Phys.: Conf. Ser.* **363** 012041
[20] Pitkin M, Gill C, Jones D I, Woan G and Davies G S 2015 First results and future prospects for dual-harmonic searches for gravitational waves from spinning neutron stars *Mon. Not. R. Astron. Soc.* **453** 4399–420



https://dspace.org/

A targeted spectral interpolation algorithm for the detection of continuous gravitational waves

Davies, Gareth S.; Pitkin, Matthew; Woan, Graham

2016-03
Institute of Physics (IOP)
https://doi.org/10.1088/1361-6382/34/1/015010
Davies GS, Pitkin M, Woan G. (2017) A targeted spectral interpolation algorithm for the detection of continuous gravitational waves, Classical and Quantum Gravity, Vol. 34, Issue 1, January 2017, Article number 015010

Downloaded from a DSpace repository. https://dspace.org/