PAPER • OPEN ACCESS

Adapting the PyCBC pipeline to find and infer the properties of gravitational waves from massive black hole binaries in LISA

To cite this article: Connor R Weaving *et al* 2024 *Class. Quantum Grav.* **41** 025006

View the article online for updates and enhancements.

Adapting the PyCBC pipeline to find and infer the properties of gravitational waves from massive black hole binaries in LISA

Connor R Weaving[1,*] ⓘ**, Laura K Nuttall**[1] ⓘ**, Ian W Harry**[1] ⓘ**, Shichao Wu**[2] ⓘ **and Alexander Nitz**[3,2] ⓘ

[1] University of Portsmouth, Portsmouth PO1 3FX, United Kingdom
[2] Max-Planck-Institut für Gravitationsphysik (Albert-Einstein-Institut), D-30167 Hannover, Germany
[3] Department of Physics, Syracuse University, Syracuse, NY 13244, United States of America

E-mail: connor.weaving@port.ac.uk

Abstract

The laser interferometer space antenna (LISA), due for launch in the mid 2030s, is expected to observe gravitational waves (GWs) from merging massive black hole binaries (MBHBs). These signals can last from days to months, depending on the masses of the black holes, and are expected to be observed with high signal to noise ratios (SNRs) out to high redshifts. We have adapted the PyCBC software package to enable a template bank search and inference of GWs from MBHBs. The pipeline is tested on the LISA data challenge's Challenge 2a ('Sangria'), which contains MBHBs and thousands of galactic binaries (GBs) in simulated instrumental LISA noise. Our search identifies all six MBHB signals with more than 92% of the optimal SNR. The subsequent parameter inference step recovers the masses and spins within their 90% confidence interval. Sky position parameters have eight high likelihood modes which are recovered but often our posteriors favour the incorrect sky mode. We observe that the addition of GBs biases the parameter recovery of masses

---

* Author to whom any correspondence should be addressed.

and spins away from the injected values, reinforcing the need for a global fit pipeline which will simultaneously fit the parameters of the GB signals before estimating the parameters of MBHBs.

1. Introduction

In preparation for the laser interferometer space antenna (LISA) [1] launch in the mid 2030s, considerable efforts are being made to design and develop the data analysis tools needed to extract and examine the numerous different types of gravitational wave (GW) signals expected to be found in the LISA band. LISA will probe the mHz GW spectrum, allowing us to explore sources such as massive black hole binaries (MBHBs) with total mass between $10^5 M_\odot$ and $10^8 M_\odot$ [2]; intermediate-mass black hole binaries with total masses in the range $10^2 M_\odot$–$10^5 M_\odot$ [3]; extreme mass-ratio inspiral (EMRI) and intermediate mass-ratio inspirals where coalescences have mass ratios of 10^{-6}–10^{-3} and 10^{-3}–10^{-1} respectively, with total masses in the range $10^3 M_\odot$–$10^7 M_\odot$ [4, 5], and more. Each of these systems, if detected and analysed properly, will help constrain and test a wide range of theories. For example, the effect of dark matter on a GW propagating from an MBHB or constraining the deviation from the Kerr metric using EMRIs (for example see [6]). Depending on the model of MBHB populations, the merger rate is estimated to be from O(2) to O(100) yr^{-1} [7–10].

In order to ensure that we can detect and analyse these signals, the LISA data challenge (LDC) [11, 12] generates mock LISA data with various GW signals injected into simulated LISA noise. The series of challenges will eventually replicate all of the complexities that the true LISA data will contain, such as gaps in the data; noise artefacts found within the data that will affect the ability to infer a system's parameters; realistic LISA spacecraft orbits; a time varying estimate of the noise to account for longer, GW signals. All of these issues will need to be addressed before LISA data can be analysed. Several groups are working on different aspects of LISA data analysis and their work is highlighted in [13–22].

In this paper, we present our pipeline which was developed to analyse the second LDC, referred to as 'Sangria' [23, 24]. The challenge consists of a training and blind data set where both the signals and the Gaussian instrumental noise are generated in the same way. We used the training data set to test and develop our pipeline as the number of signals with their injected parameters were given with the data. We then ran our analysis on the blind data, which contains an unknown number of signals. Our results for the blind data challenge were submitted to the LDC in October 2022. The blind challenge has now finished and the injected parameter values have been released publicly and can be accessed in [25].

There are two types of GW signals injected into Sangria. The first type of signal is galactic white dwarf binaries (GBs) [26] which are expected to be the most abundant source of GW signals present in LISA data. The accumulation of these signals forms a 'confusion noise' which dominates the 0.5–3 mHz range of the frequency spectrum. A subset of these are known as verification binaries [27], which are signals that are known in advance due to electromagnetic observations. The second type of signal in the data are MBHBs. These are generated using the IMRPhenomD waveform model [28, 29]. IMRPhenomD produces two GW polarisations which encode the effect a GW has on the relative distance between particles, the derivation of which is well known and can be found in [30]. These polarisations are then projected onto the LISA constellation to produce an arm response. The arm response from LISA is known to
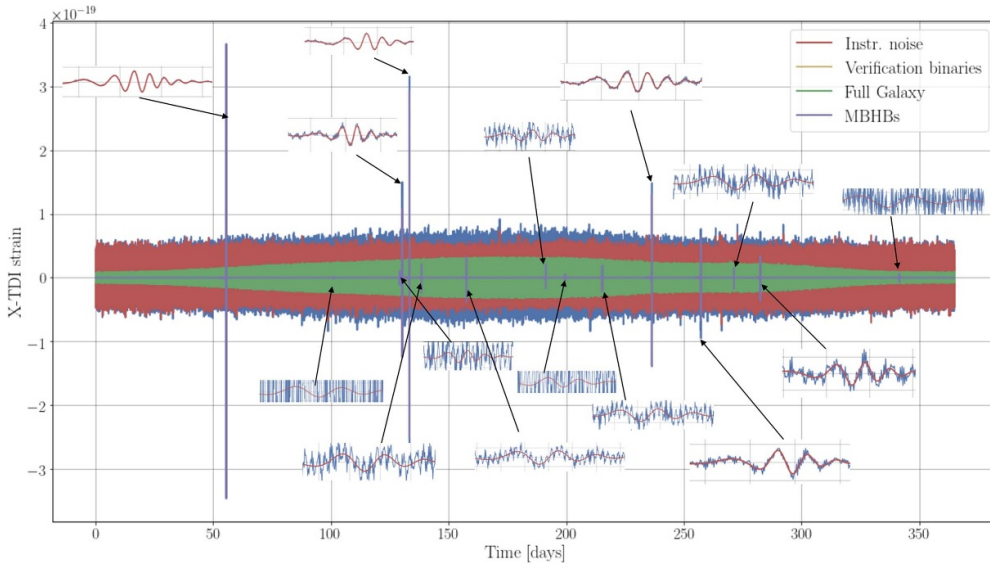
Figure 1. Timeseries of *X* TDI variable from the Sangria training data set. The data consists of a year's worth of data sampled at 5 s, with 15 MBHB signals injected into the data. This data also contains Gaussian instrumental noise and 30 million GBs, of which 17 are VBs. Reproduced with permission from [23].

be dominated by laser frequency noise and therefore time delay interferometry (TDI) [31, 32] techniques must be used in order to extract GW information from the raw data streams. The Michelson TDI variables *X*, *Y* and *Z* are provided in this data challenge.

The data set spans a year's worth of data sampled at 5 s, which is the expected sample rate for LISA. Figure 1 shows a time series of the *X* TDI data stream of the Sangria training data set. An overview of previous LDCs can be found in [33–35].

The tools used in our analysis have been developed within PyCBC [36]. PyCBC is an open source GW data analysis software package and is one of the main pipelines used to detect and characterise compact binary mergers from LIGO, Virgo and KAGRA (LVK) detectors [37–39, 54]. PyCBC has been used to detect and analyse $\mathcal{O}(100)$ GW signals from compact binary mergers to date, an overview of which can be found in [40–42]. We have adapted the techniques developed for LVK searches and applied them to LISA-like signals, in this case MBHBs.

It is important to note that there is far more work to do to achieve a reliable analysis of MBHBs in realistic conditions. The tools made available through this paper are a first attempt at the bigger picture solution, which will require many more iterations before being considered 'production ready'.

The paper is laid out as follows. In section 2, we introduce PyCBC and the components which make up our pipeline. We give the basic details needed to understand how PyCBC approaches detecting and recovering GW system parameters. In section 3, we outline how we have adapted PyCBC in order to detect and infer MBHBs. In section 4, we present the results our search and each stage of inference leading to the analysis of the blind Sangria data. Finally in section 5 we discuss the challenges in developing these tools, future work and our conclusions.

2. PyCBC: search and inference

We can split the PyCBC analysis pipeline up into two distinct sections, the search and the inference. The search portion of the pipeline is responsible for identifying candidate GW signals within the data, and the inference portion returns a statistical estimate of the parameters of the GW signals identified. Below we outline which tools within PyCBC have been used and adapted for MBHB signals.

2.1. The search

The main component of the search is the application of matched filtering. The matched filter signal-to-noise ratio (SNR) is designed to indicate when a particular signal we are searching for could be present in data [43]. For data in the form $d(t) = h(t) + n(t)$, where $h(t)$ is a modelled signal we are looking for and $n(t)$ is the detector noise, we can choose a linear filter, $F(t)$, such that the output SNR is maximised for the signal $h(t)$. We can represent this filtering acting in the time or frequency domain, as shown in equation (1),

$$\phi = \int_{-\infty}^{\infty} F(t)\, d(t)\, \mathrm{d}t = \int_{-\infty}^{\infty} \tilde{F}^*(f)\, \tilde{d}(f)\, \mathrm{d}f \tag{1}$$

where tilde represents the Fourier transform. The optimal SNR defined as $\rho_{\mathrm{opt}} = S/N$, where S is the value of ϕ where a signal $h(t)$ is present, and N is the root mean squared value of ϕ with no signal.

If we assume the noise to be Gaussian and stationary, we can define the one-sided power spectral density (PSD) via S_n in equation (2),

$$\langle \tilde{n}(f)\, \tilde{n}^*(f') \rangle = \frac{1}{2} S_n(|f|)\, \delta(f - f') \tag{2}$$

where $f \geqslant 0$ and the square brackets represents the ensemble average. By introducing the inner product of two real time series $a(t)$ and $b(t)$ in equation (3)

$$(a|b) = 4\Re \int_0^{\infty} \frac{\tilde{a}^*(f)\, \tilde{b}(f)}{S_n(f)}\, \mathrm{d}f, \tag{3}$$

we can express the optimal SNR as

$$\rho_{\mathrm{opt}} = \frac{(u|h)}{(u|u)^{1/2}}, \quad \text{where} \quad \tilde{u}(f) = \frac{1}{2}\tilde{F}(f)\, S_n(f). \tag{4}$$

In order to maximise ρ_{opt} it is clear that $\tilde{F}(f)$ should be proportional to $h(t)$

$$\tilde{F}(f) = A\, \frac{\tilde{h}(f)}{S_n(f)}, \tag{5}$$

where A is some arbitrary constant. This results in the optimal SNR taking the form

$$\rho_{\mathrm{opt}} = (h|h)^{1/2}. \tag{6}$$

In order to filter the data for a given signal, we would compute the matched filter SNR in equation (7) which is distinct from ρ_{opt}

$$\rho_{\text{mf}} = \frac{(d|h)}{(h|h)^{1/2}}. \tag{7}$$

To generate an SNR timeseries, we can inverse Fourier transform the matched filter as shown in equation (8),

$$(d|h)(t) = \int_{-\infty}^{\infty} \frac{\tilde{d}(f)|\tilde{h}(f)|^*}{S_n(f)} \exp[-2\pi ift]\,df, \tag{8}$$

where t is time. Equation (8) shifts the waveform $h(t)$ to the defined time. By computing across the entire time of the data, the SNR timeseries returned indicates at which times the signal $h(t)$ is likely to be present by peaks in the SNR timeseries.

We use a model for signals, $h(t)$, which we will now refer to as templates, that represents the response of a detector given a GW passing through the instrument. PyCBC can project the GW polarisations produced by varying waveform models onto different ground-based detectors to return an arm response. For some GW emitting systems, many parameters are required to define the full signal response. Given such a broad parameter space of values to explore, we require a large number of templates to search the data in order to potentially detect a GW. PyCBC generates template banks which can contain $\mathcal{O}(100\,000)$ templates for an LVK-like search to explore the parameters which affect GW emission from compact binary coalescing systems. These parameters are split into intrinsic, which define the masses of both compact objects and their spins, and extrinsic, which describes the system's sky position, the luminosity distance to the binary and the orientation of the binary relative to the detector [44]. Details of the various methods used to generate template banks can be found in [45–49]

When generating a template bank, the match, as defined in equation (9), is computed between templates

$$\mathcal{M}(h_1, h_2) = \max_{\phi_{\text{c}}, t_{\text{c}}} [(h_1|h_2)], \tag{9}$$

given two templates h_1 and h_2, the match tells us how similar to each other these templates are while maximising over the coalescence time t_{c} and the phase at coalescence ϕ_{c}. Templates are first normalised such that the inner product gives $(h_1|h_1) = 1$. Using the normalised templates in the match returns a number in the range [0,1] where 1 states that the two normalised templates are identical.

The matched filter is then applied to all templates and the data we are searching over, giving an SNR timeseries per template. Each timeseries is then passed to a clustering algorithm, to return the highest SNR template per time in the data. A trigger is defined by a peak in the SNR timeseries reaching, or surpassing, a certain threshold. Additional steps are then applied to search for coincidences over multiple detectors and calculate their significance [37, 50].

2.2. The inference

PyCBC can infer signal parameters through sampling the posterior probability defined by Bayes theorem

$$p(\underline{\theta}|d) = \frac{\mathcal{L}(d|\underline{\theta})p(\underline{\theta})}{p(d)} \tag{10}$$

where $\mathcal{L}(d|\underline{\theta})$ is the likelihood of the data d given the parameters $\underline{\theta}$, $p(\underline{\theta})$ is the prior of the parameters and $p(d)$ is the evidence and $\underline{\theta}$ represents an integer N number of parameters $\underline{\theta} = \{\theta_1, \ldots, \theta_N\}$. The main component of equation (10) to compute is the likelihood which under the assumptions of Gaussianity and stationarity takes the form:

$$\log \mathcal{L}(d, \underline{\theta}) \propto (d|h_\theta) - \frac{1}{2}(h_\theta|h_\theta) - \frac{1}{2}(d|d), \tag{11}$$

where $h_\theta \equiv h(\underline{\theta})$.

We can speed up likelihood calculations by enforcing some assumptions about our GW

$$\tilde{h}(f; \underline{\theta}, \phi) = A(f, \underline{\theta}) \exp[i(\Psi(f, \underline{\theta}) + \phi)]. \tag{12}$$

We use IMRPhenomD (2, 2) mode only signals, and express our GW strain, $\tilde{h}(f)$, as a product of smoothly varying amplitude and phase factors, where f is frequency, $A(f, \underline{\theta})$ is the amplitude of the GW, Ψ is the phase of the waveform and ϕ is some arbitrary phase constant.

With the GW strain expressed in this form, we use the application of heterodyning (sometimes referred to as relative binning) which assumes a smooth signal which varies in amplitude and phase. Heterodyning is a signal processing technique used to separate data into slowly and rapidly varying frequency components. Our inference application of heterodyning allows us to greatly speed up likelihood calculations by first finding a high likelihood template with our template bank search. The corresponding template is then used as a reference to explore the region of high likelihood. A derivation of this method applied to GWs can be found in [51–53].

In order to explore the parameter space, there are several samplers in PyCBC which implement different methods, including Markov chain Monte Carlo and nested sampling. An overview of PyCBC's inference setup can be found in [54].

3. Adapting PyCBC's search and inference for MBHBs

3.1. Templates

In order to perform a matched filter search, we need to be able to generate templates that represent LISA's arm response given a GW signal passing through the instrument. The data we are searching over is given in terms of TDI variables; we therefore must put the templates in this format.

A vital part of this development is the use and implementation of the open source software BBHx waveform generator [28, 29, 55–60], which we use to generate the expected strain from MBHB systems. BBHx generates the GW polarizations, projects them onto LISA to form an arm response and then creates the TDI channels. BBHx implements the IMRPhenomHM [60, 61], an aligned spin waveform model which produces GW polarisations based on IMRPhenomD, but includes other sub-dominant modes. In this work, we set the IMRPhenomHM to generate just the (2, 2) mode in order to match the Sangria convention. We transform both the data and the TDI templates into noise independent data streams referred to as A, E and T [62] via the transforms in equation (13)

$$A = (Z - X)/\sqrt{2}, \quad E = (X - 2Y + Z)/\sqrt{6}, \quad T = (X + Y + Z)/\sqrt{3}. \tag{13}$$

The data streams A and E have equal SNR across the entire LISA frequency band. This is not true for the T TDI stream as at lower frequencies the SNR contribution from T does not

Table 1. The parameter ranges used to generate the template bank. This covers the full range of potential systems that are defined within the conventions of the Sangria challenge. M_T is detector frame total mass of the black holes; q is mass ratio where $m_1 > m_2$ and m_1 is the primary mass and m_2 the secondary; a_1 and a_2 are the aligned spins of the two black holes; β is ecliptic latitude; λ is ecliptic longitude; ψ is polarisation angle; ι is inclination.

Parameter	Lower bound	Upper bound
M_T ($M_\odot$)	2×10^5	2×10^7
q	1	4
a_1, a_2	-0.99	0.99
β (rad)	$-\pi/2$	$\pi/2$
λ (rad)	0	2π
ψ, ι (rad)	0	π

compete with A and E. However, T does contribute SNRs that can be greater than A or E at higher frequencies in the LISA band and is therefore included in the inference analysis [63]. BBHx generates templates A, E and T in the same convention as in equation (13).

3.2. Template bank

This work explores whether a template bank based search is a viable approach to finding MBHBs in LISA data, so to reduce complexity we only perform the search on the A TDI stream. We generated a template bank for the A TDI data stream, which was used for the filtering stage of the search on just the A TDI data. As the A and E TDI streams have equal SNR across the LISA frequency band, searching through just the A or just E data streams would be equivalent. We do lose power when searching over one instead of both of the TDI streams, but due to the SNRs of these signals being so large, we find that the information lost is negligible. Due to the SNRs of MBHBs being so large, we populated a less dense template bank when compared with LIGO template bank searches to balance accuracy of the signal trigger time versus computation cost of producing the bank.

For the bank generation we use stochastic placement [64–66]. Specifically we use the PyCBC algorithm 'brute bank' with the parameters defined in table 1 and a minimal match of 0.9 which resulted in a bank size of 50 templates. The minimal match defines how densely the parameter space is sampled. For example, with a minimal match of 0.9, if any newly generated template matches with any other template in the bank above the minimal match, it is rejected and another template is generated. The process repeats until a pre defined number of templates have been rejected from the bank placement.

The luminosity distance D_L is marginalised over in the matched filter calculations and is set to an arbitrary value when generating templates. The time of coalescence t_c is sampled with one month from the beginning and end of the year removed. This is to ensure no edge corruption artefacts are present in the data when performing matched filtering, which could cause false triggers.

When computing the match, maximising equation (9) over time of coalescence t_c implies that we do not take into consideration the varying response function from LISA over time. LISA's response function varies over time due to the antennas sky position changing within the duration of the MBHB signal. However, because the majority of the SNR from merging

Table 2. A list of all parameters and their ranges given to the prior, $p(\theta)$. M_c, q, a_1, a_2, t_c, ψ and D_L are distributed uniformly. β, λ and ι are distributed isotropically. t_{ref} is the coalescence time found in the search.

Parameter	Lower bound	Upper bound
M_c ($M_\odot$)	87 055	8 705 505
q	1	4
a_1, a_2	-0.99	0.99
t_c (s)	$t_{ref} - 2$ days	$t_{ref} + 2$ days
β, λ, ψ (rad)	0	$\pi/2$
ι (rad)	0	π
D_L (Mpc)	10^3	10^6

MBHBs is within the final $\mathcal{O}(hour)$, and these systems have SNRs $\mathcal{O}(1000)$, we are safe to assume LISA is approximately stationary during this time period.

As t_c is directly related to LISA's orbital position in the case of BBHx, we ensure that the template bank can generate templates at different orbital positions by varying t_c. The actual recovery of t_c is done via the filter output time series from the inner product, shown in equation (8), as t_c is not an explicit template bank parameter.

3.3. Sampling, priors and sky folding

We sample the heterodyne likelihood using the nested sampling tool Dynesty [67]. Table 2 outlines the parameters used with their corresponding distributions in the inference. M_c is the chirp mass of the system [68].

The importance of multi-modality and degeneracies when considering extrinsic parameters are summarised in detail in [59] and are vital to understand for our application of sampling we introduce here. Considering a (2, 2) mode signal within the LISA frame, and applying the frozen low-f approximation which neglects both time and frequency dependence in the instrument response, there are eight exact degeneracies for sky location. Four longitudinal at $\{\lambda_L + \frac{\pi}{2}(0, 1, 2, 3), \psi_L + \frac{\pi}{2}(0, 1, 2, 3)\}$ and two latitudinal at $\{\pm\beta_L, \pm\cos\iota_L, \pm\cos\psi_L\}$, where the subscript L represents the extrinsic parameters in the LISA reference frame

$$
\begin{aligned}
\lambda_L^{(r)} &= \lambda_L, \\
\beta_L^{(r)} &= -\beta_L, \\
\psi_L^{(r)} &= \pi - \psi_L, \\
\iota_L^{(r)} &= \pi - \iota_L, \\
\phi_L^{(r)} &= \phi_c.
\end{aligned}
\tag{14}
$$

The BBHx waveform model that we use in this work, uses the frozen approximation and frequency-dependent instrument response which neglects LISA's motion. This breaks the eight-fold degeneracy into an approximate two-fold degeneracy in the reflected mode shown in equation (14).

Instead of sampling the entire parameter space, we restrict our sky location parameters to one of eight octants of the sky. After drawing a sample from our defined octant, we compute the

likelihood at the other seven modes. We marginalise over the eight sky modes by summation during the inference process. We then normalise them to sum to one. A post-processing step is then implemented to 'unfold' the sky positions by randomly selecting one of the eight sky modes based on the likelihood of each. This effectively samples the entire parameter space without having to sample other parts of the sky as we compute the likelihood directly at the analytical location of the other sky modes.

All samples for inference are taken within the LISA reference frame, not the Solar System barycentre (SSB) frame, even though the injected parameters in Sangria are all with reference to the SSB frame. The techniques described above for sampling the sky position are used in every inference run in this work.

3.4. PSDs

When GBs are included in the data, we assume them to be part of the noise $n(t)$. As there are several thousands of these signals of varying amplitude and phase, we can justify their accumulation to be Gaussian (but not stationary) by the central limit theorem. When constructing the PSDs for all three TDI variables, we implemented Welch's method over 31 segments of the year's worth of data. This gave frequency bins with a resolution of 10^{-6} Hz, which were narrow enough to model the structure of the PSD accurately. The same number of segments is used to generate all other PSDs for the different inference stages and the search. We generate a PSD per TDI channel at each inference stage.

By constructing the PSDs in this way we have ignored the fact that the PSDs are time-dependent and therefore cannot be stationary. This is also true in the case where the GBs are not included in the noise due to LISA's motion across the sky. We prioritised the LISA high-frequency information over the entire year's worth of data by reading all of the data and using 31 segments instead of a quasi-stationary PSD which is estimated on smaller chunks of data.

4. Results

In this section we present the results of our search and multiple inference stages. The associated data release which contains all of the code and instructions on how to reproduce the results presented can be found in [69]. It contains the results for all analysis of the search and inference in all of their various stages. Documentation for ongoing developments in PyCBC for LISA related searches and inference can be found in [70].

4.1. Search results

To quantify how effective the template bank search was, we computed the match between the highest SNR template per trigger against the injected waveform parameters. All matches are greater than 0.92 and are presented in table 3 for each candidate.

The template parameters from the search and the corresponding injected waveform parameters for signal 0 can be found in table 4. We attempted to produce template banks at higher minimal matches but the time to complete was $\mathcal{O}(days)$ instead of $\mathcal{O}(minutes)$ in the 0.9 minimal match case. A template bank search as we have applied it worked exactly how we needed it when considering the balance of computational cost verses accuracy of the returned template.

Table 3. The match defined in equation (9) between the highest SNR template per trigger and the waveforms injected value of the template bank search over the Sangria blind data.

Signal index	Match
0	0.945 10
1	0.926 66
2	0.973 44
3	0.940 91
4	0.945 62
5	0.924 39

Table 4. Comparison with template parameters found from the search verses the injected waveform parameters of signal 0 in the Sangria blind data.

Parameter	Injected	Recovered
m_1 ($M_\odot$)	1.76×10^6	2.02×10^6
m_2 ($M_\odot$)	1.30×10^6	7.20×10^5
a_1	0.51	0.27
a_2	0.14	-0.09
t_c (s)	6712 124	6712 065
ι (rad)	1.31	2.27
ψ (rad)	1.06	5.51
β (rad)	-0.35	0.52
λ (rad)	2.31	2.78
ϕ_c (rad)	4.23	3.14

4.2. Inference results

In this section, we present our results from the inference. All of the runs have the reference parameters for the heterodyne likelihood set to the injected parameter values. Each inference run uses all three TDI data streams. Running the heterodyne likelihood with the reference values found in the search gave no significant bias when comparing to the reference parameters being set to the injected signal values.

When building the inference tools for this challenge, there were several stages of additional complexities which were successively built upon to ensure that each step performed as expected. We performed inference with increasing complexity as follows:

- Injections with BBHx in zero noise.
- Injections with BBHx in Sangria training noise without GBs.
- Injections with BBHx in Sangria training noise with Sangria training GBs.
- Sangria blind data.

When generating data with BBHx injections for the with and without GB cases, we used the Sangria training data set. The Gaussian LISA noise and GBs for Sangria blind are generated in the same way as the training data set, but are unique to the blind data set. The Sangria data has waveform injections that were created with the LDC code base and not injected with BBHx.
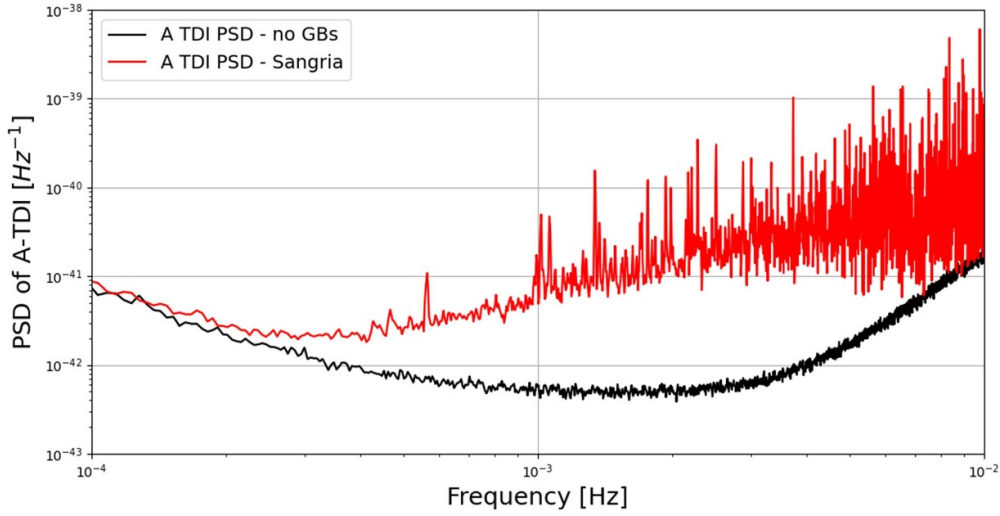
Figure 2. A figure showing the different PSDs we use in the results section. The red curve is the *A* TDI PSD of the Sangria data and the black curve is the *A* TDI PSD of Sangria without GBs.

We will now present the results of each stage of this development. The parameters used to create the BBHx injections are identical to that of the signals in the Sangria blind data. We ran analysis on all signals injected with BBHx and Sangria blind, but will concentrate our discussion on the first signal (from time zero) which we refer to as signal 0. We will distinguish the Sangria blind injection of the first signal as 'blind 0' as it is not produced with BBHx and is in the Sangria blind noise and GBs data. Quoted parameters found in the inference are the maximum log-likelihood sample from the posterior.

All of the BBHx inference runs were performed with the heterodyne parameters set to the true injected parameters. The final Sangria runs reference parameters were set to the parameters found in the search section of the pipeline. Figure 2 shows the PSDs used for the first three stages before analysing the Sangria blind data. The same sampler settings are used for each analysis to ensure a fair comparison.

For each stage of development, we performed two sets of inference. One set of inference contained only signal 0 injected into the data, shown in blue in subsequent figures, and the other contained all six signals from Sangria, shown in orange. All injections are created in a year's worth of data sampled at 5 s intervals to match the convention of the Sangria challenge.

Note that all of these parameters are given in the detector frame. This is the standard return of BBHx. For the interest of those wanting to inspect parameters in the source frame, the redshift for signal 0 injections is $z = 3.72$. Finally, we highlight the degeneracy, $h(\underline{\theta}, \psi) = h(\underline{\theta}, \psi + \pi)$, which can be observed throughout the results section.

4.2.1. Zero noise, BBHx injection. Figure 3 presents the recovered intrinsic parameter posteriors for this analysis, and figure 4 the extrinsic parameters, along with their 90% confidence intervals. The black PSD curve from figure 2, was used to scale the signals, to give SNRs of comparable sizes to subsequent analyses. We can see for the single and multiple injection cases
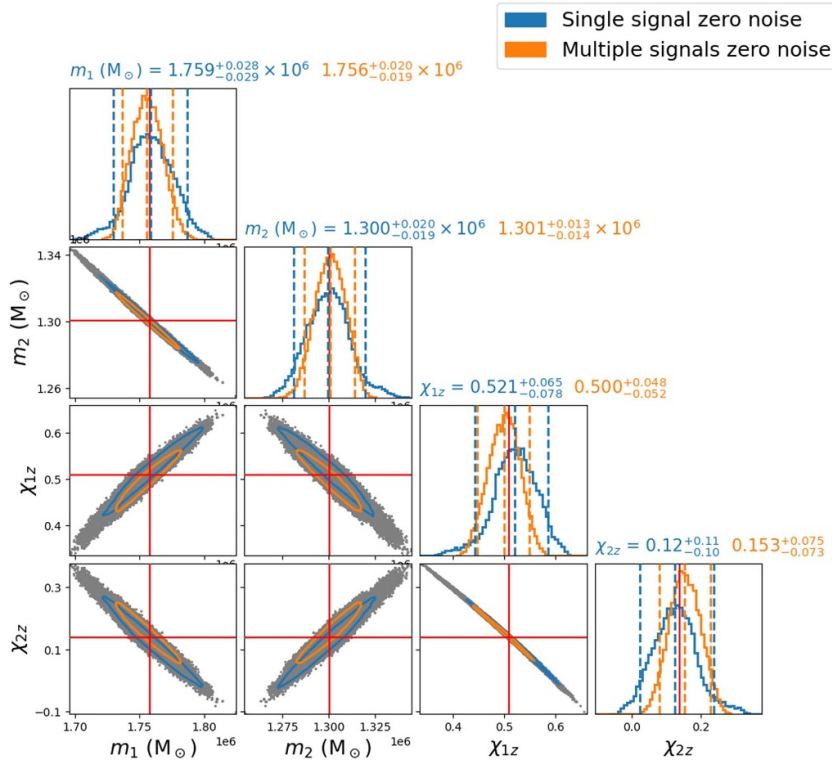
Figure 3. A corner plot showing the recovered intrinsic parameters posteriors of signal 0 in zero noise. The blue curves represent a year's worth of data containing only signal 0. The orange curves represent data containing all six signals. The red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

that the intrinsic parameters are recovered within the 90% confidence interval of the injected values. However, most extrinsic parameters do not fall within the 90% confidence interval; the sky position β has found the correct sky mode but the injected value falls just outside the peak of both single and multiple signal runs. The correct mode for λ is also recovered, however, in both the single and multiple signal cases the 90% confidence intervals do not capture the injected value.

4.2.2. Sangria noise without GBs, BBHx injections. These signals are injected into Gaussian, stationary noise using the Sangria training data. The black curve in figure 2 is the PSD used for this stage.

As figure 5 shows, the inclusion of Gaussian noise when comparing to the zero noise case for both multiple and single injections has no negative bias to the inferred intrinsic parameters. All intrinsic parameters fall within the 90% confidence interval in both cases. The extrinsic parameter recovery, shown in figure 6, shows no notable differences when comparing to the zero noise case. We do start to see some preference to other likelihood modes in sky position
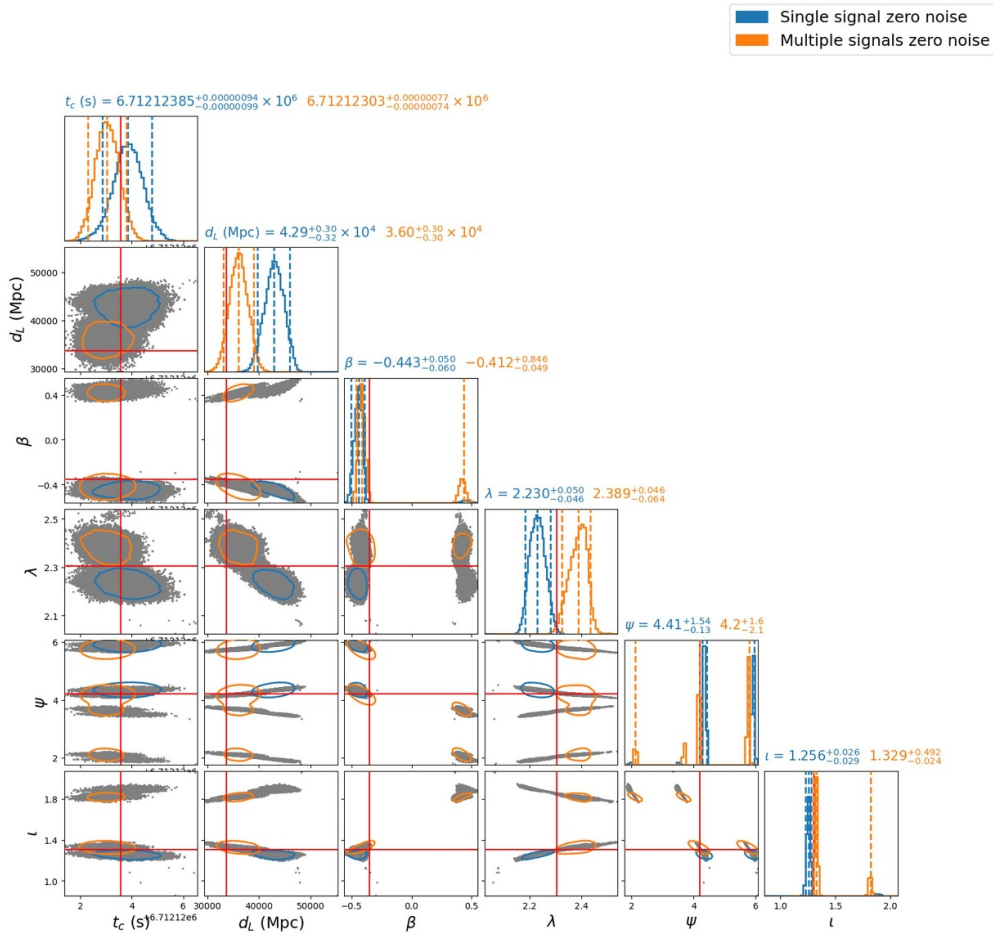
Figure 4. Extrinsic parameter posterior corner plot of injected BBHx signal in zero noise for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

as in two cases (signals 2 and 4) the inference preferred the other high likelihood mode of β instead of the true injected mode. This is true in both the single injection case and the multiple injection case.

4.2.3. Sangria noise with GBs, BBHx injections.   The red curve in figure 2 is the PSD used for this stage and contains all of the noise from the GBs. We see that this significantly changes the shape of the PSD in the 10^{-3}–10^{-2} Hz region of frequency space.

The posteriors in figures 7 and 8 are broader which is expected given the noise curve with GBs present. Figure 7 demonstrates that the intrinsic parameters still fall within the 90% confidence interval.
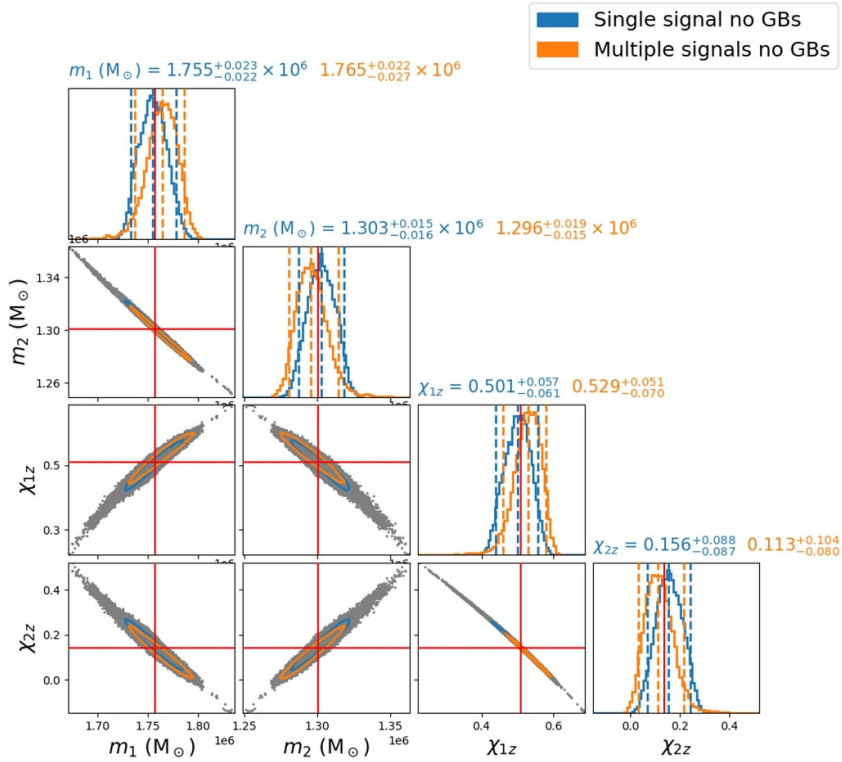
Figure 5. Intrinsic parameter posterior corner plot of injected BBHx signal(s) in Gaussian stationary noise without GBs for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

GBs have had the most significant effect on inference thus far as the peaks in the posterior likelihood have shifted from the injected value when compared to the no GBs case. We also see a lot more structure in the posteriors that includes GBs. In table 5 we have calculated the percentage of samples from the posterior with a higher likelihood than that of the likelihood of the injected waveform. If the assumptions of Gaussianity and stationarity are valid, these values for a collection of posteriors from different signals/injections should be uniformly distributed. We see for the case of Gaussian stationary noise without GBs that the distribution of values does appear to be uniform. The test performed here would have benefited from $\mathcal{O}(100)$ posteriors to properly conclude that the distribution is uniform. When including GBs, we can see that the distribution is no longer uniform. The effect of GBs being non-stationary therefore has biased the inferred parameters away from the injected value. Even with only six signals we can clearly see the bias when the data contains GBs and when inspecting the full Sangria
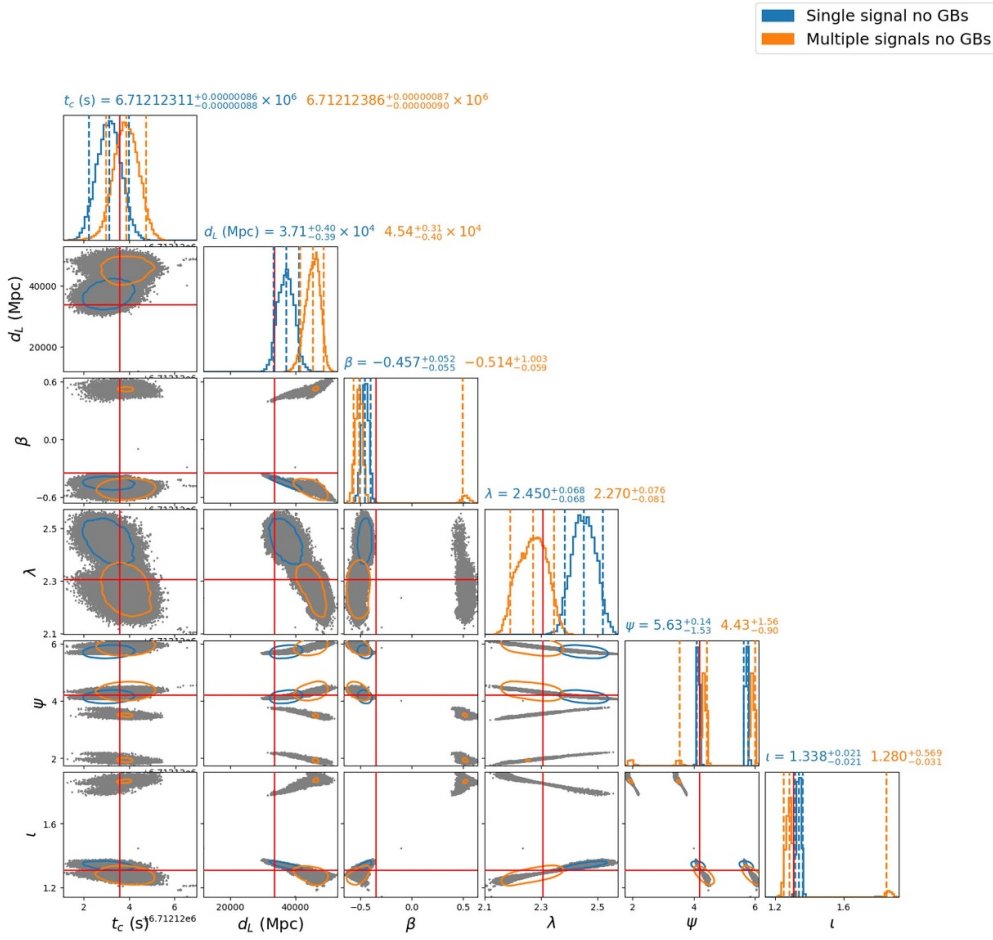
Figure 6. Extrinsic parameter posterior corner plot of injected BBHx signal(s) in Gaussian stationary noise without GBs for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

data. We have presented the other five signal parameter injections in appendix. We can clearly see that figures A1–A5 show a bias in intrinsic parameter recovery as signal 0.

As for the extrinsic parameters for signal 0, the inferred values represent the injected waveform well. This is not true for the other signals in the data as most struggle to find the correct sky mode for β and/or λ.

4.2.4. Sangria blind. The full parameter recovery of blind 0 in Sangria is presented in table 6, which also displays the injected parameter values. The intrinsic parameters in figure 9 fall within the 90% confidence interval. However, this is not true for all signals in the data set.
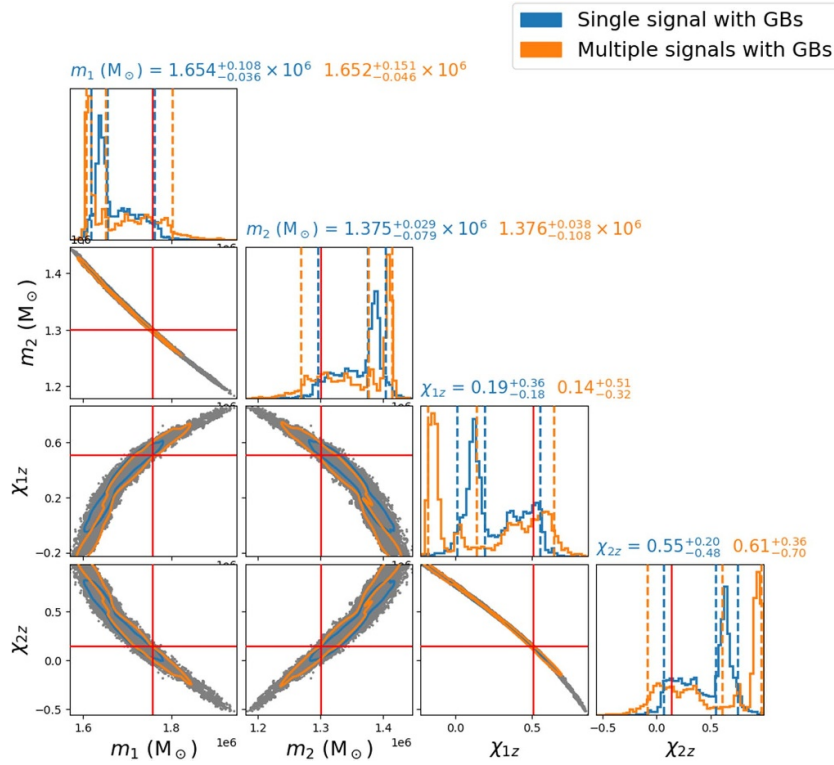
Figure 7. Intrinsic parameter posterior corner plot of injected BBHx signal(s) in Gaussian stationary noise with GBs for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

There is one case where all intrinsic parameters peak at roughly the 88% confidence interval. Overall, the peaks in likelihood are far less prominent when compared to the previous cases.

Extrinsic parameters also have far less prominent peaks than previous cases and suffer from the same sky position degeneracy issues where the wrong mode is commonly favoured. For half of the cases, D_L is recovered within the 90% interval and for all cases, t_c and ι are recovered in the 90% confidence interval.

Figure 10 shows the extrinsic recovery of our reference signal and is the only example out of the six signals that recovers the correct mode for λ. Extrinsic parameter recovery would be greatly improved when analysing signals with higher order modes [59]. As this data challenge uses just (2, 2) mode injections, it was expected that the extrinsic parameters would be the most difficult to recover correctly.
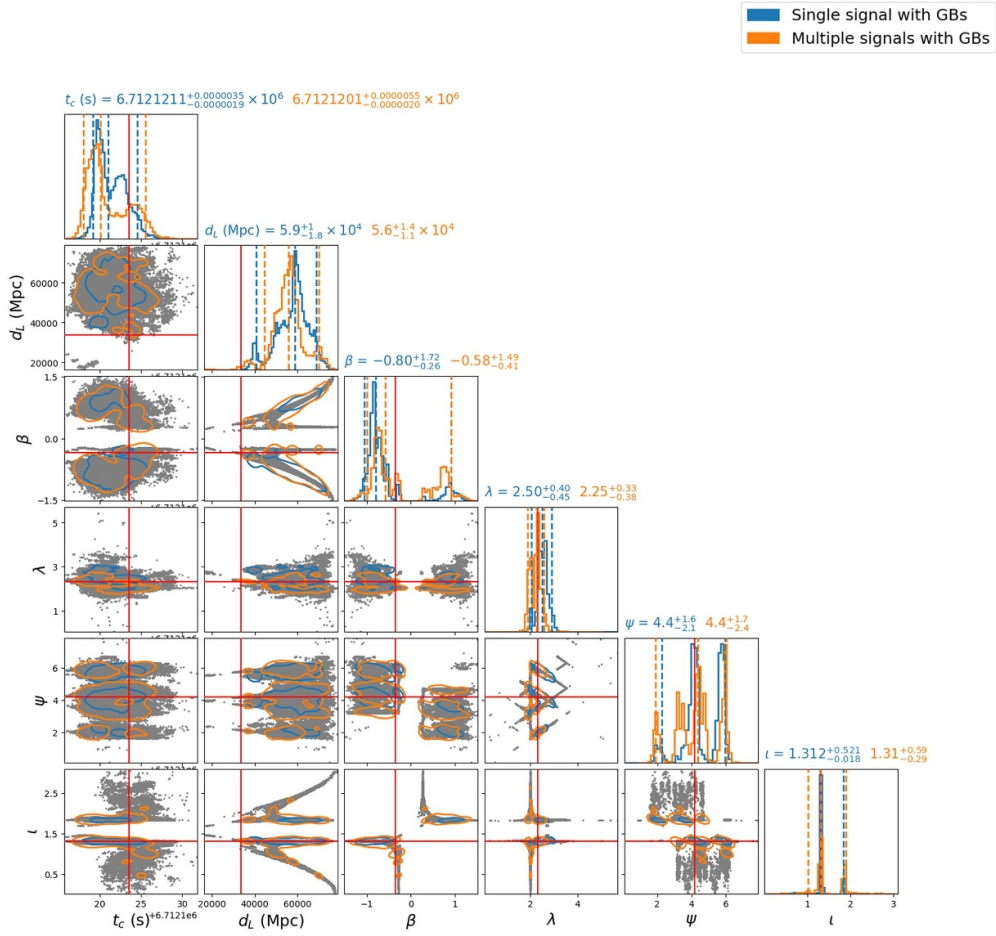
Figure 8. Extrinsic parameter posterior corner plot of injected BBHx signal(s) in Gaussian stationary noise with GBs for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

Table 5. Each percentage represents the amount of samples whose likelihood value is greater than the likelihood of the corresponding injected waveform.

Signal	Sangria noise without GBs	Sangria noise with GBs	Sangria blind
0	78%	34%	93%
1	41%	84%	98%
2	20%	44%	99%
3	2.7%	84%	100%
4	36%	84%	81%
5	18%	6.7%	86%

Table 6. The injected and recovered parameters for signal 0 in the blind Sangria dataset. The recovered values are stated with the 90% confidence regions.

	Blind 0	
Parameters	Injected value	Recovered value
m_1 ($M_\odot$)	1.76×10^6	$1.70^{+0.13}_{-0.11} \times 10^6$
m_2 ($M_\odot$)	1.30×10^6	$1.34^{+0.083}_{-0.089} \times 10^6$
a_1	0.51	$0.39^{+0.32}_{-0.56}$
a_2	0.14	$0.29^{+0.66}_{-0.49}$
t_c (s)	$6712\,124$	$6712\,466^{+103}_{-793}$
ι (rad)	1.31	$1.32^{+0.52}_{-0.02}$
ψ (rad)	1.06	$1.44^{+1.03}_{-1.08}$
β (rad)	-0.35	$-0.05^{+0.77}_{-0.59}$
λ (rad)	2.31	$2.11^{+2.40}_{-0.79}$
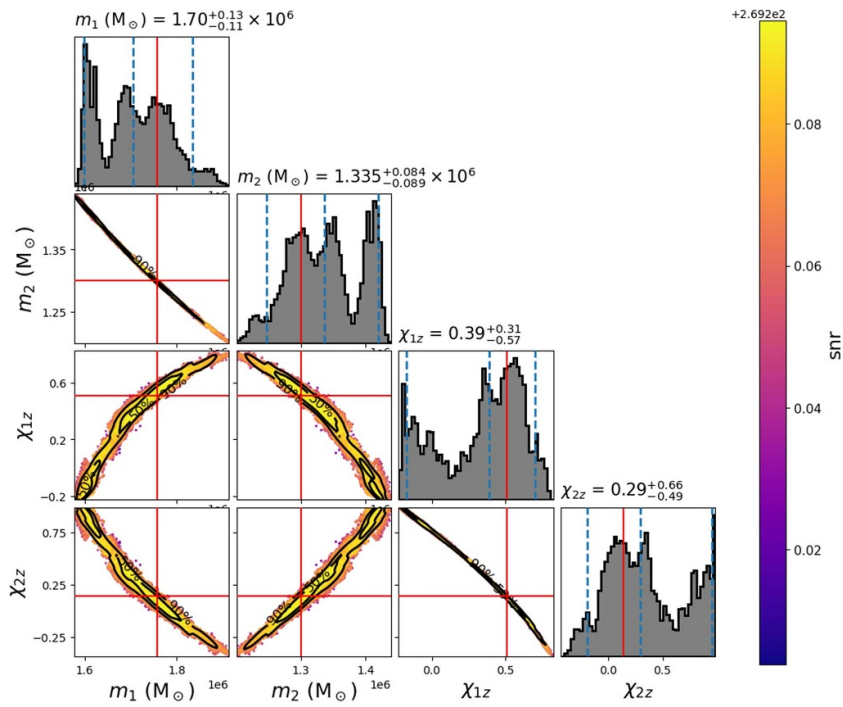
Figure 9. Intrinsic parameter posterior corner plot of Sangria blind data for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.
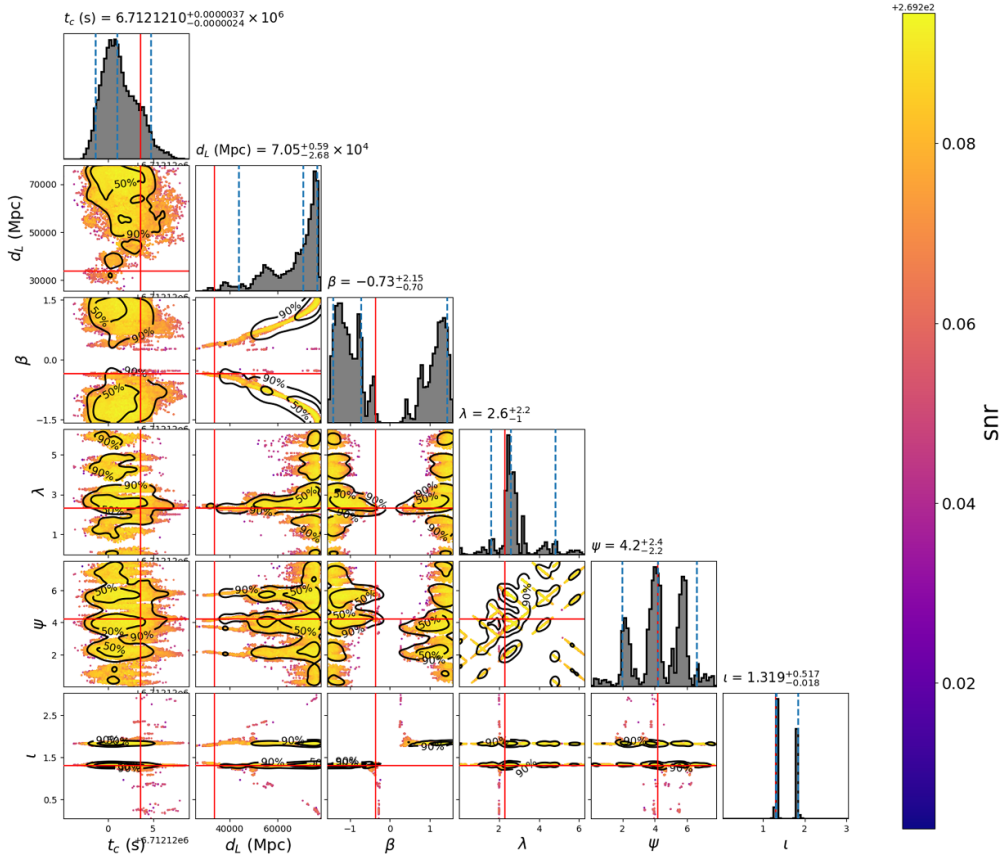
Figure 10. Extrinsic parameter posterior corner plot of Sangria blind data for signal 0. Red lines represent the injected parameter values of the signal. The tiles at the top of each column display the median and 90% range for each inference. The contour in the 2D tiles represents the 90% confidence intervals.

5. Conclusion

The application of a template bank search was able to identify all MBHB signals in the Sangria mock data challenge. Although the search for MBHB signals is relatively trivial due to the high SNR, being able to provide reference parameters for the heterodyne likelihood is the main advantage when taking this approach. We can produce a template bank $\mathcal{O}(minutes)$ to fulfil a minimal match of 0.9. Given that the SNR recovered is greater than 92%, we find that this technique is a viable approach to quickly search through LISA data for MBHBs. At the same time, the templates returned from the search provide appropriately high likelihood reference parameters for heterodyning which do not bias parameter estimation. To expand and improve upon the search, we could consider a coincident search over both the A and E TDI data streams. This would filter each TDI stream separately and then check for coincident triggers.

We have shown that the introduction of different forms of noise can have an effect on both intrinsic and extrinsic parameter recovery, although the extrinsic parameter recovery suffers more than the intrinsic. Something to consider about this data set is how the injected signals only contain the dominant (2, 2) mode. Other works have shown that sky position recovery

is vastly improved as other sub dominant modes are introduced to parameter estimation [59]. A further study could be to investigate the inference of a multi-modal signal, in the presence of LISA instrumental noise. We could then go further by introducing non-stationarities in the data, with the inclusion of GBs.

The intrinsic parameter recovery of MBHBs in the presence of confusion noise caused by GBs can bias the result away from the true value. Although the injected values in our work did fall within the 90% confidence intervals, the peak likelihood has moved away from the injected value when compared to the no GB inference case, see figures 5 and 7. This motivates the need for a full global fit pipeline, that can identify and infer the properties of the GBs and then remove them from the data before running inference on MBHBs. This was first described in [13].

An area of further investigation concerns how we estimate PSDs. As stated previously, in this work we estimate our PSDs by reading in the entire year's worth of data. Due to LISA's motion across the sky, this makes the PSDs non-stationary. One possible method is to construct our PSD by assuming quasi-stationarity on smaller time chunks of the year's worth of data, and assemble these into one PSD using the time/frequency track of the reference signal in question. It would also be interesting to investigate what effects the PSD being non-stationary has on the search and inference of such high SNR signals.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://github.com/gwastro/mbhbs-with-pycbc.

Acknowledgments

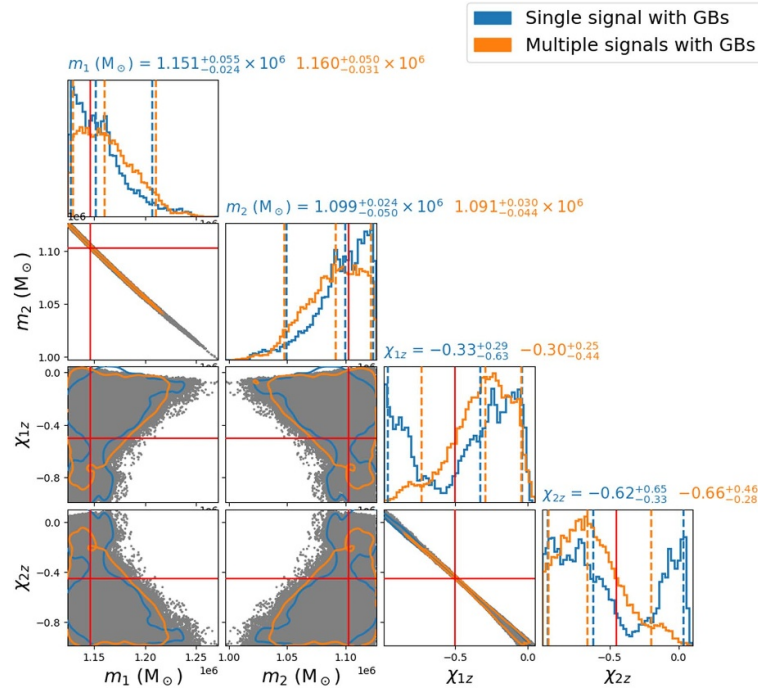## Appendix. Intrinsic parameter inference plots with GBs

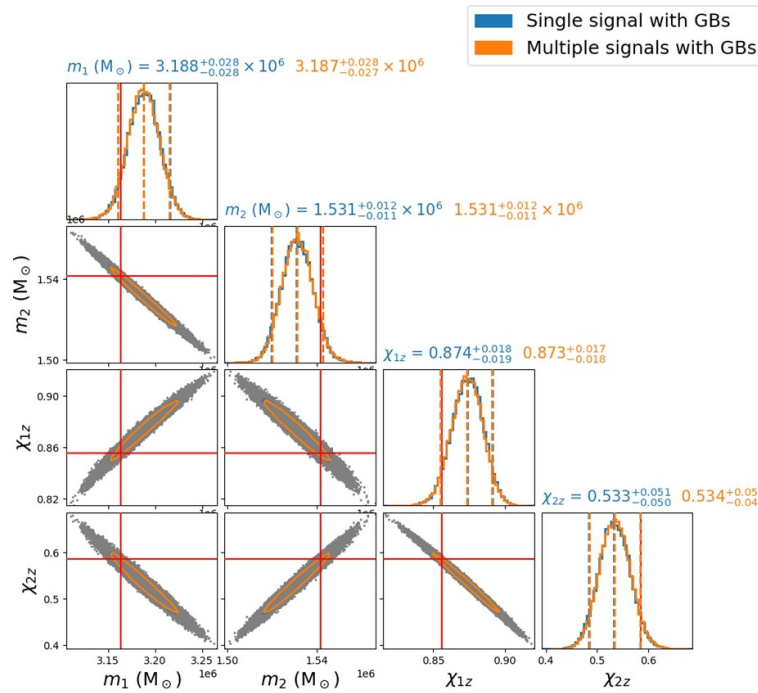**Figure A1.** Inference for BBHx injection of signal 1 from Sangria blind over intrinsic parameters.

Figure A2. Inference for BBHx injection of signal 2 from Sangria blind over intrinsic parameters.
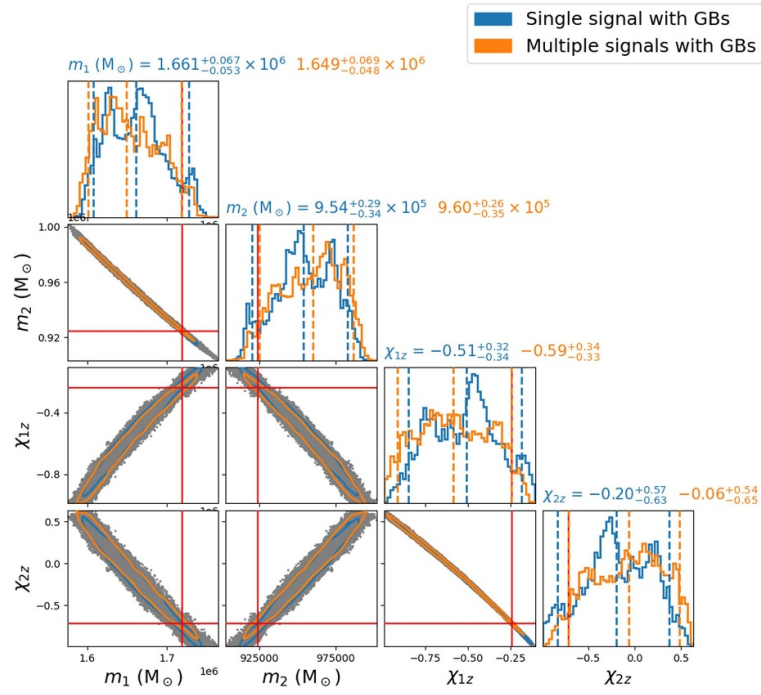
21

Figure A3. Inference for BBHx injection of signal 3 from Sangria blind over intrinsic parameters.
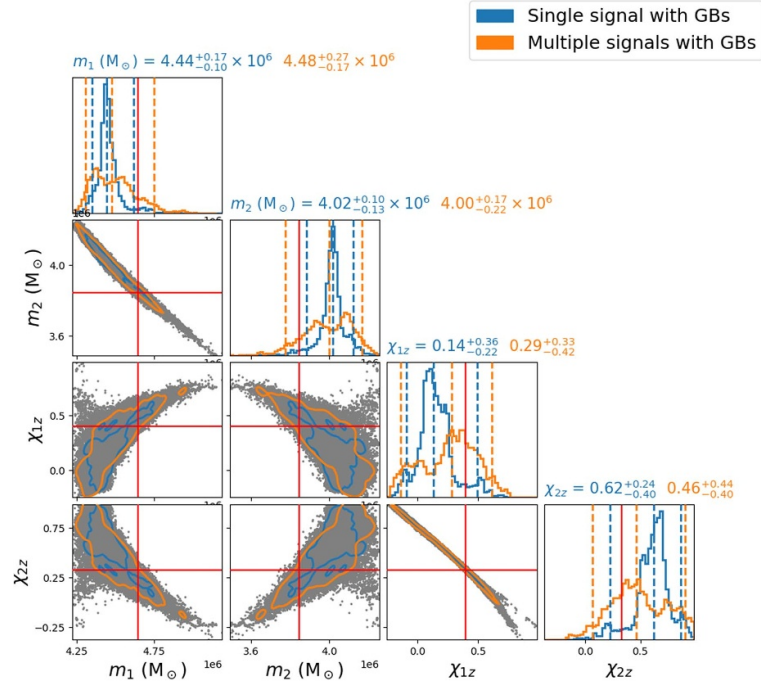

Figure A4. Inference for BBHx injection of signal 4 from Sangria blind over intrinsic parameters.
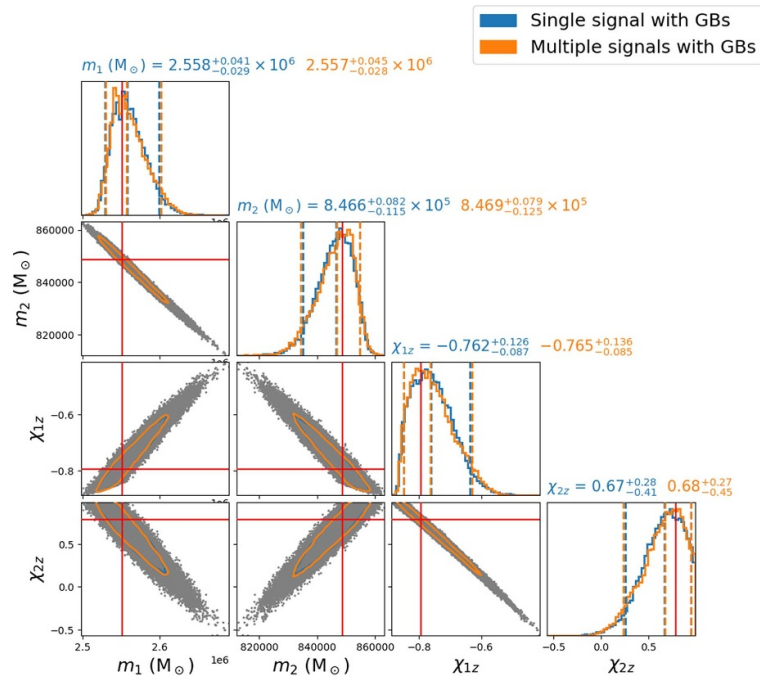
Figure A5. Inference for BBHx injection of signal 5 from Sangria blind over intrinsic parameters.

ORCID iDs

Connor R Weaving ⬤ https://orcid.org/0009-0008-2697-2998
Laura K Nuttall ⬤ https://orcid.org/0000-0002-8599-8791
Ian W Harry ⬤ https://orcid.org/0000-0002-5304-9372
Shichao Wu ⬤ https://orcid.org/0000-0002-9188-5435
Alexander Nitz ⬤ https://orcid.org/0000-0002-1850-4587

References

[1] Amaro-Seoane P *et al* (LISA) 2017 arXiv:1702.00786
[2] Wise J H 2023 arXiv:2304.09311
[3] Strokov V, Fragione G and Berti E 2023 arXiv:2303.00015
[4] Barack L and Pound A 2019 *Rep. Prog. Phys.* **82** 016904
[5] Gair J R, Babak S, Sesana A, Amaro-Seoane P, Barausse E, Berry C P L, Berti E and Sopuerta C 2017 *J. Phys.: Conf. Ser.* **840** 012021
[6] Barausse E *et al* 2020 *Gen. Relativ. Gravit.* **52** 81
[7] Li K, Bogdanović T, Ballantyne D R and Bonetti M 2022 *Astrophys. J.* **933** 104
[8] Li K, Bogdanović T, Ballantyne D R and Bonetti M 2022 arXiv:2207.14231
[9] Steinle N, Middleton H, Moore C J, Chen S, Klein A, Pratten G, Buscicchio R, Finch E and Vecchio A 2023 arXiv:2305.05955
[10] Fang Y and Yang H 2022 arXiv:2209.14509
[11] LISA Data Processing Group 2023 (available at: https://lisa-ldc.lal.in2p3.fr/)
[12] Baghi Q (LDC Working Group) 2022 The LISA data challenges *56th Rencontres de Moriond on Gravitation*
[13] Cornish N J and Crowder J 2005 *Phys. Rev.* D **72** 043005

[14] Vallisneri M 2009 *Class. Quantum Grav.* **26** 094024
[15] Karnesis N, Katz M L, Korsakova N, Gair J R and Stergioulas N 2023 arXiv:2303.02164
[16] Bayle J B, Hartwig O, Lilley M, Hees A, Chapman-Bird C, Woan G and Wolf P 2023 End-to-end simulation and analysis pipeline for LISA *57th Rencontres de Moriond on Gravitation*
[17] Cornish N J and Hellings R W 2003 *Class. Quantum Grav.* **20** 4851–60
[18] Rubbo L J, Cornish N J and Poujade O 2004 *Phys. Rev.* D **69** 082003
[19] Petiteau A, Auger G, Halloin H, Jeannin O, Plagnol E, Pireaux S, Regimbau T and Vinet J Y 2008 *Phys. Rev.* D **77** 023002
[20] Littenberg T B and Cornish N J 2023 *Phys. Rev.* D **107** 063004
[21] Bayle J B and Hartwig O 2023 *Phys. Rev.* D **107** 083019
[22] Digman M C and Cornish N J 2022 *Astrophys. J.* **940** 10
[23] LISA Data Processing Group 2023 LISA data challenge 2a: Sangria (available at: https://lisa-ldc.lal.in2p3.fr/challenge2a)
[24] LISA Data Processing Group 2023 LISA data challegne 2a, Sangria manual (available at: https://lisa-ldc.lal.in2p3.fr/static/data/pdf/LDC-manual-Sangria.pdf)
[25] 2022 LISA data challenge—Sangria results and data files (available at: https://lisa-ldc.lal.in2p3.fr/accounts/login/?next=/file)
[26] Nelemans G, Yungelson L R and Portegies Zwart S F 2001 *Astron. Astrophys.* **375** 890–8
[27] Finch E, Bartolucci G, Chucherko D, Patterson B G, Korol V, Klein A, Bandopadhyay D, Middleton H, Moore C J and Vecchio A 2023 *Mon. Not. R. Astron. Soc.* **522** 5358–73
[28] Husa S, Khan S, Hannam M, Pürrer M, Ohme F, Jiménez Forteza X and Bohé A 2016 *Phys. Rev.* D **93** 044006
[29] Khan S, Husa S, Hannam M, Ohme F, Pürrer M, Jiménez Forteza X and Bohé A 2016 *Phys. Rev.* D **93** 044007
[30] Maggiore M 2008 *Gravitational Waves: Volume 1: Theory and Experiments* vol 1 (Oxford university press)
[31] Shaddock D A, Tinto M, Estabrook F B and Armstrong J W 2003 *Phys. Rev.* D **68** 061303
[32] Tinto M, Estabrook F B and Armstrong J W 2004 *Phys. Rev.* D **69** 082001
[33] Arnaud K A *et al* 2006 *AIP Conf. Proc.* **873** 619–24
[34] Babak S *et al* (Mock LISA Data Challenge Task Force) 2010 *Class. Quantum Grav.* **27** 084009
[35] Arnaud K A *et al* 2007 *Class. Quantum Grav.* **24** S551–64
[36] PyCBC Development Team and the LIGO/Virgo Collaborations 2023 PyCBC (available at: https://pycbc.org/) (Accessed 13 June 2023)
[37] Usman S A *et al* 2016 *Class. Quantum Grav.* **33** 215004
[38] Nitz A *et al* 2022 gwastro/pycbc: v2.0.2 release of PyCBC (v2.0.2) *Zenodo* (https://doi.org/10.5281/zenodo.6324278)
[39] Abbott B P *et al* (LIGO Scientific, Virgo) 2016 *Phys. Rev.* D **93** 122003
[40] Abbott R *et al* (LIGO Scientific, VIRGO) 2021 arXiv:2108.01045
[41] Abbott R *et al* (LIGO Scientific, VIRGO, KAGRA) 2021 arXiv:2111.03606
[42] Nitz A H, Kumar S, Wang Y F, Kastha S, Wu S, Schäfer M, Dhurkunde R and Capano C D 2023 *Astrophys. J.* **946** 59
[43] Helstrom C W 1968 *Statistical Theory of Signal Detection* (Pergamon)
[44] Allen B, Anderson W G, Brady P R, Brown D A and Creighton J D E 2012 *Phys. Rev.* D **85** 122006
[45] Owen B J 1996 *Phys. Rev.* D **53** 6749–61
[46] Owen B J and Sathyaprakash B S 1999 *Phys. Rev.* D **60** 022002
[47] Babak S, Balasubramanian R, Churches D, Cokelaer T and Sathyaprakash B S 2006 *Class. Quantum Grav.* **23** 5477–504
[48] Keppel D 2013 *Phys. Rev.* D **87** 124003
[49] Privitera S, Mohapatra S R P, Ajith P, Cannon K, Fotopoulos N, Frei M A, Hanna C, Weinstein A J and Whelan J T 2014 *Phys. Rev.* D **89** 024003
[50] Davies G S, Dent T, Tápai M, Harry I, McIsaac C and Nitz A H 2020 *Phys. Rev.* D **102** 022004
[51] Cornish N J 2010 arXiv:1007.4820
[52] Zackay B, Dai L and Venumadhav T 2018 arXiv:1806.08792
[53] Cornish N J 2021 *Phys. Rev.* D **104** 104054
[54] Biwer C M, Capano C D, De S, Cabero M, Brown D A, Nitz A H and Raymond V 2019 *Publ. Astron. Soc. Pac.* **131** 024503
[55] Katz M L, Marsat S, Chua A J K, Babak S and Larson S L 2020 *Phys. Rev.* D **102** 023033
[56] Katz M L 2022 *Phys. Rev.* D **105** 044055

[57] Katz M 2021 mikekatz04/BBHx: first official public release *Zenodo* (https://doi.org/10.5281/zenodo.5730688)

[58] Marsat S and Baker J G 2018 arXiv:1806.10734

[59] Marsat S, Baker J G and Dal Canton T 2021 *Phys. Rev.* D **103** 083011

[60] London L Khan S, Fauchon-Jones E, García C, Hannam M, Husa S, Jiménez-Forteza X, Kalaghatgi C, Ohme F and Pannarale F 2018 *Phys. Rev. Lett.* **120** 161102

[61] Abbott B P *et al* (LIGO Scientific Collaboration and Virgo Collaboration) 2017 *Phys. Rev. Lett.* **116** 061102

[62] Tinto M and Dhurandhar S 2021 *Living Rev. Relativ.* **24** 1

[63] Tinto M and Dhurandhar S V 2005 *Living Rev. Relativ.*. **8** 4

[64] Babak S 2008 *Class. Quantum Grav.* **25** 195011

[65] Harry I W, Allen B and Sathyaprakash B S 2009 *Phys. Rev.* D **80** 104014

[66] Ajith P, Fotopoulos N, Privitera S, Neunzert A and Weinstein A J 2014 *Phys. Rev.* D **89** 084041

[67] Speagle J S 2020 *Mon. Not. R. Astron. Soc.* **493** 3132–58

[68] Blanchet L, Damour T, Iyer B R, Will C M and Wiseman A G 1995 *Phys. Rev. Lett.* **74** 3515–8

[69] Weaving C R and Wu S 2023 GitHub to mbhbs-with-pycbc (available at: https://github.com/gwastro/mbhbs-with-pycbc)

[70] 2023 LISA parameter estimation for simulated massive black hole binary (available at: http://pycbc.org/pycbc/latest/html/inference/examples/lisa_smbhb_ldc_pe.html) (Accessed 13 June 2023)