MASTER IN ASTROPHYSICS AND SPACE SCIENCE

Erasmus Mundus Master
in Astrophysics and Space Science

Master Thesis

Search for gravitational waves associated with the Vela pulsar glitch of 2024 April 29

Development of a burst gravitational-wave Bayesian inference pipeline

Supervisors

Marie-Anne Bizouard, Ph.D.
Artemis, Observatoire de la Côte d'Azur

Prof. Anđelka Kovačević, Ph.D.
Department of Astronomy, University of Belgrade

Author

Marco Immanuel Bayle Rivera

Academic Year 2023/2024

TOR VERGATA UNIVERSITÀ DEGLI STUDI DI ROMA
UNIVERSITY OF BELGRADE
University of Bremen
UNIVERSITÉ CÔTE D'AZUR
Funded by the European Union

This Master thesis is submitted in partial fulfillment of the requirements for the degree Master Physique Fondamental et Applications, parcours MASS as part of a multiple degree awarded in the framework of the Erasmus Mundus Joint Master in Astrophysics and Space Science −− MASS jointly delivered by a Consortium of four Universities: Tor Vergata University of Rome, University of Belgrade, University of Bremen, and Université Côte d'Azur, regulated by the MASS Consortium Agreement and funded by the EU under the call ERASMUS-EDU-2021-PEX-EMJM-MOB.

Abstract

Gravitational waves (GW) have been observed by ground-based laser interferometer detectors for almost a decade now. The only GW source observed so far is a compact binary coalescence that produces a well-modeled waveform. Another type of gravitational waveform that has been theorized is the burst GW which is sourced by stellar core collapses or changes in differential rotation of compact objects. However, the waveforms of burst sources are difficult to model, and the supplied model might not give explicit information on the nature of the source. In this thesis, we aim to infer the source properties of a burst-type GW in a Bayesian manner. More precisely we use the Bayes factor, a statistical figure-of-merit which compares evidences or marginalized likelihoods of different models. The Bayesian inference pipeline we introduce here is based on `bilby`, a `python`-based Bayesian inference library. Our pipeline compares models by performing nested sampling on GW strain data. This method allows us to calculate evidences and posteriors on the fly. We adapt the pipeline to a comparison scenario where the data is modeled to contain disparate burst-type GW signals. The Bayes factor helps us make inferences on which model to prefer; although to fully determine the nature of a burst waveform's source, the Bayes factor should be used in conjunction with other statistical inference tests. The performance of our Bayesian inference pipeline seems promising and it is expected to be used for analysis of real GW events. As an application, we analyze the GW strain data recorded during the Vela pulsar glitch event last 29 April 2024. The pipeline is adapted to a detection scenario where we use the signal-to-noise Bayes factor for comparing two models – one where the data contains a damped sinusoid signal (from the excitation of the neutron star's fundamental modes) and another model where the data does not contain this signal. We infer a reasonable limit for the amplitude of a damped sinusoid signal so that the current and design sensitivity LIGO-Virgo detector configurations can successfully detect such signals.

Table of Contents

Abstract i

1 Introduction 1
- 1.1 General Relativity and Gravitational Waves . . . 1
- 1.2 GW Detection principles . . . 4
  - 1.2.1 Laser Interferometry and the LIGO-Virgo detectors . . . 5
  - 1.2.2 GW strain data . . . 8
  - 1.2.3 Observing scenario and source catalogues . . . 9
- 1.3 GW sources and types . . . 10
  - 1.3.1 Burst-type sources . . . 11
- 1.4 Objective of the study . . . 13

2 Bayesian statistics: Theory and applications to GW Astronomy 15
- 2.1 Brief introduction to Bayesian statistics . . . 15
  - 2.1.1 Stochastic sampling using Nested Sampling . . . 16
- 2.2 Parameter estimation in the context of GW Astronomy . . . 18
- 2.3 Model selection using Bayes factors . . . 19
- 2.4 Burst GW inference pipeline development using `bilby` . . . 20

3 Application of the Bayesian inference pipeline on the Vela pulsar glitch of 2024 April 29 24
- 3.1 The Vela pulsar glitch of 2024 April 29 . . . 24
- 3.2 Methods . . . 25
  - 3.2.1 Modelling the GW emission from a Vela pulsar glitch . . . 25
  - 3.2.2 Pipeline and data description . . . 29
  - 3.2.3 Sampling and parameters' priors . . . 30
- 3.3 Results . . . 32
  - 3.3.1 Simulated Gaussian noise data analysis . . . 32
  - 3.3.2 LIGO-Virgo O4 data analysis . . . 37
  - 3.3.3 Removal of noise lines in real data ASD . . . 42
- 3.4 Discussion and conclusion . . . 43

4 Burst model comparison using the Bayesian inference pipeline 45
- 4.1 Models used for burst GW source identification . . . 46
- 4.2 Results . . . 47
  - 4.2.1 Neutron star f-mode excitation versus cosmic string cusp . . . 48
  - 4.2.2 Damped sinusoid versus sine-Gaussian waveforms . . . 50
- 4.3 Discussion and Conclusions . . . 56

A Burst GW inference pipeline in `python` **59**

Bibliography **65**

Acknowledgments **73**

List of Figures

1.1 Amplitude spectral density (ASD), or $\sqrt{S_n(f)}$ of the advanced LIGO and Virgo detectors at their design sensitivity, where $S_n(f)$ is the Power spectral density (PSD) 7

1.2 Observing scenario for the past three observing runs and the current fourth observing run, plus some predictions on the fifth observing run. Taken from `https://dcc.ligo.org/LIGO-G2002127/public` 10

2.1 An example of the LIGO Hanford (blue), LIGO Livingston (red), and Virgo (green) GW strain time series. The "fake on-source window" refers to the chunk of data where waveform injections are done. 22

3.1 BNS range as function of time on April 29th 2024 for the LIGO Hanford (red), LIGO Livingston (blue) and Virgo (purple) detectors. 25

3.2 The f-mode GW polarizations h_+ and $h_\times$ as written in Eq. (3.2). In this example $h_0 = 2.5 \times 10^{-21}$, $\tau = 0.5\,\mathrm{s}$, $f_0 = 1.6\,\mathrm{kHz}$, and $t_0 =$ 1398459500.25 GPS time. 26

3.3 Left: the half-Tukey and full Tukey windows are compared. Right: Comparing the half-Tukey window with the unwindowed damped sinusoid waveforms. 27

3.4 Corner plot of the damped sinusoid parameters' posterior distributions for a damped sinusoid signal with parameters $h_0 = 1 \times 10^{-20}$, $\tau = 0.5\,\mathrm{s}$, $f_0 = 2.7\,\mathrm{kHz}$ added to simulated data of the LIGO Hanford, LIGO Livingston and Virgo detectors (at their design sensitivity). 33

3.5 Trace plot of the sampler whose corner plot is shown in Figure 3.4. The left column shows the "live points" that roam the prior space as the prior volume $\ln X$ decreases. The right column shows the final one-dimensional marginalized distributions of each parameter 34

3.6 The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the damped sinusoid signal amplitude injected onto simulated data. For $\ln BF_{S/N} > 0$, the data prefers the signal model, $s(t) = n(t) + h(t)$, while negative $\ln BF_{S/N}$ means that the data prefers the noise-only model, $s(t) = n(t)$. 35

3.7 Reconstructed 90% credible interval for an injected damped sinusoid waveform with $h_0 = 10^{-20}$ and $\ln BF_{S/N} \approx 21772$, together with the injected and maximum likelihood waveforms. 36

3.8 Left: Reconstructed 90% credible interval for an injected damped sinusoid waveform with $h_0 = 3.45 \times 10^{-22}$ and $\ln BF_{S/N} \approx 0.95$ together with the injected and MLE waveforms. Right: A zoom-in of the figure on the left. 36
3.9 Network SNR and $\ln BF_{S/N}$ as a function of the (Vela-oriented) h_{rss} of the injected signal. The red points correspond to runs with $\ln BF_{S/N} > 8$ 37
3.10 Another set of injection runs where both the injected amplitude and frequency varies for each run. The colors of $\ln BF_{S/N}$ diverge from the threshold value $\ln BF_{S/N} = 8$, with red points indicating a strong preference for the existing signal model. 38
3.11 O4 ASD superimposed onto Figure 1.1. The O4 ASDs are calculated using 128 seconds of background data from the Hanford (H1), Livingston (L1), and Virgo (V1) interferometers, Tukey windowed and Welch-averaged every 4 seconds with 0.4-second overlaps. . . . 39
3.12 The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the amplitude of the damped sinusoid injected onto LIGO-Virgo O4 data. The minimum $\ln BF_{S/N}$ recorded was around 126, which is a really strong evidence for the presence of a damped sinusoid. 40
3.13 Corner plot for a damped sine with amplitude $h_0 = 1.0 \times 10^{-23}$ injected to O4 data, with a corresponding $\ln BF_{S/N} \approx 126$. For a low-amplitude injection, its Bayes factor is suspiciously high, meaning the sampler is picking up a damped sinusoid signal at a frequency of about 1.01 kHz. 41
3.14 O4 noise ASD within the limited bandwidth from $2.5 - 3.0$ kHz. This data is considered to understand the effect of the experimental lines in the real GW data. 41
3.15 The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the corresponding damped sinusoid amplitude injected onto real LIGO-Virgo O4 data. The samplers used in this plot used a limited frequency prior, $f_0 \in [2.5, 3.0]$ kHz. The amplitude threshold where the data has strong preference for the signal model is at $h_0 = 5.878 \times 10^{-22}$. 42
3.16 Notch-filtered noise ASD. 43
3.17 The signal-to-noise Bayes factorln $BF_{S/N}$ as a function of the damped sinusoid amplitude injected to the notched LIGO-Virgo O4 data (blue points), superimposed on Figure 3.15 (red points). 44

4.1 Corner plot for a cosmic string cusp sampler when there is a damped sinusoid with parameters $h_0 = 10^{-21}$, $\tau = 0.5$ s and $f_0 = 2.7$ kHz injected onto simulated Gaussian strain data. The orange lines are the injected damped sinusoid parameters. 49
4.2 Log Bayes factor comparing damped sinusoid signals with power-law signal (from a cosmic string cusp) $\ln BF_{DS/CS}$ as a function of the damped sinusoid's amplitude injected onto simulated Gaussian strain data. 50
4.3 Comparing the signal-to-noise $\ln BF_{S/N}$ from the damped sinusoid and cosmic string cusp samplers when there is a damped sinusoid injected onto simulated Gaussian strain data. 51

4.4 Corner plot for the sine-Gaussian sampler with a damped sinusoid injected on the strain data. The orange lines signify the frequency of the injected damped sinusoid. 52
4.5 Left: The frequency-domain representation of the injected damped sinusoid with $h_0 = 10^{-20}$ together with the maximum likelihood estimate (MLE) sine-Gaussian waveform. Right: The equivalent time-domain representation. 53
4.6 Damped sine vs. sine-Gaussian log Bayes factor $\ln BF_{DS/SG}$ as a function of the damped sinusoid signal amplitude injected onto simulated Gaussian strain data. 54
4.7 Comparison of the signal-to-noise $\ln BF_{S/N}$ from the damped sinusoid and sine-Gaussian samplers when there is a damped sinusoid injected onto simulated Gaussian strain data. 55
4.8 The signal-to-noise$\ln BF_{S/N}$ of a damped sinusoid sampler as a function of the sine-Gaussian peak amplitude h_{rss} injected onto simulated Gaussian strain data. 55

Chapter 1

Introduction

In this chapter we introduce the main properties of gravitational waves (GW) which is a prediction of general relativity (GR) [1, 2]. We then talk about the principles of GW detection as explored in [3–5]. We also list some of the sources of GW including the focus of this thesis which are the short-duration unmodeled (burst) GW sources. Then we outline the objective and content of this thesis.

1.1 General Relativity and Gravitational Waves

The general theory of relativity (GR) is Albert Einstein's way of describing gravity that does not violate his special theory of relativity, in contrast to Newtonian gravity. Its core equation, the Einstein field equations, relates the distribution of matter and energy to the curvature of space-time:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = 8\pi T_{\mu\nu}. \tag{1.1}$$

Here and for the rest of the chapter, we consider natural units where $G = c = 1$, unless otherwise specified. The right-hand side of Eq. (1.1) describes the physical distribution of matter and energy in one tensor known as the stress-energy tensor $T_{\mu\nu}$. The solution of this set of equations is called the metric $g_{\mu\nu}$, a geometric object that quantifies distances and angles in a given manifold. The left-hand side of Eq. (1.1) is composed of a combination of the metric and its derivatives as embedded in the Ricci tensor $R_{\mu\nu}$ and Ricci scalar R. These two are derived from the more general Riemann tensor $R^{\alpha}_{\beta\mu\nu}$, which consists of Christoffel symbols $\Gamma^{\alpha}_{\beta\nu}$ that reflect the variations of angles and distances on a curved manifold.

One fascinating result of GR involves the wave nature of gravity: using proper approximation techniques, a wave equation arises from this set of field equa-

tions. In a four-dimensional system with a metric signature $x^\alpha \equiv (t, x, y, z) \equiv (-, +, +, +)$ we can expand the equations in the weak-field limit by using

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}, \qquad |h_{\mu\nu}| << 1. \tag{1.2}$$

This describes a small perturbation $h_{\mu\nu}$ to the background Minkowski (flat-field) metric $\eta_{\mu\nu}$ in the space-time geometry. To further simplify the field equations, we can change our coordinates such that a Lorenz gauge transformation is made. We make infinitesimal changes in coordinates, leaving the Riemann tensor (and subsequently the Ricci tensor) unchanged. This is achieved by forcing the divergence of $\bar{h}_{\mu\nu} \equiv h_{\mu\nu} - h\eta_{\mu\nu}/2$ (also called the transverse-traceless perturbation) to be null, i.e. $\partial_\beta \bar{h}^{\alpha\beta} = 0$. This simplifies the field equations to

$$\Box \bar{h}_{\mu\nu} \equiv \eta^{\alpha\beta}\partial_\alpha\partial_\beta \bar{h}_{\mu\nu} = -16\pi T_{\mu\nu}. \tag{1.3}$$

We can now see that $\bar{h}_{\mu\nu}$ are solutions to a (sourced) wave equation, hence we call them “gravitational waves". If we let $T_{\mu\nu} = 0$ (which means we are considering solutions in vacuum), we can easily solve the wave equation by giving a simple solution in the form of a monochromatic plane wave, $\bar{h}_{\mu\nu} = A_{\mu\nu}e^{ik_\alpha x^\alpha}$. Additional gauge conditions can simplify the form of $A_{\mu\nu}$, in addition to the Lorenz gauge that asserts its orthogonality with the wavevector k^μ, i.e. $k^\mu A_{\mu\nu} = 0$. In particular, we can write these constraints including the Lorenz gauge as follows:

$$k^\mu \bar{h}_{\mu\nu} = 0 \tag{1.4}$$

$$\bar{h}_{\mu\nu}U^\mu = 0 \tag{1.5}$$

$$\bar{h}^\mu_\mu = 0 \tag{1.6}$$

where U^μ is the 4-velocity components of an observer in the Lorenz gauge that are experiencing the plane waves. In this so-called *traceless-transverse* (TT) gauge, $\bar{h}_{\mu\nu} = h_{\mu\nu}$, hence the name given to the perturbation. If, for example this observer in the TT and Lorenz gauge has a 4-velocity $(0, 0, 0, 1)$ then the $A_{\mu\nu}$ terms simplify to

$$\begin{pmatrix} A_{tt} & A_{tx} & A_{ty} & A_{tz} \\ A_{xt} & A_{xx} & A_{xy} & A_{xz} \\ A_{yt} & A_{yx} & A_{yy} & A_{yz} \\ A_{zt} & A_{zx} & A_{zy} & A_{zz} \end{pmatrix} \longrightarrow \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & A_{xx} & A_{xy} & 0 \\ 0 & A_{yx} & -A_{xx} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{1.7}$$

We see that the choice of gauges limits the GW polarizations to the two physical terms, A_{xx} and A_{xy}. This greatly simplifies our picture of a gravitational wave.

How do sources produce these waves? In the presence of a source, $T_{\mu\nu} \neq 0$. We can then solve Eq. (1.3) by using the Green's function for the box operator and retaining only the retarded-time solution to conserve causality. We can write the solution in a Cartesian coordinate system in Lorenz gauge $x^\alpha : (t, x, y, z)$ as

$$h_{\mu\nu}(t, \mathbf{x}) = 4 \int d\mathbf{x}' \; T_{\mu\nu} \left(t - r(\mathbf{x}'), \mathbf{x}'\right) \frac{1}{r(\mathbf{x}')} \tag{1.8}$$

where $r \equiv |\mathbf{x} - \mathbf{x}'|$ is the distance between the field point $\mathbf{x}$' and the source point $\mathbf{x}$, and that boldface represents a 3-dimensional space vector. We can further approximate this solution by considering a source that is much farther away from an observer, confined spatially in a finite volume of characteristic length L, and moving at an adiabatic rate such that $1/T_{\mu\nu}(\partial T_{\mu\nu}/\partial t) << L$. Using these approximations, $r \approx |\mathbf{x}|$ and can be considered a constant inside the integral; hence

$$h_{\mu\nu}(t, r) \simeq \frac{4}{r} \int d^3\mathbf{x}' \; T_{\mu\nu}(t - r, \mathbf{x}'). \tag{1.9}$$

However, it is rarely the case when we can define the stress-energy tensor analytically, which makes the integration difficult. Instead, we use the properties of the stress-energy tensor: conservation laws require it to be divergence-free which allows us to link the temporal (0) and spatial ($i \in [1, 3]$) components of the stress-energy tensor. Then we can further simplify by assuming a non-relativistic source $\partial x^i/\partial x^0 << 1 \longrightarrow T_{00} \simeq \rho$ (where ρ is the mass density) and rewrite the previous equation as

$$h_{ij}(t, r) \simeq \frac{2}{r} \int d^3\mathbf{x}' \; x'_i x'_j \frac{\partial^2 \rho}{\partial t^2} \tag{1.10}$$

$$= \frac{2G}{rc^4} \ddot{I}_{ij} \left(t - \frac{r}{c}\right) \tag{1.11}$$

where we used the notation $I_{ij} \equiv \int x_i x_j \rho(t, \mathbf{x}) d^3\mathbf{x}$ to denote the mass quadrupole tensor of the source, and the overhead dot denotes differentiation with respect to time. We also restore the physical units to emphasize the small amplitude of the metric perturbation. Hence we see that in order to produce gravitational waves,

we must have at least a change in the quadrupole moment, as the dipole moment cancels out for mass.

How do GWs interact with matter? We can observe the effect of GW on matter by considering two free-falling particles (e.g. two test-mass mirrors) and using the local inertial frame of one mirror instead of the TT and Lorenz gauge which follows the motion of the particle. In this local inertial frame the vector ξ^α connecting the geodesics (free-fall paths) of the mirrors will have an acceleration according to the nonzero components of the Riemann tensor in first order $h_{\mu\nu}$, which is proportional to the perturbation tensor h_{ij} (here, we use latin indices to indicate the labels 1-3 such that $x^i \equiv (x, y, z)$). Integrating the equations of motion for ξ^α will give $\xi_j = L_j + \frac{1}{2}h_{jj}L_j$, where L_j is separation displacement in the jth direction, and h_{jj} is one of the polarizations of the metric perturbation in the TT gauge (hereby called the *plus* polarization, or h_+). The other polarization's effect on matter is the same, although the correlation between the perpendicular directions x and y in the perturbation h_{xy} means that it will perturb the masses in an anti-correlated manner along these directions (hence the name *cross* polarization, or $h_\times$). The effective difference in the separation length will be given to first order in h_{jj} by

$$\frac{\delta L_j}{L_j} = \frac{1}{2}h_{jj} \tag{1.12}$$

which is also the same magnitude for the cross polarization.

In this regard, the GW can be thought of as imparting energy on the system of masses; hence it should carry energy from the source. The flux of a gravitational wave is a measure of the energy it carries through a unit surface per unit time. It is given (in SI units) by [2, 4]

$$F_{GW} = \frac{dE_{GW}}{dAdt} = \frac{c^3}{32\pi G}\langle \dot{h}_{ij}^{TT}\dot{h}_{ij}^{TT}\rangle = \frac{c^3}{16\pi G}\langle \dot{h}_+^2(t) + \dot{h}_\times^2(t)\rangle \tag{1.13}$$

where the angled brackets indicate time averaging over one period T.

1.2 GW Detection principles

While GR theory gives us a good grasp of how a GW is produced, how it behaves, and how it interacts with matter, the detection of GW is another interesting story. Embedded in the Eqs. (1.1), (1.3), and (1.11) is the physical

factor $G/c^4 \sim 10^{-44}$ in SI units, which makes the amplitude of the perturbation $h_{\mu\nu}$ really small. How are GW astronomers able to detect such a small perturbation?

1.2.1 Laser Interferometry and the LIGO-Virgo detectors

Just a century after the initial publication of the existence of GWs [1], these waves have been directly observed in the two detectors that form the Laser Interferometer Gravitational-wave Observatory (LIGO) [6] in the United States on September 2015. Labeled as GW150914 (for the day it was detected), this event is the merger of two inspiraling black holes [7]. Another GW detector in Italy called the Virgo detector [8] joined the LIGO detectors in 2017 and contributed to the observation of a binary neutron star merger labeled GW170817 [9], which allowed electromagnetic telescopes to observe the aftermath of this coalescence in an event that kickstarts multi-messenger astronomy [10]. In this thesis we work with data from these three GW detectors.

LIGO and Virgo (and other modern GW observatories such as the Japanese KAGRA detector [11]) use laser interferometry, a technique that requires the interference of light (in this case, by coherent stimulated emission) to provide information on the most minute of motions. Laser interferometers are usually configured to have a layout that splits the incoming laser beam into two using a beam splitter mirror. Then these beams travel through perpendicular arms that have mirrors as endpoints, reflecting them back to the beam splitter which recombines them. This configuration is called a Michelson interferometer [5], and its perpendicular arms are perfect for observing the effect of GW on matter.

In fact, for the setup leading to Eq. (1.12) it is assumed that the separation length between the test masses or mirrors can be changed. In laser interferometry, it is the phase shift of the light emitted from the beam splitter that can provide information on some passing GW. This phase shift can be written as [3]

$$\Delta\phi = \frac{2\pi}{\lambda}(2L_x - 2L_y) \tag{1.14}$$

where we have assumed that the optical paths are placed in the x and y axis of our TT gauge coordinates, and that λ is the wavelength of the photons emitted from the beam splitter. Then, assuming a passing GW along the $z-$direction with a plus polarization of the form $h_+ \equiv h_{xx} = h_0 \cos(2\pi f_{GW} t)$, we can calculate L_x by calculating the optical path along one arm and using the null light ray geodesic

(since we're dealing with a path "connected" by photons),

$$ds^2 = 0 = (\eta_{\mu\nu} + h_{\mu\nu})dx^\mu dx^\nu \tag{1.15}$$

$$= -dt^2 + (1 + h_{xx})dx^2 + (1 + h_{yy})dy^2 \tag{1.16}$$

which gives an estimate for L_x (when $dy = 0$ or for L_y when $dx = 0$):

$$L_x = \int dt = \int_0^L \sqrt{1 + h_{xx}(t - x)}dx \tag{1.17}$$

$$= L + \frac{1}{2\pi f_{GW}} \sin(2\pi f_{GW} L) h_0 \cos(2\pi f_{GW}(t - L)) \tag{1.18}$$

where we have assumed L to be the total length of the arm placed along the x-axis and used trigonometric identities and binomial approximations [12]. In terms of the "change in arm length", this amounts to $\Delta L/L \sim h_0/2$ which provides the probe to gravitational waves. Hence, the data obtained from these interferometers are usually called "GW strain data". In using laser interferometry, what is actually measured is the phase shift of the photons which makes GW detectors behave more like clocks than rulers [12].

However, in order to probe the small GW amplitudes the length of the arms should be proportional to the GW frequency and the speed of light c. For a typical binary neutron star source with a typical merger frequency of about 1500 Hz, the length of the arm should be about 50 km, which is too long. Hence advanced interferometers are equipped with enhancements that allow us to shorten the arm lengths. The LIGO detectors located at Hanford and Livingston have arm lengths of 4 km placed in a perpendicular L-shaped configuration, while the Virgo detector located in Cascina has an arm length of 3 km (also placed in L-shaped configuration). In addition to the end mirrors, a semi-transparent mirror is added at the beginning of each arm to create a Fabry-Perot cavity that enhances the optical path. Despite the high quality of the coatings of the mirrors, the high power laser circulating in the cavities generate thermal deformation that are compensated with heat regulators to prevent the cavities to be uncontrolled.

In observing the phase shift it is not automatically determined what caused it. Actually many sources of noise generate phase shift – some of these noise sources are related to the detector itself. An obvious example is that photons do not arrive at the mirror one-by-one. The number of photons arriving can be realistically described by a Poisson distribution that exerts a length shift between the two

equal arms that is called the *shot noise*. This noise is part of the class of noises called *quantum noise*, which also include the radiation pressure noise produced by the photons at low frequency. Furthermore, the ground-based detectors have to deal with the motion of the Earth's surface that appear as seismic noise. And finally in the mid-frequency range, the detectors are limited by thermal noise from sources such as the Brownian motion of atoms in the mirror coatings. The fundamental noise sources give the colored noise curve showed in Figure 1.1.

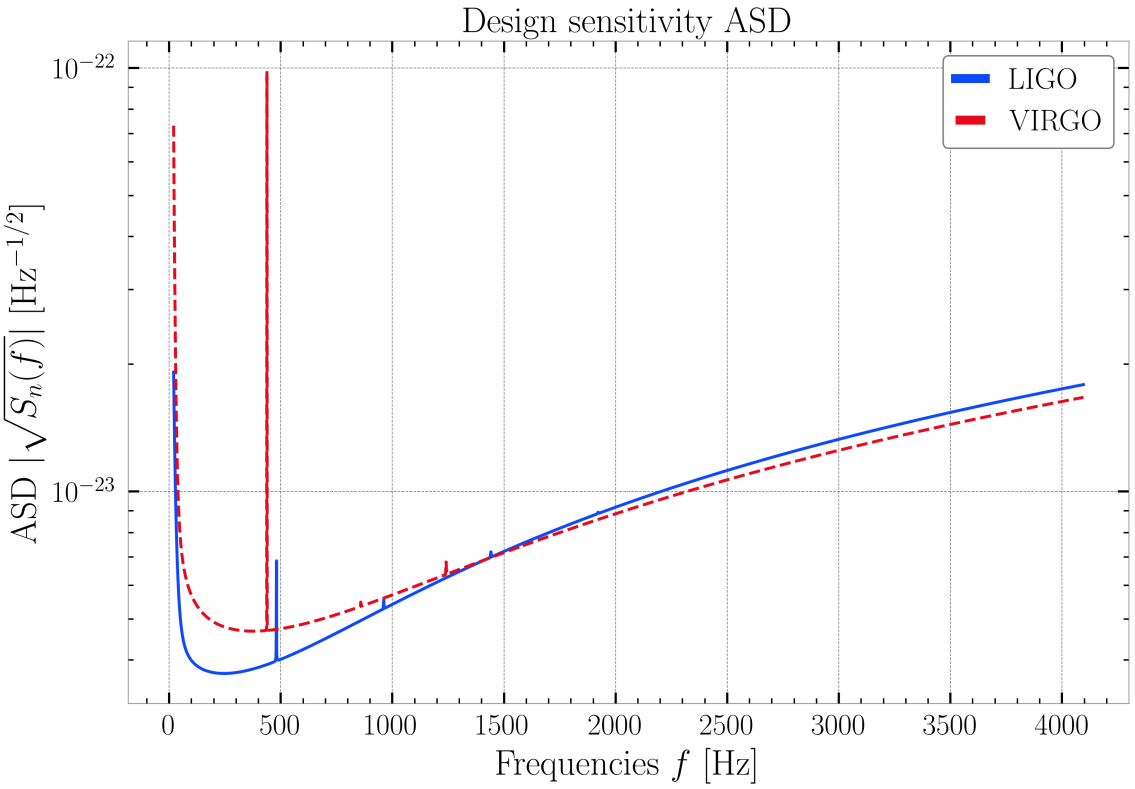

Figure 1.1: Amplitude spectral density (ASD), or $\sqrt{S_n(f)}$ of the advanced LIGO and Virgo detectors at their design sensitivity, where $S_n(f)$ is the Power spectral density (PSD)

In this figure, the noise amplitude spectral density (ASD) that current detectors should reach circa 2029 [13] is shown as a function of frequency. Here, the different noise sources contribute to the increase of amplitudes at different frequencies. In real data, many noise sources limit the sensitivity of the detectors, together with non-Gaussian transient noise sources collectively known as "glitches". There are also narrow spectral features called "lines" that are present in real GW spectrum. They are due to mechanical resonances and/or electromagnetic noise coupling (such as the 60 Hz AC power line for US). All of these are permanent noise sources and can be confused with a genuine (continuous) GW signal. The goal of many engineers and physicists in the collaboration is to bring the current detectors

to their design sensitivity, which means almost all of the noise sources could be mitigated except for the most fundamental ones described above.

1.2.2 GW strain data

How do GWs appear in laser interferometer detectors? The GW detectors work like radio antennas, where any astrophysical source from any direction in the sky can be superimposed within the noise. However, there are only certain areas of the sky where the GW detectors work properly; to the rest of the sky, they are practically "deaf". The main data output of each detector is a time series (labeled as $s(t)$). The GW signal appears as a time series $h(t)$ that is projected onto the detector. This projection depends on the sky location of the source (given in right ascension α and declination δ) and the relative position of the detector at the time of arrival. The projected GW can be written as

$$h(t) = h_+(t)F_+(\alpha, \delta, \psi) + h_\times(t)F_\times(\alpha, \delta, \psi) \tag{1.19}$$

where F_+ and $F_\times$ are called the "antenna pattern" functions and ψ is one of the Euler angles that rotates the source frame into the detector frame. The antenna pattern functions encode the information on which parts of the sky can GWs be detected. This time series form of the signal is assumed to be added onto the detector noise $n(t)$ such that the total data output, also called "strain", is given by

$$s(t) = n(t) + h(t). \tag{1.20}$$

There is a short time delay between the arrival time of GWs in the geocenter and in the detectors (of the order of $1 - 10$ ms). This time delay of the signal travelling between the detectors allows us to disentangle a real astrophysical signal from a terrestrial one.

In practice, GW strain data can be fetched as a discrete time series that is sampled at 16384 Hz. In this thesis, we use `gwpy` [14], a `python`-based package that allows users to fetch GW strain data either directly from the proper channels in LIGO-Virgo detectors or from publicly available websites such as the Gravitational Wave Open Science Center (GWOSC). [1]

[1] https://gwosc.org/

In order to properly analyze strain data the noise $n(t)$ must be characterized. The detector noise is the sum of all noise processes including the fundamental noise sources described earlier. For short timescales we can assume that noise is stationary [2]. Stationary noise allows us to work in the frequency domain where the noise is uncorrelated between frequency bins [5]. Here, the autocorrelation function (which contains information on the statistical properties of the noise) in time domain can be written as the Power Spectral Density (or PSD, also denoted by $S_n(f)$) in the frequency domain via a Fourier Transform [5]. From the PSD we can derive the ASD and other quantities that can characterize the noise and the signal [15].

One way of quantifying the detectability of a waveform $h(t)$ is by calculating the optimal signal-to-noise ratio (SNR or ρ), a quantity that is specific to a detector:

$$\rho^2 = 4\Re \int_0^\infty df \frac{|\tilde{h}(f)|^2}{S_n(f)} \tag{1.21}$$

where $S_n(f)$ is the Power Spectral Density (PSD) of the detector and $\Re$ denotes taking the real part of the complex quantity [3]. Since we are working with discrete time series that is sampled at a frequency f_s, this integral will be replaced with a summation where $\Delta f = 1/T$ and the frequency representation of our data will be limited by the Nyquist frequency $f_s/2$. In considering multiple detectors, we calculate instead the quadrature sum called the network SNR, which is the root-sum-square of the SNR from the ith detector: $\rho_{net} = \sqrt{\sum_i \rho_i^2}$.

1.2.3 Observing scenario and source catalogues

The scientists behind LIGO, Virgo and KAGRA detectors form a collaboration (collectively called LVK Collaboration). They operate each detector and organize joint data collection periods called “observing runs". As of the time of writing, the LVK is in its fourth observing run (O4) where the range of distance to sources are shown in Figure 1.2. It also shows the recently updated timeline for the observing runs, including the detectors' respective BNS detection ranges which is the average distance to detect a binary neutron star (BNS) merger with a SNR of 8 [13].

For the past three observing runs, the LVK collaboration have reported about 90 candidates with high probability of astrophysical origin. These candidates are

[2] in this case stationarity refers to the unchanging properties between segments of time series.
[3] since the Fourier transform $\tilde{h}(f) \equiv \mathcal{F}[h(t)]$ is generally complex

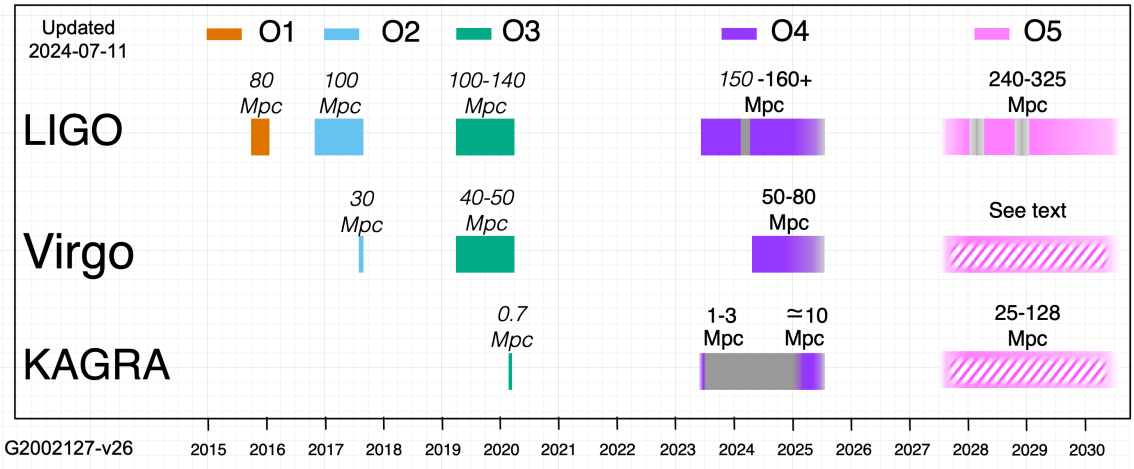

Figure 1.2: Observing scenario for the past three observing runs and the current fourth observing run, plus some predictions on the fifth observing run. Taken from `https://dcc.ligo.org/LIGO-G2002127/public`

sourced by compact binary coalescences (CBC) which are classified into binary black hole (BBH) mergers, one binary neutron star (BNS) merger, and three neutron star-black hole (NSBH) merger [16]. These successful detections were catalogued in three separate instances which are built upon each other [17, 18]. The current observing run O4 is not expected to finish until June 2025 and the next catalogue will be released in August 2025 for half O4 data (called O4a). In O4 it is expected that the number of detected GW sources will increase reaching up to $\sim$ 300 CBC sources. It is also expected that more types of sources will be detected, including the fabled burst-type GW which will be discussed in the next section.

1.3 GW sources and types

Gravitational waves can be classified into four types, depending on their duration and the source. The well-known source and the only one observed so far is the compact binary coalescence (CBC), which are the BBH, BNS, and NSBH mergers mentioned earlier. The CBC source provides transient (short-lived) and well-modeled (or well-approximated) GW. The continuous wave (CW) is a long-duration GW that is also well-modeled, emitted by rotating neutron stars. The long-duration unmodeled source is called the stochastic GW background (SGWB), which could also provide information from the early universe. And lastly, the transient unmodeled source is called burst GW, and it will be the focus of this study.

1.3.1 Burst-type sources

Recall that a GW is produced by a change in quadrupolar moment of the gravitational source. One particular type of GW that has been theorized but not been discovered yet is the burst-type GW. These waves could be sourced by differential rotation and/or explosion, both of which can realign the mass inertia moments and provide a transient change in the quadrupolar moment of the source.

Burst GWs are characterized by the root-sum-square h_{rss} of the amplitudes of both polarizations [2, 19] which can be used to determine the energy output of the source:

$$h_{rss}^2 = \int_{-\infty}^{\infty} dt\ (|h_+(t)|^2 + |h_\times(t)|^2) \tag{1.22}$$

$$= 2\int_{0}^{\infty} df\ (|\tilde{h}_+(f)|^2 + |\tilde{h}_\times(f)|^2), \tag{1.23}$$

where $\tilde{h}$ denotes the Fourier transform of the waveform polarizations. Since they are short duration, they are well-localized in time and also well-localized in frequency, which changes the integral limits further. Here we list down some sources that have been theorized to produce burst GWs.

Core-collapse supernova

Perhaps the most studied source of burst-type GW, a stellar core collapse happens for massive stars – those that have masses $> 8M_\odot$ – at the end of their life, when radiation pressure from nuclear fusion can no longer compensate the inward gravitational pressure. The core contracts until it collapses in a fraction of a second. This process can release burst-type GW through its asymmetric (nonspherical) collapse or other mechanisms that might change the quadrupole moment such as differential rotation. Numerical relativity simulations involving nuclear physics predict the GW emissions of this process. The core-bounce part of this core-collapse process [20] can be modeled using a single or combination of sine-Gaussian wavelets [21].

Pulsar glitches and the Vela pulsar

A pulsar is a rapidly rotating, magnetized neutron star that emits beamed electromagnetic radiation which are detected when the beam crosses the line of sight of the detector (hence the name). The emission takes away energy from the

source, which causes it to slow down until the pulsar 'turns off' its pulsed beams. However, these compact objects can experience sudden changes in rotational frequency that can be caused by an internal reconfiguration of angular momentum or energy distribution [22]. This event is called a glitch, and it can be observed as irregular pulses within a rather regular periodic pulse emission. Glitches are mostly observed in younger pulsars with higher spin-down rates, such as the Vela pulsar (PSR B0833-45), Crab pulsar (PSR B0531+21), PSR J0537-6910, and PSR B1737-30. However, the rate of these events happening is low and it varies from one pulsar to another. These events can be seen in radio and high-energy EM radiation, but there is also a chance to see a glitching millisecond pulsars inlcuded in GW pulsar timing observations [23].

Pulsar glitches are events that could let us study the interiors of compact stars [4]. Together with magnetar flares and gamma-ray bursts (GRB) these sources have been hypothesized to produce GW bursts in the form of damped sinusoids that can be manifestations of the excitation of the star's fundamental oscillation modes (known as f-modes) [24–27].

One particular glitching pulsar of interest is the Vela pulsar. Located in the Vela constellation, it is well localized and well observed, having a rotational period of about 0.089 seconds. In addition to precise sky location and distance measurement, Chandra X-ray observations of the pulsar wind nebula torus provide accurate determination of the orientation of the Vela spin axis [28]. The inclination angle ι between the pulsar spin axis and the line of sight, as well as the polarization angle ψ of the spin axis projection on the plane of the sky are derived from those observations [29], although these values are undetermined within 180° as their effects on the pulse strength only show as cosines. All parameters that are used for our analysis are given in Table 1.1.

Right ascension (α)	128.3861 °
Declination (δ)	-45.1764 °
Inclination angle (ι_V)	63.6 ± 0.6. °
Polarization angle (ψ_V)	130.629°.

Table 1.1: Some of the the Vela pulsar parameters that are used for the analysis in Chapter 3. The sky location is obtained from the XMM-Newton catalog [30], while the inclination and polarization angles can be derived from observations made by Mignani et. al [29].

Since 1969 there have been 25 recorded Vela pulsar glitches every 2-3 years

[4] This field of study is called asteroseismology [24].

[31, 32]. Thanks to these observations there has been several models that could be tested to explain the phenomenon, including a two-layer structure (crust and core) [33]. Due to the frequency of its glitches, the Vela pulsar is also actively studied for searches of GW bursts [34] and even continuous waves [35].

Cosmic strings

A cosmic string is a topological defect from cosmological phase transitions that can produce gravitational waves. It is expected that the burst-type GWs can come from string features called cusps and kinks [36–38], which come from cosmic string loops that periodically oscillate. The expected frequency-domain appearance of the GW emission for the string features are power laws, such that $\tilde{h}(f) \propto h_0 f^{-\alpha}$ where α is the power-law index with value $\alpha = -4/3$ for a cosmic string cusp and $\alpha = 5/3$ for a cosmic string kink. There is also a high-frequency cutoff term f_{high} [39] that is related to the shape of the GW emission and string tension. Just like the other burst-type GWs, they have not yet been observed, but O3 searches have given us constraints on the models [37].

1.4 Objective of the study

Different searches for these GW burst signals have been conducted [40], but so far no burst signal from any source has been detected. However when a burst-type GW is detected, it will be important to characterize the source. Unlike CBC searches (e.g. matched filtering [41]) where the nature of the source is known *a priori* from the waveform, most of burst signals are poorly modelled. While waveform predictions exist like the ones listed earlier, they are not precise enough to be used in burst GW search pipelines which search for excess power with few to no assumptions about the source [3].

In this thesis we use Bayesian inference to give statistically-supported statements on the nature of a burst GW signal's source. We develop a Bayesian inference pipeline with the goal of comparing several possible GW burst models (chapter 4). We assume that the strain (time-series) data can be represented by well-defined models, which means we do not need to work with time-frequency maps that are commonly used in burst GW search pipelines [20]. This study of comparing waveforms has been done previously: In [38] they compared BBH merger hypothesis with cosmic string loops and cusps hypotheses using Bayes factor as a figure-of-merit. However, on 29 April 2024 the Vela pulsar was reported to

glitch while LIGO and Virgo were acquiring data. It turns out that our Bayesian inference pipeline could be easily adapted to search for a GW signal associated to the Vela glitch (chapter 3).

Our pipeline uses the Bayes factor as a figure-of-merit, which has been used extensively in GW Astronomy for model selection [42, 43] as well as detection of glitches and burst signals [21]. We construct the pipeline with the help of `bilby`[44] and `gwpy`[14] which are special `python` packages for Bayesian inference and input/output of GW strain data, respectively. We focus on sampling the parameters of the burst model's GW signal morphology rather than the parameters of specific physical models like equations-of-state (EOS).

In chapter 2 we provide the background for Bayesian inference methodology, then we also construct the general pipeline used for the thesis. Then in chapter 3 we report our search for a GW signal associated with the Vela pulsar glitch last 29 April 2024 using our Bayesian inference pipeline. In chapter 4 we compare different GW burst signal models and their detectability. We wrap up and discuss future steps in Conclusions and Recommendations.

Chapter 2

Bayesian statistics: Theory and applications to GW Astronomy

Bayesian statistics is the framework in which GW astronomers are able to understand and interpret their observations. It is through Bayesian statistics that “events" turn into “confirmed detections" with physically meaningful properties. The scarcity of GW observations, the weakness of the GW signal compared to noise, and the solid foundation of GW modelling all imply the suitability of Bayesian statistics as compared to frequentist statistics. In this chapter, we follow the theory of Bayesian statistics from the following references: [42, 45, 46]. We start with Bayes' theorem which is the basis of all of Bayesian inference, then we explore its application on parameter estimation and model selection – two statistical inference methods commonly used in GW Astronomy. We close the chapter by outlining the Bayesian inference pipeline that is used for the next two chapters.

2.1 Brief introduction to Bayesian statistics

The entirety of Bayesian formalism stems from the definition of joint probability for $P(\mathbf{D} \cap \theta | M)$, otherwise known as the probability of obtaining both the data $\mathbf{D}$ and (a set of) parameters θ, and the definitions of symmetry for probability:

$$P(\mathbf{D}|\theta, M)P(\theta|M) = P(\mathbf{D} \cap \theta|M) \ = P(\mathbf{D}|M)P(\theta|\mathbf{D}, M) \tag{2.1}$$

$$\mathcal{L}(\theta)\pi(\theta) = P(\mathbf{D} \cap \theta|M) \ = ZP(\theta). \tag{2.2}$$

Here $P(\mathbf{D}|\theta, M)$ is the probability of obtaining data given a set of parameters θ and underlying model M, $P(\theta|M)$ is the probability of obtaining the parameters given the model, $P(\mathbf{D}|M)$ is the probability of obtaining the data based on the marginalized probability of all parameters, and $P(\theta|\mathbf{D}, M)$ is the probability of

obtaining a set of parameters given some data and an underlying model. In the context of Bayesian statistical inference, these terms are respectively called the likelihood $\mathcal{L}(\theta)$, prior $\pi(\theta)$, evidence or marginalized likelihood Z, and posterior distribution $P(\theta)$.

This equivalence in joint probability definitions allows us to write Bayes' theorem in the context of statistical inference as

$$P(\theta) = \frac{\mathcal{L}(\theta)\pi(\theta)}{Z} \tag{2.3}$$

from which we can infer that $Z \equiv \int \mathcal{L}(\theta)\pi(\theta)d\theta$ since probability distributions should be normalized. It is important to note that Eq. (2.3) is written with an underlying model in mind – later on we will see how this plays a role in model selection. The next two sections focus on the two main Bayesian inference methods we use in this thesis: parameter estimation and model selection.

2.1.1 Stochastic sampling using Nested Sampling

In Bayesian statistics, the uncertainty of the estimated parameters is included within the posterior distribution. However, estimating the posterior, through the estimation of the likelihood poses a challenge especially in the computational sense. It is not exactly known how the data should look like given some parameters, especially when we are considering approximants. For each parameter, we consider multiple values within the chosen prior range, and the more parameters we try to infer, the more difficult the estimation becomes. This is where stochastic sampling comes in handy.

Stochastic sampling is characterized by the use of random walkers (known as samplers) that explore the parameter spaces. Their randomness is encoded in their jump/transition proposal, which is defined by the probability to change steps according to some rules. Stochastic samplers are generally classified into two: on one hand, the Markov Chain Monte Carlo (MCMC) methods directly generate samples from the posterior (up to a multiplicative factor) with the jump probability usually attributed to Metropolis and Hastings [47], while on the other hand Nested Sampling divides the parameter space into iso-likelihood spaces and lets the random walkers sample from these smaller spaces, building the posterior in a "nested" manner by setting the transition probability to depend on the lowest value of calculated likelihood. In this thesis, we use Nested Sampling as a means of estimating the parameters.

Originally, Nested sampling was intended to solve the evidence integral (which we recall here)

$$Z = \int d\theta \ \mathcal{L}(\theta)\pi(\theta) \tag{2.4}$$

by redefining the integral as a function of the prior volume X,

$$X(\lambda) = \int_{\Omega_\theta : \mathcal{L}(\theta) \geq \lambda} \pi(\theta) \ d\theta \tag{2.5}$$

which is in turn a function of λ, the threshold minimum value for the iso-likelihood contour $\mathcal{L}(\theta)$. This transforms the evidence to

$$Z = \int_0^1 \mathcal{L}(X) dX \tag{2.6}$$

where the limits naturally appear as the normalization condition on X, with the total prior volume being $X = 1$. The jump proposal is then given by the omission of the random sampler that is within the iso-contour of the lowest value of $\mathcal{L}(X)$ (labeled λ). This makes the samplers proceed by eliminating the walker living on the lowest iso-likelihood contour and replacing it with a walker living on a randomly chosen higher iso-likelihood contour, thereby decreasing the prior volume and moving on to higher likelihood regions.

As compared to MCMC samplers whose stopping criterion is arbitrary, Nested Sampling also provides a natural stopping criterion by considering an estimated upper limit for the evidence at each iteration: the likelihood of the remaining prior volume is assumed to be the highest likelihood of the live points. This can be translated to a difference of evidences per iteration, given by $\Delta \ln Z_i = \ln(Z_i + \Delta Z_i) - \ln Z_i < \varepsilon$ for some small ε [1]. As a by-product, Nested Sampling is also able to build up the posterior $P(\theta) \propto \mathcal{L}(\theta)\pi(\theta) \approx \mathcal{L}(\theta)X(\pi(\theta))$ by using the importance weights.

There are several implementations of Nested Sampling in `Python`. For our purposes, `dynesty` [48] suffices, thanks to its modular nature and the ability to switch between “static" and “dynamic" nested sampling, depending on the priority of the user to calculate either the evidence or posterior.

[1] In `dynesty`[48], this ε is specified as a keyword argument `dlogz`

2.2 Parameter estimation in the context of GW Astronomy

Since GW signal amplitudes scale as $G/c^4 \sim 10^{-44}$, the signals can be easily buried in the noise. We assume that the signal is linearly added onto the noise, such that the total strain data output of the detector is $s(t) = n(t) + h(t)$, where h is the astrophysical GW signal and n is the sum of all noise sources. This is akin to the 'null' and 'alternative' hypothesis scenarios in 'frequentist' statistics, where the 'null' hypothesis asserts that there is only noise, $h(t) = n(t)$, while the 'alternative' hypothesis asserts that there is a signal added to the noise, $h(t) = n(t) + s(t)$.

Using Bayesian statistics, we can infer the parameters (of an underlying model) given the "likelihood" of the data being produced from those parameters and our belief on how the parameters are distributed ("prior"). In GW astronomy, the combination of several random noises in the data is assumed to produce a probability distribution that is Gaussian [2]. Hence we use the Gaussian likelihood

$$\mathcal{L}(d|\theta;\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{(d-\mu(\theta))^2}{\sigma^2}\right) \tag{2.7}$$

where μ and σ^2 are the mean and variance of the Gaussian distribution. For continuous GW data, this means μ is a template built on several parameters θ. As discussed in Chapter 1, the strain data is usually a discrete, finite time series with a specific duration T sampling rate f_s. The likelihood can then be rewritten as a function of a single frequency bin j with size Δf (using the notation in [42]) as

$$\mathcal{L}(d_j|\theta) = \frac{1}{2\pi(PSD)_j} \exp\left(-2\Delta f \frac{|d_j - \mu_j(\theta)|^2}{(PSD)_j}\right) \tag{2.8}$$

where PSD is the (one-sided) noise power spectral density defined in Chapter 1. Thus we need to convert both the time series d and template μ to the frequency domain (see [15]), which gives the additional $2\Delta f$ factor in the exponential. Then we can multiply each likelihood computed from all frequency bins to get the total. This multiplicativity also applies to the likelihoods obtained from different detectors.

[2] There are specific instances of non-Gaussianity in the form of glitches, but this is beyond the scope of this thesis.

In practice, likelihoods can get extremely big or extremely small. Instead it is easier to work with the log-likelihood; not only does it take care of extreme magnitudes, it also converts any multiplication to a summation, making the total log-likelihood easier to calculate. The total log-likelihood is given by

$$\log \mathcal{L}(d|\theta) = \sum_j \log \mathcal{L}(d_j|\theta) = -\frac{1}{2}\sum_j \log(2\pi(PSD)_j) - 2\Delta f \sum_j \frac{(d_j - \mu_j)^2}{(PSD)_j}. \tag{2.9}$$

The other quantity we need to specify for Eq. (2.3) is the prior, $\pi(\theta)$. The prior distribution takes into account the belief on the distribution of the parameters before any evidence is considered. Priors can range from uninformative to informative, depending on the modeler's beliefs on the distribution of the parameters. In this thesis, we use uninformative or flat priors to reflect the indifference of a GW burst detection to any model. An example is the Uniform prior, $\pi(\theta) \sim \text{Unif}(a, b)$, where the parameter θ has equal chances of being any value within the range $\theta = a$ to $\theta = b$. Another example is the Log-Uniform prior which is used when there is no *a priori* information about the magnitude of the parameter [42].

2.3 Model selection using Bayes factors

Now that the posterior distribution of parameters is taken cared by Nested sampling, our next focus is on model selection. Recall that in building Eq. (2.3), there was an assumption that the parameters belong to a certain model. An example of a model could be the existence of the signal within the data, for which an evidence Z_S and priors for its parameters π_S can be calculated. Changing this model will allow us to define different evidences from different sets of priors. For example, the other model could be the 'noise model' where the signal is not present in the data, with an entirely different evidence $Z_N \equiv \mathcal{L}(d|\theta = 0)$ [3]. We can then compare two models (generally denoted by A and B) by using the odds ratio,

$$\mathcal{O}^A_B = \frac{Z_A}{Z_B}\frac{\pi_A}{\pi_B} \tag{2.10}$$

[3]In this case, we can set the priors to be the same as the signal model, but in general the priors could be different.

where the second term consists of the prior odds which describes the degree of belief about the relative likelihood of the two models, while the first term is called the “Bayes factor", which compares the evidences (or marginalized likelihoods) of the two models. The (natural log of the) Bayes factor can be calculated as: [42, 49]

$$\ln\ BF_{A/B} = \ln\left(\frac{Z_A}{Z_B}\right) = \ln Z_A - \ln Z_B \tag{2.11}$$

For our purposes, we use Jeffrey's definition [50] for a significant decision: if $|\ln BF| \geq 8$ then one model is definitively preferred by the data, with the sign of $\ln BF$ telling us which model is preferred [42].

2.4 Burst GW inference pipeline development using `bilby`

In this thesis, our primary methodology consists of comparing “models" by calculating the Bayes factors for several scenarios. The first one can be a detection scenario: the ‘signal model’ is a model for which the astrophysical GW signal is present in the data, versus the ‘noise model’ where the data contains only noise [42]. The other scenarios are comparisons of disparate signal models, e.g. comparing a damped sinusoid signal (from a magnetar/pulsar glitch) against a power-law signal (from cosmic string cusps/kinks) or against sine-Gaussian signals. In all cases, the evidence or marginal likelihood is a necessary calculation, which prompts us to develop a pipeline based on Nested Sampling.

There are several parameter estimation softwares used and developed by the LVK collaboration; one of these is called `bilby`, a Bayesian inference library [44]. As a modular package, `bilby` can do several things: (1) load real GW data or even create synthetic ones from specific GW interferometers, (2) wrap and call different stochastic samplers like `dynesty`, (3) calculate and provide several likelihoods and priors (such as Eq. (2.9) which is implemented internally), and most importantly, (4) allow the user to create their own objects that can be used for Bayesian inference.

We describe our Bayesian inference pipeline developed with `bilby`. The input for the pipeline is either simulated or real GW strain data and the primary output is the Bayes factor for a given scenario. In the “detection scenario" described earlier, the pipeline will calculate a (log) Bayes factor that compares the “signal

model" and the "noise model" (hereby labeled $\ln BF_{S/N}$). For the signal model comparison scenario, the pipeline will calculate a Bayes factor that compares signal model A to signal model B (labeled $\ln BF_{A/B}$). The pipeline will also output the posterior distributions of the sampled parameters that are user-specified.

To test the performance of this pipeline we inject burst-type GW signals to either simulated or real data. The process follows a generic form which can be summarized as follows:

1. Construct the GW burst signals' waveform polarizations h_+ and $h_\times$, then feed this into a `WaveformGenerator` object inside `bilby`.

2. For simulated Gaussian data, load the ASD files as a `FrequencySeries` from the `gwpy` package. `bilby` will process this ASD and create a noise realization from it. For real data, fetch the data (with the correct frames) using `gwpy`'s `TimeSeries.get` from the appropriate detector channels. Both simulated and real data are preprocessed and loaded onto `bilby`'s `Interferometer` objects. An example of GW strain is shown as a time series in Figure 2.1.

3. Inject the waveform polarizations to the `Interferometer`'s strain data (which we call the "fake on-source window" in Figure 2.1) by adding the detector-frame frequency-domain waveform $\tilde{h}(f)$ [4] onto the frequency-domain strain data $\tilde{s}(f)$.

4. Construct the log likelihood, Eq. (2.9) by using `bilby`'s likelihood object `GravitationalWaveTransient`, which accepts the `Interferometer` (for $(PSD)_j$) and `WaveformGenerator` (for constructing multiple $\mu(\theta)$)

5. Construct the prior for each parameter to be sampled using `bilby`'s `PriorDict`, which is a `python` dictionary containing prior objects from `Prior` objects

6. Launch the sampler by using `bilby`'s `run_sampler` method where the user will specify the Likelihood and Prior objects and the chosen sampler with its corresponding settings. This will return a JSON file that contains information about the posterior distribution of the sampled parameters, noise and signal evidence, the signal-to-noise Bayes factor, the injected parameters, likelihood and importance weights, etc.

[4] Derived from Eq. (1.19) using a Fast Fourier Transform (FFT)

An example `python` code that implements the steps above is shown on Appendix A. This pipeline can adapt to different GW signals we want to test.

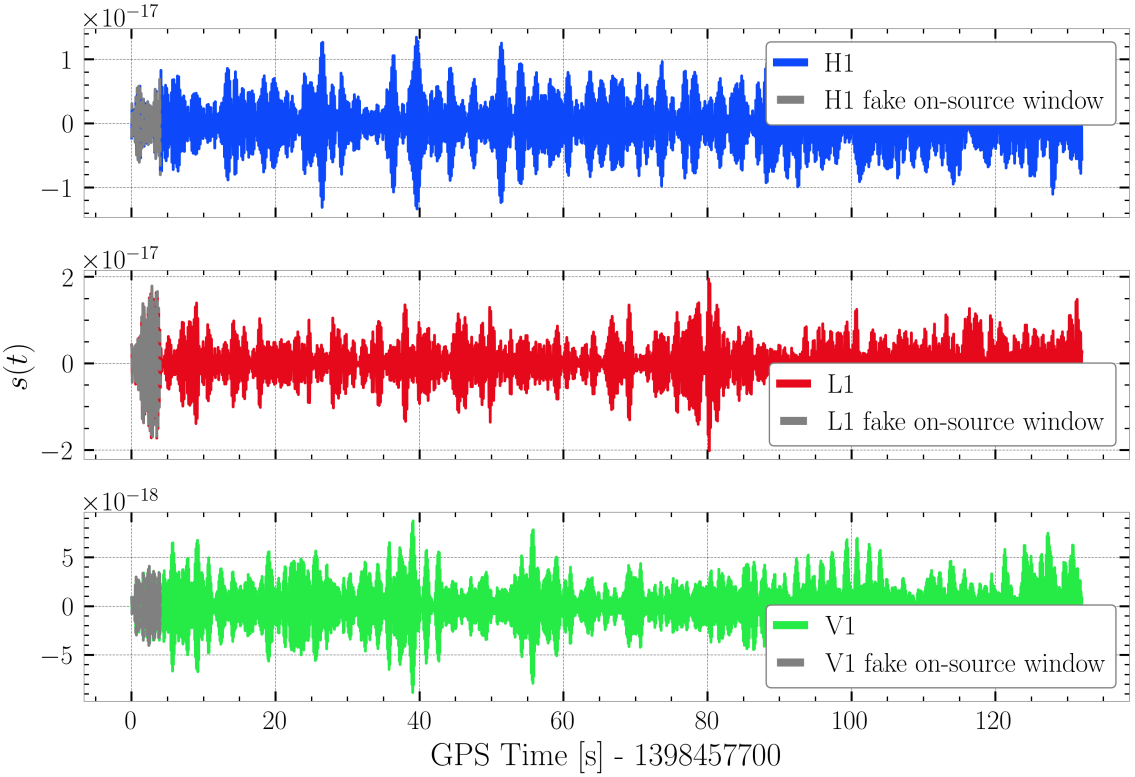

Figure 2.1: An example of the LIGO Hanford (blue), LIGO Livingston (red), and Virgo (green) GW strain time series. The "fake on-source window" refers to the chunk of data where waveform injections are done.

To evaluate the sampling results, one can check the 'trace plots' of a stochastic sampler. This plot shows how the samplers explored the parameter space over time, and whether convergence to a certain value/distribution was achieved. Since we are using `dynesty`'s (static) Nested Sampling, the exploration over time is shown as the sampler's progression with the decrease in prior volume. We can also evaluate the performance of the sampler by visualizing the posterior distributions themselves. This is usually shown as a 'corner plot', where the one-dimensional marginalized posterior distributions are shown in the diagonals and the two-dimensional marginalized posterior distributions (containing combinations of different parameters) are shown in the off-diagonals.

In the process of stochastic sampling, the goal is to recover the injected waveform within the error estimates given by the posterior distribution of the input parameters. We can visualize this goal by obtaining the posterior samples (or even just a subset) from the results' JSON file, reconstructing the waveform for each of these parameter sets, and calculating the error bounds for each point in

time/frequency from the waveform. For example, if we want to calculate the 2σ waveform errors, we need to calculate the 5^{th} and 95^{th} percentile values at each point in time/frequency for all the reconstructed waveforms. We can then visualize both the error ranges of the possible waveform reconstructions and embed the injected waveform.

Another waveform reconstruction quality measurement is given by the mean absolute error [51],

$$MAE = \frac{1}{n}\sum_{i=0}^{n} |h_i(t) - \hat{h}_i(t)| \tag{2.12}$$

where i is the individual timesteps, h_i is the injected waveform, $\hat{h}_i$ is the maximum likelihood estimate (MLE), and n is the total number of waveform points.

Chapter 3

Application of the Bayesian inference pipeline on the Vela pulsar glitch of 2024 April 29

In this chapter we apply the `bilby`-based inference pipeline described in section 2.4 to LIGO-Virgo data collected during the recent Vela pulsar glitch event that occured on April 2024 (first reported in Astronomer's telegram [52, 53]). As a disclaimer, the strain data used in this thesis does not involve the data corresponding to the glitch event itself but the data surrounding the event. The pipeline we introduced can be adapted to a detection scenario where we model the data (assuming a given signal exists) and calculate its corresponding Bayes factor $\ln BF_{S/N}$. This "Bayesian search" for a GW signal is conducted using the O4 data recorded by the two LIGO and Virgo interferometers. We first summarize of the event's properties, then we outline the steps taken in conducting what is known as a "closed-box analysis". We also explain how such analysis prepares GW scientists for a real signal search.

3.1 The Vela pulsar glitch of 2024 April 29

The most recent glitch of the Vela pulsar was first reported by the PuMA Collaboration using the two 30 m antennas of the Argentine Institute of Radio astronomy [53]. The glitch was detected by tracking the residuals of the pulsar timing, where the residuals are obtained from fitting a timing solution. This timing solution is a model of the previous pulses received at regular time intervals. They took into account the data recorded before the epoch (there reported as between MJD 60428.96 or 28 April 2024 23h UTC and MJD 60431.84 or 1 May 2024 20h UTC). The reported relative change in pulsar rotation frequency F_0 is

$$\frac{dF_0}{F_0} = 2.3 \times 10^{-6} \tag{3.1}$$

Other observations have been reported and the glitch's epoch uncertainty is reduced to MJD $\pm 3.84691 \times 10^{-5}$, centered around MJD 60429.869615 (29 April 2024 20:52 UTC). This uncertainty of ± 3 [s] allows us to define a short-duration on-source window (OSW) which makes the event a perfect candidate for Bayesian data analysis. Considering the Vela pulsar's distance of 293 parsecs, we hope to find a corresponding GW signal.

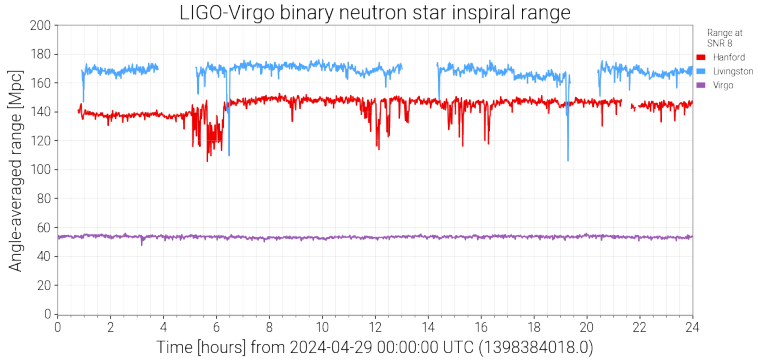

Figure 3.1: BNS range as function of time on April 29th 2024 for the LIGO Hanford (red), LIGO Livingston (blue) and Virgo (purple) detectors.

3.2 Methods

How loud should a burst GW signal be in order for us to "hear" it within the noise? In this chapter, we aim to answer this question by applying the Bayesian inference pipeline on real data surrounding the time of the Vela pulsar glitch event. We assume that this GW emission can be modeled by a damped sinusoid [25, 26], then we adapt our Bayesian pipeline to compare two models: one where a damped sine exists in the data and another where the data is purely noise.

3.2.1 Modelling the GW emission from a Vela pulsar glitch

The transient nature of the pulsar glitch suggests that a burst-type GW could be released [1]. In this thesis, we consider the GW emission of a pulsar glitch as an

[1] Although we assume that a burst GW is released, efforts have been made to search for continuous gravitational waves from pulsar glitches. These searches use pipelines such as `cwinpy` [35].

excitation of fundamental modes (f-modes) that we model using an exponentially damped sinusoid [25]. The GW propagates from the Vela pulsar to the Earth such that the two polarization amplitudes depend on the inclination angle ι between the source's spin axis and the line of sight, [2]

$$\begin{aligned} h_+(t) &= \frac{1+\cos^2\iota}{2}\,\Theta(t-t_0)\,h_0\,e^{-(t-t_0)/\tau}\,\cos(2\pi f_0 t+\varphi_0) \\ h_\times(t) &= \cos\iota\,\Theta(t-t_0)\,h_0\,e^{-(t-t_0)/\tau}\,\sin(2\pi f_0 t+\varphi_0). \end{aligned} \tag{3.2}$$

Here $\Theta(t-t_0)$ is an Heaviside step function which defines the time t_0 at which the GW amplitude h_0 is maximum. Aside from the amplitude, the other intrinsic GW parameters are the characteristic frequency f_0, and the damping time τ which controls how fast the wave loses energy (related to the rate at which the pulsar comes back to its regular spin-down rate). Furthermore, the GW emission is assumed to be well-localized in the sky, such that its source position is constant throughout the emission duration. This allows us to use the sky location, inclination, and polarization angles listed in Table 1.1 [3].

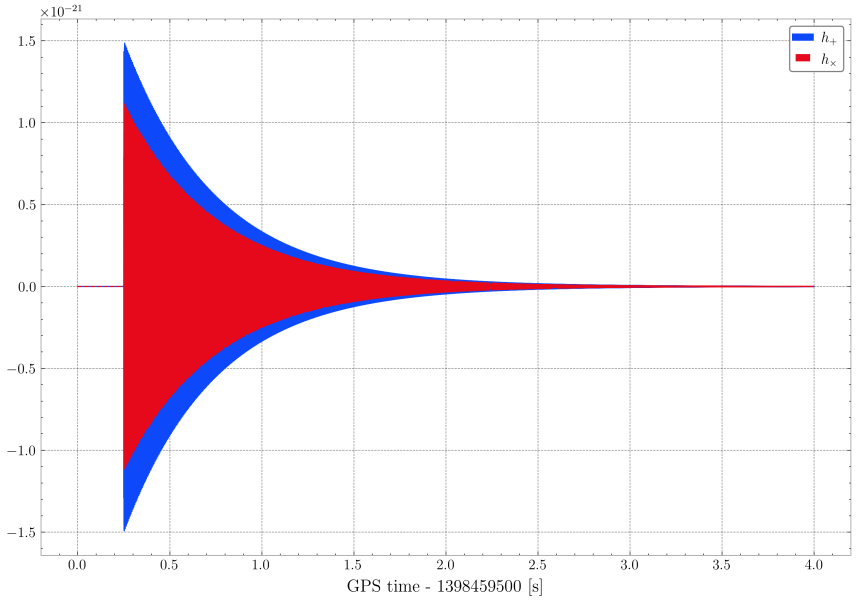

Figure 3.2: The f-mode GW polarizations h_+ and $h_\times$ as written in Eq. (3.2). In this example $h_0 = 2.5\times10^{-21}$, $\tau = 0.5\,\mathrm{s}$, $f_0 = 1.6\,\mathrm{kHz}$, and $t_0 = 1398459500.25$ GPS time.

Fig. 3.2 shows an example of h_+ and $h_\times$ polarizations as function of time. To avoid spectral leakage effect when we go from time to frequency domain, we redefine $\Theta(t-t_0)$ and apply a half-Tukey window with a shape parameter α that

[2] This emission is assumed to be dominated by the fundamental $\ell = |m| = 2$ GW mode.
[3] We also assume that the phase at the time of arrival φ_0 is zero.

depends on the characteristic frequency f_0 of the damped sinusoid [4]. On the left panel of Figure 3.3, the half-Tukey window is represented and the right panel shows how the window modulates the time series amplitudes.

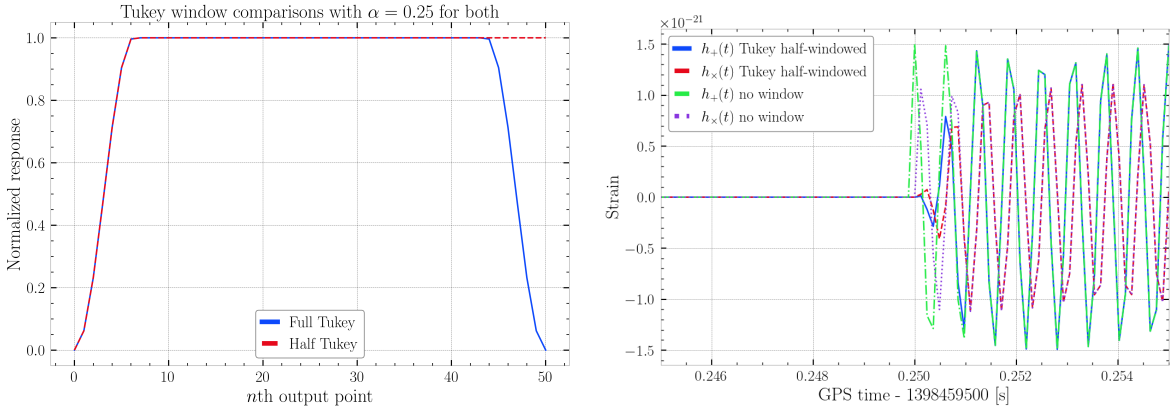

Figure 3.3: Left: the half-Tukey and full Tukey windows are compared. Right: Comparing the half-Tukey window with the unwindowed damped sinusoid waveforms.

There are several models that explain how excitation of the f-modes generate burst-type GWs. In this thesis, we use the empirical formulas from [24, 27] that limit the range of f_0 and τ:

$$\begin{aligned} f_0(\text{kHz}) &\approx 0.22 + 2.16 \left(\frac{M/(10M_\odot)}{R/(10\text{km})} \right)^{1/2} \\ \frac{1}{\tau(\text{s})} &\approx \left(\frac{(M/(10M_\odot))^3}{(R/(10\text{km}))^4} \right) \left[22.85 - 14.65 \left(\frac{M/(10M_\odot)}{R/(10\text{km})} \right) \right] \end{aligned} \tag{3.3}$$

Using these formulas and assuming a neutron star mass and radius range of $M \in [0.8, 2.5]\ M_\odot$ and $R \in [8, 20]$ km, as well as considering both f_0 and τ to be positive and sensible such that the damping time does not last too long, the range of f_0 and τ is limited to

$$\begin{aligned} f_0 &\in [0.797, 4.25]\ \text{kHz} \\ \tau &\in [0.087, 10]\ \text{s} \end{aligned} \tag{3.4}$$

These limits provide the range to be used for our uniform prior beliefs.

We can also find a suitable range for our amplitude prior. From Eq. (1.13), we convert the emitted GW energy density per unit time to a spectral energy

[4] Spectral leakage happens when box window are used in time domain analysis (time series are always finite) as the Fourier transform of a box window is $\sin(f)/f$. The power spectral density of a box windowed time series scales as f^{-2} [5].

density (SED), given as the energy density per unit frequency. Since we are dealing with a transient source that is well-localized in time and frequency, we can calculate the SED by averaging the flux Eq. (1.13) over one period T (assuming the burst signal's duration $\leq T$) and converting the squared time derivatives using $(-\imath 2\pi)^2 f^2 |h(f)|^2$. Together with Parseval's theorem and the definition for differential solid angle, the time-averaged spectral flux density is given at the quadrupolar approximation by [19]

$$F_{GW} = \frac{d^2 E_{GW}}{dAdt} = \frac{\pi c^3}{4G}\frac{1}{T}\int_{-\infty}^{\infty} df\, f^2(|\tilde{h_+}(f)|^2 + |\tilde{h_\times}(f)|^2) \tag{3.5}$$

To obtain the total GW energy emitted by the source we need to integrate this averaged flux over a sphere of radius r which is the distance to the source. Since the inclination angle ι_V is known for the Vela pulsar, we do not receive this total energy so we cannot integrate over the whole area of the sphere. Instead we factorize the ι terms of Eq. 3.2 to obtain

$$\begin{aligned} E_{GW} &= \frac{\pi c^3}{4G}\int dA \int df\ f^2(|\tilde{h}_+(f)|^2 + |\tilde{h}_\times(f)|^2) \\ &= \frac{\pi c^3}{4G} r^2 \left(\frac{(1+\cos^2\iota_V)^2}{4} + \cos^2\iota_V\right) \\ &\int_{-1}^{1} d\cos\iota \int_0^{2\pi} d\phi \int_{-\infty}^{\infty} df\ f^2(|\tilde{h}_+(f)|^2 + |\tilde{h}_\times(f)|^2) \\ &= \kappa \frac{\pi^2 c^3}{G} r^2 \int_{-\infty}^{\infty} df\ f^2(|\tilde{h}_+(f)|^2 + |\tilde{h}_\times(f)|^2) \end{aligned} \tag{3.6}$$

where $\kappa \equiv \left(\frac{(1+\cos^2\iota_V)^2}{4} + \cos^2\iota_V\right)$. Considering ι_V=63.30 °, we have κ = 0.5630. Using a monochromatic signal (i.e. dominated by one frequency f_0) and a definition for the optimally-oriented ($\iota = 0$) source h_{rss} from Eq. (1.23), Eq. 3.6 becomes

$$E_{GW} = \kappa \frac{\pi^2 c^3}{G} r^2 f_0 h_{rss}^2 \tag{3.7}$$

which is about half the energy released of a narrowband isotropic emission [19]. We can relate this description of energy to the peak amplitude h_0 [26]:

$$h_0 = \frac{1}{\pi d_L f_0}\left(\frac{5G}{c^3}\frac{E_{GW}}{\tau}\right)^{1/2} \tag{3.8}$$

which takes into account the (luminosity) distance to the source d_L and gives us a range of estimates for the peak amplitude h_0, assuming that the glitch energy is entirely carried away by the GW emission.

3.2.2 Pipeline and data description

The Bayesian inference pipeline described in Section 2.4 allows us to statistically compare two models. It also recovers the parameters of a model waveform. To quantify the performance of its parameter estimation capabilities, we simulate various damped sinusoid signals and inject them to GW strain data. Referring to Figure 2.1, the injection is done inside `bilby` and can be summarized as follows: (1) after the time-domain waveform function is constructed, its frequency-domain (FD) equivalent is calculated using a Fast Fourier transform (FFT); (2) The projection of this FD waveform to the interferometer is calculated, then (3) the projected FD waveform is added onto the FFT of a "fake" on-source window. This fake OSW allows us to perform a "blind search" which consists of tuning and testing the search efficiency without being biased by the presence of a genuine GW signal in the OSW. The fake OSW is chosen not too far away from the real OSW (see Figure 2.1 where the data for the fake OSW is highlighted in gray). The fake OSW is used by `bilby` as the data d in the log-likelihood calculations of Eq. (2.9). Furthermore, the template μ described in Eq. (2.9) is the projected FD waveform with parameters sampled from the prior distributions. This shows how the log-likelihood weighs the different waveforms that can be made from the parameters space, such that the waveform with the maximum likelihood pertains to the waveform with parameters close to the injected one (at best, it is exactly the injected one).

As a test, we first inject the damped sinusoid waveform on simulated Gaussian noise data before applying the pipeline to real LIGO-Virgo O4 data. To obtain simulated Gaussian noise, `bilby` generates a random Gaussian noise realization by respecting the design sensitivity amplitude spectral density (ASD) [13] and creating a time series from a specified duration and sampling frequency. The noise ASD to be used is exactly the one showed in Figure 1.1.

To obtain real LIGO-Virgo O4 data, we need to fetch the GW strain time series from the calibrated strain channels of each detector which sample at 16384 Hz, having a Nyquist frequency of 8192 Hz [5]. We fetch three sets of time series

[5] In LVK nomenclature, the detectors are named "H1" for the LIGO-Hanford detector, "L1" for

from three detectors; one set of 4-second data serves as the fake OSW, another 128-second time series serves as the off-source window (labeled as “background" in Figure 2.1) from which the noise ASD will be estimated. The third 6-second time series fetched is the “real" OSW (which is about 20 minutes after the fake OSW in Figure 2.1) as this corresponds to the data surrounding and including the trigger time that might possibly contain the real burst GW. This real OSW data will be used once we have fully tuned the search.

The following pre-processing procedures are applied to the fetched O4 data: we apply a (full) Tukey window over the whole 132-second time series and resample it to 8192 Hz to gain a Nyquist frequency of 4096 Hz. The O4 noise ASD is then calculated from the 128-second background data by applying a Fourier transform on the autocorrelation of each data point (assuming stationarity). For each 4-second interval in this 128-second strain, a Tukey window is also applied. The intervals are overlapped with each other with a total of 0.4 seconds per overlap, then the final ASD is averaged from all the 4-second ASDs using Welch's method [5].

In this chapter, we consider two injection sets. One set consists of 40 damped sinusoid waveforms where the amplitude varies within the range $h_0 \in [10^{-23}, 10^{-20}]$ with all other injected parameters kept constant: frequency $f_0 = 2.7$ [kHz], damping time $\tau = 0.5$ [s], and geocentric arrival time $t_0 = 0.25$ seconds after the fake OSW start. To check the frequency dependence, another set of 40 injections are made, varying both the injected amplitude (same range as before) and frequency $f_0 \in [0.9, 4.2]$ [kHz]. We inject both sets onto simulated and O4 data and recover the parameters plus signal and noise evidences for each.

3.2.3 Sampling and parameters' priors

The pipeline samples the parameters of a generic signal that can come from a real GW source or an injected simulated signal. In this chapter, we adapt the pipeline such that we sample the parameters of the damped sinusoid injections. We assume that the damped sinusoid model does not refer to any equations-of-state (EOS) that relates the mass and radius of the neutron star, but rather, we focus on sampling the phenomenological parameters $\{h_0, \tau, f_0, t_0\}$. Here we assume the time t_0 to be the time of arrival of the maximum amplitude h_0 at the center of the earth (i.e. the geocentric time).

the LIGO-Livingston detector, and “V1" for the Virgo detctor.

To complete the Bayesian sampler, we specify the likelihood and prior distributions. The multi-dimensional likelihood is given by Eq. (2.9), which is calculated for each of our parameters $\theta = \{h_0,\ \tau,\ f_0,\ t_0\}$. We also specify un-informative priors since we want to generalize our model as much as possible (albeit the constraints on the possible amplitude, frequency, and damping times). The priors are given as follows:

- Amplitude h_0: Log-uniform from 10^{-24} to 10^{-19}.
- Damping time τ: Uniform from 0.087 to 10 s.
- Frequency f_0: Uniform from 792 to 4250 Hz.
- Geocentric time of arrival t_0: Uniform on the whole fake OSW.

The extrinsic parameters — those listed in Table 1.1 and the extra phase $\varphi = 0$ — are assumed by `bilby` to be constant throughout the sampling process. They are stored inside `bilby` as a `DeltaFunction` prior object. We also make sure that the injected waveforms' parameters are contained within the prior, else the sampler would not be able to recover the correct parameters properly.

We use `bilby`'s `dynesty` wrapper to facilitate the sampling process. This sampler can be specified further by using keyword arguments that control how the sampler behaves. For all runs in this thesis, we use the same parameters: Number of initial live points `nlive=1000`, stopping criterion `dlogz=0.1`, sampling method `sample="acceptance-walk"` (sets all MCMC chains to the same length), and Poisson distribution mean `naccept=60` for the number of accepted steps. [54]

`bilby` calculates the noise evidence $Z_N \equiv \mathcal{L}(d|\theta = 0)$ even before it starts to sample. In this chapter, the noise (log) evidence will be compared with the signal evidence,

$$Z_S = \int d\theta\ \mathcal{L}(d|\theta)\pi(\theta) \tag{3.9}$$

where the integral is carried over the whole parameter space spanned by our prior. Using the signal-to-noise (log) Bayes factor,

$$BF_{S/N} = \frac{Z_S}{Z_N} \tag{3.10}$$

$$\ln BF_{S/N} = \ln Z_S - \ln Z_N. \tag{3.11}$$

This quantity will be the criterion we consider to assess the search sensitivity to a signal, to test if the data prefers the model that “the signal is present".

3.3 Results

To show the pipeline's performance, we check the results for a particular simulated data injection, with the injected damped sine's parameters $h_0 = 1 \times 10^{-20}$, $\tau = 0.5\,\mathrm{s}$, $f_0 = 2.7\,\mathrm{kHz}$. The sampler for this particular run produced $\ln BF_{S/N} \approx 21772$ which is really above the threshold value of 8. The pipeline tells us that the data containing this high amplitude injection strongly prefers the model with the signal present. Figure 3.4 shows the posterior distributions for this particular injection. In this figure, we see that the one-dimensional marginalized distributions follow the Gaussian likelihood of Eq. (2.7), which means the sampler has succeeded in finding a good estimate of the (injected) parameters. Indeed, the maximum likelihood estimates (MLE, given on top of the corner plots) are very close to the injected parameters. We also notice the presence of a degeneracy, given by the shape of the two-dimensional distribution between h_0 and τ. This means there is some dependence between the two parameters, which is expected since they both affect the amplitude of the signal and not the phase.

We can further assess the sampler's convergence by checking the trace plot, which gives a rough idea of how the live points explore the parameter space. Figure 3.5 shows the trace plot for the same run whose corner plot is shown in Figure 3.4. We notice the behavior of the points as they converge to certain values in the parameter space. The final points are then collected to obtain the posterior.

3.3.1 Simulated Gaussian noise data analysis

In this section we summarize the results of the pipeline that sampled from simulated Gaussian noise data. For the set of injections that varied amplitude and kept the rest of the parameters constant, we sample all parameters and compute $\ln BF_{S/N}$ of each injection run. Figure 3.6 shows a plot of $\ln BF_{S/N}$ as a function of the injected amplitude, where each injection and sampling run corresponds to one point.

We find that there's a certain amplitude where the log Bayes Factor starts to become positive and greater than the threshold for strong evidence $\ln BF_{S/N} = 8$, which means that from this amplitude upwards, the data prefers the signal model.

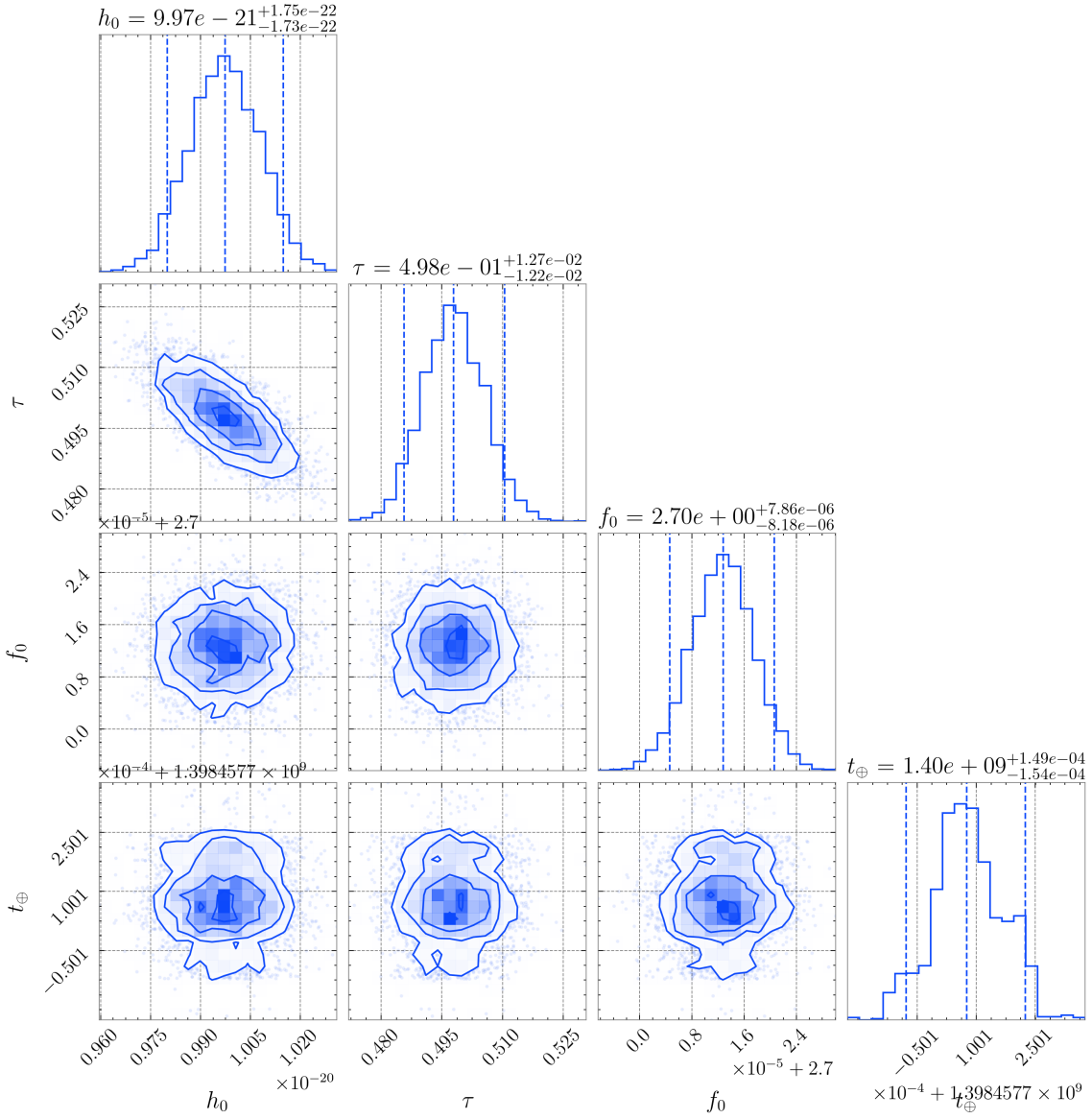

Figure 3.4: Corner plot of the damped sinusoid parameters' posterior distributions for a damped sinusoid signal with parameters $h_0 = 1 \times 10^{-20}$, $\tau = 0.5\,\mathrm{s}$, $f_0 = 2.7\,\mathrm{kHz}$ added to simulated data of the LIGO Hanford, LIGO Livingston and Virgo detectors (at their design sensitivity).

Conversely, below this amplitude threshold all log Bayes Factor values are negative and have almost the same value, which means the data prefers the noise model no matter which amplitude we inject. The sampler says that the injected signals will be buried in the noise and it prefers that the data is described by a noise-only model. For injections onto simulated data, this amplitude threshold is between 3.455×10^{-22} (with a corresponding $\ln BF_{S/N} \approx 0.95$) and 4.125×10^{-22} (with a corresponding $\ln BF_{S/N} \approx 10$); we label this amplitude $h_{0,t} \sim 4 \times 10^{-22}$.

Figure 3.7 shows the injected waveform with amplitude $h_0 = 10^{-20}$, superimposed with the results of the sampler with $\ln BF_{S/N} \approx 21772$. The maximum

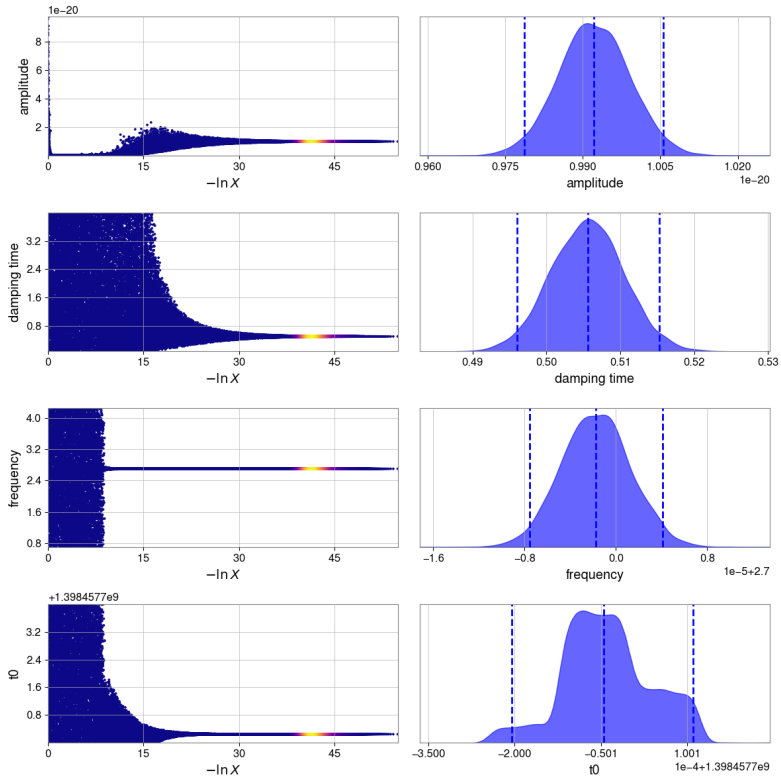

Figure 3.5: Trace plot of the sampler whose corner plot is shown in Figure 3.4. The left column shows the “live points" that roam the prior space as the prior volume $\ln X$ decreases. The right column shows the final one-dimensional marginalized distributions of each parameter

likelihood estimate (MLE) from the sampler leads to a faithful reconstruction of the injected signal; both the injected and MLE signals are well within the 2σ point-wise estimate of waveform values (sampled from the posterior distributions in Figure 3.4). The green areas (called the credible interval) show the possible values of the waveform estimates derived from the posterior. The injected amplitude here is way above the amplitude threshold $h_{0,t}$ set earlier. The injected signal is loud enough so that the sampler can recover its parameters properly, which reflects the corresponding corner plot in Figure 3.4.

On the other hand, the reconstructed waveform of the injection run with $h_0 = 3.455 \times 10^{-22}$ and $\ln BF_{S/N} = 0.9500$ is compared to the injected waveform in Figure 3.8. We see that even if the $\ln BF_{S/N}$ value is positive, the maximum likelihood waveform does not entirely match the injected waveform. Even then, both these waveforms are well within the point-wise 90% credible interval. The sampler says that the data prefers the signal model, but the preference is not strong i.e. there is not enough evidence to prove that a signal really exists within the data.

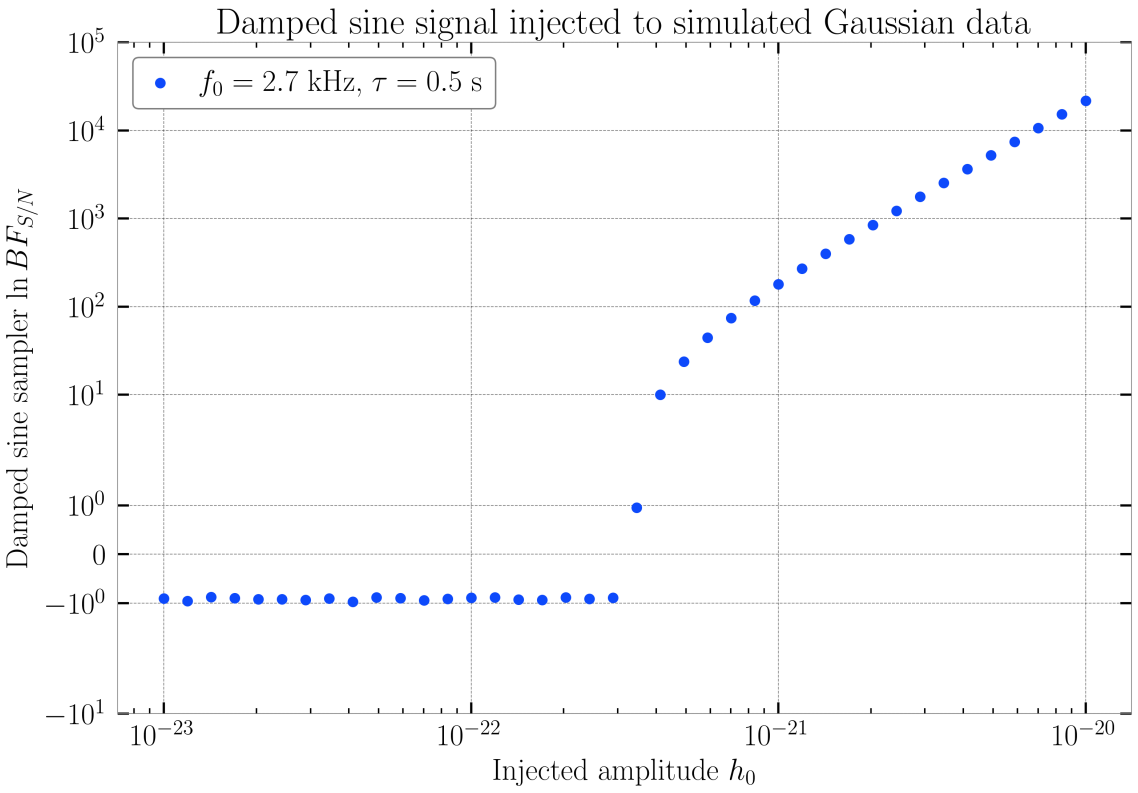

Figure 3.6: The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the damped sinusoid signal amplitude injected onto simulated data. For $\ln BF_{S/N} > 0$, the data prefers the signal model, $s(t) = n(t) + h(t)$, while negative $\ln BF_{S/N}$ means that the data prefers the noise-only model, $s(t) = n(t)$.

It is also interesting to track the log Bayes Factor with respect to the root-sum-square amplitude h_{rss} and signal-to-noise ratio (SNR) – quantities that are directly related to the energy output and detectors' sensitivity, respectively. Figure 3.9 shows this trend. The color bar corresponds to the value of $\ln BF_{S/N}$, where the colors diverge from the central value of $\ln BF_{S/N} = 8$ (the threshold value for which the signal model is "strongly preferred" over the noise model). We see that for $h_{rss} > 10^{-22}$ and about SNR > 8 the data starts to prefer the signal model. As a reference, the previous calculations on magnetar f-mode parameter estimation in simulated data quote $h_{rss} \approx 3 \times 10^{-23}$ as the threshold for damped sine signals to have $\ln BF_{S/N} > 8$ at a SNR $= 8$ [27].

Another test we can carry out is to vary the injected signal frequency aside from the amplitude. We develop another set of injections that uses the same amplitude range as the runs before, while the frequency for each injection is drawn from a uniform distribution that ranges from 0.8 to 3.9 kHz. We run the same injection and sampling procedures outlined in Section 2.4 and recover $\ln BF_{S/N}$ for each run. Figure 3.10 shows how the log Bayes Factor changes with respect to the injected

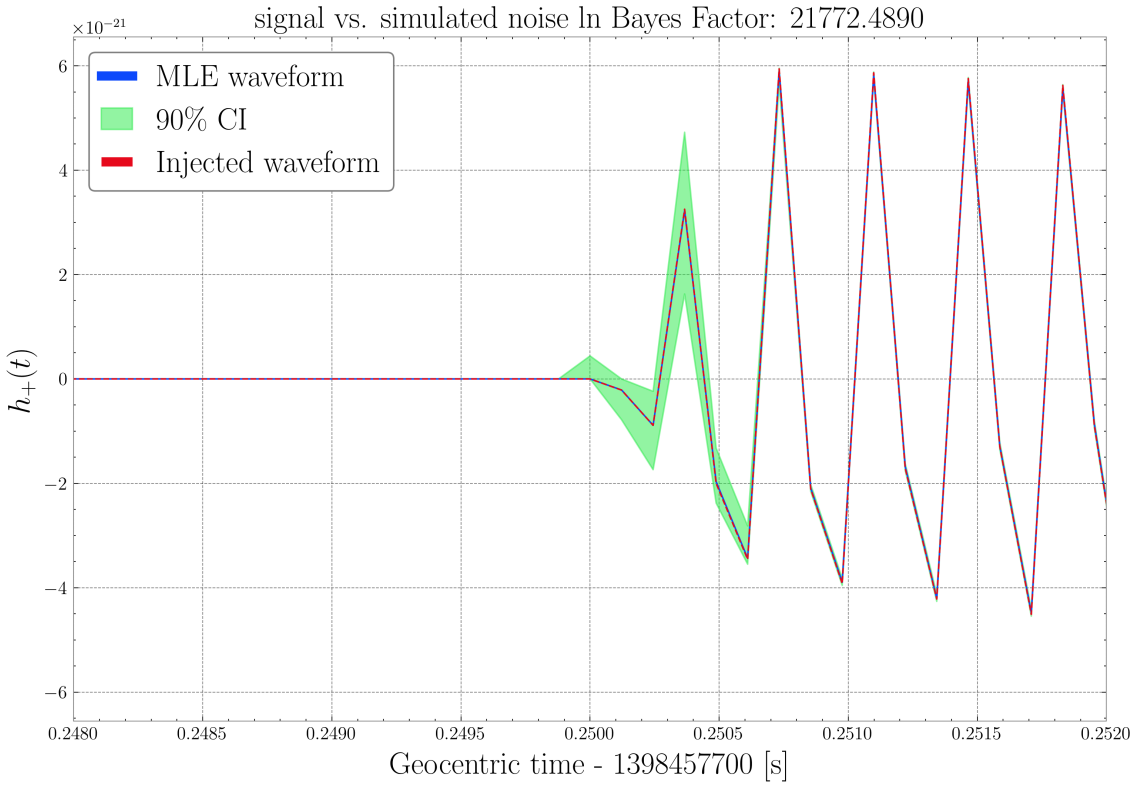

Figure 3.7: Reconstructed 90% credible interval for an injected damped sinusoid waveform with $h_0 = 10^{-20}$ and $\ln BF_{S/N} \approx 21772$, together with the injected and maximum likelihood waveforms.

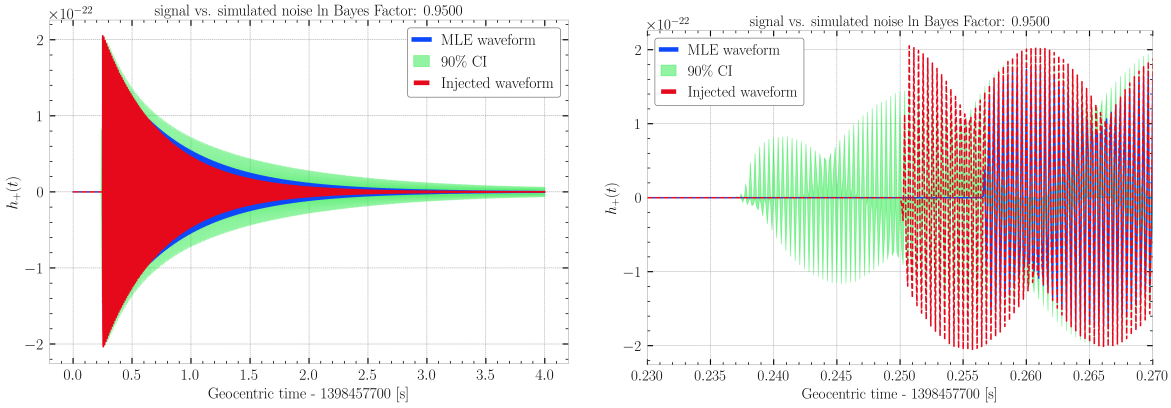

Figure 3.8: Left: Reconstructed 90% credible interval for an injected damped sinusoid waveform with $h_0 = 3.45 \times 10^{-22}$ and $\ln BF_{S/N} \approx 0.95$ together with the injected and MLE waveforms. Right: A zoom-in of the figure on the left.

amplitude and frequency. We see that there is a slight frequency dependence for $\ln BF_{S/N}$ when the amplitude is kept constant. This dependence is due to the effect of the noise PSD. In Figure 1.1 we see that the amplitude of the noise curve increases with frequency. This means that when we add a monochromatic signal with an ever-increasing frequency and keeping the amplitude constant, the signal is more and more buried in the noise, with a SNR decreasing as the

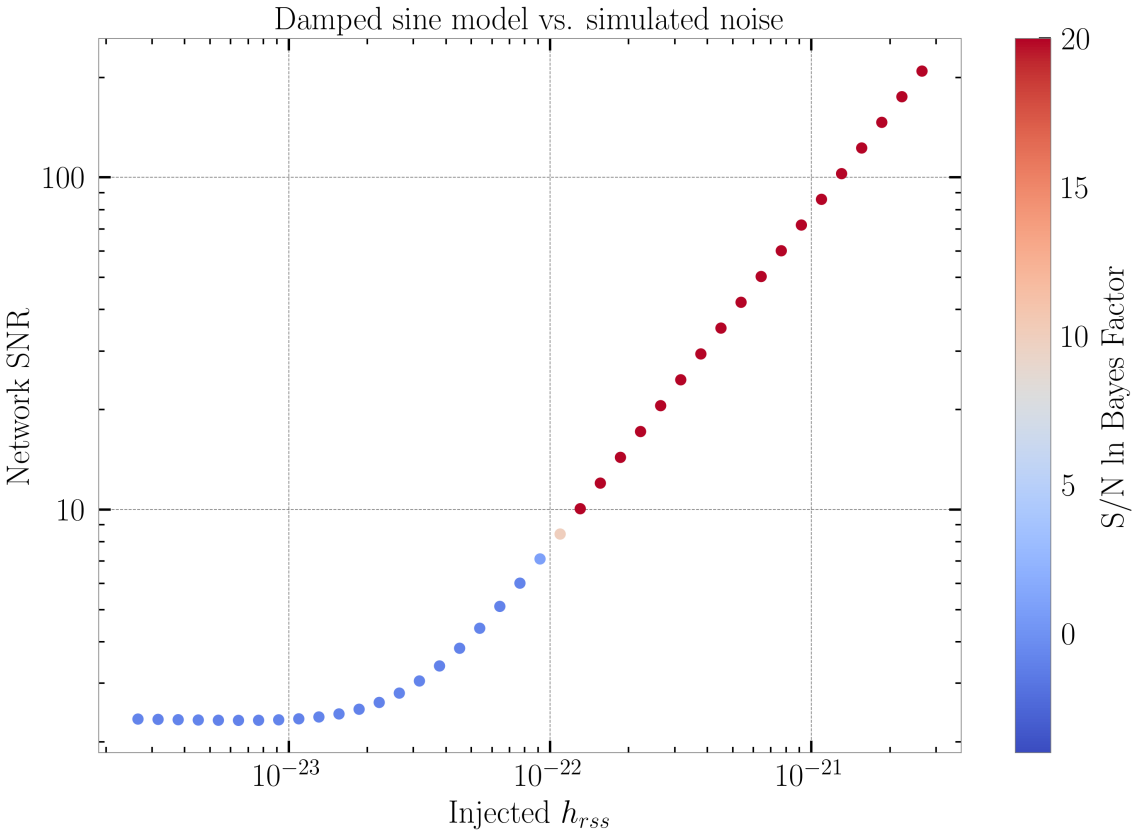

Figure 3.9: Network SNR and $\ln BF_{S/N}$ as a function of the (Vela-oriented) h_{rss} of the injected signal. The red points correspond to runs with $\ln BF_{S/N} > 8$

3.3.2 LIGO-Virgo O4 data analysis

Now that our pipeline is fully characterized with simulated Gaussian noise, we can turn to LIGO-Virgo O4 data. In this section, we label the analysis as 'closed-box' as we inject the waveforms on a fake OSW (recall the gray datapoints in Figure 2.1) which is a segment of real data having the same duration as the simulated data but taken a few minutes before the OSW start. The practice in LVK collaboration is that scientists do not "open" the box (the data from real OSW that potentially contains a GW signal) until closed-box results have been confirmed and verified. [6]

To see the difference between simulated Gaussian and real GW strain data, a comparison between their ASDs is shown in Figure 3.11. Indeed, the real data ASD contains more noise sources than the design sensitivity curves. These noise sources show up as vertical lines in the ASD that correspond to different types of resonances of the detectors. For instance, the lines around 1000 Hz correspond to

[6] This means that the data used for this analysis does not involve the GW strain data data corresponding to the time of the event.

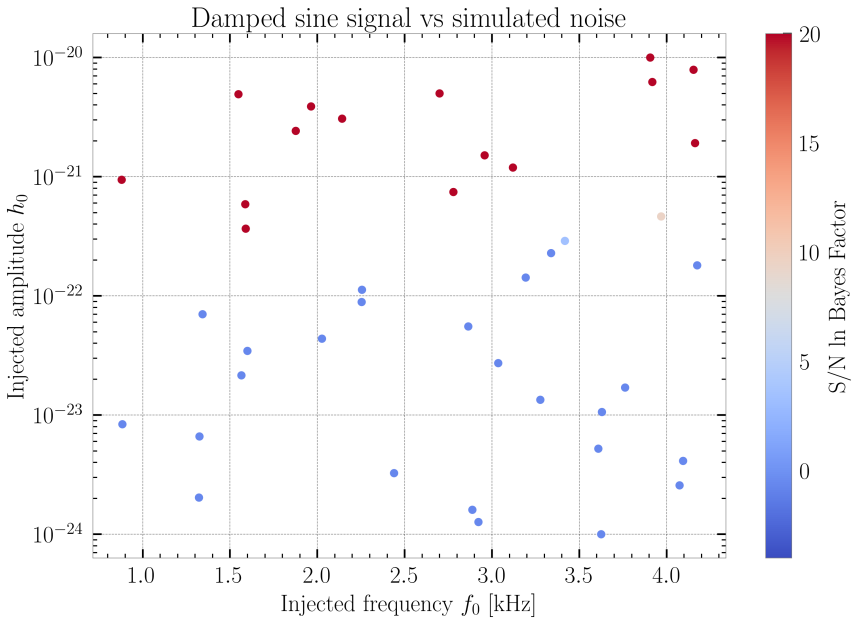

Figure 3.10: Another set of injection runs where both the injected amplitude and frequency varies for each run. The colors of $\ln BF_{S/N}$ diverge from the threshold value $\ln BF_{S/N} = 8$, with red points indicating a strong preference for the existing signal model.

the violin oscillation modes of the arm mirrors. It is also interesting to note that the Virgo data has a limited sensitivity with respect to the LIGO data; its ASD is also higher than the expected design sensitivity. For this reason we perform the injection and sampling only on LIGO H1 and L1 data.

As in the study with simulated data, we inject several damped sinusoid waveforms of varying amplitudes to the fake OSW of the LIGO O4 data, then we recover the posterior and $\ln BF_{S/N}$ and establish a lower bound from which the data strongly prefers the injected signal model. Initially, a set of 100 injections of varying amplitude ($h_0 \in [10^{-23}, 3 \times 10^{-21}]$) was sampled using the same prior ranges and injection parameters (except for f_0 which is now 1.6 kHz) as described in Section 3.2.3. We obtain $\ln BF_{S/N}$ from each run and plot them against the injected amplitudes in Figure 3.12.

We notice an exponential trend once the amplitude is above a certain threshold. However, unlike in the simulated data where the threshold set is between a positive and negative $\ln BF_{S/N}$ value, this time the $\ln BF_{S/N}$ values do not go below 126, which is way above the threshold $\ln BF_{S/N} = 8$. Even if the injected amplitude gets lower than the established threshold $h_{0,t} \sim 4 \times 10^{-22}$, having a value of $\ln BF_{S/N} > 126$ means that the data is still preferring the existence of a signal, no matter how small the injected amplitude is.

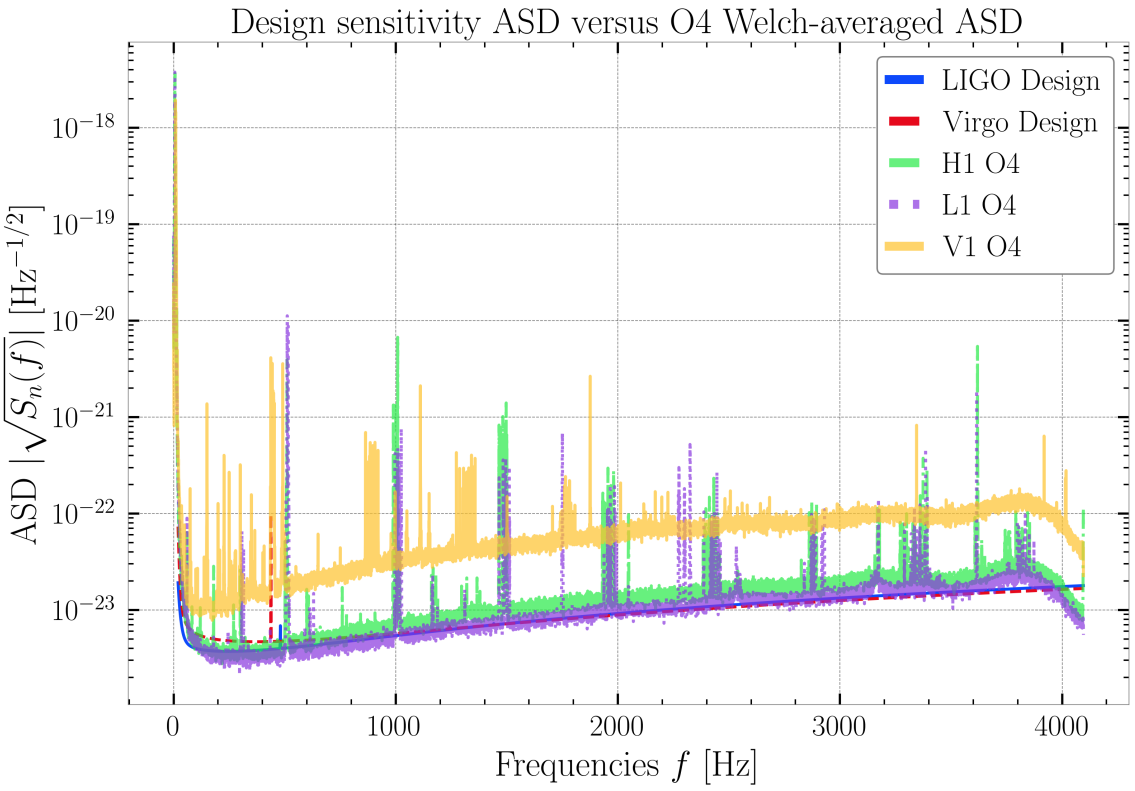

Figure 3.11: O4 ASD superimposed onto Figure 1.1. The O4 ASDs are calculated using 128 seconds of background data from the Hanford (H1), Livingston (L1), and Virgo (V1) interferometers, Tukey windowed and Welch-averaged every 4 seconds with 0.4-second overlaps.

To figure out the problem, we check one of the corner plots of these low-amplitude injections. Figure 3.13 shows the corner plot for an injected waveform of amplitude $h_0 = 1.0 \times 10^{-23}$ and frequency $f_0 = 1.6$ kHz. From the corner plot, it seems that the sampler succeeded in recovering a damped sinusoid with a maximum likelihood amplitude of around 4.41×10^{-22} and frequency of around 1.01 kHz which are different from the injected values. The frequency range near 1.01 kHz (as shown in Figure 3.11) actually corresponds to a region where lines are present in the detector's PSD. That means the sampler is recovering a noise line that is mistaken as a damped sinusoid signal (i.e. one of the violin modes can be modeled as a damped sine time series). This recovery is quantified by a high $\ln BF_{S/N}$, which means the data strongly prefers the signal being present from what are physically noise lines.

To test this hypothesis, we limit the frequency prior inside `bilby` such that the frequency range does not include any high amplitude lines. This prompts `bilby` to sample the frequencies inside this limited bandwidth. We work with the bandwidth $2.5 - 3.0$ kHz (with the detectors' corresponding noise ASD shown in

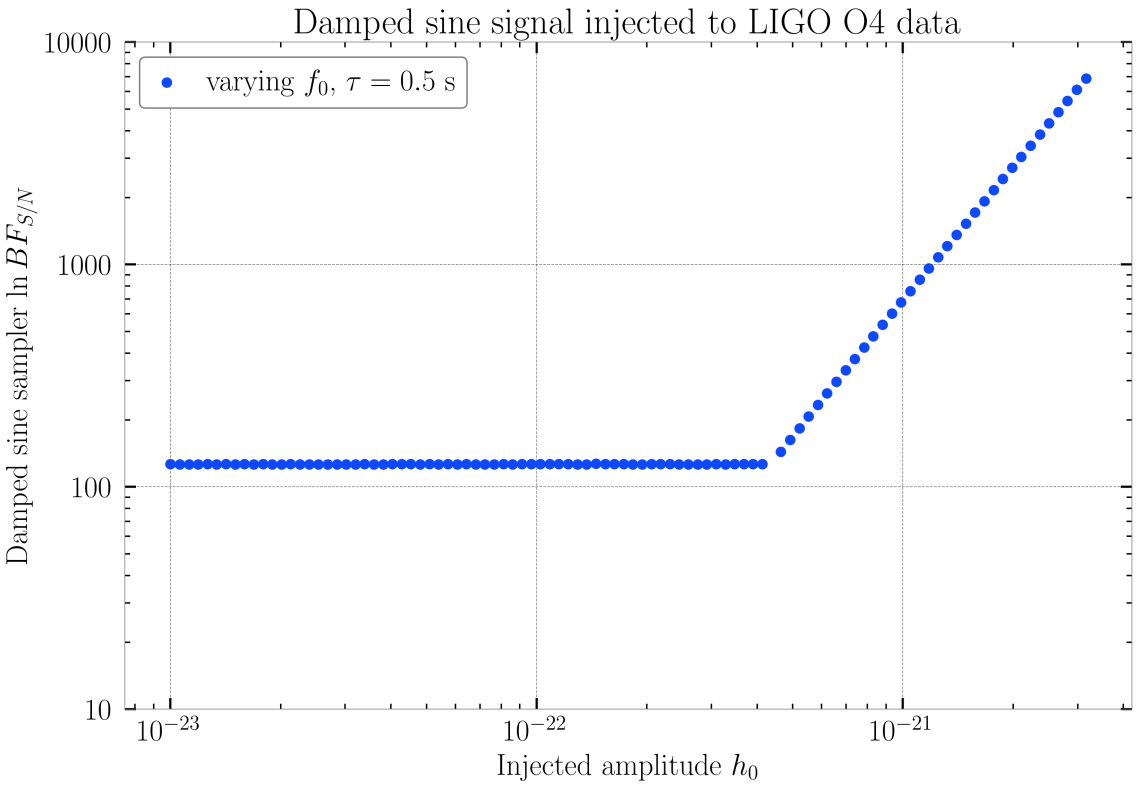

Figure 3.12: The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the amplitude of the damped sinusoid injected onto LIGO-Virgo O4 data. The minimum $\ln BF_{S/N}$ recorded was around 126, which is a really strong evidence for the presence of a damped sinusoid.

Figure 3.14) and inject waveforms in the frequency 2.7 kHz. Then a set of 40 damped sinusoid signals of varying amplitudes $h_0 \in [10^{-23}, 10^{-20}]$ was sampled.

We plot the recovered signal-to-noise $\ln BF_{S/N}$ against the injected waveform's amplitude for the reduced frequency prior runs in Figure 3.15. We see that the lowest positive $\ln BF_{S/N}$ value belongs to the injection run with $h_0 = 5.878 \times 10^{-22}$ with its corresponding $\ln BF_{S/N} = 6.59$. This value is below the strong evidence threshold, but its positive value means that the data already prefers the signal over the noise model. Below this amplitude, the $\ln BF_{S/N}$ is already negative. Checking the corner and trace plots of this particular injection, we see that all sampled parameters h_0, τ, f_0, t_0 are recovered properly and that their MLE are close enough to the injected values. The next data point is an injection with $h_0 = 7.443 \times 10^{-22}$ with a corresponding $\ln BF_{S/N} = 22.19$. We can conclude that the amplitude threshold for the O4 data surrounding the Vela pulsar glitch should be around $6 - 7 \times 10^{-22}$ in order for the sampler to conclude a strong preference for the damped sinusoid signal to exist within the data. Note that this is higher than the threshold set for simulated data $h_{0,t} \sim 4 \times 10^{-22}$ which reflects the effect

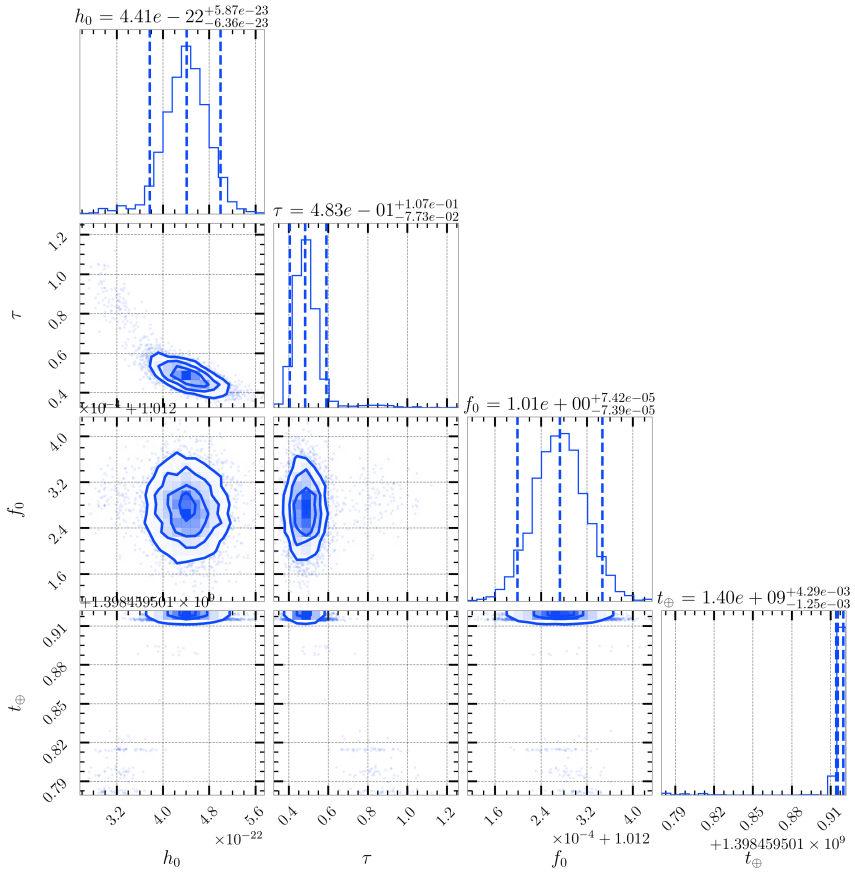

Figure 3.13: Corner plot for a damped sine with amplitude $h_0 = 1.0 \times 10^{-23}$ injected to O4 data, with a corresponding $\ln BF_{S/N} \approx 126$. For a low-amplitude injection, its Bayes factor is suspiciously high, meaning the sampler is picking up a damped sinusoid signal at a frequency of about 1.01 kHz.

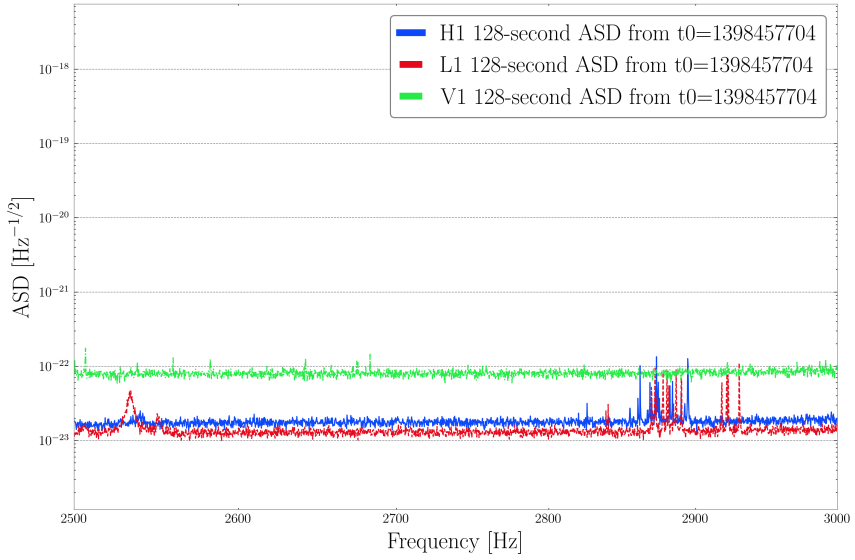

Figure 3.14: O4 noise ASD within the limited bandwidth from $2.5 - 3.0$ kHz. This data is considered to understand the effect of the experimental lines in the real GW data.

of the higher ASD values from O4 than the design sensitivity data as shown in Figure 3.11.

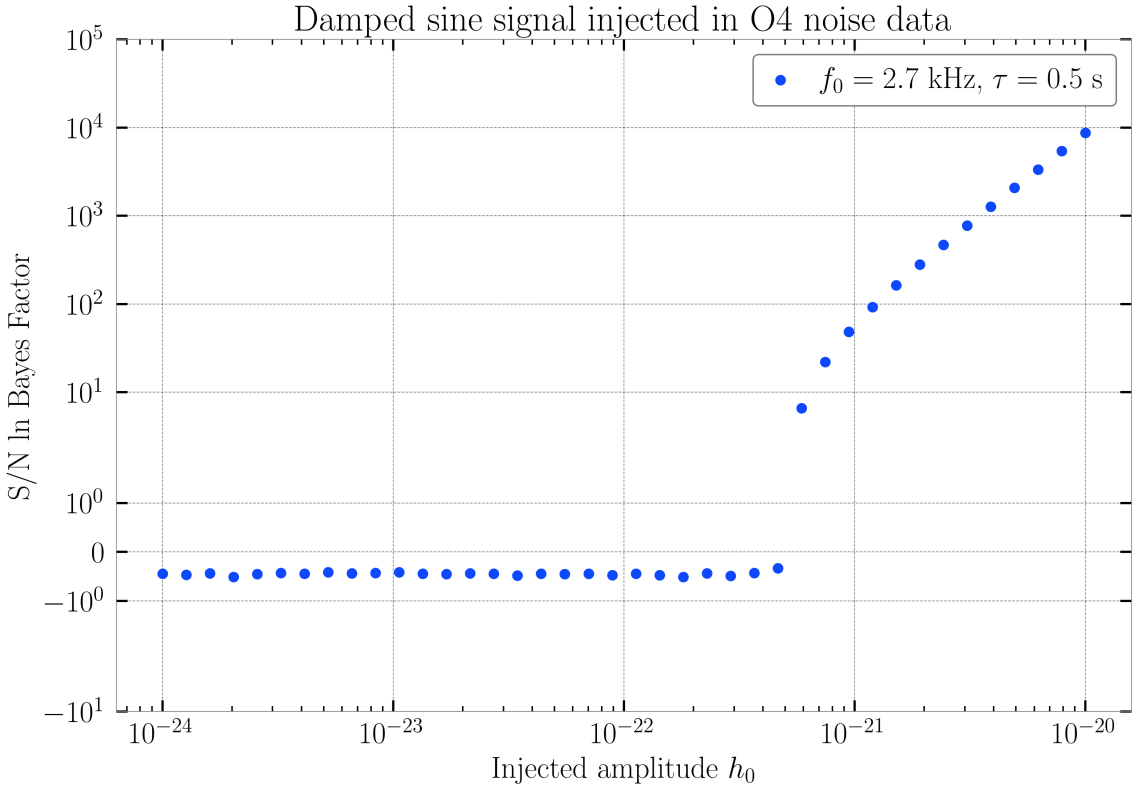

Figure 3.15: The signal-to-noise Bayes factor $\ln BF_{S/N}$ as a function of the corresponding damped sinusoid amplitude injected onto real LIGO-Virgo O4 data. The samplers used in this plot used a limited frequency prior, $f_0 \in [2.5, 3.0]$ kHz. The amplitude threshold where the data has strong preference for the signal model is at $h_0 = 5.878 \times 10^{-22}$.

3.3.3 Removal of noise lines in real data ASD

To carry out the search in O4 data, we need to mitigate the effects of the lines present. In literature there have been attempts on systematically removing the noise lines in ASD of real GW strain data; one example uses MCMC to fit several Lorentzians to the lines and adapt each of them [55]. Several attempts on lines removal have been made on our end but the following approach seems the most promising. We use a data preprocessing method called “notch filtering". This method removes certain frequencies from the analysis such that the ASD curve of the strain data goes to zero in these frequencies. In effect, the notch filter will mask the frequency bins and remove them in the likelihood calculation of Eq. (2.9). `bilby` provides such a function for notching certain frequency ranges, however the ranges must be manually given by the user. We supply the range of

frequencies where we see the forest of noise lines as illustrated in Figure 3.11. The masked ASD for the two LIGO interferometers is shown in Figure 3.16. We have not cleared all the lines within the prior range $f_0 \in [0.8, 4.0]$ kHz, but later we show that this process suffices.

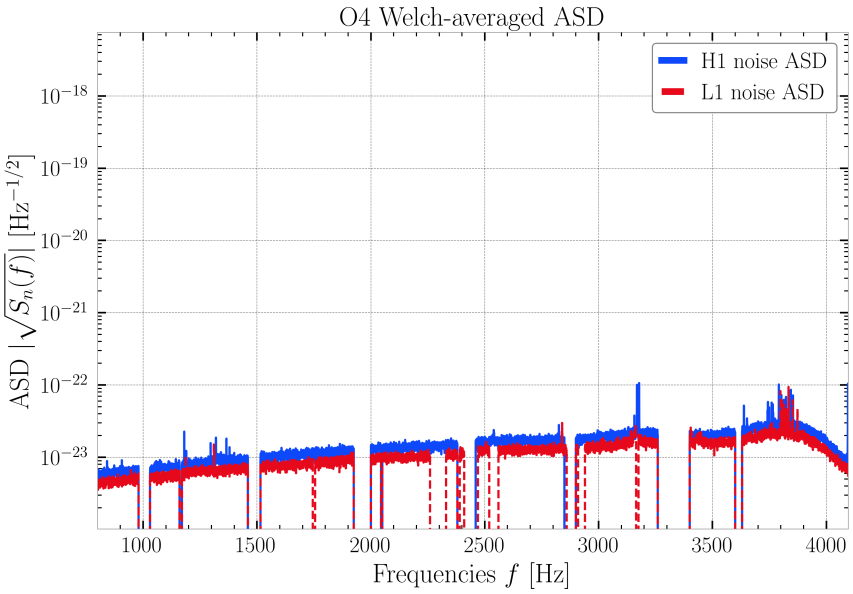

Figure 3.16: Notch-filtered noise ASD.

Then, we can let the sampler search throughout the uniform prior $f_0 \in [0.8, 4.0]$ kHz again. After the pipeline is applied, we obtain $\ln BF_{S/N}$ and compare it with the injected amplitude in Figure 3.17. We also superimpose the $\ln BF_{S/N}$ values obtained with the reduced frequency prior from Figure 3.15. The trend and the injection run where the $\ln BF_{S/N}$ becomes positive is the same for both processes (albeit the lower $\ln BF_{S/N}$ values for the notched O4 data), so it is advisable to mask the frequencies of the prevalent noise lines.

3.4 Discussion and conclusion

In this chapter, we use the signal-to-noise Bayes Factor $\ln BF_{S/N}$ to evaluate whether the data prefers a model that a GW signal is present or when the data prefers a noise-only model. The closed-box results shown in this chapter are meant to serve as data quality checks for the Bayesian search of a GW burst signal in O4 data surrounding the Vela glitch. Indeed, the burst inference pipeline we introduced can reconstruct a certain signal with physically meaningful parameters from O4 data. Our closed-box results also give us an idea of what to expect when a burst GW search is conducted. We find two amplitude thresholds –

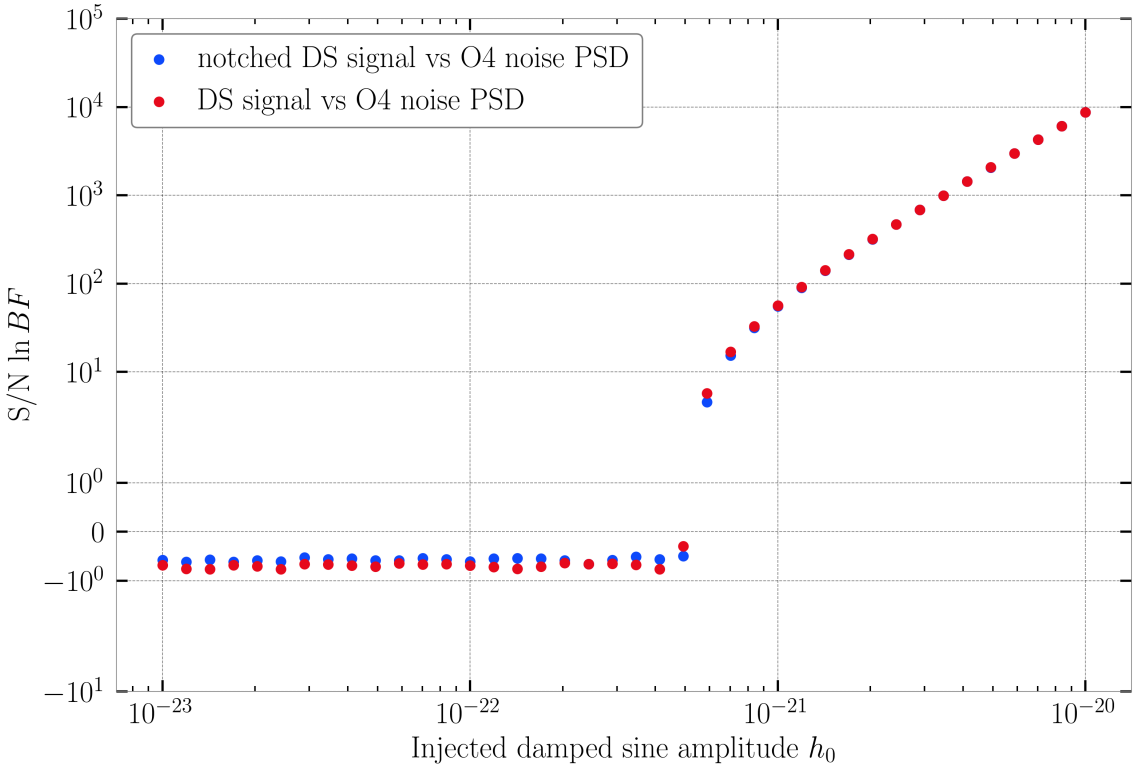

Figure 3.17: The signal-to-noise Bayes factorln $BF_{S/N}$ as a function of the damped sinusoid amplitude injected to the notched LIGO-Virgo O4 data (blue points), superimposed on Figure 3.15 (red points).

one for the simulated Gaussian data ($h_{0,t} \sim 4 \times 10^{-22}$) and another for the O4 data ($h_{0,t} \sim 7 \times 10^{-22}$). If the burst GW signal from the Vela pulsar glitch has an amplitude above this threshold, then we can expect to find it within the collected O4 data. The pipeline made this inference by analyzing the dependence of $\ln BF_{S/N}$ on the injected amplitude.

The next step is to apply this pipeline to a real OSW to search for a certain signal (such as “opening the box" for the Vela glitch). In this step, we do not need to inject a signal; rather, we model the data such that it contains a signal, then we can calculate this signal's $\ln BF_{S/N}$ and compare the results with other detection statistics. In case of a $\ln BF_{S/N}$ result below the strong evidence threshold, we will estimate the 90% confidence level on the GW energy calculated with Eq. (3.7) from the posterior distribution on h_0.

Chapter 4

Burst model comparison using the Bayesian inference pipeline

In this chapter, we aim to determine the nature of a burst GW detection in a Bayesian manner. It was established in Chapter 1 that, due to the vagueness and few assumptions of the detection methods for burst-type GWs, we are not able to determine which source produced the signal as easily as for CBCs. We propose to use the Bayes factor to compare different signal models given a particular waveform signal exists in the data. For two disparate signals labeled as A and B, we can compute their evidences using

$$\begin{aligned} Z_A &= \int d\theta \ \mathcal{L}(\theta)\pi_A(\theta) \\ Z_B &= \int d\vartheta \ \mathcal{L}(\vartheta)\pi_B(\vartheta) \end{aligned} \tag{4.1}$$

where θ and ϑ are the signal parameters for signal model A and B, respectively. [1] It is assumed that we use the same (log) likelihood, Eqs. (2.7) and (2.9), for both models. Then the model comparison Bayes factor is given by $BF_{A/B} = Z_A/Z_B$.

We reuse the pipeline described in Sections 2.4 and 3.2.2. The strain data are processed in the same manner, except that we sample different kind of waveforms which are disparate from the injected GW signal. This changes the signal template μ in the likelihood calculation of Eq. (2.9). Then, to solve the posterior in Eq. (2.3) and the signal evidence in Eq. (4.1) we supply the prior of this disparate signal template. In theory, this process mimics an unmodeled search: we are not really sure what to look for so we test different burst models. But since we injected the waveform ourselves we know what to expect, and the sampler determines if the sampled waveform can be inferred from the data.

[1] It does not matter if the number of parameters are different for the two models.

4.1 Models used for burst GW source identification

In this section, we list down some of the possible burst GW models we can use together with their possible sources. So far in this thesis we have used the damped sinusoid (labeled *DS*) waveform as a model of the GW emission due to the f-mode excitation of neutron stars. Another burst signal that can be used for multiple models is the sine-Gaussian (labeled *SG*) which is implemented in `bilby` as [44]

$$h_+(h_{\rm rss},\tau,f_0) = \frac{h_{\rm rss}\sqrt{\pi}\tau}{2\sqrt{\alpha(1+e^{-Q^2})}}\left(e^{-\pi^2\tau^2(f+f_0)^2}+e^{-\pi^2\tau^2(f-f_0)^2}\right) \tag{4.2}$$

$$h_\times(h_{\rm rss},\tau,f_0) = \frac{ih_{\rm rss}\sqrt{\pi}\tau}{2\sqrt{\alpha(1-e^{-Q^2})}}\left(e^{-\pi^2\tau^2(f+f_0)^2}-e^{-\pi^2\tau^2(f-f_0)^2}\right) \tag{4.3}$$

where

$$\tau = \frac{Q}{\sqrt{2}\pi f_0} \tag{4.4}$$

$$\alpha = \frac{Q}{4\sqrt{\pi}f_0}. \tag{4.5}$$

This makes the sine-Gaussian a frequency-domain model (which is actually preferred by `bilby` as the likelihood calculations are carried out in the frequency domain). The sine-Gaussian has been used as a wavelet basis for decomposing complicated burst-type GW emissions as well as modelling transient noise [21]. It can also serve as a wavelet basis for the core-bounce part of a core-collapse supernova (CCSN) [20]. We will see later how its generic nature can complicate burst GW identification.

Yet another class of burst-type GW signals comes in the form of power-law emissions. This type of burst signal may be indicative of a cosmic string, which have cusps and kinks in its loops. These cusps and kinks produce gravitational radiation which is implemented in `LALSuite` [39] as

$$\begin{aligned}
\tilde{h}_+(f;h_0,f_{high}<f,\alpha,f_{low},t_0) &= h_0 f^{-\alpha}\left(1+\frac{f_{low}^2}{f^2}\right)^{-4}\exp\left(1-\frac{f}{f_{high}}\right)\exp(-2\pi\imath f t_0)\\
\tilde{h}_+(f;A,f_{high}\geq f,\alpha,f_{low},t_0) &= h_0 f^{-\alpha}\left(1+\frac{f_{low}^2}{f^2}\right)^{-4}\exp(-2\pi\imath f t_0)\\
\tilde{h}_\times(f) &= 0.
\end{aligned} \tag{4.6}$$

Here, we modify the power-law model such that we include a phase factor for the time of arrival t_0. For cosmic string cusps, $\alpha = 4/3$ while for cosmic string kinks, $\alpha = 5/3$.

In general, any waveform can be applied to our pipeline for sampling and inference. This includes the phenomenological waveforms used for CBC parameter estimation (e.g. the Inspiral-Merger-Ringdown family of waveforms [56], with a general form $h(f) = \mathcal{A}(f)e^{\imath\Phi(f)}$) or the waveforms with changing frequency (e.g. in the post-merger of a BNS inspiral [57]). The flexibility of the pipeline allows us to compare the signal evidences of these waveforms with one another.

4.2 Results

In this section, we present some results from the application of our burst inference pipeline to the comparison of signal models. In particular, we adapt the pipeline from Section 2.4 to a comparison scenario, where we inject a damped sinusoid and then sample a custom cosmic string cusp waveform and a sine-Gaussian waveform.

To make things simpler, we use simulated Gaussian strain data with a damped sinusoid waveform injected (the same dataset from Chapter 3). Then we call an instance of the sine-Gaussian waveform generator from `bilby` and write our own instance of the cosmic string cusp generator. We also supply the sampler with the corresponding priors for the different signal models. Aside from the posterior distributions of the cosmic string cusp and sine-Gaussian waveforms' parameters, the results give the natural logarithm of the signal evidence in Eq. (4.1) and then we can compare these new evidences with the signal evidence obtained from previously sampling a damped sinusoid in Chapter 3. This creates a different (log) Bayes factor that is used for model comparison,

$$\ln BF_{DS/B} = \ln Z_{DS} - \ln Z_B \tag{4.7}$$

where $\ln Z_{DS}$ is the previously calculated damped sinusoid log signal evidence and $\ln Z_B$ can be the log signal evidence of the sine-Gaussian SG or the cosmic string cusp CS signal models.

4.2.1 Neutron star f-mode excitation versus cosmic string cusp

We compare the cosmic string power-law GW signals with the damped sinusoid GW signals from neutron star f-mode excitations. This analysis has been done before [38] where they sampled a (confirmed) BBH merger using cosmic string models to test some model comparison hypotheses. In particular, we compare the cosmic string cusp with power-law index $\alpha = 4/3$. From Eq. (4.6) we see that the waveform is linearly polarized such that only h_+ is nonzero. The parameters that will be sampled include the waveform amplitude h_0, high frequency cutoff f_{high}, low frequency cutoff f_{low}, and time of arrival t_0 [2]. Recall from Chapter 1 that the high frequency cutoff f_{high} dictates the geometry of the GW emission [58], while f_{low} is not physically motivated; it controls the high-pass filter term.

We use the same dataset where damped sinusoids of varying amplitude $h_0 \in [10^{-23}, 10^{-20}]$ are injected onto simulated data, with all other parameters kept constant: $f_0 = 2.7$ kHz, $\tau = 0.5$ s and t_0 set on the fake OSW. Then we supply the waveform (4.6) to `bilby` together with the corresponding priors:

- Amplitude h_0: Uniform $\in [10^{-25}, 10^{-19}]$
- High frequency cutoff f_{high}: Uniform $\in [30, 4096]$ Hz
- Low frequency cutoff f_{low}: Uniform $\in [0, 16]$ Hz
- Time of arrival (extra phase factor) t_0: Uniform $\in [0, 4]$ s

After running the pipeline, we check the posterior recovery by looking at a particular injection's result. The posterior distributions for one particular run are shown as a corner plot in Figure 4.1. This data has an injected damped sinusoid with $h_0 = 10^{-21}$. We see that there is no sufficient convergence, i.e. the sampler cannot retrieve the parameters of a cosmic string cusp waveform that fits the data. The one-dimensional marginalized distributions on the diagonals further show that the posterior almost looks like the priors we assigned, and according to Bayes' theorem Eq. (2.3) this means the likelihood of the cosmic string cusp appearing in the data is very low.

After sampling the cosmic string cusp from all 40 GW strain data with damped sinusoid injections of varying amplitude, we collect the log signal evidence $\ln Z_{CS}$

[2] Even though we are using the same notation as some of the damped sine parameters, it does not mean that these parameters are defined in the same way.

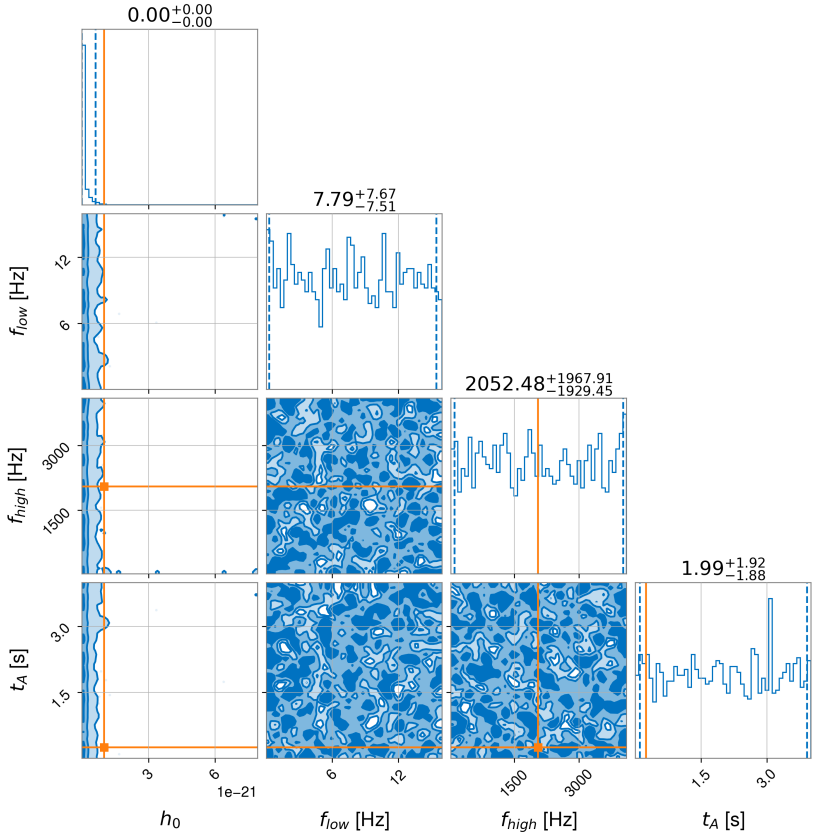

Figure 4.1: Corner plot for a cosmic string cusp sampler when there is a damped sinusoid with parameters $h_0 = 10^{-21}$, $\tau = 0.5$ s and $f_0 = 2.7$ kHz injected onto simulated Gaussian strain data. The orange lines are the injected damped sinusoid parameters.

and compare it with the log signal evidence of a damped sinusoid $\ln Z_{DS}$ from the damped sinusoid sampler. In Figure 4.2 we show $\ln BF_{DS/CS}$ as a function of the injected damped sinusoid amplitude. We see that after a certain amplitude, the log Bayes factor becomes positive. This certain amplitude is the same amplitude at which the signal-to-noise $\ln BF_{S/N}$ for the damped sinusoid becomes positive, i.e. $h_0 = 3.45 \times 10^{-22}$ (see Figure 3.6). Thus Figure 4.2 reflects the amplitude threshold for which the damped sinusoid signal starts to be observed. It implies that from this amplitude upwards, the damped sinusoid model is preferred.

However, the converse is not true: just because $\ln BF_{DS/CS} < 0$ for some injection runs does not mean the data prefers the cosmic string cusp model. We know this since we have not injected any cosmic string cusp. It might be misleading to use a signal vs. signal Bayes factor alone as it might cause confusion in burst searches where there is no injection and no *a priori* information on the source. We need to show the difference between the samplers by comparing their respective signal-to-noise $\ln BF_{S/N}$ values in Figure 4.3. Recall that $\ln BF_{S/N}$ compares the

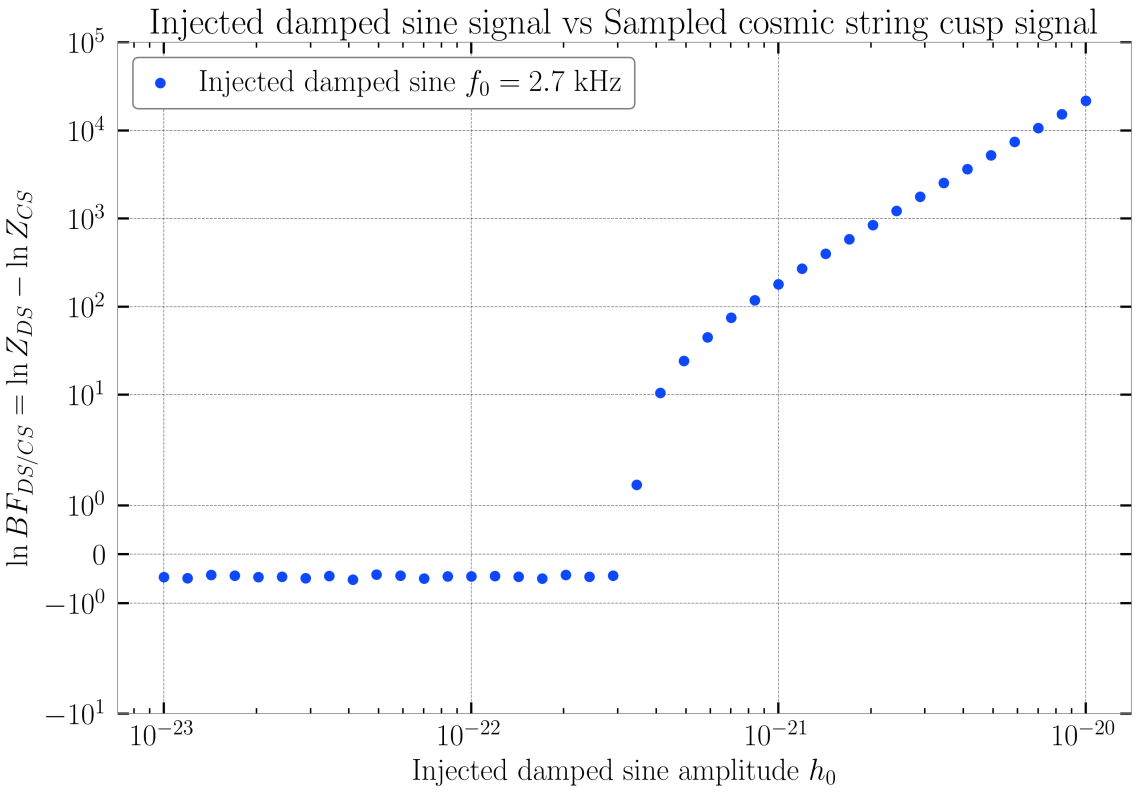

Figure 4.2: Log Bayes factor comparing damped sinusoid signals with power-law signal (from a cosmic string cusp) $\ln BF_{DS/CS}$ as a function of the damped sinusoid's amplitude injected onto simulated Gaussian strain data.

signal model against a noise model. We see that there is no change in $\ln BF_{S/N}$ for the cosmic string cusp, further supporting the fact that the power-law waveform cannot be recovered from a data with a damped sinusoid waveform injected onto it. This implies that power-law waveforms do not look like damped sinusoid waveforms regardless of the loudness of the signal. We can only deduce from our analysis that our pipeline can definitely distinguish the power-law signal from a damped sinusoid one if the signal is loud enough, otherwise the pipeline would have a difficult time telling us which model to prefer.

4.2.2 Damped sinusoid versus sine-Gaussian waveforms

In this section we compare a single sine-Gaussian waveform with the injected damped sinusoid waveform from NS f-mode excitations. This comparison is more relevant for the Vela pulsar glitch since sine-Gaussians are extensively used as wavelet basis for burst waveforms. In this study we focus on using one sine-Gaussian waveform (instead of combining multiple wavelets) to compare against a damped sinusoid waveform injected onto both simulated Gaussian noise and O4

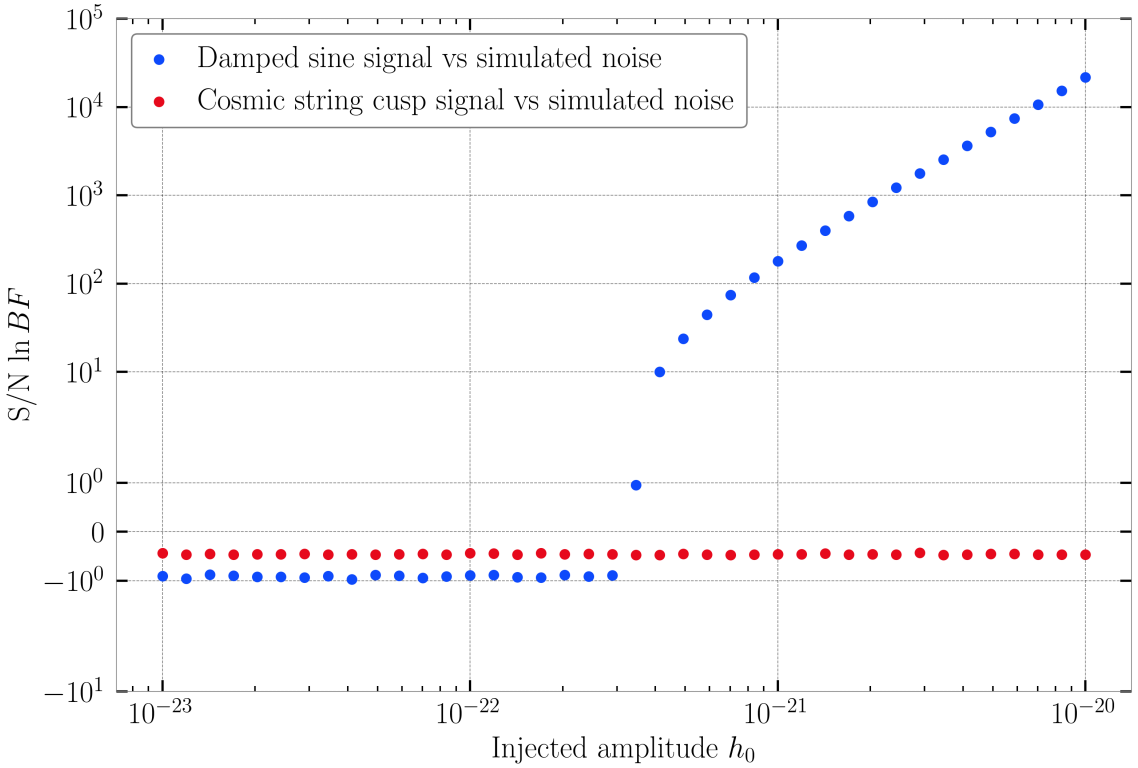

Figure 4.3: Comparing the signal-to-noise $\ln BF_{S/N}$ from the damped sinusoid and cosmic string cusp samplers when there is a damped sinusoid injected onto simulated Gaussian strain data.

data surrounding the Vela pulsar glitch. We use the same set of 40 injections where the damped sine amplitude varies from 10^{-23} to 10^{-20}. This set is similar to the one used for Chapter 3 and the cosmic string sampler; the other injection parameters are kept constant: $f_0 = 2.7$ kHz, $\tau = 0.5$ s, and t_0 is 0.25 seconds after the fake OSW start time.

In this case, the parameters that `bilby` will sample are $h_{\rm rss}, Q, f_0$ which correspond to the peak amplitude, quality factor (which dictates the number of sine cycles within the Gaussian envelope), and the central frequency of the Gaussian envelope (as viewed in frequency domain) respectively. We also sample one extrinsic parameter, the geocentric time of arrival t_0. The priors for each parameter are listed as follows:

- Peak amplitude h_{rss}: Log-uniform $\in [10^{-24}, 10^{-19}]$
- Quality factor Q: Log-uniform $\in [10^{-2}, 10^{6}]$
- Frequency f_0: Uniform $\in [0.8, 4.0]$ kHz
- Geocentric peak time t_0: Uniform on the whole fake on-source window (OSW)

This set of parameters and the range of their corresponding priors is independent of those set for the damped sinusoid. We run the sampler for each injection and check the sampler's convergence using an example corner plot shown in Figure 4.4. We see that for a very loud damped sinusoid signal with $h_0 = 10^{-20}$, the sine-Gaussian sampler picks up the same frequency as that of the injected damped sinusoid. We also see that the sampler converges for all other parameters, and together with the very high $\ln BF_{S/N}$ of this result, this means that a loud damped sine signal might be mistaken as a sine-Gaussian. In Figure 4.5 the injected damped sinusoid waveform is plotted together with the MLE reconstructed waveform for the sine-Gaussian. We see that the sampler picked out the main feature of the damped sinusoid in frequency domain (i.e. the single frequency peak), but the shape of the damped sinusoid is not that visible in the time domain. It turns out that when we zoom further in the time domain, the oscillations from the maximum likelihood estimate (MLE) sine-Gaussian follow those which are in the injected damped sine, reflecting the sampler's ability to recover the frequency.

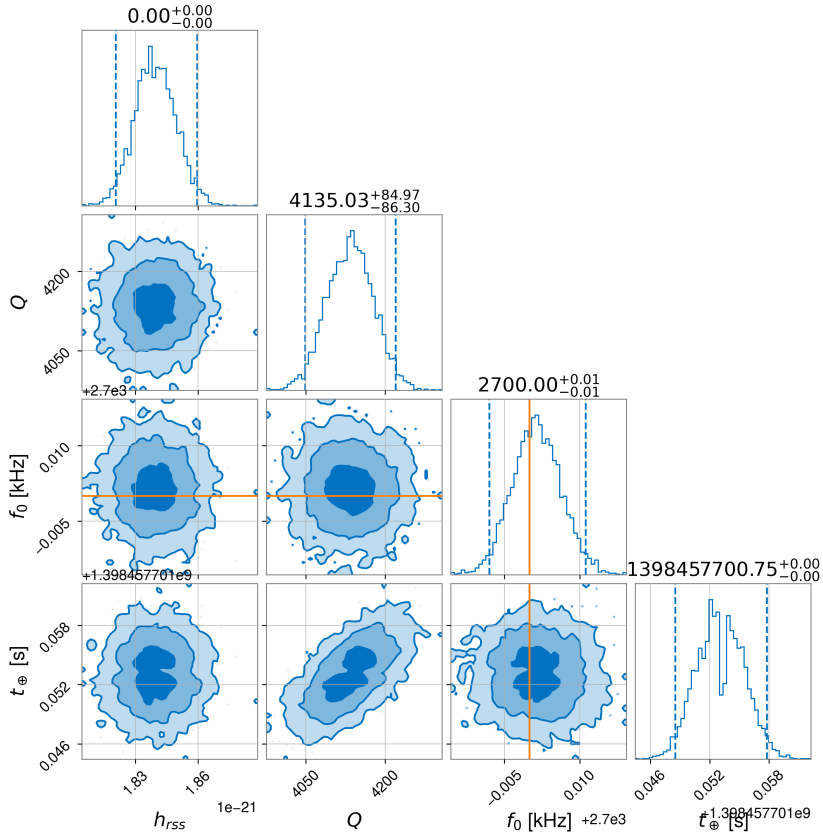

Figure 4.4: Corner plot for the sine-Gaussian sampler with a damped sinusoid injected on the strain data. The orange lines signify the frequency of the injected damped sinusoid.

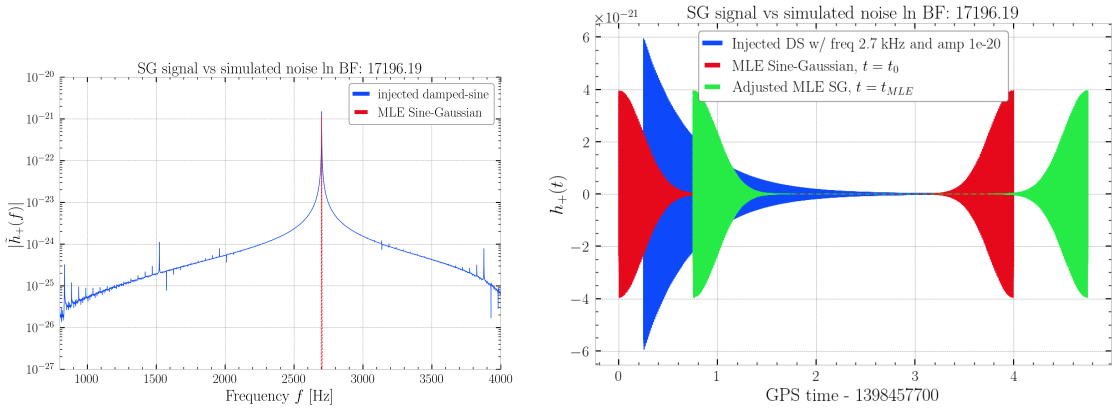

Figure 4.5: Left: The frequency-domain representation of the injected damped sinusoid with $h_0 = 10^{-20}$ together with the maximum likelihood estimate (MLE) sine-Gaussian waveform. Right: The equivalent time-domain representation.

To check which damped sinusoid injections might be mistaken for a sine-Gaussian, we collect the signal evidences for the sine-Gaussian sampler $\ln Z_{SG}$ and calculate the Bayes factor $\ln BF_{DS/SG}$. We plot this value against the injected signal amplitude in Figure 4.6. Recall that when $\ln BF_{DS/SG} > 0$ the damped-sine model is preferred over the sine-Gaussian model. This positive log Bayes factor trend starts to occur at the injection run with $h_0 = 3.455 \times 10^{-22}$ with its corresponding $\ln BF_{DS/SG} = 1.843$. [3] However, it is in the next louder injection with $h_0 = 4.124 \times 10^{-22}$ that the log Bayes factor crosses the threshold for strong evidence, with $\ln BF_{DS/SG} = 9.62$.

Recall that the damped sine sampler's threshold $h_{0,t}$ from Chapter 3 occurs at the injection run with $h_0 = 3.45 \times 10^{-22}$, so we guess that there is a difference in the samplers' signal-to-noise $\ln BF_{S/N}$. Indeed, this difference is shown in Figure 4.7. The same trend is observed for both samplers but with higher $\ln BF_{S/N}$ values for the damped sine sampler, indicating the preference for the damped sine model over the sine-Gaussian model. Since both of their sampler's $\ln BF_{S/N}$ values become positive, this implies that signals can be easily mistaken for one model over another. However since the damped sinusoid's $\ln BF_{S/N}$ values are higher than the sine-Gaussian, the preference is given to the damped sinusoid (which makes sense as we know there is a damped sinusoid in the data). This further solidifies the argument that the signal-vs-signal Bayes factor cannot be used on its own; we need to use other statistical information on the nature of the source that

[3] It is true that below this amplitude there are injection runs with a positive value of $\ln BF_{DS/SG}$, but the magnitude of their log Bayes factor ranges from $0.001 - 0.01$, so the chosen model is indeterminate.

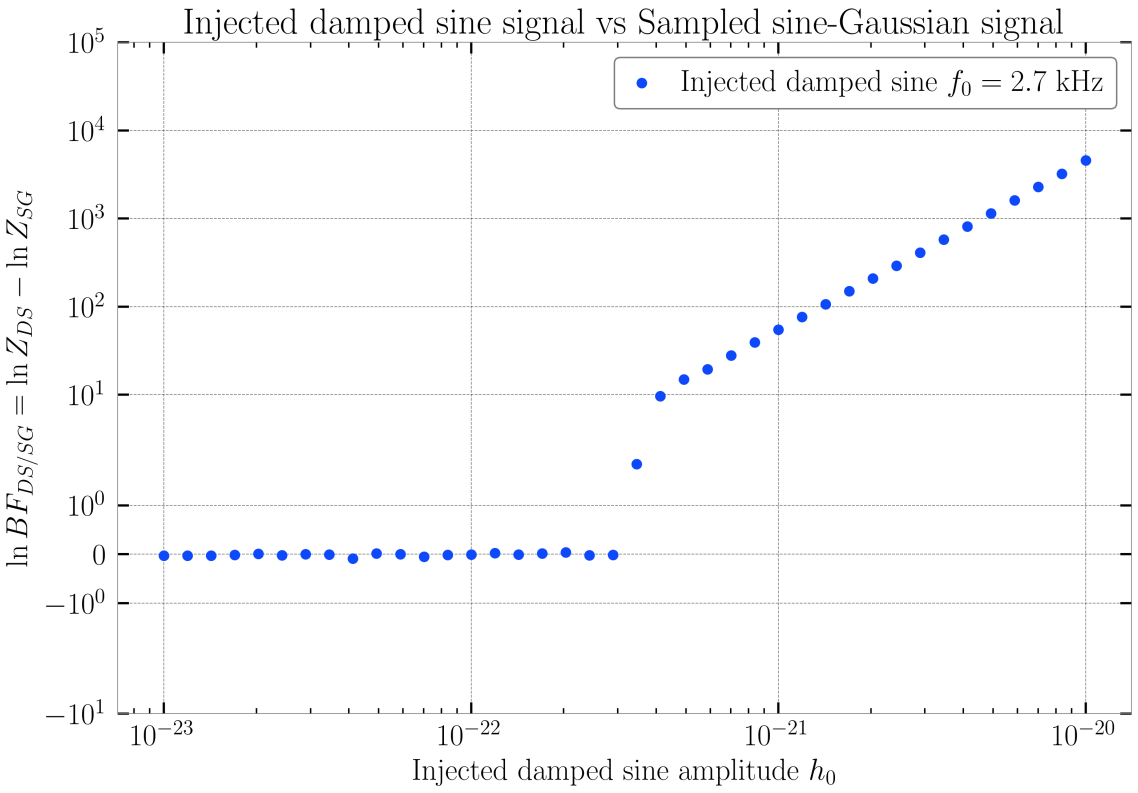

Figure 4.6: Damped sine vs. sine-Gaussian log Bayes factor $\ln BF_{DS/SG}$ as a function of the damped sinusoid signal amplitude injected onto simulated Gaussian strain data.

would compliment the results of the model comparison study.

To test the consistency of the sampler, we also inject 40 sine-Gaussians with different root-sum-square amplitudes $h_{rss} \in [10^{-23}, 10^{-20}]$, then using our pipeline we sample a damped sinusoid for each injection run (basically the inverse of the pipeline where we injected a damped sine then sampled a sine-Gaussian) and collect the posteriors and signal and noise evidences. The plot of signal-to-noise $\ln BF_{S/N}$ of the damped sinusoid samplers as a function of the injected sine-Gaussian h_{rss} is shown in Figure 4.8. Just like the previous result, we see that the sampler prefers the existence of the damped sinusoid from an injected amplitude of $h_{rss} = 3.45 \times 10^{-22}$. When we have a loud signal, the pipeline can determine if we can recover it from the data. But running several waveform samplers is important if we want to determine the nature of a burst GW source as we can compare their results using the signal-vs-signal Bayes factor.

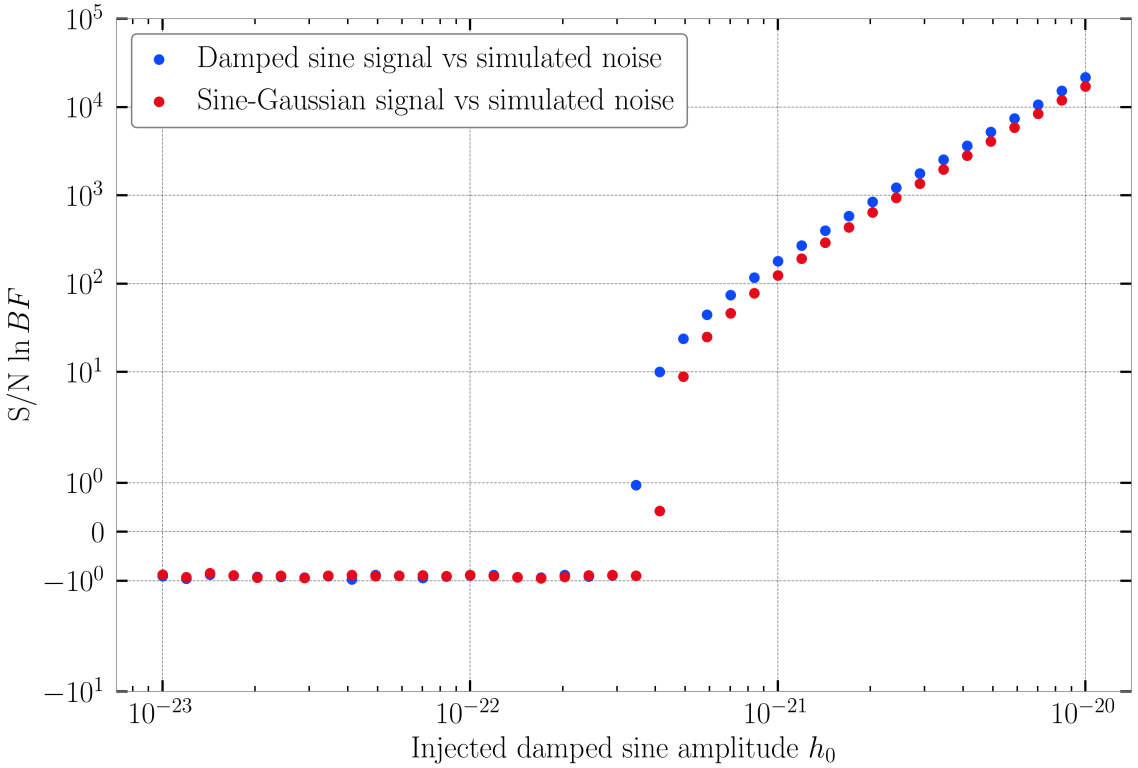

Figure 4.7: Comparison of the signal-to-noise $\ln BF_{S/N}$ from the damped sinusoid and sine-Gaussian samplers when there is a damped sinusoid injected onto simulated Gaussian strain data.

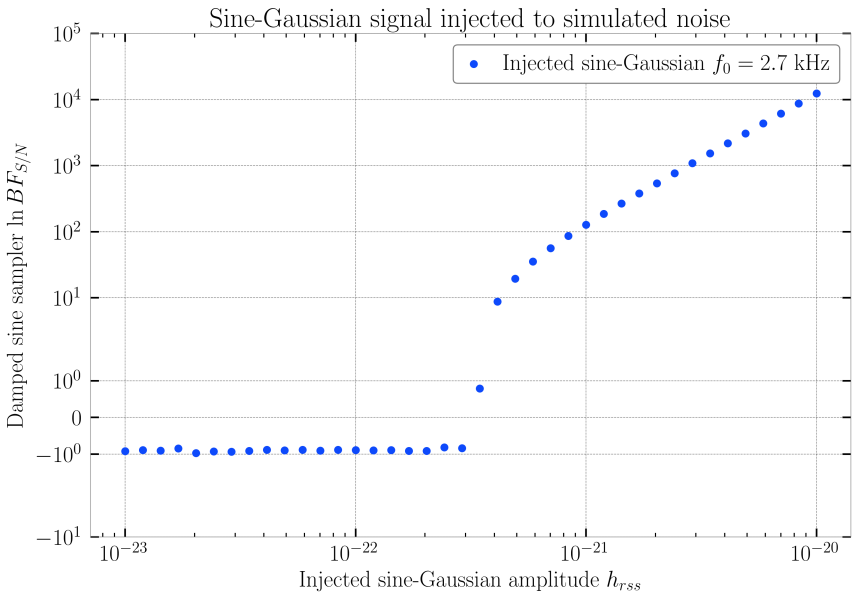

Figure 4.8: The signal-to-noiseln $BF_{S/N}$ of a damped sinusoid sampler as a function of the sine-Gaussian peak amplitude h_{rss} injected onto simulated Gaussian strain data.

4.3 Discussion and Conclusions

In this chapter, we tested the pipeline's ability to distinguish between different classes of GW burst signals. We have shown two cases: a case where the waveforms are deemed different by the sampler (damped sine vs. power-law) and another where the waveforms can look the same for the sampler (damped sine vs. sine-Gaussian). Although the analysis here only involves injections to simulated Gaussian data where we know the signal's true model *a priori*, we can use the signal vs. signal Bayes factor to quantify which is the preferred signal (especially when we perform unmodeled searches). We also confirm the limitation of the Bayes factor where it cannot decisively choose between two signal models when its magnitude is low. This is confirmed by the negative log Bayes factors with magnitudes near zero, where we cannot immediately conclude that the cosmic string cusp or sine-Gaussian model is preferred as they do not exist in the data *a priori*.

Furthermore, the results involving the sine-Gaussian reveal the generic nature of this waveform. It implies that when we only sample data using a sine-Gaussian, we can always recover a sine-Gaussian signal. To properly use the pipeline, another waveform sampler must be considered. This emphasizes the importance of testing several models as the Bayes factor does not provide the complete picture. Just like in [38], we can re-analyze confirmed CBC detections using sine-Gaussian wavelets for example. We can further extend the capabilities of our pipeline by considering combinations of models in what is known as a trans-dimensional MCMC algorithm, something that was proposed in the `Bayeswave` algorithm [21].

Conclusions and Recommendations

In this thesis, we present a Bayesian inference pipeline that compares models using the Bayes factor as a figure-of-merit. The Bayes factor compares the evidences (or marginalized likelihoods) of two models, providing insight into which model is more supported by the data. Our analyses involve several steps: (1) injecting waveforms onto both simulated and real data, (2) sampling different waveforms from this data, and (3) obtaining the posterior distributions and evidences. The flexibility of the pipeline allows it to accommodate a variety of waveforms, making it adaptable to both detection (signal versus noise) [4] and comparison scenarios (signal versus signal).

We applied our pipeline to a closed-box analysis of the recent Vela pulsar glitch, where we placed a limit on the amplitude of a damped sinusoid waveform generated by the pulsar glitch's excitation of the neutron star f-mode. This amplitude limit is derived from the corresponding value of the signal-to-noise Bayes factor, where a threshold of $\ln BF_{S/N} = 8$ indicates a strong preference for a model in which the signal is present, $s(t) = n(t) + h(t)$. This limit on the amplitude can also be translated into a limit on the energy released. If the Vela pulsar glitch of 2024 April 29 released energy — primarily in the form of gravitational waves — above the threshold, then a signal could be detectable in the GW strain data. This energy threshold will be computed from the damped sine amplitude's posterior distribution and considered for future work.

Additionally, we tested the pipeline's capability to compare disparate signal models. The sign of the logarithm of the Bayes factor calculated between two signal models indicates the preferred model, while its magnitude tells how much it prefers one model over the other. From our analysis the generic nature of sine-Gaussian is observed, and why it is the preferred basis for phenomenological

[4] In contrast to an unmodeled search, we assume the signal is well-modeled.

modelling of burst-type GW. We also conclude that while the Bayes factor provides valuable insights into burst GW model selection, it alone does not provide a complete picture of the burst signal's origin. It does, however, remain a useful tool for determining which waveform model better fits a confirmed detection.

In summary, the Bayesian inference pipeline developed in this thesis offers a versatile framework for gravitational wave data analysis, capable of both detection and model comparison tasks. While the Bayes factor provides a robust measure of model preference, further work is needed to refine our understanding of the physical implications behind these preferences, particularly with respect to energy release and source characterization in gravitational wave events. Future research will focus on applying this framework to more complicated models and extending the analysis to include more complex waveform models (not only phenomenological ones) as well as transdimensional analysis.

Appendix A

Burst GW inference pipeline in python

In this appendix, we show an example `python` code for our Bayesian inference pipeline. Specifically, we inject a damped sinusoid in this code to simulated Gaussian data; hence we need to fetch the design sensitivity ASD from [13]. In injecting and sampling multiple waveforms, it is advised to use multiprocessing methods such as HTCondor. [1]

```
01 | import numpy as np
02 | import matplotlib.pyplot as plt
03 | import pandas as pd
04 | import bilby
05 | from bilby.gw.detector.psd import PowerSpectralDensity
06 | from gwpy.frequencyseries import FrequencySeries
07 | from scipy.signal.windows import tukey
08 | import sys
09 | import json
10 | import argparse
11 |
12 | # parser arguments for reading txt files of parameters and
       assigning
13 | # job numbers in HTCondor
14 |
15 | parser = argparse.ArgumentParser(description="Inference on f-
       mode damped sinusoid parameters (shape parameters only).",)
16 | parser.add_argument("parameter_file", type=str, help='Filename
       of the parameter list. List must be three-column format for `
       amplitude`, `damping_time`, and `frequency` respectively')
17 | parser.add_argument("jobid", type=str, help='Job ID to be used
       for condor-dag.')
18 | parsed_args = parser.parse_args()
19 | jobid = int(parsed_args.jobid)
20 |
21 | ################## Functions ############################
22 | # define the half-Tukey window
23 | def half_tukey(N, alpha=0.5):
24 |     full = tukey(N, alpha)
25 |     half_tukey = np.ones(N)
```

[1] https://htcondor.readthedocs.io/en/latest/

```
26 |      half_tukey[:N//2] = full[:N//2]
27 |
28 |      return half_tukey
29 |
30 |  # define the time-domain damped sinusoid model
31 |  def fmode_strain(time, amplitude, damping_time, frequency,
       cosincl, t0, phase):
32 |      r"""
33 |      Damped sinusoid waveform (elliptically polarized, Tukey
       windowed).
34 |      Accepts waveform parameters directly.
35 |
36 |      .. math::
37 |          h_+(t) = \dfrac{1}{2} (1+(\cos \iota)^2) A \exp (- (t-
       t_0)/\tau) \sin(2 \pi f_0 (t-t_0))
38 |          h_\times(t) = \cos(\iota) A \exp (- (t-t_0)/\tau) \sin(2
        \pi f_0 (t-t_0))
39 |
40 |      Parameters
41 |      ----------
42 |      time: array-like
43 |          The times at which to evaluate the model. This is
       required for all
44 |          time-domain models.
45 |      amplitude: float
46 |          Waveform amplitude (dimensionless)
47 |      damping_time: float
48 |          Scaling factor for the damping exponential term (in
       seconds)
49 |      frequency: float
50 |          Characteristic (constant) frequency (in kHz)
51 |      cosincl: float
52 |          (cosine of the) polarization angle between h_+ and h_x
53 |      phase: float
54 |          Phase offset due to start time (in rad)
55 |
56 |      Returns
57 |      -------
58 |      dict:
59 |          A dictionary containing "plus" and "cross" entries.
60 |
61 |      """
62 |      incl_term = 0.5*(1+cosincl**2)
63 |      plus = np.zeros(len(time))
64 |      tidx = (time >= t0)
65 |      Nidx = len(time[tidx])
66 |      alpha = 1/(frequency*1000) # tapering fraction for Tukey
       window
67 |      window = half_tukey(Nidx, alpha)
68 |      plus[tidx] = (
69 |          window*incl_term*amplitude # amplitude: h0, not A = h0
       exp(-t0/tau)
70 |          * np.exp(-(time[tidx] - t0) / damping_time)
71 |          * np.cos((2 * np.pi * (frequency*1000) * (time[tidx])) +
        phase)
72 |      )
73 |      cross = np.zeros(len(time))
74 |      cross[tidx] = (
```

```
75 |          window*cosincl*amplitude
76 |          * np.exp(-(time[tidx] - t0) / damping_time)
77 |          * np.sin((2 * np.pi * (frequency*1000) * (time[tidx])) +
       phase)
78 |      )
79 |
80 |
81 |      return {"plus": plus, "cross": cross}
82 |
83 |  # define the function for reading the parameters from txt files
84 |  def read_params(parameter_file, jobid):
85 |      with open(parameter_file, "r") as f:
86 |          lines = f.readlines()
87 |      line = lines[int(jobid)+1].strip() # +1 to skip header
88 |      amp, tau, nu = list(map(float, line.split()))
89 |      return amp, tau, nu
90 |
91 |  print("Starting ...")
92 |
93 |  ###################### Constants ##########################
94 |  # set up duration and sampling frequency
95 |  duration = 4
96 |  sampling_frequency = 8192
97 |  ts_duration = 132 # Full TimeSeries duration
98 |
99 |  # parameters from txt file
100 |  amp, tau, nu = read_params(parsed_args.parameter_file, jobid)
101 |
102 |  # set up random seed number
103 |  bilby.core.utils.random.seed(190521)
104 |
105 |  # set up known source parameters
106 |  trigger_time = 1398458982 # GPS Time [s] from Matt Pitkin
107 |  OSW_start_time = 1398457700 # fake OSW start
108 |  OSW_stop_time = OSW_start_time + duration
109 |
110 |  # extrinsic params
111 |  phase = 0.0
112 |  ra = np.deg2rad(128.83606354)
113 |  dec = np.deg2rad(-45.17643181)
114 |  cosincl = np.cos(np.deg2rad(63.30)) # 63.30 or 116.70 deg, from
       Patrick Sutton
115 |  psi = np.deg2rad(130.629) # deg, from Patrick Sutton
116 |
117 |  # other parameters
118 |  roll_off = 0.4
119 |  win_alpha = 2 * roll_off / duration # for PSD/TimeSeries Tukey
       window
120 |  min_frequency = 128.0
121 |  max_frequency = 8192.0
122 |
123 |  # create dictionary for injection parameters
124 |  injection_parameters = dict(
125 |      amplitude=amp,
126 |      damping_time=tau,
127 |      frequency=nu,
128 |      t0=OSW_start_time+0.25, # time at which the waveform's
       amplitude is max
```

```
129 |       cosincl=cosincl,
130 |       ra=ra,
131 |       dec=dec,
132 |       geocent_time=OSW_start_time+0.25,
133 |       psi=psi,
134 |       phase=phase,
135 |   )
136 |
137 |   # set up bilby output
138 |   outdir = f"simdata-oneamp-DSsampler/freq_{nu}"
139 |   label = f"DSsampler-freq_{nu}"
140 |   logger = bilby.core.utils.logger
141 |   bilby.core.utils.check_directory_exists_and_if_not_mkdir(outdir)
142 |   bilby.core.utils.setup_logger(outdir=outdir, label=label)
143 |
144 |   ############### WaveformGenerator ##########################
145 |   waveform = bilby.gw.waveform_generator.WaveformGenerator(
146 |       duration=duration,
147 |       sampling_frequency=sampling_frequency,
148 |       time_domain_source_model=fmode_strain,
149 |       parameter_conversion=lambda parameters: (parameters, list())
          ,
150 |       start_time=OSW_start_time, # define the 4s waveform from the
          start of the OSW
151 |   )
152 |
153 |   # WaveformGenerator check if nonzero
154 |   td_plus = waveform.time_domain_strain(injection_parameters)['
          plus']
155 |   td_cross = waveform.time_domain_strain(injection_parameters)['
          cross']
156 |   hrss = np.sqrt(np.sum((np.abs(td_plus)**2 + np.abs(td_cross)**2)
           * 1/sampling_frequency))
157 |   logger.info(f"injection hrss = {hrss:.6e}")
158 |   # with open('simdata-amp-hrss.txt', 'a') as file:
159 |   #     # file.write(f"# injected_amplitude calculated_hrss \n")
160 |   #     file.write(f"{amp:.6e} {nu:.6e} {hrss:.6e}\n")
161 |
162 |   ########################## ASD files ##########################
163 |   LIGO_ASD = FrequencySeries.read('aLIGO18_ASD.txt', unpack=True)
164 |   VIRGO_ASD = FrequencySeries.read('aVirgo18_ASD.txt', unpack=True
          )
165 |   # must contain frequency (col. 0) and ASD (col. 1)
166 |
167 |   ############## Inteferometer object construction #############
168 |   ifo_names = ["H1", "L1", "V1"]
169 |   ifos = bilby.gw.detector.InterferometerList(ifo_names)
170 |
171 |   # Assign ASDs
172 |   ifos[0].power_spectral_density = PowerSpectralDensity(
          frequency_array=LIGO_ASD.frequencies.value,   asd_array=
          LIGO_ASD.value)
173 |
174 |   ifos[1].power_spectral_density = PowerSpectralDensity(
          frequency_array=LIGO_ASD.frequencies.value, asd_array=
          LIGO_ASD.value)
175 |
176 |   ifos[2].power_spectral_density = PowerSpectralDensity(
```

```
        frequency_array=VIRGO_ASD.frequencies.value, asd_array=
        VIRGO_ASD.value)
177 |
178 |  # Limit the frequency range for sampling and construct fake OSW
        strain data
179 |  for ifo in ifos:
180 |      # based on the PSD array data
181 |      ifo.minimum_frequency=min_frequency
182 |      ifo.maximum_frequency=max_frequency
183 |      ifo.set_strain_data_from_power_spectral_density(
        sampling_frequency=sampling_frequency, duration=duration,
        start_time=OSW_start_time)
184 |
185 |  mf_snr = np.zeros(len(ifos))
186 |  time_delay = []
187 |
188 |  # Inject signal and calculate matched-filter SNR and geocentric
        time delays for each interferometer
189 |  for i, ifo in enumerate(ifos):
190 |      ifo.inject_signal_from_waveform_generator(
191 |          waveform_generator=waveform,
192 |          parameters=injection_parameters,
193 |          # raise_error=False,
194 |      )
195 |      time_delay.append(ifo.time_delay_from_geocenter(
        injection_parameters['ra'], injection_parameters['dec'],
        injection_parameters["geocent_time"]))
196 |      mf_snr[i] = np.abs(ifo.matched_filter_snr(ifo.
        get_detector_response(waveform.frequency_domain_strain(
        injection_parameters), injection_parameters)))
197 |
198 |  net_snr = np.sqrt(np.sum(mf_snr**2))
199 |  logger.info(f"H1-L1-V1 network SNR: {net_snr:.3f}")
200 |  logger.info(f"Saving frequency-domain plots for noise ASD and {
        duration}-second strain ASD at {outdir}")
201 |  ifos.plot_data(outdir=outdir, label=label)
202 |
203 |  ################ Bayesian inference part
        ##########################
204 |  #  create the priors
205 |  prior = bilby.prior.PriorDict()
206 |  for key in ["psi", "ra", "dec", "cosincl", "geocent_time", "
        phase"]:
207 |      prior[key] = injection_parameters[key] # sets DeltaFunction
        priors
208 |
209 |  prior["amplitude"] = bilby.core.prior.LogUniform(1e-24, 1e-19, r
        "$h_0$")
210 |  prior["damping_time"] = bilby.core.prior.Uniform(0.087, 10, r"$\
        tau$", unit=r"s")
211 |  prior["frequency"] = bilby.core.prior.Uniform(0.8, 4.0, r"$f_0$"
        , unit="kHz")
212 |  # prior["cosincl"] = bilby.core.prior.Uniform(0, 1, r"$\cos \
        iota$")
213 |  prior["t0"] = bilby.core.prior.Uniform(OSW_start_time,
        OSW_stop_time, r"$t_\oplus$", unit="s")
214 |  # prior["phase"] = bilby.core.prior.Uniform(-np.pi / 2, np.pi /
        2, r"$\phi$")
```

```
215 |
216 |  # create the likelihood
217 |  likelihood = bilby.gw.likelihood.GravitationalWaveTransient(ifos
        , waveform)
218 |
219 |  # testing the likelihood
220 |  prior_sample = prior.sample()
221 |  likelihood.parameters=prior_sample
222 |  logger.info(f"example log-likelihood value calculated from prior
         sample: {likelihood.log_likelihood_ratio():.6e}")
223 |
224 |  # launch sampler
225 |  result = bilby.core.sampler.run_sampler(
226 |      likelihood,
227 |      prior,
228 |      sampler="dynesty",
229 |      sample='acceptance-walk',
230 |      nlive=1000,
231 |      walks=50,
232 |      naccept=60,
233 |      dlogz=0.1,
234 |      npool=16,
235 |      outdir=outdir,
236 |      label=label,
237 |      injection_parameters=injection_parameters,
238 |      # rstate=rstate,
239 |      # conversion_function=lambda parameters:
240 |  )
241 |
242 |  # Make corner plot
243 |  sampled_params = ["amplitude","damping_time", "frequency", "t0"]
244 |  truth_params = {key: injection_parameters[key] for key in
        sampled_params}
245 |  logger.info(f"Saving corner plots for {' '.join([str(param) for
        param in sampled_params])} on {outdir}")
246 |  result.plot_corner(quantiles=(0.025,0.975), title_fmt='.2e') #
        2-sigma
247 |
248 |  # log the hrss, SNR, and ln BF values
249 |  with open('simDSoneamp-nu-hrss-net_snr-lnBF.txt', 'a') as file:
250 |      file.write(f"{nu} {hrss} {np.abs(net_snr)} {result.
        log_bayes_factor}\n")
251 |
252 |  # list MLE of parameters
253 |  samples = result.posterior
254 |  param_samples = {param: samples[param].values for param in
        sampled_params}
255 |  df = pd.DataFrame(param_samples)
256 |  ml_params = df.iloc[samples.log_likelihood.idxmax()].to_dict()
257 |  logger.info(f"Injected parameters: {injection_parameters}")
258 |  logger.info(f"Maximum Likelihood Estimates of parameters: {
        ml_params}")
259 |  # with open('simDSoneamp-mle-params.txt', 'a') as file:
260 |      # file.write(f"{nu} {ml_params} \n")
261 |
262 |
263 |  # calculate Mean Absolute Error (MAE)
264 |  mle_dict = injection_parameters.copy()
```

```
mle_dict.update(ml_params)
mle_plus = waveform.time_domain_strain(mle_dict)['plus']
mle_cross = waveform.time_domain_strain(mle_dict)['cross']
MAE_plus = np.sum(np.abs(td_plus - mle_plus))/len(td_plus)
MAE_cross = np.sum(np.abs(td_cross - mle_cross))/len(td_cross)
logger.info(f"Mean Absolute Error (MAE) for hplus = {MAE_plus},
    hcross = {MAE_cross}")
with open('simDSoneamp-mean-abs-error.txt', 'a') as file:
    file.write(f"{nu:.6e} {MAE_plus:.6e} {MAE_cross:.6e}\n")

# collect the time-series strain data (injected and MLE) then
    save
waveform_data = np.column_stack((waveform.time_array, td_plus,
    td_cross, mle_plus, mle_cross))
np.savetxt(f"{outdir}/freq-{nu}-waveforms.txt", waveform_data)
```

Bibliography

[1] A. Einstein, Approximative Integration of the Field Equations of Gravitation, *Sitzungsber. Preuss. Akad. Wiss. Berlin (Math. Phys.)* **1916**, 688 (1916).

[2] M. Maggiore, *Gravitational Waves: Volume 1: Theory and experiments* (OUP Oxford, 2007).

[3] A. Macquet, Ph.D. thesis, Université Côte d'Azur (2021).

[4] J. Creighton and W. Anderson, *Gravitational-Wave Physics and Astronomy* (Wiley, 2011).

[5] B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, C. Affeldt, M. Agathos, et al., A guide to LIGO–Virgo detector noise and extraction of transient gravitational-wave signals, *Classical Quant. Grav.* **37**, 055002 (2020).

[6] J. Aasi, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari, et al., Advanced LIGO, *Classical Quant. Grav.* **32**, 074001 (2015).

[7] Abbott, Benjamin P and Abbott, Rich and Abbott, Thomas D and others (LIGO Scientific Collaboration and Virgo Collaboration), Observation of Gravitational Waves from a Binary Black Hole Merger, *Phys. Rev. Lett.* **116**, 061102 (2016).

[8] F. Acernese, M. Agathos, K. Agatsuma, D. Aisa, N. Allemandou, A. Allocca, J. Amarni, P. Astone, G. Balestri, G. Ballardin, et al., Advanced Virgo: a second-generation interferometric gravitational wave detector, *Classical Quant. Grav.* **32**, 024001 (2014).

[9] B. P. Abbott, R. Abbott, T. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, V. B. Adya, et al., GW170817: obser-

vation of gravitational waves from a binary neutron star inspiral, *Phys. Rev. Lett.* **119**, 161101 (2017).

[10] B. P. Abbott et al. (LIGO Scientific, Virgo, Fermi GBM, INTEGRAL, IceCube, AstroSat Cadmium Zinc Telluride Imager Team, IPN, Insight-Hxmt, ANTARES, Swift, AGILE Team, 1M2H Team, Dark Energy Camera GW-EM, DES, DLT40, GRAWITA, Fermi-LAT, ATCA, ASKAP, Las Cumbres Observatory Group, OzGrav, DWF (Deeper Wider Faster Program), AST3, CAASTRO, VINROUGE, MASTER, J-GEM, GROWTH, JAGWAR, CaltechNRAO, TTU-NRAO, NuSTAR, Pan-STARRS, MAXI Team, TZAC Consortium, KU, Nordic Optical Telescope, ePESSTO, GROND, Texas Tech University, SALT Group, TOROS, BOOTES, MWA, CALET, IKI-GW Follow-up, H.E.S.S., LOFAR, LWA, HAWC, Pierre Auger, ALMA, Euro VLBI Team, Pi of Sky, Chandra Team at McGill University, DFN, ATLAS Telescopes, High Time Resolution Universe Survey, RIMAS, RATIR, SKA South Africa/MeerKAT), Multi-messenger Observations of a Binary Neutron Star Merger, *Astrophys. J. Lett.* **848**, L12 (2017), arXiv:1710.05833.

[11] K. Somiya et al., Detector configuration of KAGRA–the Japanese cryogenic gravitational-wave detector, *Classical Quant. Grav.* **29**, 124007 (2012).

[12] M. Arène, Ph.D. thesis, Université Paris Cité (2021).

[13] R. Abbott, B. Abbott et al., Prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo and KAGRA, *Living Rev. Relativ.* **21**, 3 (2018).

[14] D. M. Macleod, J. S. Areeda, S. B. Coughlin, T. J. Massinger, and A. L. Urban, GWpy: A Python package for gravitational-wave astrophysics, *SoftwareX* **13**, 100657 (2021), ISSN 2352-7110.

[15] C. J. Moore, R. H. Cole, and C. P. Berry, Gravitational-wave sensitivity curves, *Classical Quant. Grav.* **32** (2015).

[16] R. Abbott, T. Abbott, F. Acernese, K. Ackley, C. Adams, N. Adhikari, R. Adhikari, V. Adya, C. Affeldt, D. Agarwal, et al., GWTC-3: Compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run, *Phys. Rev. X* **13**, 041039 (2023).

[17] B. P. Abbott, R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al., GWTC-1: a gravitational-wave transient catalog of compact binary mergers observed by LIGO and Virgo during the first and second observing runs, *Phys. Rev. X* **9**, 031040 (2019).

[18] R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al., GWTC-2: compact binary coalescences observed by LIGO and Virgo during the first half of the third observing run, *Phys. Rev. X* **11**, 021053 (2021).

[19] P. J. Sutton, A rule of thumb for the detectability of gravitational-wave bursts, *arXiv preprint arXiv:1304.0210* (2013).

[20] J. Powell, S. E. Gossan, J. Logue, and I. S. Heng, Inferring the core-collapse supernova explosion mechanism with gravitational waves, *Phys. Rev. D.* **94**, 123012 (2016).

[21] N. J. Cornish and T. B. Littenberg, Bayeswave: Bayesian inference for gravitational wave bursts and instrument glitches, *Classical Quant. Grav.* **32**, 135012 (2015).

[22] D. Antonopoulou, B. Haskell, and C. M. Espinoza, Pulsar glitches: observations and physical interpretation, *Reports on Progress in Physics* **85**, 126901 (2022).

[23] J. McKee, G. Janssen, B. Stappers, A. Lyne, R. Caballero, L. Lentati, G. Desvignes, A. Jessner, C. Jordan, R. Karuppusamy, et al., A glitch in the millisecond pulsar J0613- 0200, *Mon. Not. R. Astron. Soc.* **461**, 2809 (2016).

[24] N. Andersson and K. D. Kokkotas, Towards gravitational wave asteroseismology, *Mon. Not. R. Astron. Soc.* **299**, 1059 (1998).

[25] W. C. Ho, D. Jones, N. Andersson, and C. M. Espinoza, Gravitational waves from transient neutron star f-mode oscillations, *Phys. Rev. D.* **101**, 103009 (2020).

[26] D. Lopez, S. Tiwari, M. Drago, D. Keitel, C. Lazzaro, and G. A. Prodi, Prospects for detecting and localizing short-duration transient gravitational

waves from glitching neutron stars without electromagnetic counterparts, *Phys. Rev. D.* **106**, 103037 (2022).

[27] M. Ball, R. Frey, and K. Merfeld, Prospects for neutron star parameter estimation using gravitational waves from f-modes associated with magnetar flares (2023), arXiv:2310.15315, https://arxiv.org/abs/2310.15315.

[28] C. Y. Ng and R. W. Romani, Fitting Pulsar Wind Tori. 2. Error Analysis and Applications, *Astrophys. J.* **673**, 411 (2008), arXiv:0710.4168.

[29] R. P. Mignani, S. Bagnulo, J. Dyks, G. L. Curto, and A. Slowikowska, The Optical Polarisation of the Vela Pulsar revisited, *Astronomy and Astrophysics* **467**, 1157 (2007), arXiv:astro-ph/0702307.

[30] M. Watson, J. P. Pye, M. Denby, J. Osborne, D. Barret, T. Boller, H. Brunner, M. Ceballos, R. Dellaceca, D. Fyfe, et al., The XMM-Newton serendipitous source catalogue, *Astronomische Nachrichten: Astronomical Notes* **324**, 89 (2003).

[31] B. Link, R. I. Epstein, and J. M. Lattimer, Pulsar constraints on neutron star structure and equation of state, *Phys. Rev. Lett.* **83**, 3362 (1999).

[32] A. Basu, B. Shaw, D. Antonopoulou, M. J. Keith, A. G. Lyne, M. B. Mickaliger, B. W. Stappers, P. Weltevrede, and C. A. Jordan, The Jodrell bank glitch catalogue: 106 new rotational glitches in 70 pulsars, *Mon. Not. R. Astron. Soc.* **510**, 4049 (2021), ISSN 0035-8711, arXiv:https://academic.oup.com/mnras/article-pdf/510/3/4049/42194660/stab3336.pdf.

[33] B. Haskell and A. Melatos, Models of pulsar glitches, *Int. J. Mod. Phys. D.* **24**, 1530008 (2015).

[34] J. Abadie, B. P. Abbott, R. Abbott, R. Adhikari, P. Ajith, B. Allen, G. Allen, E. Amador Ceron, R. S. Amin, S. B. Anderson, et al. (The LIGO Scientific Collaboration), Search for gravitational waves associated with the August 2006 timing glitch of the Vela pulsar, *Phys. Rev. D.* **83**, 042001 (2011).

[35] M. Pitkin, CWInPy: A Python package for inference with continuous gravitational-wave signals from pulsars, *Journal of Open Source Software* **7**, 4568 (2022).

[36] A. K. Divakarla, E. Thrane, P. D. Lasky, and B. F. Whiting, Memory effect or cosmic string? Classifying gravitational-wave bursts with Bayesian inference, *Phys. Rev. D.* **102**, 23010 (2020), ISSN 24700029, arXiv:1911.07998.

[37] R. Abbott, T. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al., Constraints on cosmic strings using data from the third advanced LIGO–Virgo observing run, *Phys. Rev. Lett.* **126**, 241102 (2021).

[38] J. C. Aurrekoetxea, C. Hoy, and M. Hannam, Revisiting the cosmic string origin of GW190521, 1–8 (2023), arXiv:2312.03860.

[39] LIGO Scientific Collaboration, Virgo Collaboration, and KAGRA Collaboration, LVK Algorithm Library - LALSuite, Free software (GPL) (2018).

[40] R. Abbott, T. Abbott, et al., All-Sky Search for Short Gravitational-Wave Bursts in the Third Advanced LIGO and Advanced Virgo Run, *Phys. Rev. D.* **104**, 102001 (2022).

[41] B. S. Sathyaprakash and B. F. Schutz, Physics, astrophysics and cosmology with gravitational waves, *Living Rev. Relativ.* **12** (2009), arXiv:0903.0338.

[42] E. Thrane and C. Talbot, An introduction to Bayesian inference in gravitational-wave astronomy: Parameter estimation, model selection, and hierarchical models, *Publ. Astron. Soc. Aust.* **36** (2019), ISSN 14486083, arXiv:1809.02293.

[43] S. Ghosh, X. Liu, J. Creighton, I. M. n. Hernandez, W. Kastaun, and G. Pratten, Rapid model comparison of equations of state from gravitational wave observation of binary neutron star coalescences, *Phys. Rev. D.* **104**, 083003 (2021).

[44] G. Ashton et al., BILBY: A user-friendly Bayesian inference library for gravitational-wave astronomy, *Astrophys. J. Suppl.* **241**, 27 (2019), arXiv:1811.02042.

[45] D. Sivia and J. Skilling, *Data analysis: a Bayesian tutorial* (OUP Oxford, 2006).

[46] N. Christensen and R. Meyer, Parameter estimation with gravitational waves, *Rev. Mod. Phys.* **94**, 025001 (2022).

[47] S. Chib and E. Greenberg, Understanding the Metropolis-Hastings algorithm, *Am. Stat.* **49**, 327 (1995).

[48] J. S. Speagle, dynesty: a dynamic nested sampling package for estimating bayesian posteriors and evidences, *Mon. Not. R. Astron. Soc.* **493**, 3132 (2020).

[49] N. Kunert, S. Antier, V. Nedora, M. Bulla, P. T. Pang, S. Anand, M. Coughlin, I. Tews, J. Barnes, T. Hussenot-Desenonges, et al., Bayesian model selection for GRB 211211A through multiwavelength analyses, *Mon. Not. R. Astron. Soc.* **527**, 3900 (2024), ISSN 13652966.

[50] H. Jeffreys, *The theory of probability* (OuP Oxford, 1998).

[51] N. Lopac, J. Lerga, and E. Cuoco, Gravitational-wave burst signals denoising based on the adaptive modification of the intersection of confidence intervals rule, *Sensors* **20**, 6920 (2020).

[52] R. Rutledge, The astronomer's telegram: A web-based short-notice publication system for the professional astronomical community (1998), https://arxiv.org/abs/astro-ph/9802256.

[53] E. Zubieta, S. A. Furlan, S. del Palacio, F. Garcia, G. Gancio, C. Lousto, J. Combi, and the PuMA Collaboration, Detection of a new giant glitch in the Vela Pulsar observed from the Argentine Institute of Radio astronomy (2024), https://www.astronomerstelegram.org/?read=16608.

[54] I. M. Romero-Shaw, C. Talbot, S. Biscoveanu, V. D'emilio, G. Ashton, C. Berry, S. Coughlin, S. Galaudage, C. Hoy, M. Hübner, et al., Bayesian inference for compact binary coalescences with bilby: validation and application to the first LIGO–Virgo gravitational-wave transient catalogue, *Mon. Not. R. Astron. Soc.* **499**, 3295 (2020).

[55] T. B. Littenberg and N. J. Cornish, Bayesian inference for spectral estimation of gravitational wave detector noise, *Phys. Rev. D.* **91**, 084034 (2015).

[56] P. Ajith, S. Babak, Y. Chen, M. Hewitson, B. Krishnan, J. Whelan, B. Brueggmann, P. Diener, J. Gonzalez, M. Hannam, et al., A phenomenological template family for black-hole coalescence waveforms, *Classical Quant. Grav.* **24**, S689 (2007).

[57] P. J. Easter, S. Ghonge, P. D. Lasky, A. R. Casey, J. A. Clark, F. Hernandez Vivanco, and K. Chatziioannou, Detection and parameter estimation of binary neutron star merger remnants, *Phys. Rev. D.* **102**, 43011 (2020), ISSN 24700029, arXiv:2006.04396.

[58] J. S. Key and N. J. Cornish, Characterizing the gravitational wave signature from cosmic string cusps, *Phys. Rev. D.* **79**, 043014 (2009).

Acknowledgments

I would like to acknowledge everyone who helped make this thesis possible:

To my supervisor at OCA, **Dr. Marie-Anne Bizouard**, for giving me the project and making sure I hit my targets; for pushing me to think harder about the problems at hand; for introducing me to the LVK Collaboration; and for tirelessly editing my codes and manuscript. I really enjoyed working with you at Artemis and I hope for more fruitful research collaborations.

To my supervisor at the University of Belgrade, **Prof. Dr. Anđelka Kovačević**, for providing me with means to think in a Bayesian way; for letting me work on Nested Sampling which really jumpstarted my love for Bayesian Statistics. I really appreciate your comments and suggestions and patience.

To my Artemis laboratory colleagues and collaborators, **Dr. Nelson Christensen, Dr. Renate Meyer, Dr. Guillaume Boileau, Dr. Shanika Galaudage, (soon-to-be Dr.) Weizmann Kiendrebeogo, Hugo Einsle**, for extending your help and answering my questions on gravitational waves, lab life, and more. Thank you for your patience on answering even the smallest questions.

To the people of **LIGO-Virgo-KAGRA Collaboration**, particularly the developers and users of `bilby` and the GWOSC data providers, for providing the strain data used in this thesis, for also answering all of my questions and for catching bugs and updating me when things are down. It is really inspiring to work with a large-scale multi-continental collaboration and I hope to contribute more to its success.

To my colleagues in MASS, and everyone I've met over the past two years, for the scientific talks and supporting one another through the bureaucracies.

To my family, friends, loved ones, colleagues in the Philippines, and God, the reasons and real motivators for my journey here in Europe.

This one's for all of you!

MIB Rivera acknowledges support through an Erasmus Mundus Joint Master (EMJM) scholarship funded by the European Union in the framework of the Erasmus+, Erasmus Mundus Joint Master in Astrophysics and Space Science – MASS. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or granting authority European Education and Culture Executive Agency (EACEA). Neither the European Union nor the granting authority can be held responsible for them.