DETECTING AND CHARACTERIZING GRAVITATIONAL WAVES WITH MINIMAL ASSUMPTIONS

by

Margaret Ann Millhouse

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

Doctor of Philosophy

in

Physics

MONTANA STATE UNIVERSITY
Bozeman, Montana

April 2018

ACKNOWLEDGEMENTS

Thank you first and foremost to my advisor, Neil Cornish. Thank you to the rest of my committee as well for their help along the way, and special thanks Nico Yunes and Dana Longcope for their comments on this Thesis.

Thank you to all my LIGO collaborators and in particular Tyson Littenberg and Jonah Kanner without whom much of this work would not be possible.

I would also like to extend my gratitude to everyone at MSU past and present who helped me get through my time here. This especially includes Margaret Jarrett for having answers to all my grad school questions, and the students who came before me in the Gravity Group for having answers to all my physics questions.

And finally thank you to my family and friends who have supported me the whole way through, especially Alex.

TABLE OF CONTENTS

TABLE OF CONTENTS – CONTINUED

TABLE OF CONTENTS – CONTINUED

TABLE OF CONTENTS – CONTINUED

LIST OF TABLES

LIST OF FIGURES

Figure Page

LIST OF FIGURES – CONTINUED

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

LIST OF FIGURES – CONTINUED

Figure Page

LIST OF FIGURES – CONTINUED

Figure Page

ABSTRACT

After many years of preparation and anticipation, we are finally in the era of routine gravitational-wave detection. All of the detected signals so far have come from merging compact objects– either black holes or neutron stars. These are signals for which we have very good waveform models, but there still exist other more poorly modeled sources as well as the possibility of completely new gravitational-wave sources. Because of this, it is important to have the ability to confidently detect gravitational-waves from a wide variety of sources. In this Thesis I will describe one particular algorithm used to detect and characterize gravitational-wave signals using Bayesian inference techniques, and minimal assumptions on the source of the gravitational wave. I will report on the methods and results of the implementation of this search in the first two observing runs of advanced LIGO. I will also discuss developments to this algorithm to improve waveform reconstruction, and target certain signals without using full waveform templates.

1

INTRODUCTION

The past few years have been a revolutionary period for gravitational-wave physics. After completing a series of upgrades, the Advanced Laser Interferometer Gravitational-wave Observatory [7] began its first Observing run in September of 2015. On September 14, 2015 the twin interferometers in Hanford, Washington and Livingston, Louisiana observed for the first ever gravitational waves from merging stellar mass binary black holes [13]. The remainder of the first Observing run, and the second Observing run in which LIGO was joined by Italy's Virgo detector [22], resulted in several more binary black hole detections, and a binary neutron star detection [12, 18–21].

Stellar mass binary black holes and neutron stars are only two examples of the gravitational-wave sources predicted. Other predicted sources range from a stochastic background of gravitational waves from black hole binaries, to continuous gravitational waves from rotating asymmetric neutron stars, to short bursts of gravitational-wave radiation from core-collapse supernovae, and more. Detecting these different signals requires a variety of gravitational-wave detectors, and different data analysis approaches. In this work, we will be focusing on LIGO and Virgo searches for short-duration gravitational-wave signals (also called bursts), which are signals that can last up to a few seconds. Some of the predicted sources that would be detectable in these searches include supernovae, cosmic string cusps, and stellar mass black hole mergers. And of course, we want to be ready to find signals from as-of-yet *unpredicted* gravitational-wave sources. Rarely have scientists opened up a new window into the universe and only seen what they were expecting to see– some

notable examples of this are the Cosmic Microwave Background, and Gamma Ray Bursts. When using gravitational-waves to probe the universe it is imperative that in addition to our modeled searches which look for specific gravitational waves from specific sources, we are also prepared for new and surprising discoveries.

This Thesis will present methods for and results from unmodeled gravitational-wave searches in Advanced LIGO's first two Observing runs, including our analysis of GW150914. We employ an algorithm that uses Bayesian inference techniques to robustly distinguish real gravitational-wave signals from instrumental noise, and reconstruct the gravitational waveform using minimal assumptions on the form that signal will take. We will further explore other methods of characterizing the signal without the use of full templates. Finally we discuss future prospects for improving our unmodeled searches and waveform reconstructions, including using targeted searches that lie somewhere between fully modeled, and fully unmodeled.

The rest of this thesis is as follows: In Chapter 2 we give an overview of gravitational waves in general relativity and the current state of gravitational-wave detectors. In Chapter 3 we introduce some of the data analysis techniques used in gravitational-wave astronomy with some examples. In Chapter 4 we introduce the *BayesWave* algorithm used to detect GW bursts. In Chapter 5 we investigate our algorithm's ability to confidently separate gravitational waves from noise artifacts. Chapter 5 is based off of [73] which was work done in collaboration with Tyson Littenberg and Jonah Kanner. To this work I contributed to the numerical experiments needed to validate our predictions. Chapter 6 describes how the *BayesWave* algorithm was incorporated into LIGO data analysis, and in this chapter as well as Chapter 7 give some of the results from the first two observing runs of advanced LIGO. This was work that I carried out within the LIGO Scientific Collaboration. Chapters 8 and 9 are based off of my own original work and examine

how we can improve the *BayesWave* algorithm to better reconstruct GW signals.

4

GRAVITATIONAL WAVES

The biggest scientific discovery this century was undoubtedly the first direct detection of gravitational waves (GWs) by the Laser Interferometer Gravitational-wave Observatory (LIGO) [13]. In this chapter we give an overview of of the formulation of GWs in general relativity, and discuss the current state of GW detectors.

General Relativity

The Theory of General Relativity (GR) was first proposed by Albert Einstein in 1915 [49]. This theory was revolutionary in that rather than treating gravity as a Newtonian force between massive objects it treats gravity as a curvature of spacetime. The curvature is influenced by massive objects, so a heavier object will create a more dramatic distortion of spacetime. In turn, the movement of objects in space is influenced by this curvature. An free-falling object will follow the shortest path, or the "geodesic", which in curved spacetime may not appear to be a straight line.

This relationship between the movement of matter and the curvature of spacetime can be represented by Einstein's field equations [5, 6, 50, 55]

$$R_{\mu\nu} - \frac{1}{2} R g_{\mu\nu} = 8\pi T_{\mu\nu} \tag{2.1}$$

where we are using geometric units so that $G = c = 1$. Here $g_{\mu\nu}$ is the metric of our space time, and $T_{\mu\nu}$ is the stress-energy tensor which encodes information about the matter in our space. $R_{\mu\nu}$ and R are the Ricci tensor and Ricci scalar respectively,

which are both derived from the Riemann curvature tensor

$$R^{\lambda}_{\rho\mu\nu} = \Gamma^{\lambda}_{\mu\rho,\nu} - \Gamma^{\lambda}_{\nu\rho,\mu} + \Gamma^{\lambda}_{\mu\alpha}\Gamma^{\alpha}_{\nu\rho} - \Gamma^{\lambda}_{\nu\alpha}\Gamma^{\alpha}_{\mu\rho} \tag{2.2}$$

with the commas representing partial derivatives, and $\Gamma^{\lambda}_{\mu\nu}$ being the Christoffel symbol

$$\Gamma^{\lambda}_{\mu\nu} = \frac{1}{2}g^{\lambda\alpha}(g_{\nu\alpha,\mu} + g_{\mu\alpha,\nu} - g_{\mu\nu,\alpha}). \tag{2.3}$$

Specifically, $R_{\mu\nu} = R^{\alpha}_{\mu\alpha\nu}$ and $R = R_{\mu\nu}g^{\mu\nu}$. Note that we have employed Einstein summation notation:

$$A^{\alpha}B_{\alpha} = \sum_{i}^{N} A^{i}B_{i}. \tag{2.4}$$

Once Eq. 2.1 is written down, we just need to solve it in order to determine the motion of massive objects in space. Unfortunately Eq. 2.1 is actually ten coupled nonlinear differential equations, and most times unsolvable. Despite the difficulty, some exact solutions to Einstein's field equations do exist. Most notable are the Schwarzschild metric [96] for stationary black holes, and the Kerr metric [61] for rotating black holes. We can also develop approximate solutions to the field equations. Here we will look at one approach following [78], where we develop a linearized theory of gravity.

First we assume that our metric is a perturbation on flat space-time:

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu} \tag{2.5}$$

where we use the Minkowski metric for flat space[1]

$$\eta_{\mu\nu} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and the perturbation is in some sense small, $|h_{\mu\nu}| \ll 1$. Using Eq. 2.5 in Eq. 2.2 and keeping only up to linear order in $h_{\mu\nu}$, we get

$$R_{\mu\nu\alpha\beta} = \frac{1}{2}(\partial_\nu\partial_\alpha h_{\mu\beta} + \partial_\mu\partial_\beta h_{\nu\alpha} - \partial_\mu\partial_\alpha h_{\nu\beta} - \partial_\nu\partial_\beta h_{\mu\alpha}). \tag{2.6}$$

Using this linearized Riemann curvature tensor, we can write a linearized version of the Einstein field equations (Eq. 2.1) as

$$\Box\bar{h}_{\mu\nu} + \eta_{\mu\nu}\partial^\alpha\partial^\beta\bar{h}_{\alpha\beta} - \partial^\alpha\partial_\nu\bar{h}_{\mu\alpha} - \partial^\alpha\partial_\mu\bar{h}_{\nu\alpha} = -16\pi T_{\mu\nu} \tag{2.7}$$

where for convenience we have introduced the definitions:

$$h \quad = \eta^{\mu\nu}h_{\mu\nu} \tag{2.8}$$

$$\bar{h}_{\mu\nu} \quad = h_{\mu\nu} - \tfrac{1}{2}\eta_{\mu\nu}h. \tag{2.9}$$

We can further simplify Eq. 2.7 by taking advantage of gauge freedom to pick the Lorenz gauge, defined by $\partial^\nu\bar{h}_{\mu\nu} = 0$, from which we get the straightforward expression,

$$\Box\bar{h}_{\mu\nu} = -16\pi\bar{T}_{\mu\nu}. \tag{2.10}$$

[1]We have chosen to use the signature (-1,1,1,1). The choice of (1,-1,-1,-1) is equally valid.

One can immediately recognize the above equation as a *wave equation*, meaning it describes how the perturbation about flat space propagates as a wave– a *gravitational wave*. Whether GWs were a real physically observable phenomenon, or simply a mathematical artifact was for many years a point of debate. While many other aspects of GR, such as the perihelion precession of Mercury [49] and deflection of light by the sun [48], passed their tests with flying colors GWs remained elusive. The first evidence of GWs came from observations of a system of binary pulsars. Their orbital radius decreased at precisely the rate predicted by GWs carrying away energy from the system, a discovery that earned a Nobel prize [59]. The final and most direct confirmation of GWs came from, as we mentioned earlier, LIGO's detection of GWs from a binary black hole system [13]. We now look at how GWs are actually detected.

Gravitational Waves

To detect a GW, we must first understand how it travels through space and time. Since we are observing GWs away from their source, i.e. in a vacuum, the stress-energy tensor vanishes, or $T_{\mu\nu} = 0$, and thus

$$\Box \bar{h}_{\mu\nu} = 0, \tag{2.11}$$

from which we can immediately write down a plane wave solution of the form

$$\bar{h}_{\mu\nu} = A_{\mu\nu}(\vec{k}) e^{i\vec{k}\cdot\vec{x}} \tag{2.12}$$

where $\vec{k}$ is the 4-wave vector, and $A_{\mu\nu}$ is a tensor of constants (the polarization tensor). If we actually plug Eq. 2.12 back into Eq. 2.11, we recover the condition that

$$k_\mu k^\mu = 0. \tag{2.13}$$

The consequence of this null vector condition is that GWs propagate at the speed of light. Further, using the Lorenz gauge as above, we get the condition

$$A_{\mu\nu} k^\nu = 0 \tag{2.14}$$

which implies that GWs are transverse– they propagate perpendicular to their polarization.

We can now enforce yet another gauge condition, the transverse-traceless gauge which enforces that the trace of $\bar{h}_{\mu\nu}$ is zero, or

$$\bar{h} = \bar{h}^i_i = 0, \tag{2.15}$$

which means that $\bar{h}_{\mu\nu} = h_{\mu\nu}$. We further implement the gauge choice that the perturbation to the metric is purely in the spatial components, or

$$h_{00} = h_{0i} = 0.^2 \tag{2.16}$$

By combining these requirements with the Lorenz gauge we note that $\partial_i h_{ij} = 0$, or in other words the metric perturbation is transverse (hence the transverse-traceless gauge).

[2]Here we employ the convention that Roman indices refer to spatial only components, while Greek indices include temporal components as well.

In the end, we can write our metric perturbation in terms of only two independent components which, for reasons that will become clear later, we call h_+ and $h_\times$:

$$h_{\mu\nu}^{\mathrm{TT}} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & h_+ & h_\times & 0 \\ 0 & h_\times & -h_+ & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cos(\omega(t-z)) \tag{2.17}$$

where we have assumed the wave propagates in the z direction ($k^\mu = (\omega, 0, 0, k)$) and we have included the TT superscript to emphasize the fact that we are in the transverse-traceless gauge. The frequency ω is simply $\omega = |\vec{k}|$. The h_+ and $h_\times$ terms are the "plus" and "cross" polarizations and are independent components of the polarization tensor.

Now we look at how a GW would theoretically affect a set of test masses as it passes by. For these, we will look at the geodesic deviation equation, which yields the equations of motion:

$$\ddot{\xi}^\mu = \frac{1}{2}\xi^\nu \ddot{h}_\nu^{\mathrm{TT}\mu}. \tag{2.18}$$

For this particular example, let us consider a purely plus-polarized wave such that $h_\times = 0$, and a set of test particles in the $x - y$. We can then write Eq. 2.17 as

$$h_{ab}^{\mathrm{TT}} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \sin(\omega t) \tag{2.19}$$

where a, b range over the x and y coordinates, and we have chosen the $t = 0$ point

such that $h_{ab}^{\mathrm{TT}} = 0$ there. Our geodesic deviation equatoins are now

$$\begin{aligned} \delta\ddot{x} &= \frac{-h_+}{2}(x(0) + \delta x)\omega^2 \sin(\omega t) \\ \delta\ddot{y} &= \frac{h_+}{2}(y(0) + \delta y)\omega^2 \sin(\omega t). \end{aligned} \tag{2.20}$$

Keeping only up to linear order in the perturbation[3], the expressions for the deviations can be expressed simply by

$$\begin{aligned} \delta x(t) &= \frac{h_+}{2} x(0) \sin(\omega t) \\ \delta y(t) &= \frac{-h_+}{2} y(0) \sin(\omega t). \end{aligned} \tag{2.21}$$

Performing a similar calculation for a purely cross-polarized wave gives deviation equations

$$\begin{aligned} \delta x(t) &= \frac{h_\times}{2} y(0) \sin(\omega t) \\ \delta y(t) &= \frac{h_\times}{2} x(0) \sin(\omega t). \end{aligned} \tag{2.22}$$

To visualize this, think of the canonical ring of massless test particles. In the case of Eq. 2.21, the ring of particles will distort up-and-down, and then sideways. See Fig. 2.1 for an illustration. In the case of Eq. 2.22, the ring of particles will exhibit the same behavior, but rotated by 45°. Note that from Eqs. 2.21 and 2.22 we can interpret a physical meaning to the perturbation h: it is the fractional change in displacement of our test particles. We call this the *gravitational wave strain*, and as we will see later in this Chapter is what we try to measure with interferometric detectors.

[3] δx and δy are $O(h_+)$

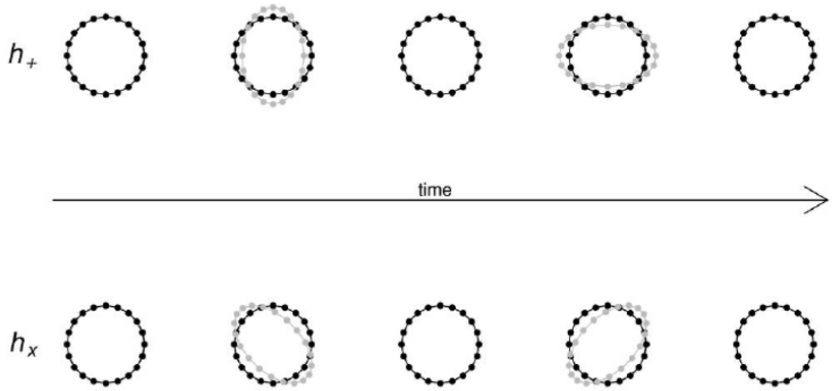

Figure 2.1: Example of how a GW with a plus polarization (top) and cross polarization (bottom) influences a ring of particles. Figure from [25].

Toy Problem: GWs from a Massive Binary System

Now let us look at an example of GWs produced from an actual system. In this case we will look at a very simple case: a system of massive objects orbiting each other in a circular orbit. For this example we follow Chapter 7 of [37].

We start with a more generalized solution to Eq. 2.10 which can be written with a Green's function as

$$\bar{h}_{\mu\nu}(\vec{x},t) = 4\int \frac{T_{\mu\nu}(t',\vec{x}')}{|\vec{x}-\vec{x}'|}d\vec{x} \tag{2.23}$$

where t' is the retarded time $t' = t - |\vec{x}-\vec{x}'|$. We assume that we the observers are sufficiently far from the GW source, and so $|\vec{x}-\vec{x}'| \sim r$, where $r = |\vec{x}|$. Eq. 2.23 reduces further to

$$\bar{h}_{\mu\nu}(\vec{x},t) = \frac{4}{r}\int T_{\mu\nu}(t-r,\vec{x}')d\vec{x}. \tag{2.24}$$

After enforcing the TT gauge and some clever rewriting (see [37] or [52] for the details) we arrive at the expression

$$h^{\mathrm{TT}ij} = \frac{2}{r}\ddot{I}^{ij}(t-r) \tag{2.25}$$

where I^{ij} is the quadrupole moment tensor

$$I^{ij}(t) = \int T^{00}(t, \vec{x}) x^i x^j d\vec{x}. \tag{2.26}$$

Now we can finally look at our example system of a binary system of two equal masses. For simplicity we assume they orbit each other in the $x - y$ plane at a radius R. We will assume their orbital velocities are much less than the speed of light so that we can use Kepler's laws to write down their orbital angular frequency as

$$\Omega = \left(\frac{M}{4R^3} \right)^{1/2}. \tag{2.27}$$

We can then write down the T^{00} component of the stress energy tensor as

$$T^{00} = M \left[\delta^3(\vec{x} - \vec{x}_1(t)) + \delta^3(\vec{x} - \vec{x}_2(t)) \right] \tag{2.28}$$

where the equations of motion $\vec{x}_1(t)$ and $\vec{x}_2(t)$ can be found with Eq. 2.27:

$$\begin{aligned} \vec{x}_1(t) &= (R\cos\Omega t, R\sin\Omega t) \\ \vec{x}_2(t) &= (-R\cos\Omega t, -R\sin\Omega t). \end{aligned} \tag{2.29}$$

A straightforward integration of Eq. 2.26 using our particular stress energy tensor leads to the metric perturbation components

$$\bar{h}_{ij}(t, \vec{x}) = -\frac{8M}{r}\Omega^2 R^2 \begin{pmatrix} \cos(2\Omega(t-r)) & \sin(2\Omega(t-r)) & 0 \\ \sin(2\Omega(t-r)) & -\cos(2\Omega(t-r)) & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{2.30}$$

In this toy problem we have kept our simplifying assumption that the GW propagates

along the the z direction only. We can generalize h_+ and $h_\times$ to a binary system of any arbitrary inclination ι between the orbital plan and the line of sight to the observer:

$$\begin{aligned} h_+ &= \frac{4M}{r}\Omega^2 R^2 \cos(2\Omega(t-r))(1+\cos^2\iota) \\ h_\times &= \frac{8M}{r}\Omega^2 R^2 \sin(2\Omega(t-r))\cos\iota \end{aligned} \tag{2.31}$$

From which we can see a GW will be strongest from a system that is either face on or face off.

We can also use the above expressions to produce an estimate of the magnitude of the strain at the observation point of

$$|h| \simeq \frac{4M}{r}\Omega^2 R^2 \tag{2.32}$$

into which we can plug in some numbers for a back of the envelope estimate of GW strain. Say our two masses are black holes with $M = 50M_\odot$, are at a separation of $R = 6M$, and are relatively close by at 10 Mpc. This gives a strain magnitude estimate of about 10^{-21}, which is extremely small! Next we will discuss methods by which we can actually detect these minuscule shifts in spacetime.

GW Detectors

Interferometric GW Detectors

Now we will look at how the LIGO and Virgo GW detectors actually measure GWs by giving an overview of interferometric GW detectors, similar to the approach in [78].

We start with a Michelson interferometer [84], which consists of a laser split by a beam splitter down two perpendicular arms of equal length. At the end of each

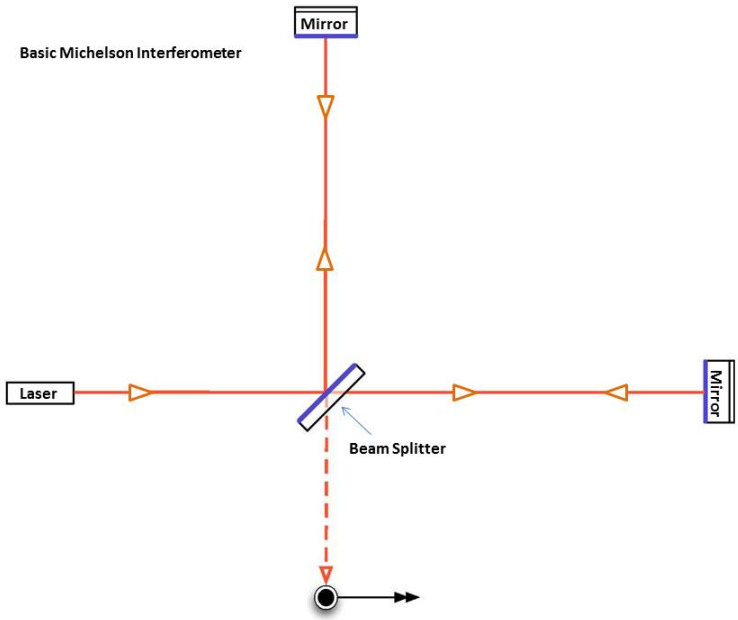

Figure 2.2: Basic schematic of a Michelson interferometer GW detector. Figure from [67].

arm is a mirror which reflects the beams back to the beam splitter where they are recombined, and aimed at a photodiode detector. A simple schematic of the LIGO interferometer is shown in Fig. 2.2.

The LIGO interferometers are set up such that the two beams interfere destructively, so the photodiode is always dark. If, however, one of the arms were to change its length the interference pattern would change, allowing some light to hit the photodiode. We will examine now how we can use this interference pattern to measure GWs.

Let the electric field emitted by the laser be

$$E_{\text{input}} = E_0 e^{-i\omega_L t + i\vec{k}_L \cdot \vec{x}} \tag{2.33}$$

where the L subscripts denotes the frequency and wavenumber of the laser light

wavelength, not to be confused with the GW wave properties. We will put the two arms of the interferometer on the x and y axes, with lengths L_x and L_y respectively. Then the electric fields at the beam splitter after the x and y beams have been reflected back down the arm are

$$\begin{aligned} E_x &= \frac{-1}{2} E_0 e^{-i\omega_L t + 2ik_L L_x} \\ E_y &= \frac{1}{2} E_0 e^{-i\omega_L t + 2ik_L L_y}, \end{aligned} \tag{2.34}$$

which gives a total output electric field of

$$E_{\text{out}} = -iE_0 e^{-i\omega_L t + ik_L(L_x + L_y)} \sin\left(k_L(L_y - L_x)\right). \tag{2.35}$$

Now using the fact that the power of a laser is proportional to the square of the electric field's amplitude, we see that

$$P \propto |E_{\text{out}}|^2 = E_0^2 \sin^2\left(k_L(L_y - L_x)\right). \tag{2.36}$$

When $L_y = L_x$, or the light travel time down the two arms is equal, the output power is zero. Now we look at a how a GW passing by our interferometer arms will affect this output power.

For simplicity, let us look at the case of a purely plus-polarized wave that is traveling in the z-direction. We then have, from Eq. 2.17

$$h(t) = h_+(t) = h_0 \cos(\omega_{GW} t) \tag{2.37}$$

where we have put our detector in the $z = 0$ plane. Here ω_{GW} denotes the frequency

of the GW. Staying in the TT framework, we can write the line element as

$$ds^2 = -dt^2 + (1 + h(t))dx^2 + (1 - h(t))dy^2 + dz^2. \tag{2.38}$$

Now say we are looking at a photon traveling down the x-arm of this setup. This means $dy = dz = 0$, and $ds^2 = 0$ because photons follow a null geodesic. We find (using $h_0 \ll 1$) that

$$dx = \pm(1 - \frac{1}{2}h(t))\ dt. \tag{2.39}$$

In the case of our Michelson interferometer, we have a photon that leaves the beam splitter at a time $t = t_0$, and reaches the mirror at location $x = L_x$ at time $t = t_1$, and is reflected back to the beam splitter where it arrives at time $t = t_2$. A simple integration of Eq. 2.39 for the photon's trip from the beam splitter to the mirror gives

$$L_x = (t_1 - t_0) - \frac{1}{2}\int_{t_0}^{t_1} h(t)dt \tag{2.40}$$

where we have chosen the positive branch of Eq. 2.39 for the first leg of the photon trip. Similarly, choosing the negative branch of Eq. 2.39 for the photon's return trip gives

$$L_x = (t_2 - t_1) - \frac{1}{2}\int_{t_1}^{t_2} h(t)dt. \tag{2.41}$$

We can use Eqns. 2.40 and 2.41 to give the total travel time of the photon in the x-arm:

$$t_2 - t_0 = 2L_x + \frac{1}{2}\int_{t_0}^{t_2} h(t)dt. \tag{2.42}$$

Note here that in the TT gauge, it is not the locations of the mirrors and beam splitters that are changing (the beam splitter is always at $x = 0$ and mirror at $x = L_x$) but

the light travel time that changes. Performing the integral above, we return

$$t_2 - t_0 = 2L_x + L_x h(t_0 + L_x) \tag{2.43}$$

where we have made the assumption $\omega_{GW} L_x \ll 1$, which is the case for kilometer-scale interferometers like LIGO and Virgo.

We now relate this back to the electric field in Eq. 2.34. Say that at the time the laser beam leaves the beam splitter, it has phase

$$E_x = \frac{-1}{2} E_0 e^{-i\omega_L t_0}. \tag{2.44}$$

Using Eq. 2.43 and the fact that the phase of the electric field is conserved as it propagates through a vacuum, we get an expression for the electric field at time t_2

$$E_x(t_2) = \frac{-1}{2} E_0 e^{-i\omega_L (t_2 - 2L_x) + i\omega_L L_x h(t_2 - L_x)}. \tag{2.45}$$

From here is useful to generalize $t_2 \equiv t$, and also to define the phase change,

$$\Delta\phi_x(t) = \omega_L L_x h(t - L_x). \tag{2.46}$$

A nearly identical calculation for photons traveling down the y-arm, but using $dy = \pm(1 + \frac{1}{2}h(t))\ dt$ in lieu of Eq. 2.39 gives the electric field and phase change of the beam traveling down the y-arm:

$$E_y(t) = \frac{1}{2} E_0 e^{-i\omega_L (t - 2L_y) - i\omega_L L_x h(t - L_y)} \tag{2.47}$$

$$\Delta\phi_y(t) = -\omega_L L_y h(t - L_y). \tag{2.48}$$

In the LIGO interferometers, L_x and L_y are designed to be as close to the same length as possible, so using $L = L_x = L_y$, we see that $\Delta\phi_x(t) = -\Delta\phi_y(t)$. Finally, the power of the output electric field at the photodiode is proportional to

$$\begin{aligned} |E_{\text{output}}|^2 &= |E_x(t) + E_y(t)|^2 \\ &= E_0^2 \left(1 - \cos(2\phi_0 + \Delta\phi_{\text{Mich}}(t))\right) \end{aligned} \tag{2.49}$$

where $\Delta\phi_{\text{Mich}} \equiv 2\Delta\phi_x(t)$ and $\phi_0 = \omega_L(L_x - L_y)$ is the initial phase, and can be determined by the experimenter. Thus we have shown that we can measure the GW strain h by measuring the change in power at the output of a Michelson interferometer.

Of course the real LIGO and Virgo interferometric detectors are much more complicated than a simple Michelson interferometer, and have features that allow us to measure the desired strain sensitivity of $h \sim 10^{-21}$. For example, each photon does not make a single round-trip down one arm and back, but is rather trapped in a Fabry-Perot cavity and reflected many times. This effectively lengthens LIGO's 4km (and Virgo's 3km) arm lengths, thus increasing our sensitivity [62]. Additionally, by collecting many photons over the period of a GW we are further able to increase our sensitivity.

Interferometer Sensitivity

We note though that the sensitivity of these interferometric GW detectors to signals is dependent on the sky location of the signal, and its frequency.

Our example above concerned only a plus-polarized GW traveling along the z-axis. We can generalize this to a GW from an arbitrary location, with some mixture of h_+ and $h_\times$ polarizations. For this it is useful to define the antenna patters, which are a way of describing how our test masses respond to signals from different locations in

the sky. These antenna patterns for the plus and cross polarizations are given by [26]

$$\begin{aligned} F_+ &= \frac{1}{2}(1+\cos^2\theta)\cos 2\phi\cos 2\psi + \cos\theta\sin 2\phi\sin 2\psi \\ F_\times &= \frac{1}{2}(1+\cos^2\theta)\cos 2\phi\sin 2\psi + \cos\theta\sin 2\phi\cos 2\psi \end{aligned} \tag{2.50}$$

where (θ, ϕ) give the location of the GW source in the sky, and ψ is the polarization angle. Note that all of the quantities are in the reference frame of the detector. Our strain can then be written in a more general sense as

$$h(t) = h_+(t)F_+ + h_\times(t)F_\times. \tag{2.51}$$

We are also limited in what types of signals we can detect by the frequency of the GWs. As one might expect of a detector capable of measuring such small displacements, it's incredible sensitivity also means it is highly susceptible to a range of noise sources as well. Some examples of noise affecting LIGO's sensitivity is seismic noise from earthquakes and human activity, thermal noise from atomic fluctuations in the test masses, and noise from slight variations in the laser's direction and intensity. More detailed descriptions of noise sources can be found in [80].

To look at how we measure the sensitivity of LIGO it is useful to define the power spectral density of the. We can assume that our detector output will consist of two parts: the signal or strain, and the noise, which we can write as

$$s(t) = h(t) + n(t) \tag{2.52}$$

where $s(t)$ is the total detector output and $n(t)$ which is the noise. While most signals for which the magnitude of $h(t)$ is significantly greater than the order of $n(t)$ should almost certainly be detectable, we can still measure signals that are buried deeper in

the noise. However to do so, we must be able to characterize our noise, which we now describe below.

First we assume that the noise is stationary, and normally distributed. This simple assumption is surprisingly accurate for LIGO data. In this case the average noise level is zero, or explicitly:

$$\langle n \rangle = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} n(t) dt = 0 \tag{2.53}$$

where T is the observation time. We can then define the one-sided noise power spectral density (PSD) $S_n(f)$ as

$$\langle \tilde{n}^*(f) \tilde{n}(f') \rangle = \frac{1}{2} \delta(f - f') S_n(f) \tag{2.54}$$

where we have used the Fourier transform of the noise:

$$\tilde{n}(f) = \int_{-\infty}^{\infty} n(t) e^{-2\pi i f t} dt. \tag{2.55}$$

Eq. 2.53 fully characterizes the noise of our instrument. Examples of Advanced LIGO's PSD from the first observing run in 2015 are shown in Fig. 2.4. We see that LIGO is most sensitive around 100Hz. Potential GW sources around this frequency include stellar mass binary black holes, binary neutron stars, core collapse supernovae, cosmic string cusps, and of course the possibility of exciting as-of-yet unpredicted GWs. However there are also predicted GW sources in frequency ranges where LIGO has no appreciable sensitivity. Next we will briefly discuss other types of GW detectors used to search for these other sources.

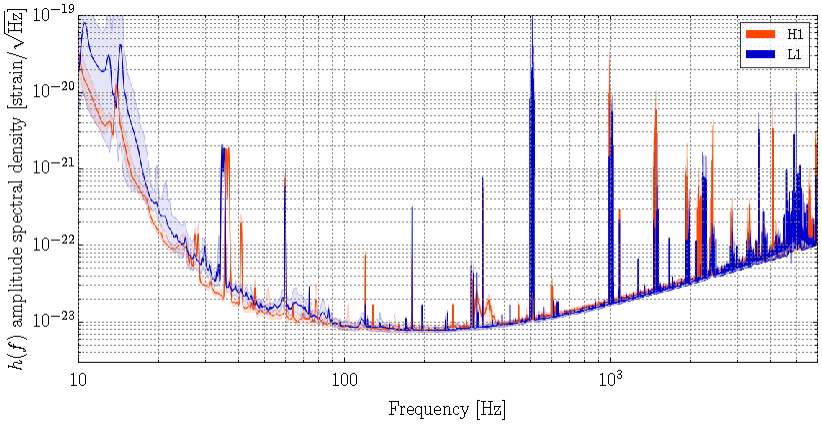

Figure 2.3: Example noise spectrum from LIGO's Hanford and Livingston detectors around the time of the GW150914 detection. The sharp peaks, or "lines" in the noise curve arise from known mechanical and electrical resonances. The noise at low frequency is dominated by seismic noise, while shot noise is the dominant source at the higher end of the frequency range. Figure from [10].

Other GW Detectors

There are two more ground-based interferometric detectors currently planned or under construction, including the KAGRA detector in Japan [24], and IndIGO in India [2]. These additional detectors will help detect more GWs, and help better estimate GW signal sky locations. Beyond ground-based interferometers, we are currently searching for GWs using pulsar timing arrays (PTAs). Pulsars are rapidly rotating neutron stars that emit a beam of electromagnetic radiation. Pulsars have very regular rotational periods making them extremely accurate timing devices. If a GW were to pass by the pulsar or earth, the distance between earth and the pulsar would shrink or grow, thus changing the arrival time of the radio pulse at earth. By carefully monitoring the pulse arrival times of a number of pulsars (or an "array") of pulsars, we can look for correlations in discrepancies between predicted and observed times of arrival, which would be indicative of a passing GW [58]. PTAs are most sensitive in the nHz frequency range, which means they are sensitive to GWs

from supermassive black hole binaries (SMBHs), and in particular are searching for a stochastic GW background (GWB) of SMBHs. Current projections estimate the first GWB detection within the next few years[4] [101].

Another planned GW detector is the Laser Interferometer Space Antenna (LISA) [3]. LISA is, as its name suggests, another interferometric detector, but will be located in space. Placing a detector in space means it can have much larger arm lengths than ground-based methods like LIGO. The longer arm lengths will make LISA more sensitive at lower frequencies, with the most sensitivity in the mHz region. This makes LISA sensitive to SMBHs, white dwarf binaries, and extreme mass ratio inspirals (i.e. a small black hole orbiting a big black hole). LISA is currently planned to launch around 2030, and based off of the astounding results from a preliminary mission to test some of the technologies needed for the full mission [27], we can expect phenomenal sensitivities and lots of discoveries from LISA.

Outlook

With LIGO's first GW detections over the past few years, we are firmly in the age of GW astronomy. The addition of PTAs and LISA promise further open this new window into the universe. For the rest of this thesis, we will focus on analysis of LIGO data, though many of the approaches and techniques could in the future be applied to other GW detection methods. In particular, we will look at how to separate signals from noise, and how to maximize our abilities to detect GWs from *any* possible source.

[4] Even in the absence of a GWB detection we are still gaining astrophysical insight.

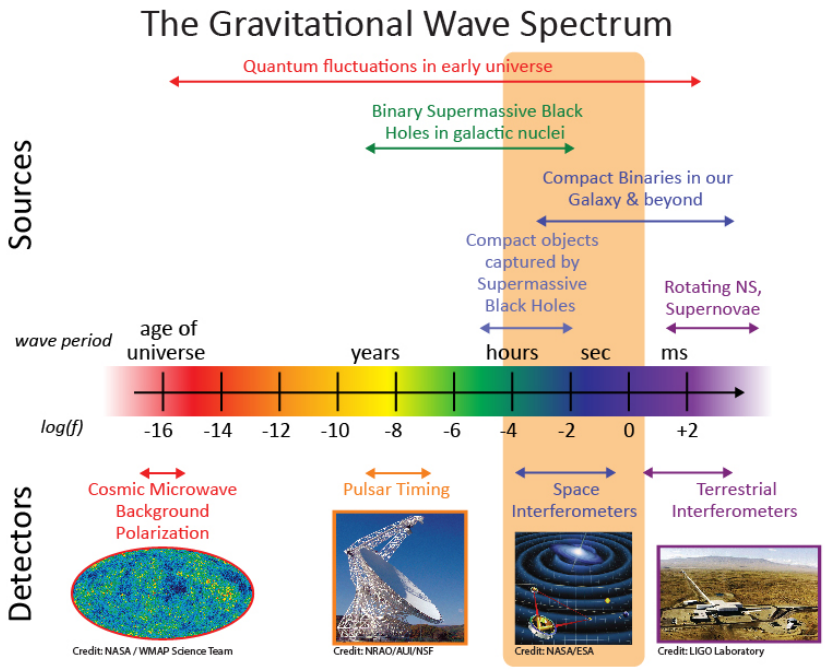

Figure 2.4: Examples of the GW frequency range and potential sources for a range of GW detection methods. Figure from [3].

24

BAYESIAN INFERENCE AND MCMC

In this chapter we will build the foundations needed to carry out GW data analysis. In particular, we introduce Bayesian probability theory, and the Markov chain Monte Carlo (MCMC) statistical method of implementing Bayesian inference.

Bayes Theorem

For any scientific experiment we have a set of data we collected, which we will call $\vec{d}$. We also have one or more hypotheses about the underlying model that produces the data, which we will call $\mathcal{H}_i$, where i refers to the specific model. These models may in turn depend on some set of parameters $\vec{\theta}$. The goal of any experiment is to produce some estimate of the parameters $\vec{\theta}$ and/or to compare the relative odds of two or more models $\mathcal{H}_i$. This can be accomplished using Bayes theorem:

$$p(\vec{\theta}|\vec{d},\mathcal{H}_i) = \frac{p(\vec{d}|\vec{\theta},\mathcal{H}_i)p(\vec{\theta}|\mathcal{H}_i)}{p(\vec{d}|\mathcal{H}_i)}. \tag{3.1}$$

Each probability in Eq. 3.1 $p(A|B)$ can be read as "the probability of A given B". Let us further examine the ingredients of this equation.

- $p(\vec{\theta}|\vec{d},\mathcal{H}_i)$ is the *posterior distribution function*, which gives us the probability that the true parameters are $\vec{\theta}$, given the observed data $\vec{d}$ and hypothesis $\mathcal{H}_i$. This probability function is used to perform parameter estimation.

- $p(\vec{d}|\vec{\theta},\mathcal{H}_i)$ is the *likelihood*. This gives the probability that we observed the data $\vec{d}$ assuming the true parameters are $\vec{\theta}$ and the model is $\mathcal{H}_i$. In other words, it is a measure of how well our data fits the given model.

- $p(\vec{\theta}|\mathcal{H}_i)$ is the *prior probability distribution* on our parameters $\vec{\theta}$. Sometimes the prior distribution of a parameter is unknown, and we use an uninformative prior such as a uniform prior over some range. Other times we can use information from previous measurements of $\vec{\theta}$ to develop a prior[1].

- $p(\vec{d}|\mathcal{H}_i) = \int p(\vec{\theta}|\mathcal{H}_i)p(\vec{d}|\vec{\theta}\mathcal{H}_i)d\vec{\theta}$ is the *marginal likelihood* or *evidence* for the model $\mathcal{H}_i$, and can also be thought of as a normalizing factor to be sure the total integrated probability of the posterior is one. The evidence is used when comparing model hypotheses.

Once one has collected the data and proposed a model, it is relatively straightforward to write down the likelihood and the prior. Calculating the posterior and the evidence in particular is not so straight forward. The evidence is a multidimensional integral (often in GW data analysis $\vec{\theta}$ can have 15 dimensions, and in our unmodeled analysis we can have hundreds!) which can be prohibitively computationally expensive. It is for this reason that though Bayes theorem has been around since Reverend Bayes in the 18th century, the frequentist approach to statistical inference, which focuses on the frequency of event occurrences, was the standard in data analysis[2]. Now however, developments in both computational methods and technology have lead to practical and clever methods of calculating the posterior and evidence.

Two of the primary approaches to sampling from a posterior distribution are Nested Sampling [98] and Markov chain Monte Carlo [57,82]. This thesis uses MCMC methods, and we now go into detail on their implementation.

[1] Using a common bit of Bayesian wisdom: "Yesterday's posterior is today's prior".

[2] For an introductory comparison between frequentist and Bayesian inference, see section 3 of [91].

MCMC

An MCMC algorithm is a method of exploring a parameter space, and returning a collection of values drawn from some distribution. The "Markov chain" part of an MCMC refers to an underlying process where the next generated data point in the chain depends only on the current location of the chain (i.e. it is memoryless). The "Monte Carlo" part of the name refers to the stochastic process that is used in generating the chain.

Metropolis-Hastings

To implement an MCMC, we use the the Metropolis-Hastings algorithm [57,83]. For the Metropolis-Hastings algorithm we first start with some target probability distribution (i.e. a posterior) from which we want to sample– call it $\pi(\vec{x})$. The rest of the algorithm proceeds as follows:

1. Pick a starting point in your parameter space, $\vec{x}_0$.

2. Pick a new point, $\vec{x}_1$ which is drawn from some *proposal distribution* $q(\vec{x}_1|\vec{x}_0)$. We will discuss the importance of choosing a good proposal distribution later in this chapter.

3. Calculate the *acceptance ratio*:

$$\alpha = \frac{\pi(\vec{x}_1)}{\pi(\vec{x}_0)} \frac{q(\vec{x}_0|\vec{x}_1)}{q(\vec{x}_1|\vec{x}_0)} \tag{3.2}$$

4. Generate a random number u from a uniform $[0, 1]$ distribution.

5. If $u \leq \alpha$, accept this jump and add the value of $\vec{x}_1$ as the new point in the chain. Otherwise reject that jump and the next point in the chain retains the

$\vec{x}_0$ value.

6. Repeat steps 2-5 until enough points in the chain are generated.

The final chain of points $\vec{x}_i$ will follow a distribution function described by $\pi(\vec{x})$. The acceptance ratio is the key to letting the chains explore the whole parameter space, returning the desired distribution. If we were to skip steps 4 and 5 and only move the chain into regions of higher probability we would simply have a "hill climber" algorithm. This would not allow us to fully map out the parameter space of the posterior distribution. Instead, the Metropolis-Hastings algorithm always accepts jumps to regions of higher probability, and *sometimes* accepts jumps to regions of lower probability. This allows the chain to spend a lot of time and map out in detail high probability regions, while also exploring lower probability regions to map the entire parameter space.

There is no one way to decide when an MCMC is complete, but typically one looks for when the chains have converged onto a stationary distribution. One way this can be tested is by splitting a chain in half, and checking that the distribution of one half matches that of the other. Another method usually employed is to discard the "burn-in" stage. This is a way of ensuring that the chain has no memory of its starting location. The convergence time and burn-in period can vary widely for different MCMC implementations and so there no standard number of iterations used in MCMC across the board. Another metric used is to calculate the correlation lengths of the chains, which gives the number of independent samples in the chains. For higher dimensional posterior distributions, we need many independent samples. In general the longer you can run your MCMC the more accurate your posterior distributions will be.

The convergence time can be highly dependent on the proposal distribution

$q(\vec{x}_0|\vec{x}_1)$. The proposal distribution is confusingly both irrelevant and the most important part of the algorithm. Any proposal distribution that covers the parameter space will eventually converge given an infinite amount of time. Unfortunately none of us have an infinite amount of time, and so we must be clever about our proposal distribution. The best proposal distribution in theory is just our target distribution, but generally we are relying on MCMC techniques because we do not *a priori* know the functional form of this distribution. We will now look at some examples of frequently used jump proposals, and some common difficulties encountered when using an MCMC.

MCMC Toy Problem

To investigate some of these MCMC details, we first pick an example target posterior distribution to try to sample from. Let us choose:

$$\pi(x,y) = \frac{16}{3\pi}\left(e^{-x^2-(9+4x^2+8y)^2} + \frac{1}{2}e^{-8x^2-8(y-2)^2}\right). \tag{3.3}$$

This distribution is plotted in Fig. 3.1, and shows that there are two interesting features: bimodality of the distribution, and an unusual shape for one of the modes. Both of these features can present a challenge for MCMC methods. Without clever jumps, an MCMC can frequently only explore one mode of a distribution, missing out key features of the posterior distribution. A curved distribution, like the lower (in terms of x-y space) mode in Eq. 3.3 can be challenging to map out in detail without similarly clever jump proposals.

Here we examine the performance of several types of MCMC jumps: Fisher matrix proposals, Langevin jumps, and parallel tempering.

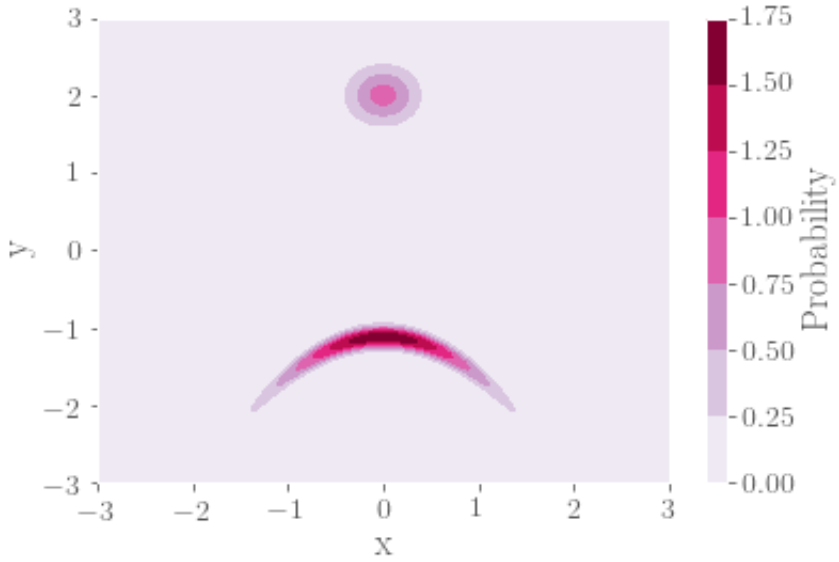

Figure 3.1: The target distribution we will be exploring.

Fisher Matrix Proposals

Perhaps the most common proposal distribution used currently in our MCMC implementations is a proposal based on the Fisher matrix of the target distribution. The Fisher matrix is most generally defined as the negative Hessian of the log of the likelihood distribution [42], or

$$\Gamma_{ij} = -\left\langle \partial_i \partial_j \log(p(\vec{x})) \right\rangle \tag{3.4}$$

where $\langle\rangle$ denotes the expectation value. The Fisher matrix is commonly described as a measure of the information about some variable, and can be used as an approximate to the inverse of the covariance matrix. This means that the Fisher information encodes information about correlation between parameters, and thus can help guide our jumps into the higher probability regions, decreasing convergence times. In this particular toy problem, we will use the Fisher matrix of the target distribution the

form

$$\Gamma_{ij} = -\partial_i \partial_j \log(\pi(\vec{x})). \tag{3.5}$$

Practically, we use the Fisher matrix to devise a proposal distribution that is a D dimensional multivariate Gaussian:

$$q(\vec{x}_{n+1}|\vec{x}_n) = \frac{\det \Gamma^{D/2}}{(2\pi)^{D/2}} e^{-\frac{1}{2}\Gamma_{ij}(x^i_{n+1}-x^i_n)(x^j_{n+1}-x^j_n)}. \tag{3.6}$$

Then working in the Fisher eigenbasis, the jump proposals will be of the form:

$$\vec{x}_{n+1} = \vec{x}_n + \frac{\delta}{\sqrt{\lambda_i}} \hat{e}_{(i)} \tag{3.7}$$

where δ is a random number drawn from a normal distribution with zero mean and unit variance, and λ_i and $\hat{e}_i$ correspond to eigenvalues and eigenvectors of the Fisher matrix. In our Fisher jumps, we select at random one eigendirection to jump along at a time.

In this toy problem we will look at two particular implementations of Fisher jumps: one where there is a single global Fisher matrix calculated once and used for all jumps, and one where the Fisher matrix is calculated at every chain iteration.

Global Fisher Matrix Proposals

In this case of a global Fisher matrix, we calculate Eq. 3.4 once, and evaluate it at the parameters that give the maximum likelihood in the target distribution. Using the distribution in Eq. 3.3 we find that Fisher matrix contains zeroes in the off diagonals, so when proposing jumps along the eigendirections we will be jumping in one parameter at a time. This makes this particular jump proposal a good example of a very basic MCMC implementation– especially because in this case $q(x_{n+1}|x_n) =$

$q(x_n|x_{n+1})$ (as seen in Eq. 3.6) so the proposals will cancel in the Hastings ratio.

Results To test how this jump proposal explores the target distribution, we ran one chain at an initial starting point $(x_0, y_0) = (1, 2)$, and one at an initial starting point $(x_0, y_0) = (0, -2)$, both for $N = 100000$ iterations. Shown in Fig. 3.2 are scatter plots of the points in both of these chains, as well as a line showing the path of the chain for the first 100 iterations. In our final distribution, these first 100 iterations would be discarded in the burn in period, but for illustrative purposed it is useful to look at them to see how the chain approaches the maxima. The first thing of note is that these two different starting points each found one of the maxima and stayed there. This is a prime example of one of the limits of basic MCMC implementations: chains can get stuck on one mode of a multimodal distribution. We also see that for the chain that explored the lower peak, it spent more time in one of the "legs" of the distribution, not fully mapping out the finer details of that mode. So the two issues we wish to approach with more well thought out jump proposals are: exploring more than one probability distribution peak, and better mapping out the peak with an unusual shape.

Local Fisher Matrix Proposals

The next proposal distribution we will implement are the other form of Fisher jumps– ones where the Fisher matrix in Eq. 3.4 is evaluated at the current chain location rather that at the global likelihood maximum. This approach is clearly more computationally expensive, but has the advantage that it takes into account the curvature of local parameter space to guide the jump directions. As with the global Fisher jump proposals, we run two chains for $N = 100000$ iterations at initial points of $(x_0, y_0) = (1, 2)$ and $(x_0, y_0) = (0, -2)$.

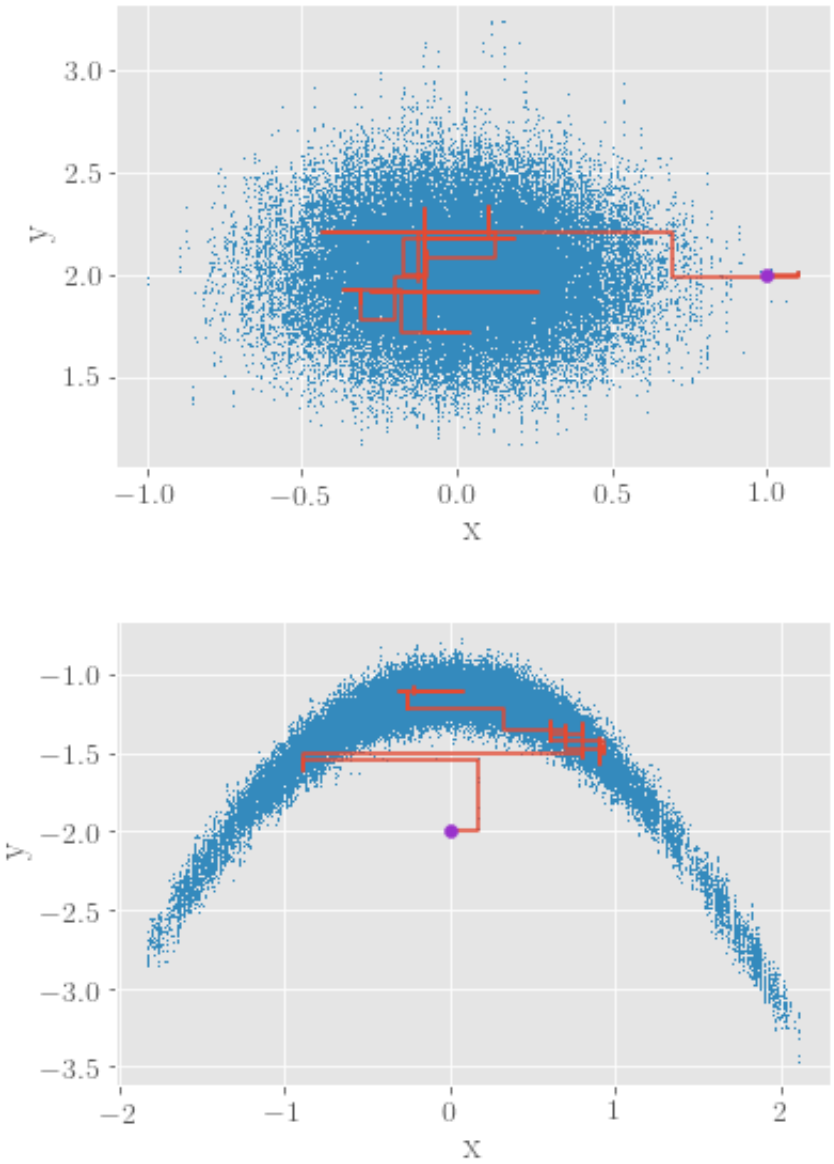

Figure 3.2: Distributions returned by the basic MCMC method with a constant Fisher matrix for starting points $(x_0, y_0) = (1, 2)$ (top) and $(x_0, y_0) = (0, -2)$ (bottom). Also shown is the path taken for the first hundred iterations.

Results Shown in Fig. 3.3 are scatter plots from these two chains, again with the path of the first 100 chain iterations overlaid. For the chain started in the vicinity of the upper maximum, we see that again the chain fails to explore the second peak. The chain started closer to the lower curved peak shows some interesting behavior though. The chain begins by exploring the lower peak, but then finds the upper one. While this is moving towards the type of behavior we want– exploration of all maxima– unfortunately in this implementation once the chain finds the upper peak it abandons the lower peak completely.

To examine why exactly this behavior occurs, we can focus on the section of the chain containing the transition from the lower to upper maximum. The oblong shape of the lower peak mess that eigenvectors in that vicinity will be long in one direction, and short in the other, as shown in Fig. 3.5. This means that it is possible to propose pretty large jumps away from that lower maximum. Typically, these jumps are proposed into areas with low likelihood (i.e. the target function there is essentially zero), which means there is practically zero probability that those jumps will be accepted. But since our target distribution has the second upper maximum, it is possible for the chain to propose a jump into that region of higher likelihood, and for that jump to be accepted. Once the chain is in the area of the upper maximum, the eigenvectors are relatively small, so there is very little probability of proposing a jump very far away from that peak, resulting in a very low (effectively zero) probability that the chain will ever return to the lower maximum.

Though the local Fisher matrix proposals do not solve the problem of sufficiently exploring multiple maxima, we can see particularly in the lower panel of Fig. 3.3 how the first hundred iterations follow the curve of the lower region. This indicates that these Fisher jump proposals can more easily explore complex parameter space, and converge more quickly than simple jumps like the global Fisher matrix proposals.

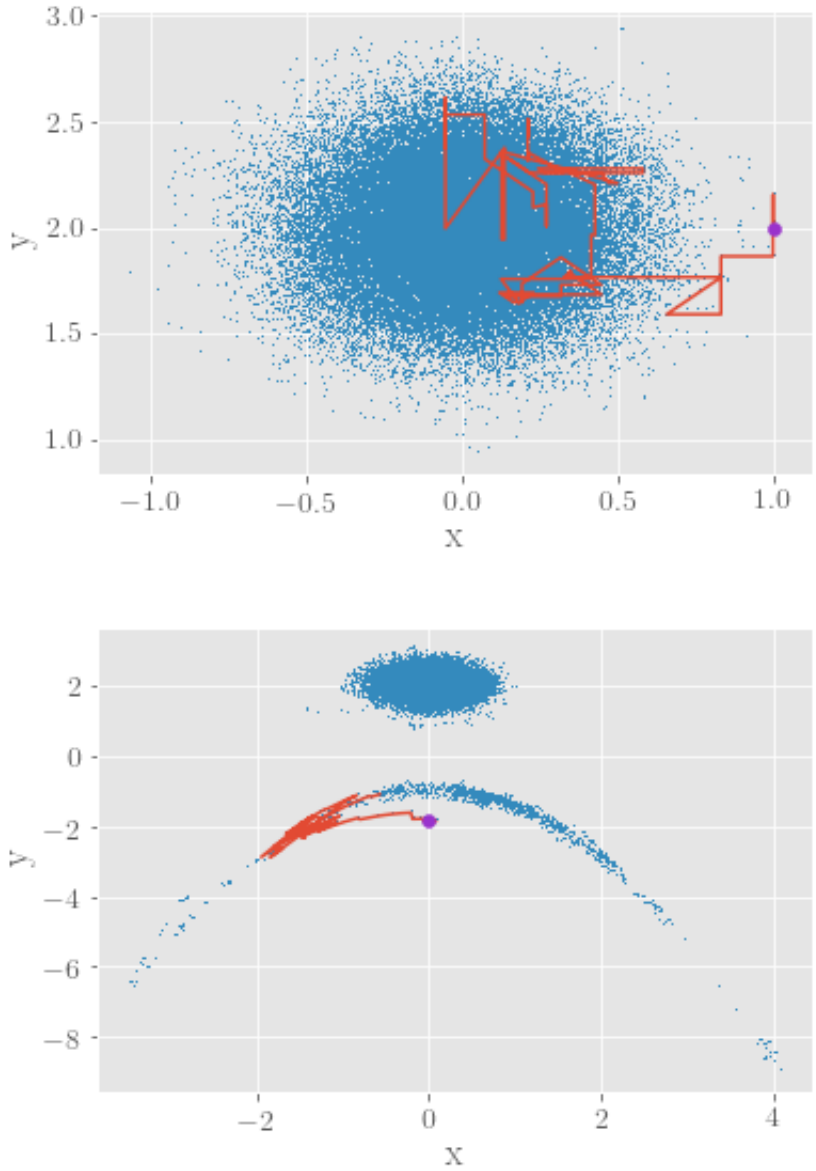

Figure 3.3: Distributions returned by the basic MCMC method with a constant Fisher matrix for starting points $(x_0, y_0) = (1, 2)$ (top) and $(x_0, y_0) = (0, -2)$ (bottom). Also shown is the path taken for the first hundred iterations.

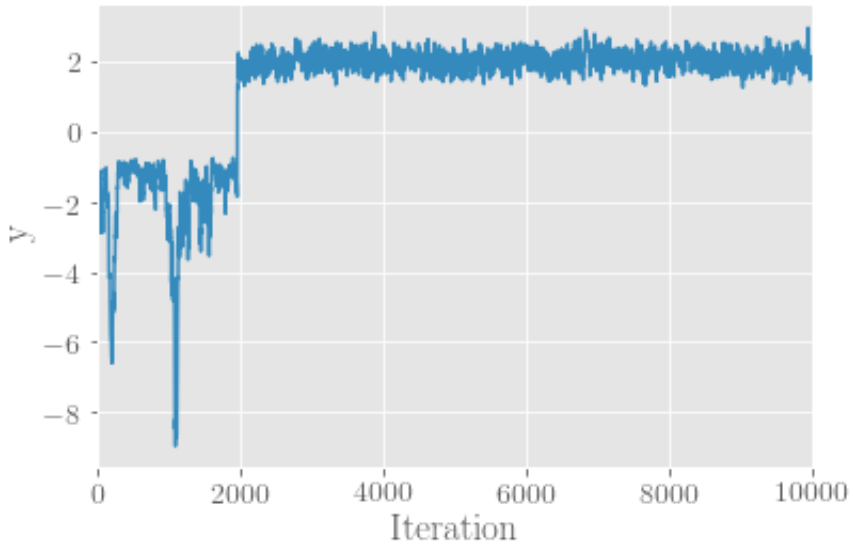

Figure 3.4: A selection of the y values of the chain from the distribution in the lower panel of Fig. 3.3, which illustrate where the chain transitions from the lower to the upper maximum.

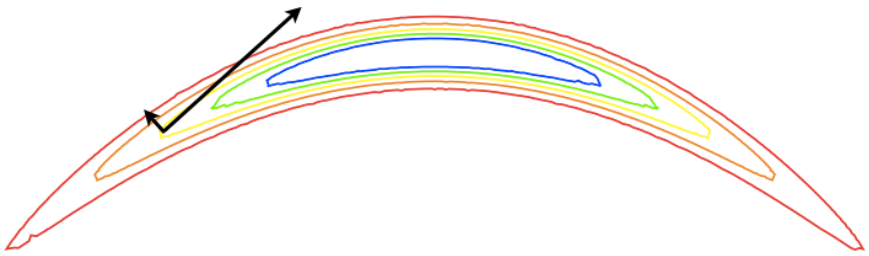

Figure 3.5: An example of the relative directions and magnitudes of the eigenvectors at a particular location on the lower peak.

Langevin

Another Markov Chain Monte Carlo method is Langevin MCMC. Langevin MCMC improves upon the basic MCMC algorithm by proposing jumps that are more likely to be in the direction of a maxima. This is done by using information based on the gradient of the target distribution. The most extensive Langevin method is the Riemann Manifold Langevin (RML) method. The RML method treats the Fisher matrix as sort of curvature metric (or Riemannian metric) in a space parameterized by our target probability density function. Instead of the full RML algorithm, here we use the somewhat simpler Manifold Metropolis Adjusted Langevin Algorithm (MMALA), in which the jumps are describe by:

$$\vec{x}_{n+1} = \vec{x}_n + \frac{1}{2}\mathbf{C}(\vec{x}_n)\nabla \log \pi(\vec{x}_n) + \sqrt{\mathbf{C}(\vec{x}_n)}\vec{z} \tag{3.8}$$

where $\vec{z}$ is a random variable drawn form a multivariate normal distribution. $\mathbf{C}$ is the Covariance matrix (inverse of the Fisher matrix). Typically the $\sqrt{\mathbf{C}}$ term is calculated using Cholesky decomposition, however this can be simplified as we will see below. Using the above jumps, the proposal distribution $q(\vec{x}_{n+1}|\vec{x}_n)$ is a multivariate normal distribution with covariance (which can be thought of as the "width" term) defined by $\mathbf{C}$ and a mean given by:

$$\mu(\vec{x}_n) = \vec{x}_n + \frac{1}{2}\mathbf{C}(\vec{x}_n)\nabla \log \pi(\vec{x}_n). \tag{3.9}$$

This proposal distribution looks very similar to the one used in the Fisher matrix proposals– however in this case the mean is shifted by a gradient term. And again, similarly to how the jumps were proposed in the previous section, we can work in the Fisher matrix eigenbasis and propose jumps along the eigendirections. In this case,

the jumps will look like

$$\vec{x}_{n+1} = \vec{x}_n + \hat{e}_{(i)} \frac{\hat{e}_{(i)} \cdot \nabla \log \pi(\vec{x}_n)}{2\lambda_i} + \frac{\delta}{\sqrt{\lambda_{(i)}}} \hat{e}_{(i)}. \tag{3.10}$$

Note that this has the benefit of eliminating the need for Cholesky decomposition. Because Langevin jumps use information based on the gradient of the current location of the chain, we expect that this method will be better than the basic MCMC at finding and exploring the local maxima of complicated distributions, such as the banana shaped maximum in our example target distribution.

Results Shown in Fig. 3.6 are the Langevin chains run for a hundred thousand iterations, with the first 100 steps illustrated, using the same starting points as in the previous section. We see the same pathology for the Langevin proposals as for the local Fisher matrix jumps: starting points close to the upper maximum explore only that peak, while initial points near the lower maximum explore that peak for a while before latching onto the upper peak. However we see in that time that the Langevin jumps appear to explore the distribution better, perhaps decreasing the time needed for convergence. One way to test this is to look at the *autocorrelation length*. The autocorrelation is defined to be

$$\rho(h) = \frac{\gamma(h)}{\gamma(0)} \tag{3.11}$$

where

$$\gamma(h) = \frac{1}{N-h} \sum_{i=1}^{N-h} (x_{i+h} - \bar{x})(x_i - \bar{x}) \tag{3.12}$$

with $\bar{x}$ the mean of the distribution. The autocorrelation length is defined as the h at which $\rho(r) \sim 0.01$. A shorter autocorrelation length means a chain that has converged more quickly. Fig. 3.7 shows the autocorrelations of x for the local Fisher jumps and Langevin jumps. For this example we used the chains that only explored

the upper peak, as to not complicate matters with the peak jumping feature. We see that the autocorrelation for the Langevin jumps is clearly much shorter. So even in the case of a very simple 2-D gaussian (like the upper peak is), the Langevin jumps provide a lot of improvement in terms of convergence.

Though the Langevin MCMC did a good job at quickly finding and exploring the weirdly shaped peak, it still only managed to locate and explore a single maximum per run. Next we will look at a way to explore both peaks at once.

Parallel Tempering

To explore both peaks at once, we will use Parallel Tempering [99]. In Parallel Tempering, we run several different chains, each at a different "temperature". To define what temperature means here, we can write our target posterior distribution as:

$$\pi(\vec{x}) \propto p(\vec{x})L(\vec{x}) \tag{3.13}$$

where $L(\vec{x})$ is the likelihood, and $p(\vec{x})$ is the prior. Then, we can write the posterior at a particular temperature T as

$$\pi_T(\vec{x}) = p(\vec{x})L(\vec{x})^{1/T}. \tag{3.14}$$

In the case of a constant prior, as we will be using here, we can set the likelihood equal to our target distribution. By running a MCMC routine at a high temperature, the chain should be able to pick out the general locations of all of the local maxima. Then we can develop chains at lower temperatures which will fill out in finer detail the shapes of the maxima. All the chains at different temperatures are run simultaneously, and at certain points jumps between chains are proposed, with inter-chain transition

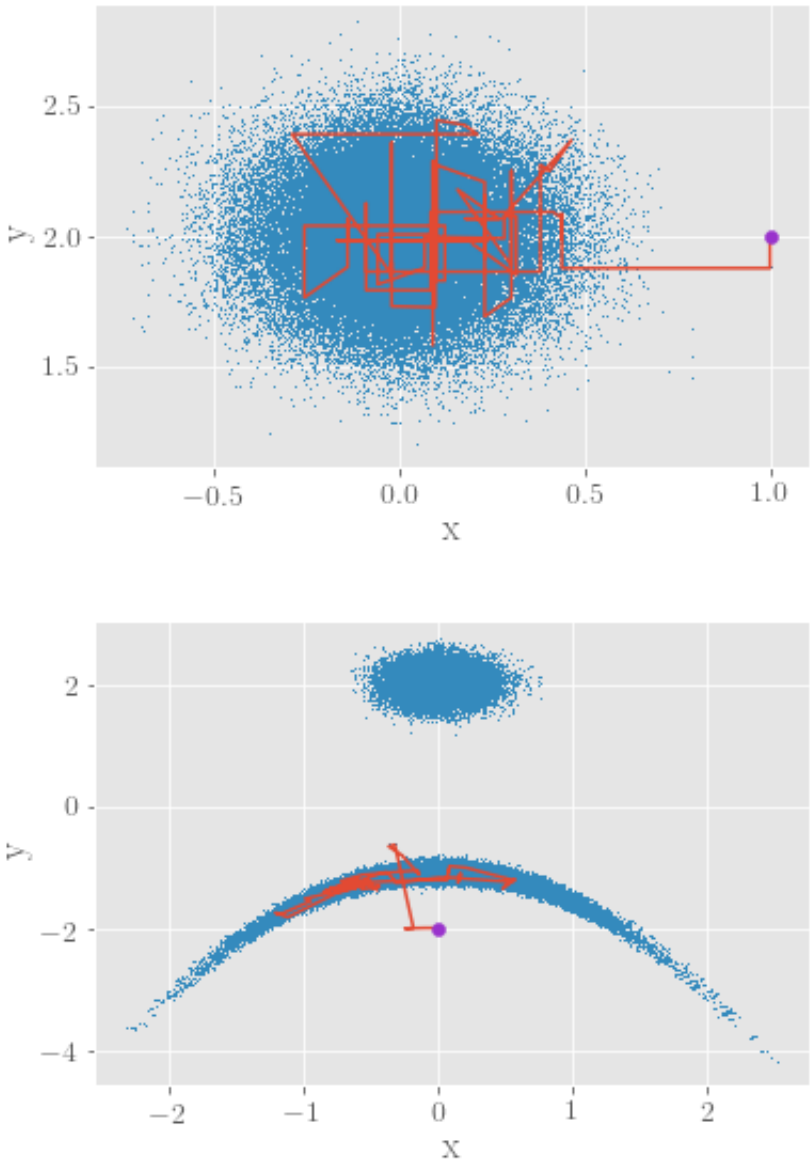

Figure 3.6: Distributions returned by the Langevin MCMC method for starting points $(x_0, y_0) = (1, 2)$ (top) and $(x_0, y_0) = (0, -2)$ (bottom). Also shown is the path taken for the first hundred iterations.

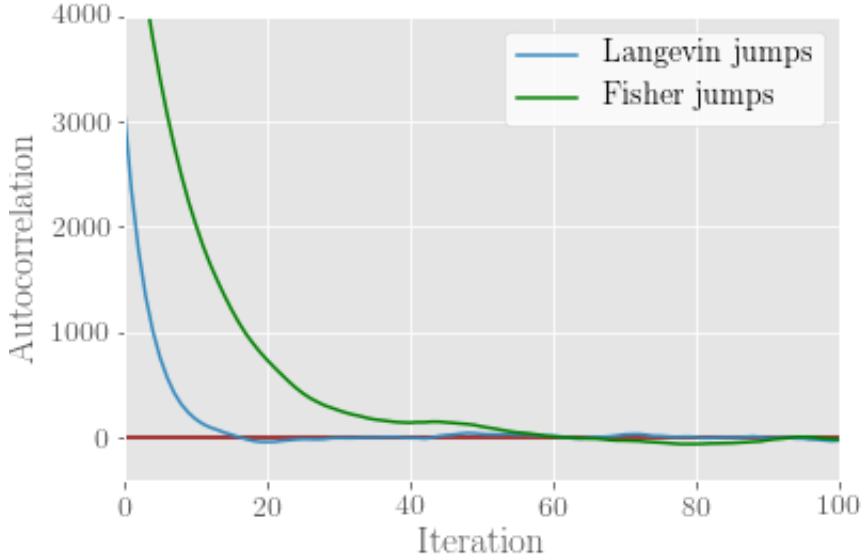

Figure 3.7: Autocorrelations for the parameter x for chains using the local Fisher matrix proposals, and Langevin proposals. Both chains were run for 100000 iterations with starting points $(x_0, y_0) = (1, 2)$. Also shown is a horizontal line a 0.01. We can clearly see the Langevin jumps have a much shorter autocorrelation length.

probability determined by the Hastings ratio:

$$H = \min\left(1, \frac{\pi_{T_i}(\vec{x}_{i+1})\pi_{T_{i+1}}(\vec{x}_i)}{\pi_{T_i}(\vec{x}_i)\pi_{T_{i+1}}(\vec{x}_{i+1})}\right). \tag{3.15}$$

If the swap is accepted, the parameters (in this case the current x and y values) of the chain at temperature T_i get swapped with those at temperature T_{i+1} (not the entire past history, just the current location). Each chain then proceeds until another inter-chain swap is proposed. This swapping means that lower temperature chains which were filling out in finer detail a specific maxima, then jump to the location of another maxima, and fill out the shape of that one. In the end, only the points from the $T = 1$ chain are used in the posterior samples.

Results

Shown in Fig. 3.8 are the cold $T = 1$ chain and a hot chain for the parallel tempering implementation. As we can see in the $T = 1$ case a single chain manages to successfully find both maxima, just as we had hoped. We can also see how the hot chain explores a *much* larger area than the cool chain. This occurs because raising the likelihood (target function) to the power of $1/T$, with $T > 1$, is essentially broadening the peaks. Fig. 3.8 shows how the maxima are broadened to the point that they end up blending together, meaning the chain will no longer be stuck on a single maximum. The interchain swaps then allow the a cooler chain which may be stuck at one peak to jump to the other.

For the run shown in Fig. 3.8, the within temperature jumps used a constant Fisher proposal; a 3:1 ratio of within temperature proposals to inter-chain swap proposals; and 12 different chains run in parallel. One could use Langevin jump proposals for the within temperature jumps, and this would allow the cooler chains to be more efficient in mapping out the finer detail of the peaks. However since the

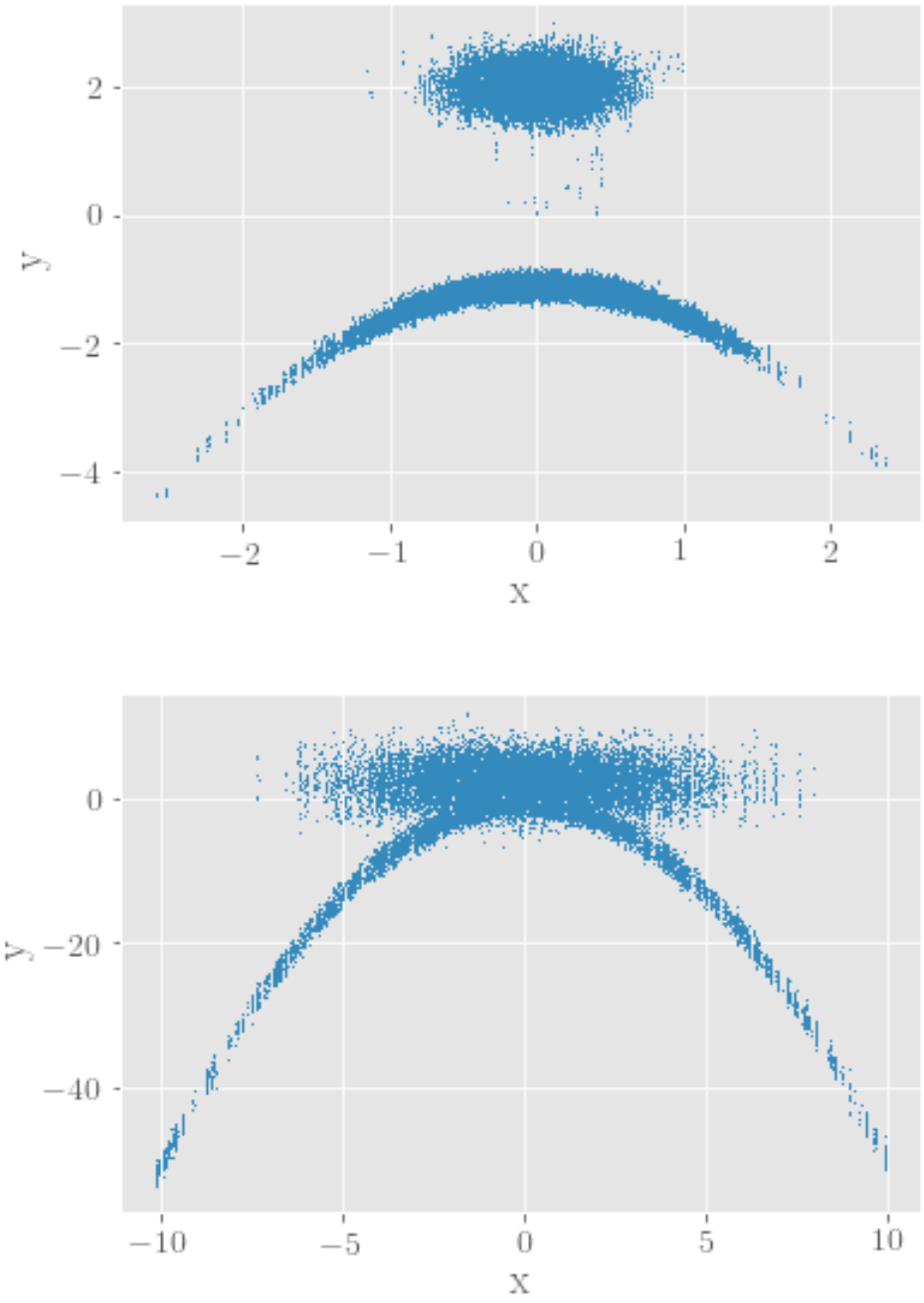

Figure 3.8: Points returned by Parallel Tempering MCMC, for the cold chain (top) and an example hot chain (bottom). The chain was run for 100000 iterations.

Langevin jumps require an evaluation of both the Fisher matrix and the gradient at each location, and we're running twelve different chains, it is somewhat more computationally expensive, and for the sake of this toy problem we will stick with computationally cheap methods.

The inter-chain swap proposals were done by selecting a random temperature, and proposing to switch current chain locations with its hotter neighbor. The temperatures of the chains are geometrically spaced, meaning:

$$T_{i+1} = cT_i$$

where c is a spacing factor greater than one. Choosing appropriate spacing is somewhat tricky. One thing to look at is how the spacing affects the inter-chain swap acceptance rate. Shown in Fig. 3.9 is the inter-chain acceptance ratio for c ranging from 1.2 to 2. Temperature spacings correspond to lower acceptance rates. This result is unsurprising: for very small spacings the likelihood function at adjacent temperatures is very similar, so we expect a high number of the inter-chain swaps to be accepted. Though we want a relatively high rate of acceptances for inter-chain swaps, we also don't want every swap to be accepted so that the cooler chains have time to fully explore and map out the different peaks. The chains shown in Figure 7 have $c = 1.5$, which has an acceptance rate of 82%.

The great triumph of Parallel Tempering is that it manages to find both peaks in a single run, and does so no matter where the chain starts. In Figure 7, $(x_0, y_0) = (0, 0)$ which lies somewhere between the two maxima. However even if the starting point of the chain is very close to either peak, the hotter chains still manage to find both maxima.

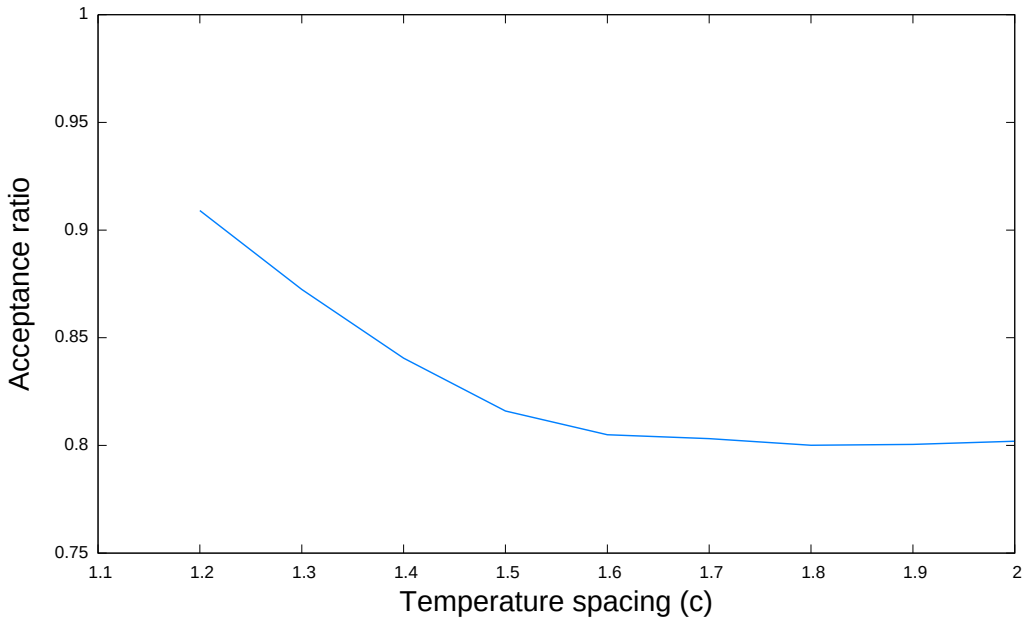

Figure 3.9: The inter-chain swap acceptance ratio, as a function of the temperature spacing.

Other Proposals

In addition to the proposal distributions described here there are other standard jumps used.

One of these is a simple prior draw. In these jumps the new proposal location is chosen from the prior range on the parameters. These jumps can aid in exploring the full parameter space.

Another proposal is a "jiggle proposal", which is simply a slight variation on the current location of the chain. These proposals can help map out the finer details of the parameter space.

Finally, a very useful jump proposal is Differential Evolution (DE). DE uses the history of the chains as a posterior. DE is not strictly Markovian, but with enough iterations it is asymptotically Markovian and so suitable for our needs. For a full

explanation see Chapter 6 of [94].

Transdimensional MCMC and Model Selection

While we have so far looked at a toy problem where we had a single model, in real scientific experiments we may have competing models that we wish to explore and compare. In this case we can use a transdimensional or Reversible Jump MCMC (RJMCMC) [56] in which the model, $\mathcal{H}_i$, is also a parameter. When proposing a jump between models in the Metropolis-Hastings algorithm, the Hastings ratio becomes (see [70])

$$\alpha_{\mathcal{H}_i \to \mathcal{H}_j} = \frac{p(\vec{d}|\vec{\theta}_j, \mathcal{H}_j) p(\vec{\theta}_j, \mathcal{H}_j)}{p(\vec{d}|\vec{\theta}_i, \mathcal{H}_i) p(\vec{\theta}_i, \mathcal{H}_i)} \frac{q(\vec{u}_i, \mathcal{H}_i)}{q(\vec{u}_j, \mathcal{H}_j)} |J_{ij}| \tag{3.16}$$

where $\vec{\theta}_i$ are the parameters of hypothesis $\mathcal{H}_i$, $\vec{u}_i$ is some randomly generated set of numbers used to generate the parameters in $\mathcal{H}_i$, and $|J_{ij}|$ is the determinant of the Jacobian:

$$|J_{ij}| = \left| \frac{\partial(\vec{\theta}_j, \vec{u}_i)}{\partial(\vec{\theta}_i, \vec{u}_j)} \right| \tag{3.17}$$

which defines the coordinate transformation from the i to the j basis. The Jacobian factor is what accounts for a different dimensionality, or volume in parameter space between the two models. In this way we see that the Occam factor naturally falls out of Bayesian inference methods, as it will be harder to jump into models with more dimensions.

In comparing the efficacy of different models, we use a quantity known as the Bayes factor,

$$B_{i,j} = \frac{p(\vec{d}|\mathcal{H}_i)}{p(\vec{d}|\mathcal{H}_j)} \tag{3.18}$$

which we can see is the ratio of the evidences of models $\mathcal{H}_i$ and $\mathcal{H}_j$. $B_{i,j}$ greater (less) than unity means the model $\mathcal{H}_i(\mathcal{H}_j)$ is preferred. Colloquially speaking, the

Bayes factor between model A and model B is the betting odds in favor of model A *assuming equal prior odds between the models.*

As mentioned earlier, calculating the evidence is difficult, and thus calculating the Bayes factor is difficult. There are a number of ways of estimating the evidence, details of which can be found in [70], and one particular example of Bayes factor estimation will be discussed in Chapter 5. Here we will look briefly at the thermodynamic integration method for calculating the evidence, as it is what is used in *BayesWave* when calculating Bayes factors.

Thermodynamic Integration

Parallel tempering not only serves as an excellent way to explore the full parameter space, but can also provide a method to calculate evidence. We do this by thinking of the marginalized likelihood as an analysis to a thermodynamics energy partition function (hence the name "thermodynamic integration") [68]. We write the evidence using Eq. 3.14 as:

$$Z(\beta) = \int p(\vec{\theta}|\mathcal{H}) p(\vec{d}|\vec{\theta}, \mathcal{H})^{\beta} d\theta \tag{3.19}$$

where $\beta = 1/T$ is the inverse of the temperature. Given that the prior $p(\vec{\theta}|\mathcal{H}_i)$ has no β dependence, we can rewrite this as

$$\frac{d}{d\beta} \ln Z(\beta) = \left\langle \ln p(d|\vec{\theta}, \mathcal{H}) \right\rangle_{\beta} \tag{3.20}$$

where $\langle \rangle_{\beta}$ is the expectation value of the β chain's likelihood. Thus the evidence can be calculated by integrating this expression over β

$$p(d|\mathcal{H}) = \int_0^1 \left\langle \ln p(d|\vec{\theta}, \mathcal{H}) \right\rangle_{\beta} d\beta \tag{3.21}$$

and there we have a direct calculation of the evidence, which can now be used to compute Bayes factors. One feature of this approach is that it can be used for RJMCMCs, but also can be used when we simply run two (or more) different MCMCs testing different models.

Bayesian Inference and GW Data Analysis

Before we continue any further we will review the current state of GW data analysis in LIGO.

The starting point for our analysis is the calibrated strain of the LIGO and Virgo detectors [17, 97]. The detector output, call $\vec{d}$, is really made up of two parts: the signal that may or may not be present in the data (call it $\vec{h}$) which depends on some parameters $\vec{\theta}$, and the noise (call it $\vec{n}$) which is always present in the data. So in order to distinguish a signal from the noise, we must have a model for both the signal and the noise. Bayes theorem then explicitly gives the posterior distribution of a GW signal as [45]

$$p(\vec{h}|\vec{d},\vec{\theta},\mathcal{H}_i) = \frac{p(\vec{d}|\vec{h},\vec{\theta},\mathcal{H}_i)p(\vec{h}|\vec{\theta},\mathcal{H}_i)}{p(\vec{d}|\vec{\theta},\mathcal{H}_i)} \tag{3.22}$$

where $\mathcal{H}_i$ is the model hypothesis for the signal (or noise).

The likelihood implemented in Eq. 3.22 is a test of how well the residuals match our noise model when we subtract our signal model from the data. This is effectively a chi-squared minimization, and can be written

$$p(\vec{d}|\vec{h},\vec{\theta},\mathcal{H}_i) = Ae^{-\frac{1}{2}(\vec{d}-\vec{h}(\vec{\theta})|\vec{d}-\vec{h}(\vec{\theta}))^2} \tag{3.23}$$

where A is some overall normalization constant and $(a|b)$ represents the noise weighted

inner product over the power spectral density $S_n(f)$

$$(a|b) = 2\int \frac{\tilde{a}(f)\tilde{b}^*(f) + \tilde{a}^*(f)\tilde{b}(f)}{S_n(f)}df. \tag{3.24}$$

Equation 3.22 will give us a posterior distribution on the GW signal $\vec{h}$. The model for $\vec{h}$ can be constructed in many ways, but LIGO and Virgo data analysis tends to be separated into two broad categories: modeled searches in which we search for GWs from particular sources, such as binary black holes; and unmodeled searches which look for astrophysical signals of any morphology. For modeled searches, we can write the prior on the waveform $\vec{h}$ as

$$p(\vec{h}|\vec{\theta}, \mathcal{H}_i) = \delta(\vec{h} - \vec{h}_i(\vec{\theta})) \tag{3.25}$$

where $\vec{\theta}$ are the parameters of the template, which in the case of searches for compact binary objects includes quantities such as chirp mass, total mass, spins, and sky location and orientation.

On the other hand, we have an unmodeled search where the signal is reconstructed as a sum of N basis functions $w(\vec{\theta})$, where $\vec{\theta}$ are the parameters of the basis function. In this case the prior is

$$p(\vec{h}|\vec{\theta}, \mathcal{H}_i) = \delta(\vec{h} - \sum_a^N w(\vec{\theta_a})). \tag{3.26}$$

As the basis function $w(\vec{\theta})$ approaches the true form of the signal, Eq. 3.26 approaches Eq. 3.25. In this sense we see that GW searches are not actually two distinct categories of modeled and unmodeled, but rather a continuum between the two.

We can also produce posterior distributions not just on the signal itself, but on the underlying parameters of the model. In the templated model example above, we

might more interested in quantities such as the masses of the black holes than just the final waveform. By marginalizing over the waveform samples, we get the expression for the posterior distribution of parameters $\vec{\theta}$:

$$p(\vec{\theta}|\vec{d},\mathcal{H}_i) = \frac{p(\vec{d}|\vec{\theta},\mathcal{H}_i)p(\vec{\theta}|\mathcal{H}_i)}{p(\vec{d}|\mathcal{H}_i)}. \tag{3.27}$$

This is how LIGO and Virgo do their parameter estimation.

The formulations of posterior distributions that we have described above are an example of Bayesian Hierarchical Modeling [69, 77, 90]. The formulation of a posterior on the signal $\vec{h}$ is an example of Level 1 Inference, while the posteriors on the parameters of the model, $\vec{\theta}$, is Level 2 Inference. If we were to move onto model selection, that would be an example of Level 3 Inference. Details of Bayesian Hierarchical Modeling can be found in Chapter 5 of [90], and a forthcoming study of how BHM is used in GW data analysis can be found is in progress [45].

It is also useful to take this opportunity to introduce a couple quantities that are used frequently in GW data analysis, and will be used throughout this thesis.

First is the match, also sometimes referred to as the overlap. This is a measure of how alike two waveforms are and is defined by

$$M(h,\bar{h}) = \frac{(h|\bar{h})}{\sqrt{(h|h)(\bar{h}|\bar{h})}}. \tag{3.28}$$

The match ranges from -1 for perfectly out of phase waveforms, to 1 for perfectly matched waveforms. We sometimes also use the maximized match, which marginalizes over any phase or time shifts between the two waveforms being compared.

Next we define the signal-to-noise ratio (SNR). The definition of SNR use can vary for different GW searches. In burst searches, and as we will use for most of this

Thesis is

$$\mathrm{SNR}^2 = (h|h). \tag{3.29}$$

This is sometimes referred to as the optimal SNR. Another definition of SNR sometimes used in GW data analysis is

$$\mathrm{SNR}^2 = \frac{(d|h)^2}{(h|h)} \tag{3.30}$$

where d is the data, and h is the GW model.

As a final note that throughout this Thesis we will sometimes refer to "whitened" data. This refers to the technique in which the frequency domain data is divided by the power spectral density, i.e.

$$d(f)_{\mathrm{white}} = \frac{d(f)}{\sqrt{S_n(f)}}. \tag{3.31}$$

This has the affect that whitened data, in the absence of a signal, will have a Gaussian distribution. We can think of the whitened data as the data that the detector "sees".

In the following Chapters we will discuss one particular unmodeled search algorithm and its implementation in LIGO data analysis. We will then look at some developments where we move this search along the unmodeled to modeled continuum and the advantages that can bring.

BAYESWAVE

One of the flagship data analysis pipelines used by the LIGO Scientific Collaboration (LSC) in the first two observing runs is the *BayesWave* algorithm [44]. *BayesWave* uses a wavelet decomposition and Bayesian inference methods to reconstruct GW signals, and distinguish between real astrophysical signals and instrumental noise. The standout feature of *BayesWave* is that, along with its companion algorithm *BayesLine* [72], it simultaneously models both the signal and the noise.

BayesWave has been under development of some form since Cornish and Montana State University first joined the LSC in 2007 [41], and the groundwork by Cornish and Littenberg for what eventually became *BayesWave* can be found in [71] and [43]. The *BayesWave* algorithm was further developed by Baker in his thesis [29]. Up to this point, the *BayesWave* algorithm used a basis of discrete orthogonal wavelets to reconstruct signals. Cornish and Littenberg then updated the algorithm to use a frame[1] of continuous sine-Gaussian (or Morlet-Gabor) wavelets [44]. The continuous wavelets have a number of advantageous features. For one, they have a dynamic shape in time-frequency space, allowing them to easily reconstruct a wide range of signals. Second, with these wavelets the entire analysis can be done in the Fourier domain, which is more convenient because the likelihood is most efficiently computed in the frequency domain. Finally, projecting the wavelets onto the active detectors is much easier with sine-Gaussian wavelets as it only requires a phase shift. This final point is the primary advantage that the continuous wavelets have over the discrete wavelets.

[1] A frame here is just a basis that is over complete

Morlet-Gabor Wavelets

Before we go much further into the description of the *BayesWave* algorithm, we will discuss our frame function. The sine-Gaussian wavelets take the following analytic form in the time domain:

$$\Psi(t; t_0, f_0, Q, A, \phi_0) = Ae^{-(t-t_0)^2/\tau^2} \cos(2\pi f_0(t - t_0) + \phi_0), \tag{4.1}$$

and in the Fourier domain:

$$\Psi(f; t_0, f_0, Q, A, \phi_0) = \frac{\sqrt{\pi}A\tau}{2} e^{-\pi^2\tau^2(f-f0)^2} e^{-2\pi i f t_0} \left(e^{i\phi_0} + e^{-i\phi_0} e^{-Q^2 f/f_0} \right). \tag{4.2}$$

For both expressions $\tau = Q/(2\pi f_0)$. The analytic expression in the frequency domain is one of the benefits of the sine-Gaussian wavelets as it allows us to do all of our analysis in the frequency domain. The wavelets (in either representation) are uniquely determined by five parameters: t_0 which gives the central time of the wavelet, f_0 which gives the central frequency, Q which is the quality factor (or how localized in time the wavelet is), A which is the amplitude, and ϕ_0 which is the overall phase offset. The wavelets can be represented in time-frequency space as ellipses with a width in time given by $\Delta t = \sqrt{2}\tau$ in time and $\Delta f = \sqrt{2}/(\pi\tau)$ in frequency. Note that $\pi\Delta t\Delta f = 1$ meaning the wavelets occupy the minimum volume in time-frequency space allowed by the Heisenberg-Gabor uncertainty principle.

BayesWave reconstructs detector data as a sum of these wavelets. Both the intrinsic parameters of the wavelets as well as the number of wavelets used are determined through a transdimensional (or reversible jump) Markov chain Monte Carlo. This wavelet MCMC is run in two different models: the glitch model and the signal model, as well as an additional model where we assume the detector data is

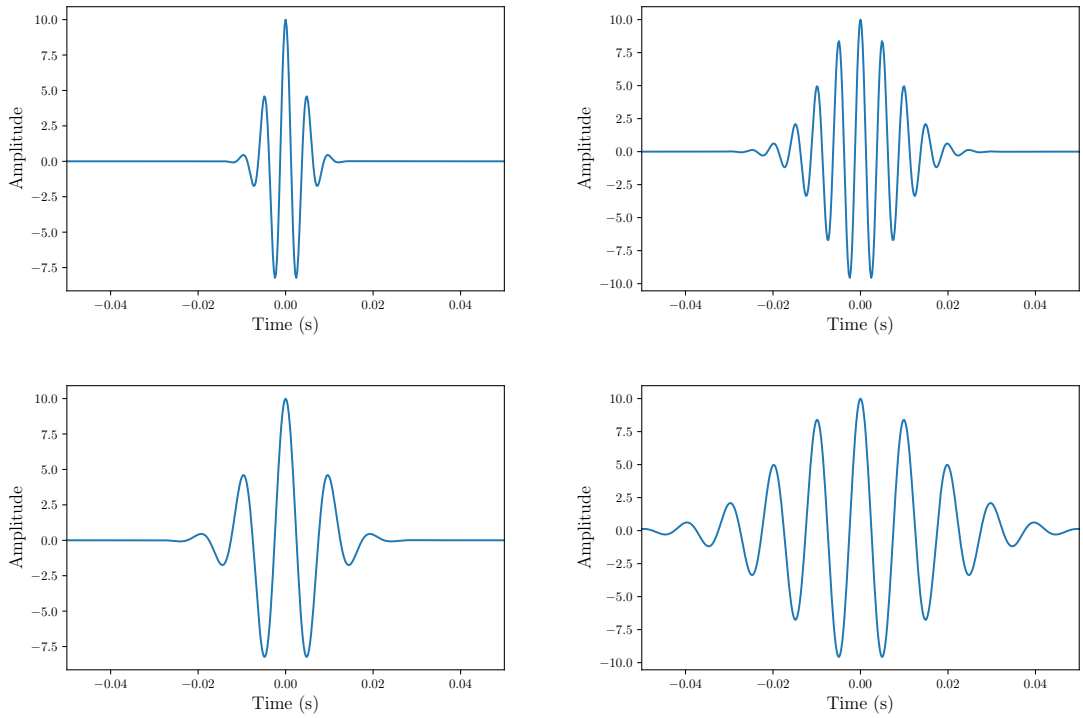

Figure 4.1: Examples of a Morlet-Gabor wavelet in the time domain, for a a selection of f_0's and Q's. Top row: $f_0 = 200$Hz. Bottom row: $f_0 = 200$Hz. Left column: $Q = 7$. Right column: $Q = 15$. For all examples, $t_0 = 0$s, $A = 10$, $\phi_0 = 0$.

Gaussian noise only. We now describe the details of the *BayesWave* analysis in these different models.

Glitch Model

When running in the glitch mode, *BayesWave* reconstructs the data independently in each detector. The strain for the glitch model in detector I is then given by

$$h^I(f) = \sum_{i=0}^{N_w^I} \Psi(f; \vec{\lambda}_i^I), \tag{4.3}$$

where N_w is the number of wavelets used, and $\vec{\lambda}$ represents the set of intrinsic parameters (t_0, f_0, Q, A, ϕ_0). We now describe the priors and proposals on our parameters N_w and $\vec{\lambda}$ used in the glitch model RJMCMC.

Priors

The prior probability distributions for t_0 and f_0 are uniform over the time-frequency volume being analyzed. This is typically on the range of 1s, and 1kHz. The priors on the phase ϕ_0 is uniform over the entire range $[0, 2\pi]$. The prior on Q is also uniform, and in the O1 and O2 analysis was in the range $[0.01, 40]$. The specifics of the prior ranges used in the actual LIGO data analysis will be described in Chapter 6, where we discuss the pipeline implementation.

The prior probability distribution of the amplitude of the wavelets A is a bit more complicated than the rest, as it is really a prior on the SNR of the wavelet. Because we know that in LIGO data quiet glitches are more common than loud glitches, we wish to construct a prior that reflects that. We chose a prior of the form

$$p(\mathrm{SNR}) = \frac{\mathrm{SNR}}{2\mathrm{SNR}_*^2(1 + \frac{\mathrm{SNR}}{2\mathrm{SNR}_*})^3} \tag{4.4}$$

where SNR_* is a chosen SNR at which the prior peaks, and the SNR of a sine-Gaussian wavelet can be estimated as

$$\mathrm{SNR}^2 \simeq \frac{A^2 Q}{2\sqrt{2\pi} f_0 S_n(f_0)} \tag{4.5}$$

where $S_n(f_0)$ is the noise power spectral density at the central frequency of the wavelet f_0. This prior function is shown graphically in Fig 4.2. We typically choose $\mathrm{SNR}_* = 5$, but again details can be found in Chapter 6. We also note that Eq. 4.4 is slightly different from the original glitch prior that is proposed in [44]. We found that in

situations of unusually loud glitches the signal model was heavily favored. This misclassification was fixed by implementing this new prior which has larger tails in higher SNR regions.

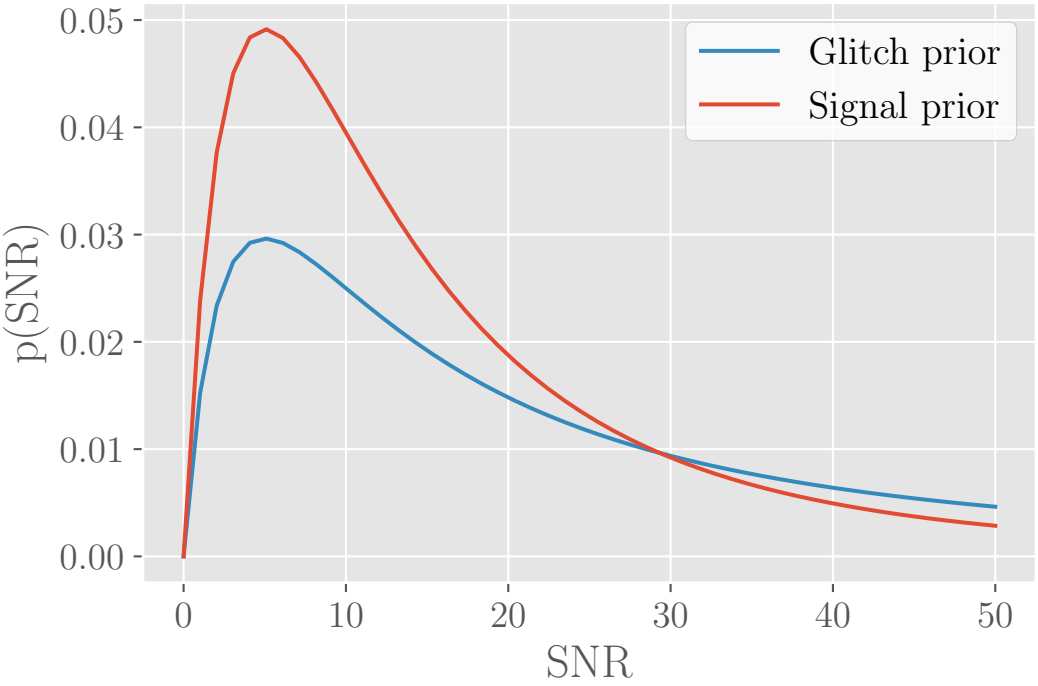

Figure 4.2: The SNR (amplitude) prior on the individual wavelets for the signal and glitch models.

Finally, we must also place a prior on the number of wavelets, N_w. In the original *BayesWave* development a uniform prior was used, and in O1 we used $N_w \in [0, 20]$ for the glitch model. In O2 this prior was updated using the information gained from the O1 analysis. Our new prior is based off of an analytic fit to the distribution of N_w for the 500 most significant background triggers in our data set, and is given by:

$$p(N_w) = \frac{N_w 4\sqrt{3}}{2\pi b^2 (3 + N_w/b)^4} \tag{4.6}$$

where $b = 2.9$ and has been empirically determined by the data. This distribution is shown in Fig 4.3. Similar to the updated glitch amplitude prior, this new prior helps in cases where there is a glitch with an unusually large SNR. These extremely loud glitches would sometimes saturate the constant wavelet prior, leading to large Bayes factors in favor of the signal model[2]. The new prior penalizes large number of wavelets, while still allowing the model to use that many wavelets if it really does fit the data better.

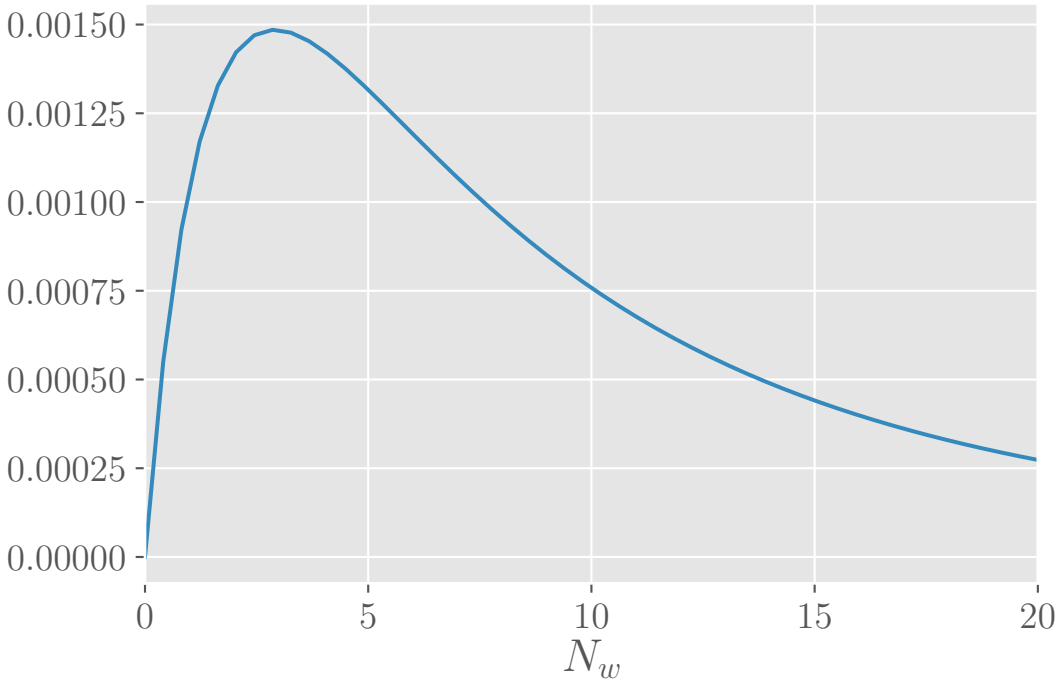

Figure 4.3: The prior on the number of wavelets that was implemented in O2.

[2]The signal to glitch Bayes factor is highly dependent on the number of wavelets used. This will be discussed in depth in Chapter 5.

Proposals

In the glitch model MCMC there are two classes of jump proposals: transdimensional jumps in which we propose to either add or remove a wavelet, and wavelet update proposals in which we propose to change the parameters of the current active wavelets.

Wavelet Updates In the wavelet update moves, we select at random one single wavelet to update at a time. There are two jump proposals used for single wavelet updates:

- **Prior draws**. These are straight forward jump proposals, in which the proposed new parameters are drawn directly from the prior probability distributions described above. The prior draw proposals can be thought of as "big" jumps, where the proposed parameters may be in a very different region of parameter space than the current parameters.

- **Fisher jumps**. In Chapter. 3 we described jump proposals based off of the Fisher matrix, which are commonly used in MCMC algorithms. For the wavelet jumps in *BayesWave*, we use jumps of the form described in Eq. 3.6. For a sine-Gaussian wavelet with parameters $t_0, f_0, Q, \ln A, \phi_0$, the Fisher matrix can be estimated[3] as:

$$\Gamma = \mathrm{SNR}^2 \begin{pmatrix} \frac{4\pi^2 f_0^2(1+Q^2)}{Q^2} & 0 & 0 & 0 & -2\pi f_0 \\ 0 & \frac{3+Q^2}{4f_0^2} & -\frac{3}{4Qf_0} & -\frac{1}{2f_0} & 0 \\ 0 & -\frac{3}{4Qf_0} & \frac{3}{4Q^2} & \frac{1}{2Q} & 0 \\ 0 & -\frac{1}{2f_0} & \frac{1}{2Q} & 1 & 0 \\ -2\pi f_0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.7}$$

[3]We note that this Fisher matrix estimation ignores correlations between neighboring wavelets.

where SNR is given in Eq. 4.5. As in Chapter 3, we propose jumps along the eigendirections scaled by the eigenvectors. The Fisher matrix proposals can be thought of as "small" jumps, where the proposed parameters are only slight deviations from the existing parameters.

Transdimensional Moves In the transdimensional moves, we propose to either add a new wavelet (birth move), or remove an existing wavelet (death move). For the death moves, an active wavelet is chosen at random, and removed from the waveform reconstruction.

For the birth moves, a new wavelet with a new set of intrinsic parameters is chosen. These parameters are either chosen from the prior ranges, as is done for the wavelet update moves described above, or are chosen from a time-frequency density proposal based off of the interferometer data. This second proposal was introduced in between O1 and O2. To make this proposal we use Q-scans– a type of spectrogram– of the data. The mechanics of our Q-scans are described in Appendix C, but the general idea is using a sine-Gaussian wavelet transform to map out the power in time-frequency space. We perform these Q-scans on the interferometer data, using sine-Gaussians as the wavelet transform basis over a range of Q's. An example of Q-scans at various resolutions for GW150914 are shown in Fig. 4.4. This gives us a probability density distribution in t_0-f_0-Q space, which we can then use as a proposal by using rejection sampling. This proposal distribution is designed to propose wavelets in regions of time-frequency space with higher power. This helps the chains to converge more quickly.

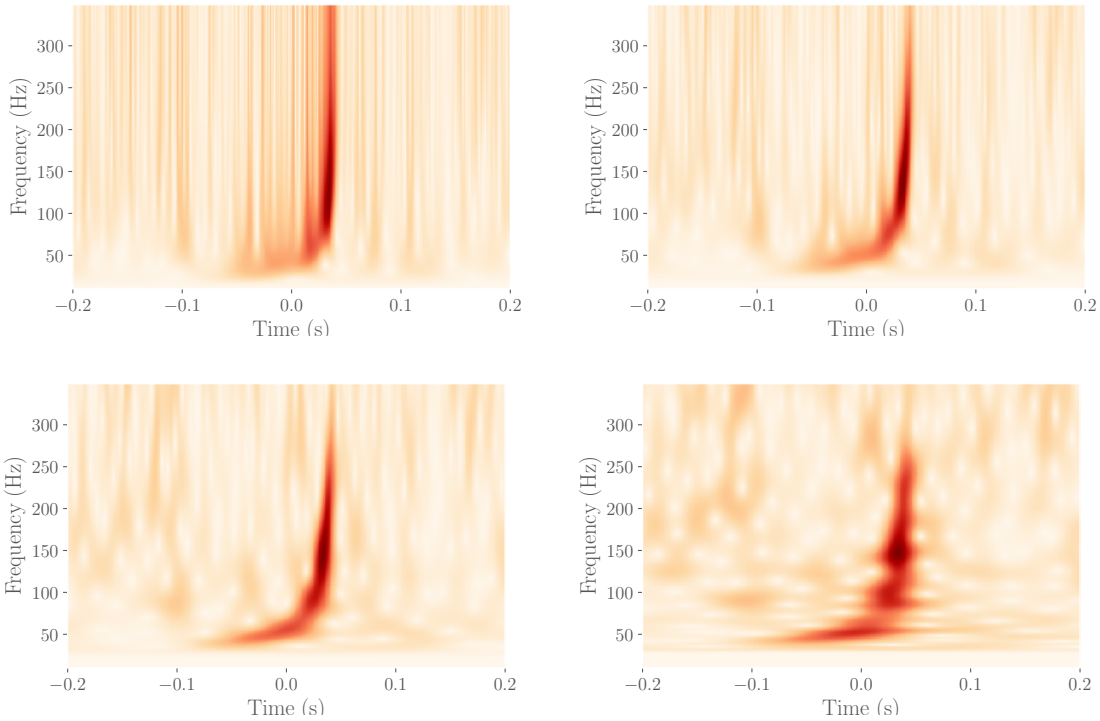

Figure 4.4: Example Q-scans of the whitened data in the Hanford detector for GW150914, at various Q resolutions. Top left: $Q = 2$; Top right: $Q = 4$; Bottom left: $Q = 8$; Bottom right: $Q = 16$.

Signal Model

The signal model differs from the glitch model in that there is only a single set of wavelets, plus four additional external parameters that give the sky location and orientation of GW source, which are used to forward model the detector output. These extrinsic parameters are the sky location (θ,ϕ), ellipticity (ϵ), and polarization angle (ψ). We will collectively refer to the extrinsic parameters as $\vec{\Omega}$.

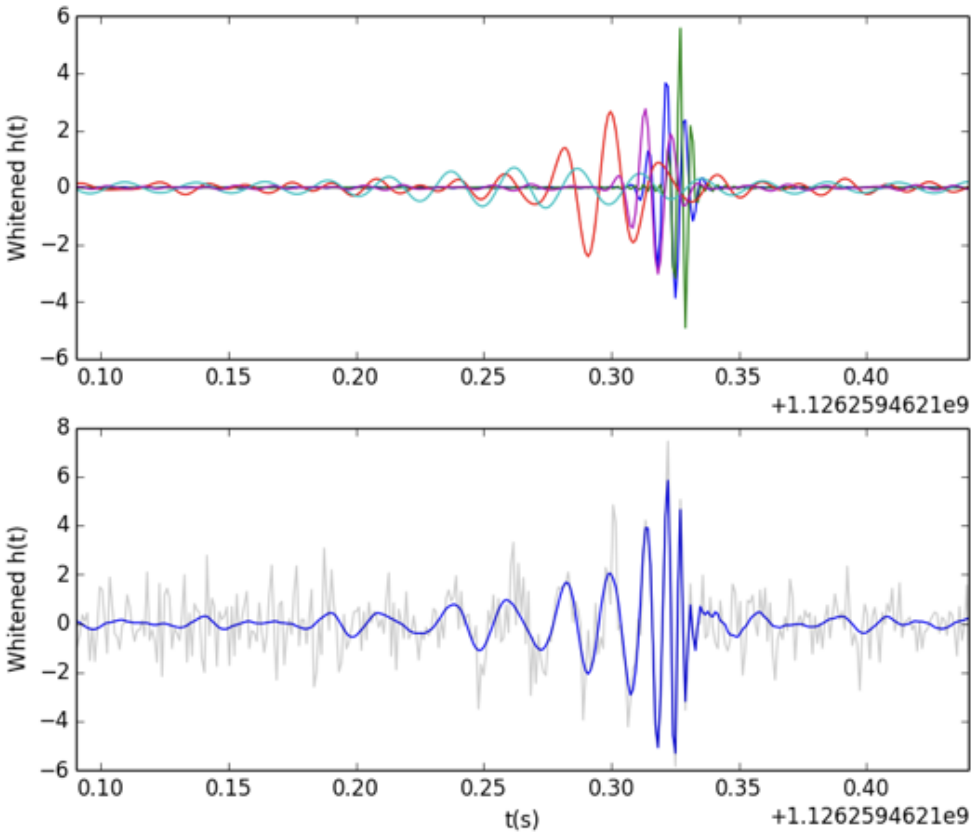

Figure 4.5: An example of the reconstruction of GW150914 using wavelets. The top panel shows the individual wavelets (plotted in various colors) from own draw of the chain, and the full waveform reconstruction in the bottom panel, over laid on the detector data from the Hanford detector.

The strain in each detector, I, is then given by

$$h^I(f;\vec{\lambda},\vec{\Omega}) = (F^{+,I}h_+(f) + F^{\times,I}h_\times(f))e^{2\pi i f\Delta t_I} \tag{4.8}$$

where Δt_I is the time delay between the detectors, which is a function of (θ,ϕ), and $F^+/F^\times$ are the antenna patters discussed in Chapter 2. h_+ and $h_\times$ are given by

$$h_+ = \textstyle\sum_i^{N_w} \Psi(f;\vec{\lambda}) \tag{4.9}$$

$$h_\times = \epsilon h_+(f)e^{i\pi/2}, \tag{4.10}$$

meaning that in the current implementation of *BayesWave* we assume an elliptical polarization. This ranges from linearly polarized as ϵ approaches 0, to circularly polarized for $\epsilon = 1$. Future developments plan to implement independent polarizations in each detector.

Most of the priors and jump proposals for wavelet updates and birth/death moves in the signal model are the same for the glitch model. Here we highlight the priors and proposals that are unique to the signal model, including those for the extrinsic parameters.

Priors

For the intrinsic parameters, the priors on t_0, f_0, Q, and ϕ_0 are uniform as they are for the glitch model. The prior on the wavelet amplitude however is slightly different than for the glitch model. The signal amplitude prior is again really a prior on the wavelet SNR, given by

$$p(\mathrm{SNR}) = \frac{3\mathrm{SNR}}{4\mathrm{SNR}_*^2(1+\frac{\mathrm{SNR}}{4\mathrm{SNR}_*})^5}, \tag{4.11}$$

where again SNR_* is some chosen peak SNR. This prior was designed to fall off as SNR^{-4} for high SNRs, as this is what we expect astrophysically: the SNR of a signal is inversely proportional to the distance to the signal, and we expect GW sources to be distributed uniformly in volume.

The priors on the extrinsic parameters are uniform over their ranges. The prior on the sky location is uniform over the sphere, so $\theta \in [0, 2\pi]$ and $\sin\phi \in [0, 1]$. The prior on the polarization angle is uniform over $\psi \in [0, \pi]$, and the ellipticity is uniform over $\epsilon \in [0, 1]$.

Proposals

Again, most of the proposals for the wavelet moves are the same in the signal model as in the glitch model, albeit for a single set of wavelets rather than a set for each detector. The proposals for the extrinsic parameters fall into three different categories:

- **Prior draws**: Similar to the prior draw proposals used for the intrinsic wavelet parameters, we simply draw from the priors on the extrinsic parameters in these jump proposals. These are generally thought of as "big" jumps as they often propose a totally new location in the sky.

- **Fisher jumps**: Again we employ jumps of the form of Eq. 3.6. In this case the Fisher matrix is derived from the reconstructed waveform at geocenter.

- **Sky ring proposals**: These jump proposals are based off of the fact that the delay in the times of arrival of a signal between two detectors uniquely defines a ring in the sky of directions from which the GW could have come. By using the current estimation of the time delay between the detectors, we propose jumps that rotate the sky location parameters along this ring.

By setting the extrinsic parameters as variable parameters in our MCMC, we also get posterior distributions on the sky location of the GW signal. These posteriors can be turned into probability density sky maps, which could in turn be used to help search for an electromagnetic counterpart to the GW signal.

In O2, we added a modification to *BayesWave* where one could set the sky location *a priori*. This is useful in the case where we have already observed an electromagnetic counterpart to a GW signal (as was the case for GW170817).

Noise Modeling

In addition to modeling the transient features in GW detector data, the *BayesWave* analysis also models the noise spectrum of the detectors. This is done using *BayesWave*'s partner noise modeling algorithm *BayesLine* [72]. *BayesLine* is a fully Bayesian RJMCMC algorithm that models the detector noise as a cubic spline and Lorentzian lines. The cubic spline models the broad-band features of the spectrum, where the number of control points used for the spline is a parameter of the RJMCMC. The Lorentzians are used to model the narrowband features that arise due to resonances from the power supply, suspensions, and other sources. The number of Lorentzians used is also a parameter of the RJMCMC. An example of how the spline and Lorentzians combine to give a noise curve is shown in Fig. 4.6.

One of the unique benefits of the *BayesLine* approach to noise modeling is that we get a full Bayesian posterior of noise curves, and actually marginalize over our noise estimates. This is an improvement over the point estimate PSDs typically used in LIGO data analysis.

In addition to calculating the evidence for the signal and glitch models, *BayesWave* calculates the evidence assuming the data contains only noise. This is then used to calculate the signal-to-noise Bayes factors, $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$.

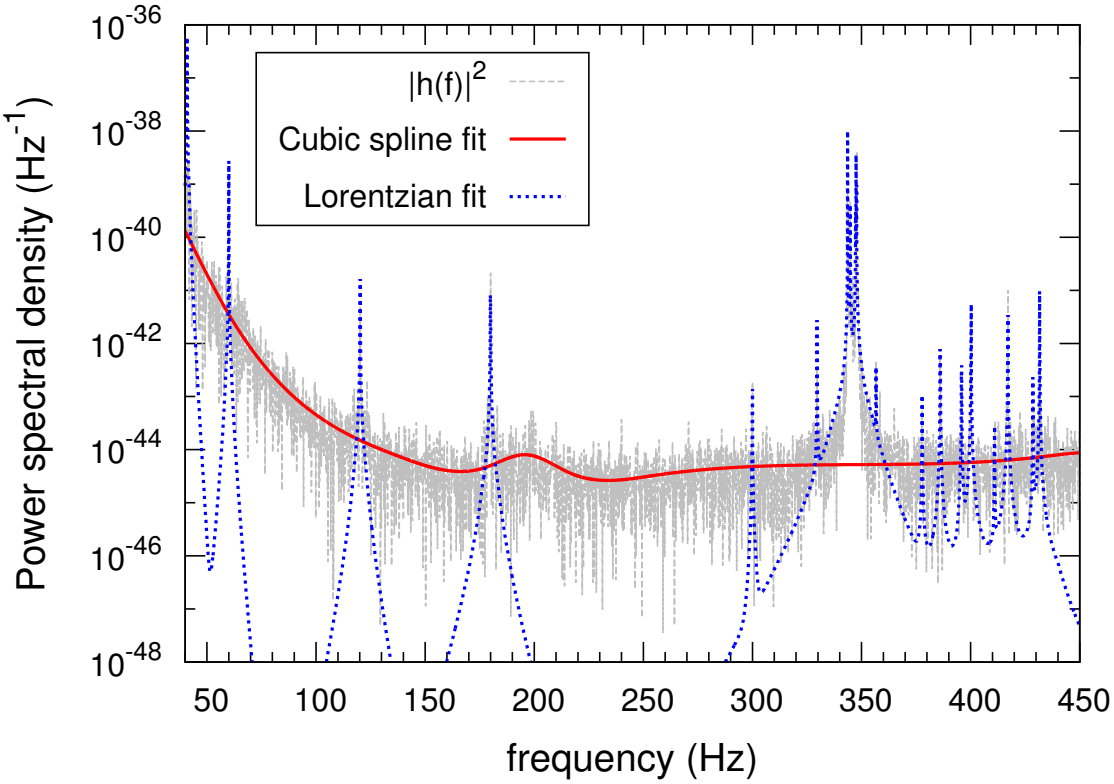

Figure 4.6: An example of the spline and Lorentzian used to reconstruct LIGO noise. Figure from [72].

Parallel Tempering

Finally we note that our RJMCMC implementation includes parallel tempering, as described in Chapter 3. Parallel tempering aids in the convergence of chains, and is also how we calculate the evidence of each model via thermodynamic integration. Once the evidence has be calculated, we can compute the Bayes factors between models. Thermodynamic integration is crucial to our analysis since in the current implementation of *BayesWave*, each model (signal, glitch, noise) is run independently, so we rely on thermodynamic integration to calculate the Bayes factors. Future developments of the algorithm will include an option to run the signal and glitch models simultaneously, to fit for coherent and incoherent power at once.

In the next chapter we examine these Bayes factors in detail.

UNDERSTANDING BAYES FACTORS

Enabling High Confidence Detections of GW Bursts

In the world of unmodeled searches in LIGO data, the biggest challenge is confidently distinguishing real astrophysical signals from instrumental noise. GW signals are very weak, with amplitudes comparable to the amplitudes of white noise in the detectors. Furthermore, detector data frequently contains non-gaussian transient noise events known as *glitches*. Here we focus primarily on separating signals and glitches.

Glitches

The Advanced LIGO interferometers are without a doubt an incredible feat of engineering. During the first Observing run, aLIGO reached a strain sensitivity of $10^{-23}/\sqrt{\text{Hz}}$ at 100 Hz [9]. This sensitivity also means that interactions between the detectors and their environments can lead to transient noise events or "glitches" in the strain channel.

Glitches can pose a significant challenge in searches for short-duration GW events, especially so in the unmodeled searches. One risk is that a glitch could happen simultaneously with a GW event, obscuring the signal and making it difficult to confidently claim a detection. This was the case for GW170817, the first detected binary neutron star merger, in which there was an extremely loud glitch in the Livingston detector. *BayesWave* has recently been employed in removing glitches like the one in the GW170817 signal [21]. Another, more common challenge, is that glitches could occur simultaneously in multiple detectors potentially mimicking a real GW signal. Burst search pipelines are particularly challenged by simultaneous glitches since even though glitches are relatively common occurrences in the detectors,

glitches that look like compact binary mergers are essentially non-existent and thus the compact binary searches do not have the same challenges.

Mitigation of the effects of glitches can be done through two main approaches: either getting rid of the glitch itself, or developing pipelines which are (relatively) robust against the presence of glitches.

For the first approach, there are entire working groups within the LIGO Scientific Collaboration dedicated to removing glitches from detector data. By monitoring over a hundred thousand auxiliary channels in the detectors, significant progress has been made in mitigating the effects of glitches in searches for GWs, both by identifying causes of glitches in order to prevent them, and introducing *data quality vetoes* (which will be further described in the next Chapter) to be used when there is well understood problematic behavior in the detectors. While in a perfect world these noise-hunting scientists would be able to identify and prevent all possible glitches, that is unfortunately not a realistic goal. Additionally, while the data quality vetoes are helpful in identifying problematic stretches of data, we do not want to rely on the vetoes too much or we risk significantly decreasing the amount of analyzable data.

And so we must develop data analysis pipelines that can confidently distinguish between signals and glitches. As mentioned in Chapter 5, *BayesWave* provides Bayes factors between the signal and glitch models by comparing their Bayesian evidences. Here we dive deeper into understanding exactly what those Bayes factors mean, and why *BayesWave*'s detection statistic is a significant improvement upon previous burst searches in LIGO.

Bayesian Hypothesis Testing

As a brief reminder, *BayesWave* runs in two different modes: the signal mode in which there is a single set of wavelets plus a common set of external parameters

which is forward modeled into detector output; and the glitch model in which there is an independent set of wavelets used to reconstruct the data in each detector.

For each model the evidence is calculated. The evidence, or marginalized likelihood is given by

$$p(d|\mathcal{H}) = \int p(d|\vec{\theta},\mathcal{H})p(\vec{\theta})d\vec{\theta} \tag{5.1}$$

where again $\mathcal{H}$ is the hypothesis, d is the data, and $\vec{\theta}$ is the parameters of the model.

The evidence can then be compared via the odds ratio:

$$\mathcal{O}_{A,B} = \frac{p(\mathcal{H}_A)}{p(\mathcal{H}_B)}\frac{p(d|\mathcal{H}_A)}{p(d|\mathcal{H}_B)} = \frac{p(\mathcal{H}_A)}{p(\mathcal{H}_B)}\mathcal{B}_{A,B} \tag{5.2}$$

where $p(\mathcal{H}_i)$ is prior probability that hypothesis i, and $\mathcal{B}_{A,B}$ is the *Bayes factor*. In other words, the Bayes factor is the betting odds between models A and B, *assuming equal priors*.

Laplace Approximation for Evidence

The likelihood is a measure of how consistent the data is with the noise model, once the signal model has been subtracted. Explicitly this likelihood is

$$p(d|\vec{\theta},\mathcal{H}) \propto \prod_{I}^{N_I} \exp\left(-\frac{1}{2}\left(d^I - h^I(\vec{\theta}|\mathcal{H})|d^I - h^I(\vec{\theta}|\mathcal{H})\right)\right) \tag{5.3}$$

with I running through the number of interferometers N_I.

For a signal with sufficiently large SNR the likelihood will be sharply peaked around the true values of the parameters $\vec{\theta} = \vec{\theta}_{max}$, meaning the integrand Eq. 5.1 will be sharply peaked. This integral can then be evaluated using the Laplace approximation, giving an estimate of the evidence as

$$p(d|\mathcal{H}) \approx \sqrt{\det C}\; p(d|\vec{\theta}_{max},\mathcal{H})p(\vec{\theta}_{max}|\mathcal{H})(2\pi)^{D/2}. \tag{5.4}$$

Here D is the dimensionality of the model and det$C$ is the determinate of the covariance matrix, which can be estimated as the inverse of the Fisher matrix.

From Eq. 5.4 we can gain some insight into how the evidence is affected by the size of our parameter space. The combination of the terms $\sqrt{\det C}(2\pi)^{D/2}$ gives the volume of the posterior, which we will call ΔV. If we use uniform priors on all of the model parameters, the prior can be written $p(\vec{\theta}_{max}, \mathcal{H}) = 1/V_{\rm prior}$, then rewriting Eq. 5.4 as

$$p(d|\mathcal{H}) \approx p(d|\vec{\theta}_{max}, \mathcal{H}) \frac{\Delta V}{V_{\rm prior}}. \tag{5.5}$$

The second term in Eq. 5.5 is simply the ratio of the posterior volume to the prior volume, or what we recognize as the *Occam factor*. The Occam factor is a measure of how much of the prior volume is taken up by the posterior, and can penalize the evidence in the case of a model with too many (or too tightly constrained parameters). However it may also be that adding one or more parameters will fit the data much better, thus significantly increasing the likelihood. In this case we might actually prefer the higher dimension model.

Returning to our approximation of the evidence, we can further estimate he likelihood when evaluated at the parameters $\vec{\theta}_{max}$ as [92]

$$p(d|\vec{\theta}_{max}, \mathcal{H}) \approx {\rm SNR}^2/2 + D/2. \tag{5.6}$$

Using the estimated Fisher matrix of a Morlet-Gabor wavelet from Chapter 4, we can replace the determinant of the covariance matrix with

$$\frac{1}{\sqrt{\det C}} = \sqrt{\det \Gamma} = \frac{\pi {\rm SNR}^5}{\sqrt{2} Q}. \tag{5.7}$$

While Eq. 5.7 is for a single wavelet, we wish to extend this to multiple wavelets as

BayesWave typically uses more than one at a time. We first test that within a single chain step the wavelets have little overlap in parameter space. Then the $\sqrt{\det C}$ term is simply a product of the determinants of all N individual wavelet covariance matrix:

$$\sqrt{\det C} \approx \prod_{i}^{N} \frac{\sqrt{2}Q_i}{\pi \mathrm{SNR}_i^5}. \tag{5.8}$$

Our final assumption is that the priors of all wavelet parameters are uniform. Typically all wavelet parameter priors are indeed uniform, except for the prior on the wavelet amplitude, but for now we examine only a simple case with all uniform priors.

Combining all these elements we now have an estimate of the evidence as:

$$\ln p(d|\mathcal{H}) \approx \frac{\mathrm{SNR}^2}{2} + \frac{5N}{2} - N\ln(V) + \sum_{i}^{N} \ln \frac{\bar{Q}_i}{\mathrm{SNR}_i^5} \tag{5.9}$$

where we have utilized the fact that the total number of dimensions is 5 times the number of wavelets (N), and for simplicity defined $\bar{Q} \equiv (2\pi)^{5/2} \frac{\sqrt{2}Q_n}{\pi}$.

While Eq. 5.9 is a general expression of the likelihood for a set of wavelets, our goal here is to compare the evidences between the signal ($\mathcal{S}$) and glitch ($\mathcal{G}$) models. For the glitch model, as stated earlier, there is an independent set of wavelets used to reconstruct the data in each interferometer independently. Thus the log evidence can be found by summing over the wavelets in all detectors (N_{IFO}):

$$\ln p(d|\mathcal{G}) \approx \frac{\mathrm{SNR}^2_{\mathrm{NET}}}{2} + \sum_{n}^{N_{\mathrm{IFO}}} \frac{5N_n^{\mathcal{G}}}{2} - N\ln(V) + \sum_{n}^{N_{\mathrm{IFO}}} \sum_{i}^{N^{\mathcal{G}}} \ln \frac{\bar{Q}^{\mathcal{G}}_{i,n}}{\mathrm{SNR}^5_{i,n}}. \tag{5.10}$$

The signal model is a bit more complicated, as the signal model uses only one set of wavelets, but has a common set of extrinsic parameters used to forward model the detector data. For clarity we will denote the common extrinsic parameters with

the subscript Ω, and intrinsic wavelet parameters with the subscript λ. Similar to the Occam factor described earlier, there will be an Occam factor for the extrinsic parameters of $\sqrt{\det C_\Omega}/V_\Omega$, and from the likelihood an additional term of $D_\Omega/2$. In the particular case of *BayesWave*, the number of extrinsic parameters is $D_\Omega = 4$, and the prior volume is $V_\Omega = 4\pi^2$. So the estimated evidence for the signal model is

$$\ln p(d|\mathcal{S}) \approx \frac{\mathrm{SNR}^2_{\mathrm{NET}}}{2} + \frac{5N^{\mathcal{S}}}{2} - N^{\mathcal{S}} \ln(V) + \sum_i^{N^{\mathcal{S}}} \frac{\bar{Q}_i^{\mathcal{S}}}{\mathrm{SNR}^5_{\mathrm{NET},i}} + \left[2 + \ln \frac{\sqrt{\det C_\Omega}}{4\pi^2}\right]. \quad (5.11)$$

Bayes Factors As a simple example, let us look at the Bayes factor (ratio of evidences) between the signal and glitch model for a fairly loud signal in two interferometers. For a simplifying assumption, we say that the signal has equal SNR in each detector. Further, we assume that for a signal with sufficient SNR both the signal and glitch models can reconstruct the data equally as well, using similar wavelets. This means we can estimate $Q^{\mathcal{G}} = Q^{\mathcal{S}} = Q$, and $N^{\mathcal{G}} = N^{\mathcal{S}} = N$. The final assumption we make is that the SNR of each individual wavelet, $\overline{\mathrm{SNR}}$ is the same: $\overline{\mathrm{SNR}} = \mathrm{SNR}_{\mathrm{NET}}/\sqrt{N}$. Using these approximations and subtracting Eq. 5.10 from Eq. 5.11, the expression for the log Bayes factor:

$$\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} \approx \frac{5N}{2} + N \ln V_\lambda + 5N \ln(\overline{\mathrm{SNR}}) - \sum_i^N \ln \bar{Q}_i + \left[2 + \ln \frac{\sqrt{\det C_\Omega}}{4\pi^2}\right]. \quad (5.12)$$

The most notable feature of this expression for $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ is that the primary scaling goes as $\mathcal{O}(N \ln \mathrm{SNR})$. This means that it is not primarily the loudness of the signal that predicts the Bayes factor, but the number of wavelets used to reconstruct the signal.

Numerical Results

Now that we have a predicted scaling of the Bayes factors as a function of the SNR and number of wavelets used, we can numerically test these predictions by running *BayesWave* on simulated GW signals. This was done in two different setups: using a single wavelet to recover the signal, and using multiple wavelets to recover the signal. Below we detail the methodology and results from both of these tests.

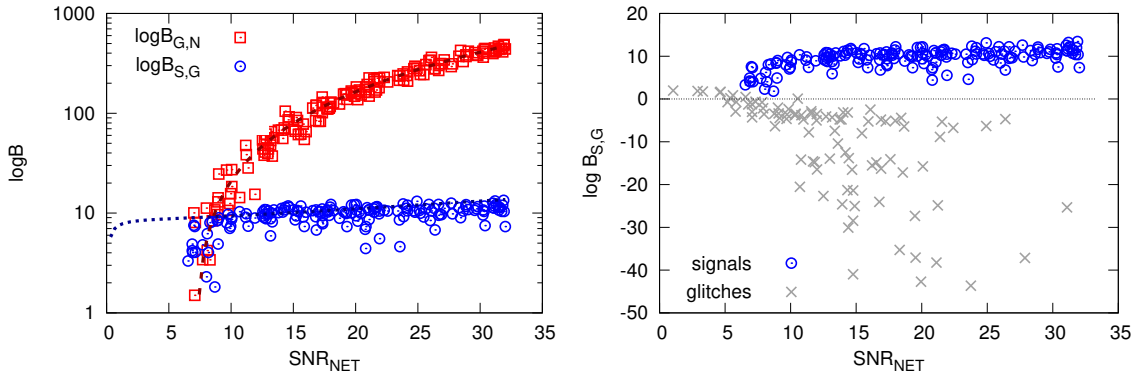

Figure 5.1: Bayes factors for a injected signals and glitches, as compared to the analytical predictions from the Laplace approximation. Figure from [73].

Single Wavelet Though the RJMCMC functionality is a key feature of *BayesWave*, it is also possible to run in a mode where the number of wavelets is set *a priori*, and the transdimensional jumps are turned off. To test the accuracy of Eq. 5.12, we set the number of wavelets used by *BayesWave* to one. The simulated GWs used were a set of sine-Gaussians signals. Both the parameters of the the sine-Gaussian and the extrinsic parameters describing the the physical location of the signal are drawn from the prior distributions described in Chapter 4.

In addition to the simulated GW signals, we simulated a set of glitches as well. Glitches were simulated by using the same prior distributions, but with independent

sets of parameters in each detector. For both sets of injections the number of interferometers used was $N_{\rm IFO} = 2$. The simulated signals and glitches were injected into data from initial LIGO's fifth and sixth science runs, with added Gaussian noise.

Fig. 5.1 shows the recovered Bayes factors and predicted scaling for simulated signals (left panel), and recovered Bayes factors for simulated glitches (right panel). Though there were a number of simplifying assumptions made, we see that the general scaling of the log signal-to-glitch an log glitch-to-noise Bayes factors generally follow the predicted scaling (Eq. 5.12 and Eq. 5.10 respectively). At lower SNR's we can see that the approximation, especially that for $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ breaks down, but this is to be expected as many of the assumptions made earlier are from the high SNR limit. The panel on the right contains the Bayes factors for both the injected signals and glitches. If $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} < 0$, it means that there is more evidence in favor of the glitch model than the signal model. The dotted gray line indicates $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} = 0$, and we see that for most events there is a clear separation between the signal and glitch models, showing that *BayesWave* can robustly reject glitches.

Multiple Wavelets Now that we have seen that in the case of a single wavelet the Bayes factors follow the general trend predicted by Eqs. 5.11 and 5.10, we can investigate how the Bayes factors scale when taking advantage of *BayesWave*'s full multi-wavelet capabilities.

The set of GW signals used in this tests includes sine-Gaussian waveforms (SG) like those described above, mergers from two $50M_{\odot}$ black holes (IMBH), and white noise bursts (WNB). The BBH waveforms were modeled from the non-spinning Effective One Body approach [34]. The WNB injections are unstructured and unpolarized excess power limited in their extent in time and frequency. The results of $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ vs the injected SNR are shown in Fig. 5.2. We can see a clear trend

in the Bayes factor for each individual waveform. To further understand these trends, we now focus on the number of wavelets used.

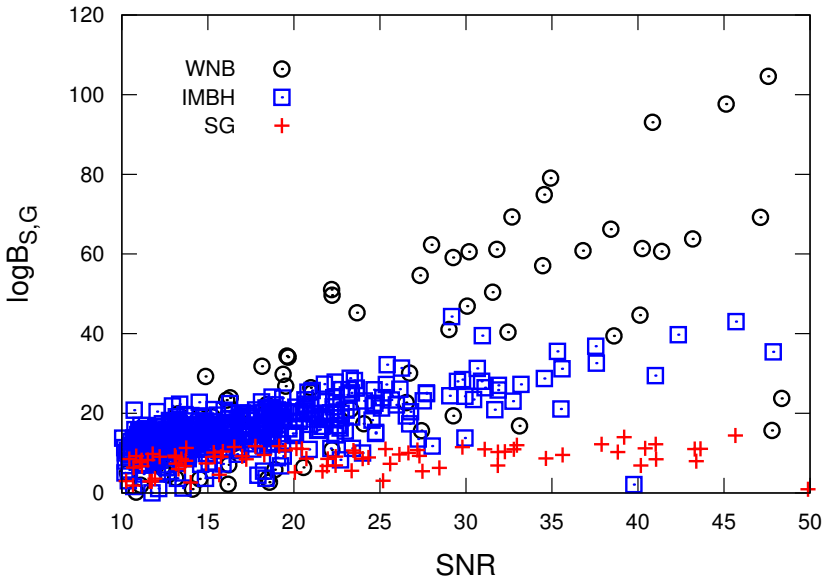

Figure 5.2: Bayes factors for a injected signals using full transdimensional jump features, for a variety of waveforms. Figure from [73].

Wavelet Number

We see in Eq. 5.12 that a main scaling of $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ is the number of wavelets used in the reconstruction. A natural follow-up to this insight is to see if we can predict at all the number of wavelets *BayesWave* will tend to use in a reconstruction. We tested this using *BayesWave* to recover a set of injections of a variety of waveforms at a range of SNR's. From the output of *BayesWave* we can extract the average (median) number of wavelets used from all iterations of the chain. For our set of injections, we again used SG, IMBH, and WNB injections as described above.

Fig. 5.3 shows the median number of wavelets used by *BayesWave* (left panel) and the average SNR per wavelet (right panel) as a function of the injected SNR. We can clearly see a generally linear relationship between the number of wavelets used

and the SNR, and further that the linear relation for each waveform morphology appears to have its own unique slope. In other words, we see that

$$N \sim 1 + \beta \mathrm{SNR} \tag{5.13}$$

where β is determined by the particular waveform morphology. The best fit β for each waveform family tested is shown in the left panel of Fig. 5.3, and the predicted linear relation using that slope is plotted over the numerical results. We can see that for the SG waveforms $\beta = 0$, and *BayesWave* tends to use a single wavelet for the reconstruction. This is an unsurprising result as *BayesWave* uses a sine-Gaussian as its basis function, and so shouldn't need more than one sine-Gaussian to reconstruct such a waveform. The slight spread in the number of wavelets used for SG waveforms is due to the fact that *BayesWave* is constantly proposing adding new wavelets, and occasionally a wavelet can grab on to a slightly anomalous bit of noise and hang around for a while.

The IMBH and WNB waveforms show a much more pronounced increase in wavelet number with increase in SNR. This is due to the fact that as a signal becomes stronger, the finer details of the waveform will become more apparent and more waveforms will be favored to capture these details (i.e. increase the likelihood). WNB waveforms tend to have more complicated structure in time-frequency space, and so we see in Fig. 5.3 that WNBs require more wavelets to reconstruct than IMBH signals (and of course SG signals).

In addition to the dimensionality of *BayesWave*'s model, we can also study how the typical SNR per wavelet is dependent on the injected signal. The righthand panel of Fig. 5.3 shows the average SNR used per wavelet vs the injected SNR. We see that

the general relationship goes as

$$\mathrm{SNR}_i \sim \alpha \mathrm{SNR}^a \tag{5.14}$$

where α and a are again unique for each waveform family. For SG injections $\alpha \sim 1$ and $a \sim 1$.

One thing to note is that in both Figs. 5.2 and 5.3 the WNB injections have more spread than the SG or IMBH injections. This is a result of WNB waveforms in general having a less consistent form, and the spread in $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ in particular can in part be attributed to the fact that WNB waveforms are unpolarized, while *BayesWave* assumes an elliptical polarization.

The combination of our predicted $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ analytical expression and numerical results of dimensionality studies leads us to one of the most unique features of *BayesWave*: *the detection statistic is not determined solely by* SNR*, but by complexity of the waveform.* This can be seen in our semi-rigorous mathematical expressions and numerical experiments, but it is really a very intuitive result. If a very complicated signal appears simultaneously in more than one detector, it is unlikely that that is pure coincidence.

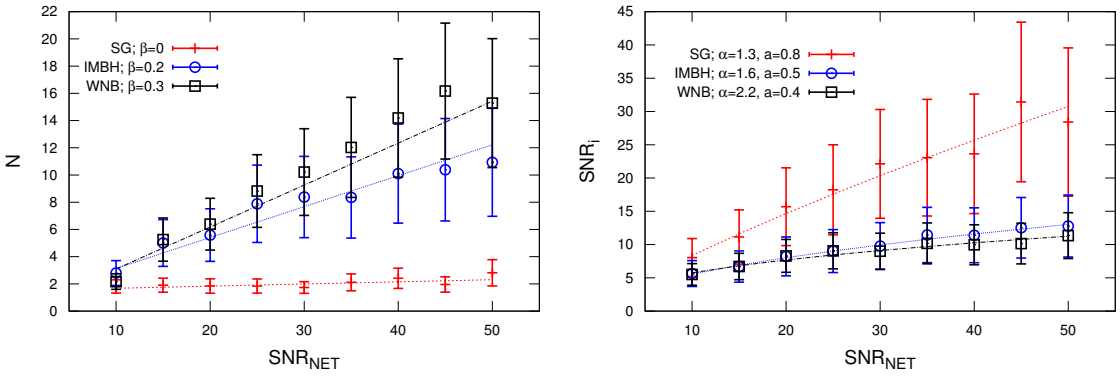

Figure 5.3: Average number of wavelets used and average SNR for a variety of waveform morphologies at a range of SNR's. Figure from [73].

BAYESWAVE SEARCH PIPELINE

Now that we have shown why *BayesWave*'s glitch rejection capabilities are robust and unique, we must actually implement it into a LIGO data analysis pipeline.

The full MCMC nature of *BayesWave* is computationally expensive, and unfortunately makes running *BayesWave* as a standalone search unfeasible[1]. A typical *BayesWave* run on about one second of data, over a frequency range of about a kiloHertz takes on the order of a day. In O1, for example, there were about 40 days worth of coincident data collected from the two LIGO detectors, which means we would need thousands of years of CPU time to run on all of that data. So instead of running on all this data, we need a method of easily identifying possibly interesting stretches of data.

In LIGO and Virgo, there are a number of low-latency search pipelines that quickly identify potentially interesting short-duration triggers. Some of these searches specifically target compact binary coalescences [11, 36, 46, 81, 102], while others look signals from any source. The latter of these types of searches, so-called "burst" searches, are sensitive to a wide range of signals, but as mentioned earlier are more susceptible to false alarms from coincident glitches.

Thus *BayesWave* can fill in a crucial role in burst searches– by following up triggers of interesting data to help enhance our confidence in detections, and better reject false alarms due to glitches.

Current LIGO Burst Searches

In the run-up to O1, there were two main burst searches under development: Omicron LALInferenceBurst (oLIB) [76], and coherent WaveBurst (cWB) [63, 64].

[1] For now.

The oLIB pipeline works by first incoherently finding excess power in each detector [66]. The pipeline then follows up with a coherent analysis that uses a single sine-Gaussian wavelet to model the GW signal, computes the Bayes factors for the signal versus noise models, and the coherent versus incoherent noise transient models (similar to *BayesWave*'s signal to glitch Bayes factor). The detection statistic used by oLIB is the joint likelihood ratio, Λ, between those two Bayes factors. While oLIB's coherent approach helps to increase the sensitivity of this burst search, it is limited by the fact that it uses only a single sine-Gaussian and so cannot capture some of the finer details of a potential GW signal.

The cWB pipeline has been used in LIGO and Virgo since the Science runs in initial LIGO. cWB operates by using a maximum-likelihood-ratio statistic to look for correlated excess power across a network of detectors. The main scaling of the cWB detection statistic, ρ, is designed to scale with SNR. cWB also includes regulators and cuts on signals from unlikely sky locations. For a full discussion see [47]. One of the main advantages of cWB is that it is capable of analyzing large amounts of data very quickly, and can successfully identify interesting triggers in the data. However because of ρ's dependence on SNR, cWB can have trouble with very loud glitches. This is an area *BayesWave*'s unique detection statistic can help increase our confidence in GW detections. By following up cWB triggers with *BayesWave*, we can better distinguish between real signals and instrumental glitches. Furthermore, cWB can return a waveform estimate, but only a point estimate, where as *BayesWave* returns a full posterior waveform distribution.

Tests of Pipeline Performance

To numerically investigate whether using *BayesWave* as a cWB follow-up would actually be beneficial, we used data obtained during the fifth and sixth Science Runs

of LIGO and Virgo, and looked at background data sets and injections, similar to what was described in the previous section.

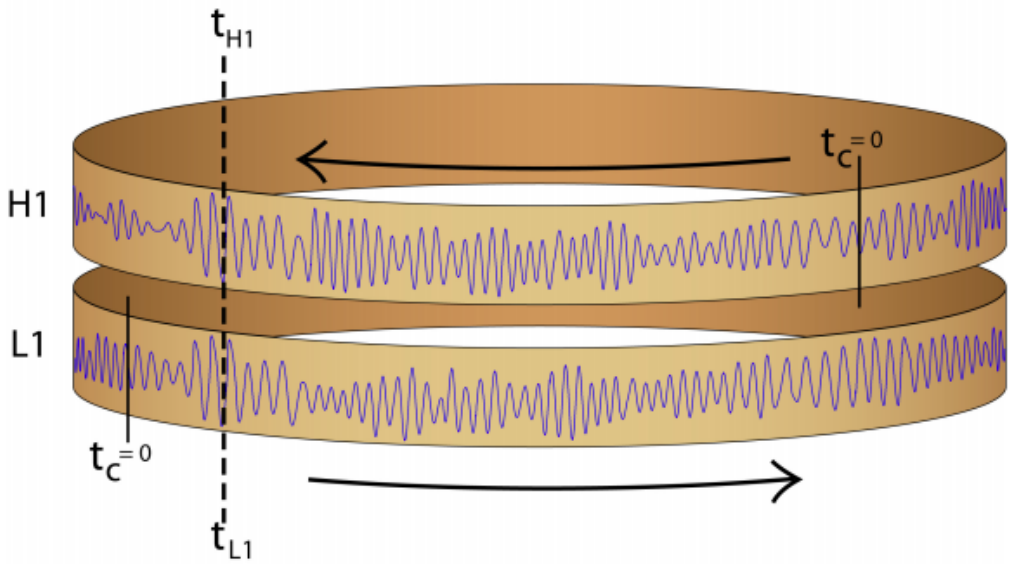

Figure 6.1: A cartoon example of the time slide method used to estimate false alarm rate. Figure from [29].

Background Analysis

A common approach to measure the false alarm rate (FAR) of events in LIGO data is the *time-slide* method, which is a way to measure how often bits of excess noise occur in multiple detectors at once, and how often those instances of noise mimic a real GW signal. To do this, we need a stretch (or really multiple stretches) of data that we are confident contains no real astrophysical signals. Since shielding the detectors from GWs is not physically possible, what we can do is take the data streams from two or more detectors and temporally shift them relative to each other by an amount greater than the light travel time across the earth. Any simultaneous events then in the resulting data sets are then purely coincidently, and definitely not a real signal. This method does assume that the rate of transient noise events is much

greater than the rate of GW events, but this is a safe assumption. The time-slides have the added advantage of extending our data set– turning days worth of data into many many years worth of data. A cartoon illustration of the time-slide method is show in Fig. 6.1

A search pipeline can run their analyses on these time-slide data sets, or *background* data set as it is called, and produce a cumulative distribution of detection statistics. To construct a cumulative distribution function (CDF) we use the same approach as constructing a basic histogram, but rather than looking at counts per bin you look at *cumulative* counts per bin, which for a distribution of variables x we write as:

$$\mathrm{CDF}(X) = N(x \leq X) \tag{6.1}$$

where $N(x \leq X)$ is the number of samples of x that are less than some threshold value X. In our background distribution construction we use $1 - CDF$, and divide the number of counts per bin by the effective live time of the background to get a FAR as a function of the detection statistic. The time-slide method is used by many pipelines in LIGO data analysis to assign a significance to GW triggers.

As an example of how the cWB+BW pipeline can improve upon a cWB only background distribution, we look at a stretch of data from initial LIGO's sixth science run (S6) that has beed "recolored" to generally look like the expected noise curve for advanced LIGO. We used 26.6 days worth of coincident data, and using time slides turned it into approximately 1896 years of effective live time [60]. cWB is first run on the entirety of this background data set in a frequency range of 16-512 Hz to identify the most significant triggers. We then set a threshold at $\rho > 8.1$, which corresponds to the 500 most significant triggers. These triggers are followed up by *BayesWave* (in the same frequency bandwidth) to develop a background distribution

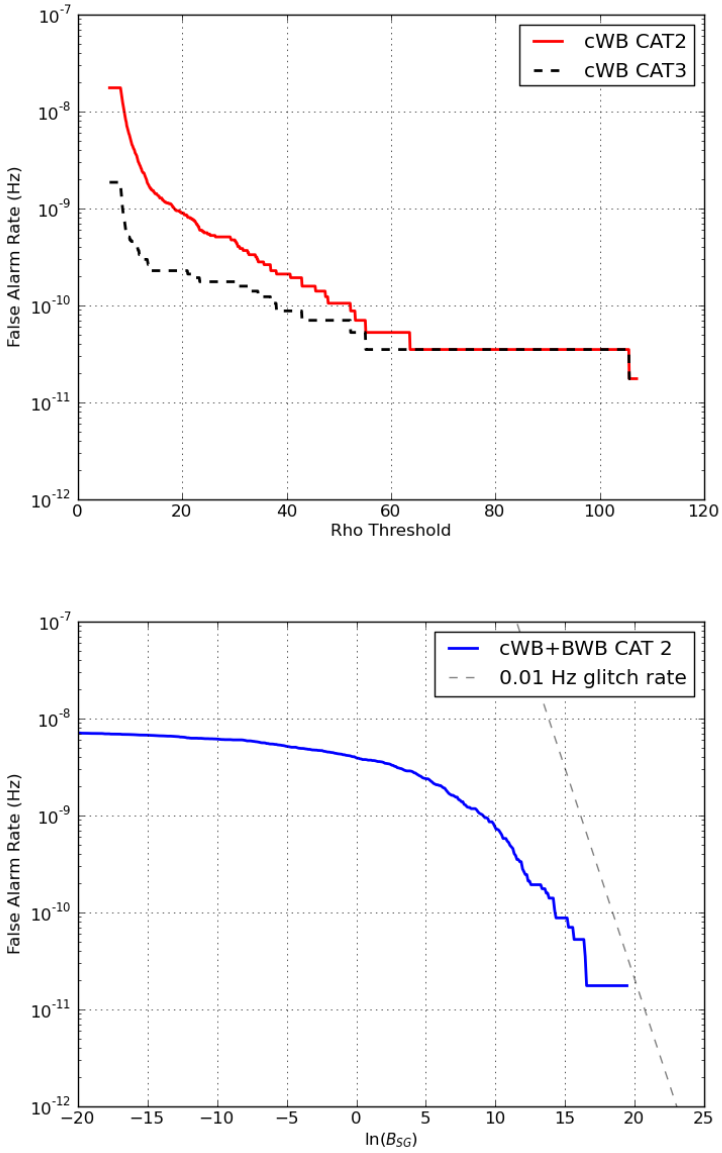

Figure 6.2: The background distributions for recolored S6 data from cWB alone (top), and cWB+BW (bottom). The cWB only background shows the distribution after the Category 2 (CAT2) and Category 3 (CAT3) data quality vetoes, and the cWB+BW distribution uses only CAT2 vetoes. Figure from [60]

of *BayesWave*'s $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ detection statistic. Finally data quality vetoes as described in [8] are applied to remove stretches of data with known issues. We will discuss data quality vetoes in detail later in this chapter. The cumulative distribution functions for these background sets are shown in Fig. 6.2.

The most notable difference between theses two background distributions is that the cWB only distributions have fairly long tails out towards higher detection statistics (ρ in this case). Long tails are problematic because they increase the false alarm rate and thus decrease the significance of potential GW triggers in the foreground data. The cWB+BW background distribution has a much sharper drop off due to the fact that *BayesWave*'s detection statistic, $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ does not simply scale with SNR.

These results alone are a promising indication that using *BayesWave* as a followup to cWB can help enhance our confidence in GW detection, and better distinguish signals from glitches. To further test this, we now look at injected simulated signals into our noise.

Injected BBH Signals

To test how *BayesWave* can aid in signal detection, we used two sets of simulated BBH mergers: one with a component masses of $50M_{\odot}$, and one with component masses of $150M_{\odot}$. The waveforms were generated using a particular implementation of the Effective One Body approximation, EOBNRv2 [87]. These waveforms were injected into the S6 recolored noise (as described above), and first recovered with cWB. Then for injections recovered with $\rho > 6$, *BayesWave* was used as a followup. Fig. 6.3 shows a scatter plot of the cWB ρ statistic versus *BayesWave*'s $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ statistic. Also shown are dashed lines marking the value of each detection statistic that corresponds to a FAR of 1/300 years. This threshold FAR was chosen to

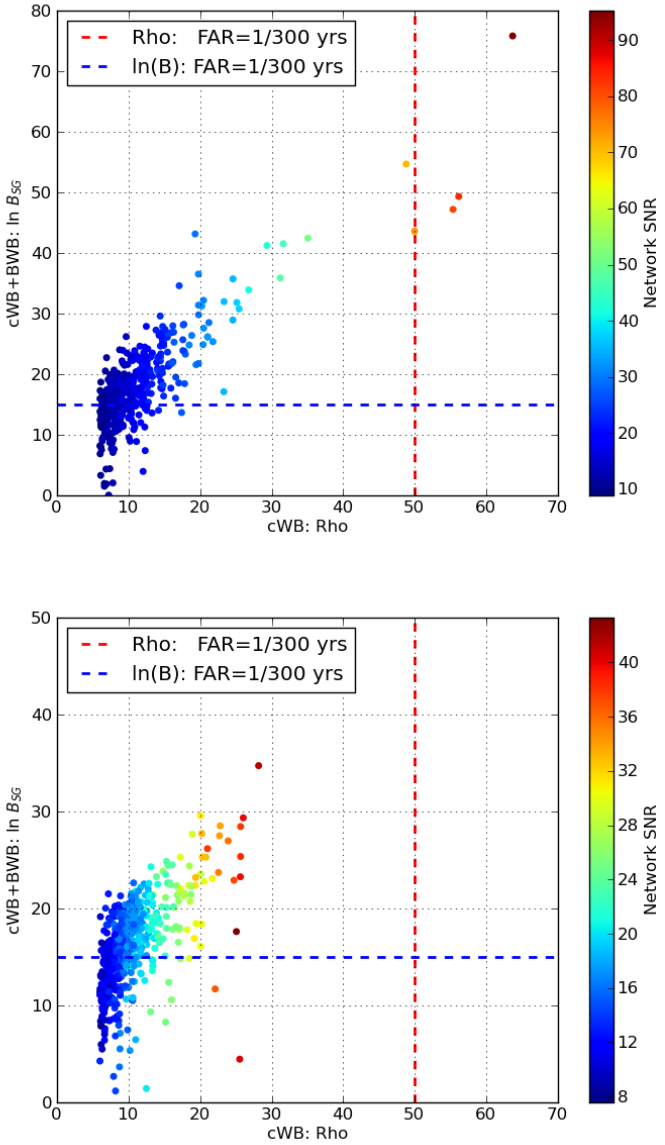

Figure 6.3: Scatter plot of cWB's ρ and *BayesWave*'s $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ detection statistic for BBH systems with component masses of $50M_{\odot}$ (top), and $150M_{\odot}$ (bottom). The dashed lines denote a FAR of 1/300 years. Figure from [60]

correspond to a 3σ level of confidence in detection.

All the points that are above the horizontal blue dashed line, but to the left of the vertical red dashed line are potential detections which would not pass the 3σ detection threshold with cWB alone, but would be found using a *BayesWave* followup to cWB. We see that for the BBH injection set *BayesWave* helps boost the confidence in a large number of detections, showing clearly that using the full cWB+BW pipeline will decrease the chance of missing one of these detections. Looking at the SNR of the injections, we see that for the $50M_{\odot}$ events we would need SNRs around 70 to detect these GWs with cWB alone.

Ad-hoc Waveforms

In addition to BBH signals, we also tested the performance of the cWB+BW pipeline using some of the ad-hoc signals that we previously described in Chapter 5. Namely, we looked at white noise burst (WNB) and sine-Gaussian (SG waveforms). For these waveforms we follow the approach from [8], which means we use a different background set than for the BBH injections. The data used to produce a background set was 51 days worth of coincident observations from S6, time slid to create an equivalent of 70 years worth of livetime for the background. Again, cWB was run on this data set first and *BayesWave* followed up on triggers with $\rho > 8$. The cWB and cWB+BW background distributions are shown in Fig. 6.4, and again show that adding the *BayesWave* followup drastically reduces the long tails on the background distribution. The WNB and SG waveforms are then injected into the S6 data, and again first recovered with cWB, and *BayesWave* is used as a followup for triggers above $\rho = 8$. The recovered detection statistics for cWB and cWB+BW are shown in Fig. 6.5, along with dashed lines marking the value of the highest detection statistic from the background data set for each search.

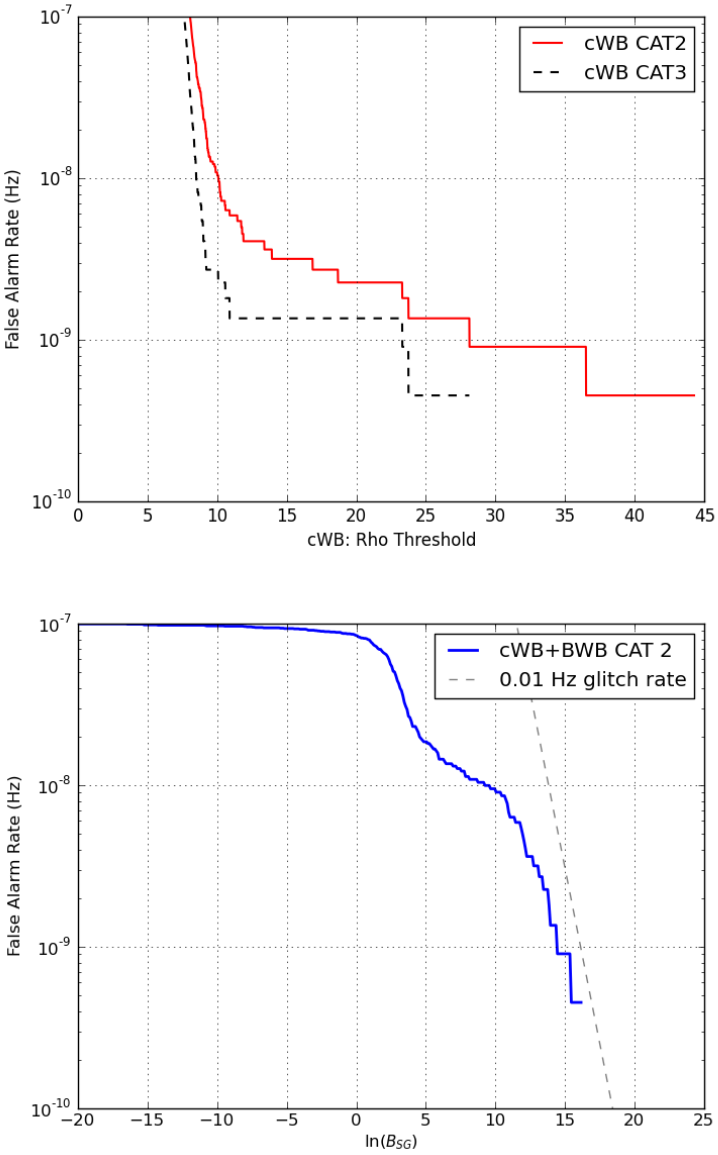

Figure 6.4: The background distributions for S6 data from cWB alone (top), and cWB+BW (bottom). The cWB only background shows the distribution after the application of Category 2 (CAT2) and Category 3 (CAT3) data quality vetoes, and the cWB+BW distribution uses only CAT2 vetoes. Figure from [60].

We see that these scatter plots look somewhat different from those in Fig. 6.3 for the BBH injections. The WNB signals still show that using a *BayesWave* followup can significantly increase our confidence in detection for a number of injections. However there is quite a bit of more scatter than the BBH cases. This is primarily a result of some of the challenges WNB waveforms pose for *BayesWave* as described in Chapter 5– such as WNB injections being unpolarized. The SG waveform injections are clearly where we see the least improvement using a *BayesWave* followup. We see in Fig. 6.5 that only a small number of SG injections are recovered at a $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ greater than the loudest background event. Looking again at Eq. 5.12, we recall that the characteristic scaling of $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ goes as the number of wavelets used, and *BayesWave* typically uses only a single wavelet to reconstruct a SG signal. This means that *BayesWave* is relativity insensitive to SG signals, and so we may need to rely on other methods to confidently detect these type of signals.

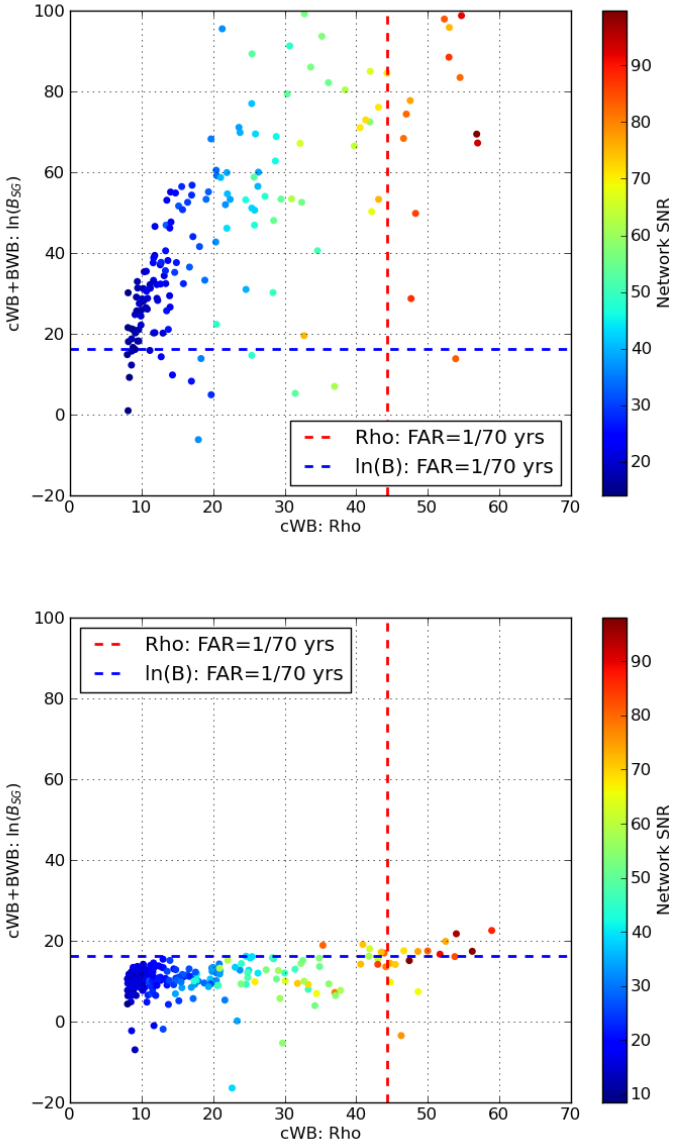

Figure 6.5: Scatter plot of cWB's ρ and *BayesWave*'s $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ detection statistic for WNB waveforms (top), and SG waveforms (bottom). The dashed lines denote a FAR of 1/70 years (the loudest background event). Figure from [60].

Setting Up the Pipeline

We have shown that using *BayesWave* as a followup to the cWB search can greatly increase our ability to distinguish between signals and glitches for a variety of waveforms. The next step is to fully implement the cWB+BW pipeline into LIGO data analysis.

The bulk of the *BayesWave* and *BayesLine* code is written in C, and uses the LIGO Analysis Library Suite [4] to properly read in the gravitational wave frame files, which are in the .gwf format. Intermediate scripts based mostly in Python are used to read in triggers produced by cWB; setup runs with the appropriate command line options; locate the frame files; and launch jobs through a high throughput computing job manager. After jobs have run, there is another Python script which compiles the Bayes factors from background runs to produce a FAR distribution. All analysis was done on the LIGO Data Grid.

Here I describe in detail *BayesWave*'s implementation running on "zero-lag" triggers (i.e. trigger from actual coincident data), and background data sets.

Zero-lag

The data is first collected at each interferometer site. In O1 this was the two LIGO sites in Hanford, WA and Livingston, LA. For the second half of O2, the Virgo detector in Italy joined as well, however the cWB search used only the data from the two LIGO detectors. For any zero-lag trigger during which Virgo data was available, we did an additional *BayesWave* run with the three detectors for waveform reconstruction, but all significance results use only the data from the two LIGO detectors. The data collected by the interferometers is then transferred to the various LIGO Data Grid computing sites, and cWB analyzes the entire stretch

of data. Selection cuts are then applied on cWB's detection statistic ρ, as well as the correlation coefficient c_c, which is a measure of how coherent the power is across a network of detectors [88]. Triggers that survive this cut are uploaded to the Gravitational-wave Candidate Event Database (GraceDB) [1].

GraceDB is an incredibly useful tool in LIGO/Virgo to serve as a central location for information on GW candidates. Each candidate event gets a unique entry in the database that contains information such as the trigger time of the candidate and an initial estimate of the FAR, and places where one can add any other information they want about the event. GraceDB is used internally within LIGO and Virgo to centralize information about potential GWs, and is also used in communication with electromagnetic observation partners to share information about probable sky locations of the trigger.

GraceDB is used in the cWB+BW followup pipeline to launch *BayesWave*'s zero lag jobs. When the cWB events are uploaded to GraceDB, *BayesWave* reads in the trigger time as well as cWB's estimate of the central frequency. *BayesWave* then sets up a run centered on cWB's estimated trigger time, with a sampling rate based on the central frequency estimate, and uses the trigger time information to locate the the gravitational wave data frames on the LIGO data grid. The details of the prior ranges used in the zero lag analyses will be discussed in the results section of this chapter.

After *BayesWave* finishes running, there are a number of pieces of information that it reports back to the GraceDB entry for the trigger of interest. *BayesWave* adds its results for the significance of the event ($\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ and $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$), some estimates of characteristics of the signal (central estimates and spread in frequency and time as will be described in detail below), and a probability sky map. Also added to the GraceDB page is a link to the *BayesWave* output page which contains all of

BayesWave's pertinant information.

Note that between O1 and O2 cWB added an extra search intended to better find GW signals that have a characteristic chirping up in frequency. In O2 *BayesWave* only followed up the events from the search for the most general signals.

Background

In addition to following up the zero-lag triggers from cWB, *BayesWave* also follows up on cWB's background sets in order to measure our coincident glitch rate.

cWB produces its background sets using the time-slide method, with the time-slides being performed in one second increments. The number of time-slides done is determined by the FAR/FAP level one is aiming to reach. Once the time-slid background data is in hand, cWB runs on the whole data set and reports the results. These results are presented in an ASCII file and includes information such as the trigger time in each interferometer, estimates of the central frequency, the detection statistic ρ, and the correlation coefficient c_c (among other things).

BayesWave can then read in this ASCII file of triggers and set up runs based off of the trigger times (and thus time lag information), and central frequency estimates. Details of the run settings can again be found in the results section of this chapter.

Finally, we must decide which of cWB's triggers *BayesWave* will actually follow-up on. As mentioned earlier in this chapter, *BayesWave* is limited by computing resources, and following up all cWB triggers is unfortunately unfeasible. So we must pick a ρ threshold, and potential down sampling rate, such that we can dig deep enough into the background without exhausting computational resources. The particular cuts used in the searches are described below in the results section.

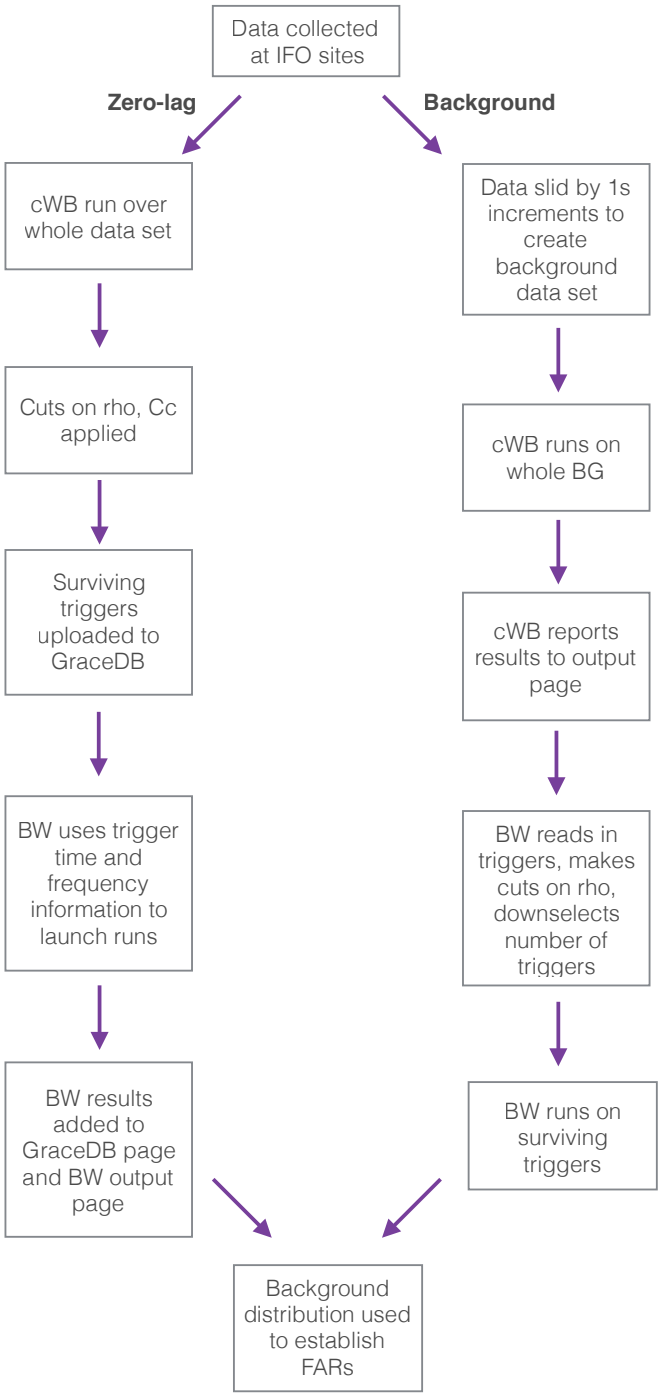

Figure 6.6: Schematic of BW+cWB pipeline

Data Quality

The last step in the analysis of triggers is applying data quality flag vetoes. LIGO and Virgo's Detector Characterization working group has studied the complex interactions between the detectors' GW strain channel and the some hundred thousand auxiliary channels in order to identify stretches of data that could be problematic [8, 10, 16]. Various data quality flags exist to mark particular issues in the data. Relevant to the short duration burst searches are three data quality flags:

- The category 1 (CAT1) data quality flag is typically applied before any analysis starts, and indicates when there is a significant issue with the current state of the detector.

- The category 2 (CAT2) data quality flag is usually applied after the initial analysis of detector data, and indicates time segments with a known noise source that can couple to the strain channel.

- The category 3 (CAT3) data quality flag is also applied after the initial analysis. This flag is generally used to mark stretches of data where there is excess noise in an auxiliary channel that can be statistically correlated with noise in the strain channel, but the exact mechanism by which they're connected is not known. CAT3 also includes times of hardware injections.

The stretches of data which are vetoed by these flags are identified by a GPS start time and a GPS stop time, and removed from the analysis. Each interferometer has its own list of vetoed data segments.

BayesWave applies data quality vetoes by checking that for each trigger none of the data in *BayesWave*'s analysis window overlaps with a data quality veto segment.

Postprocessing

Once the full *BayesWave* algorithm has been run on a trigger, the next most important thing to do is present our results in a concise and informative way. The main outputs of *BayesWave* are the Bayes factors between models, and the waveform reconstructions. However there are other metrics we can use to characterize the signal, as well quantities we can look at that may help us troubleshoot issues if necessary.

We compiled our postprocessing information into output webpages that are generated automatically for real GW triggers. These output pages were first put together by Jonah Kanner and Francesco Pannerale, and have since undergone some significant redesign– both cosmetically and in terms of content.

Here we will describe the information included in each section (practically a "tab") of the webpage for the version used in LIGO's O2 run.

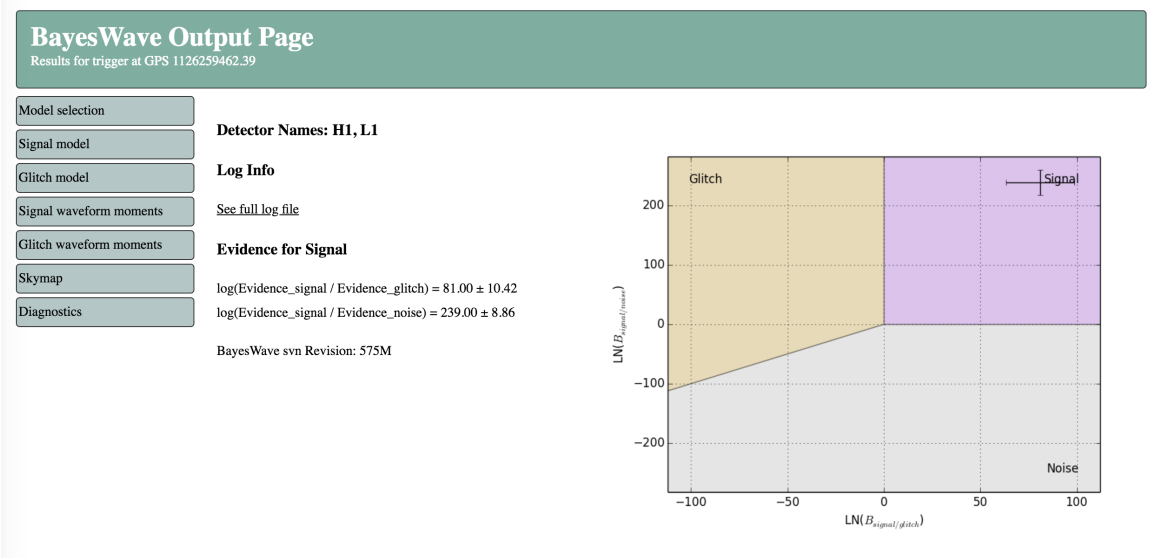

Figure 6.7: An example of the output page for a *BayesWave* run. This shows the home page which is the "Model selection" page.

Model Selection This serves as the "home page" for the *BayesWave* output pages, and immediately informs the reader as to whether or not the signal model is preferred over the glitch or noise model. This information is reported both in text (giving $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ and $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$), as well as in a plot. An example of this homepage and model selection plot is shown in Fig. 6.7. This tab also has some practical information such as which detectors were used in the analysis, and a link to the GraceDB page if applicable.

Waveform Reconstruction After model selection, we look at the waveform reconstructions in various forms. The signal and glitch models each have their own tabs that show:

- Whitened time domain waveform reconstruction, in each interferometer. These plots show the median waveform, as well as the 50% and 90% credible intervals. For reference, the whitened detector data is also shown.

- Frequency domain waveform reconstructions, and PSDs in each interferometer. Again the median reconstructed waveform is shown with 50% and 90% credible intervals. The PSD also has the credible intervals shown. This is one thing that distinguishes *BayesWave* from other noise modeling methods: we actually marginalize over our noise model rather than just using a point estimate. Examples of both the time and frequency domain waveforms are shown in Fig. 6.8.

- Spectrograms of the median waveform, the actual data, and the residuals of the median waveform subtracted from the data. These spectrograms are made using Q-scans as described in Appendix C, using $Q = 4, 8, 16$. These are used to give a visual representation of where the power is in time-frequency space,

and the residual spectrograms in particular can be a very good check of whether or not the reconstruction captured all of the excess power.

- Time-frequency tracks in each interferometer. This is another way of visualizing the evolution of the frequency of the signal over time. These will be described in detail in Chapter 7.

Waveform Moments In addition to the waveform reconstructions themselves, there are certain quantities we can measure from the waveform reconstructions– so-called *waveform moments* first proposed by Patrick Sutton and described in [44]. For these waveform moments, we use the normalized densities given by

$$\rho(t) = \frac{h_+^2(t)+h_\times^2(t)}{h_{\rm hrss}^2} \tag{6.2}$$

$$\rho(f) = \frac{|\tilde{h}_+(f)|^2+|\tilde{h}_\times(f)|^2}{h_{\rm hrss}^2} \tag{6.3}$$

where $h_{\rm hrss}$ is the root-mean-square strain:

$$h_{\rm rss} = \int_{-\infty}^{\infty}(h_+^2(t)+h_\times^2(t))dt = \int_0^\infty(|\tilde{h}_+(f)|^2+|\tilde{h}_\times(f)|^2)df. \tag{6.4}$$

We can then define time and frequency moments by

$$\langle t^n\rangle = \textstyle\int_{-\infty}^{\infty} t^n\rho(t)dt \tag{6.5}$$

$$\langle f^n\rangle = \textstyle\int_0^{\infty} f^n\rho(f)df. \tag{6.6}$$

These can be used to give estimates of physically interesting quantities like the central time $t_0 = \langle t^1\rangle$, central frequency $f_0 = \langle f^1\rangle$, duration $\Delta t = \sqrt{\langle t^2\rangle - \langle t^1\rangle^2}$, and bandwidth $\Delta f = \sqrt{\langle f^2\rangle - \langle f^1\rangle^2}$.

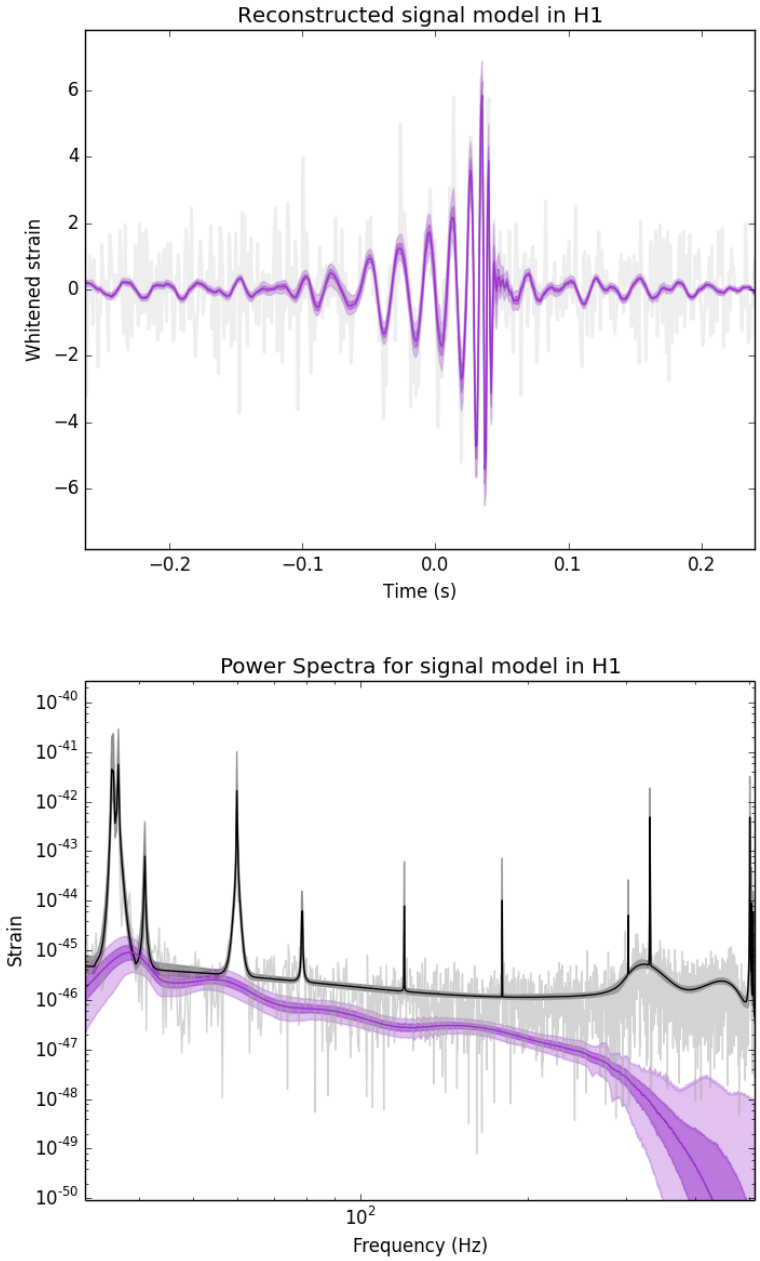

Figure 6.8: Example waveform reconstructions used in the output pages in the time domain (top) and frequency domain (bottom) for GW150914.

The signal model and glitch model each have a "waveform moments" tab, which contains histograms of these quantities in each interferometer. Also included are histograms of the recovered SNR in each interferometer. An example of some of these histograms are shown in Fig. 6.9.

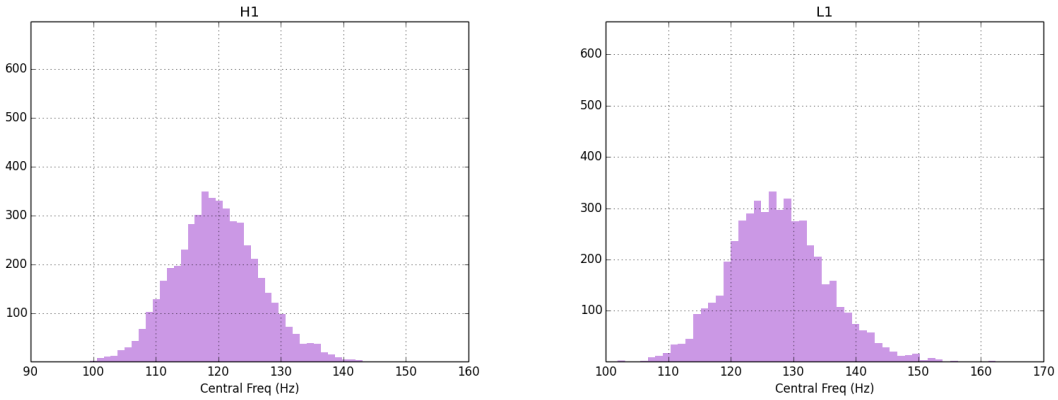

Figure 6.9: Example waveform moments histograms shown in the *BayesWave* output pages. This particular example, from GW150914, shows histograms of the central frequency in each interferometer.

Sky Maps As described in Chapter 4, by setting the extrinsic sky location parameters as parameters in our MCMC, we end up with a posterior distribution on the GW signal's sky location. We use the Sky Area tool set [51] to turn our posterior distributions into credible intervals of probability regions, and produce Healpix [54] maps of the probability region. This sky map is shown in a tab on the *BayesWave* output pages. An example of the sky map for GW150914 is shown in Fig. 6.10.

Diagnostics The final tab on the output pages is used primarily to get a sense of how the chains are behaving. One of the plots included in this page is the log likelihood for all of the different temperature chains used in the MCMC, for all three different models. This serves as a way to get an intuitive sense of the Bayes factors–

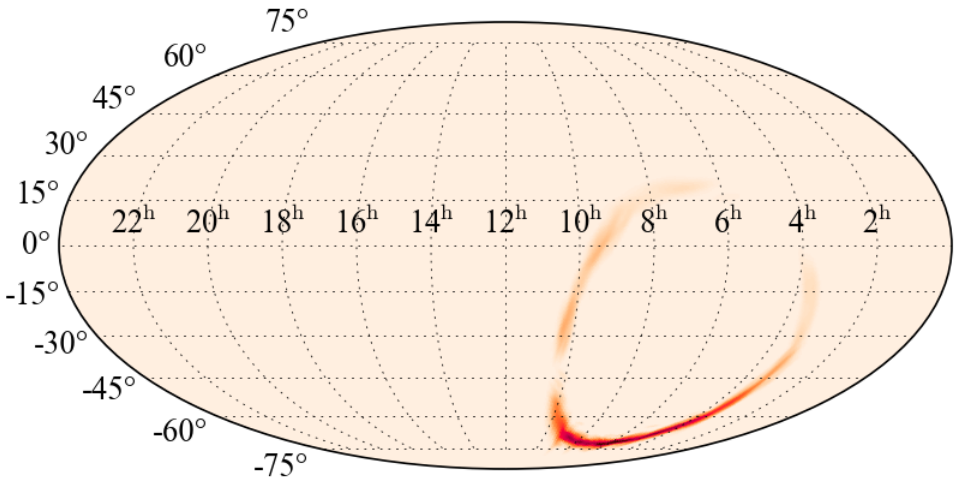

Figure 6.10: Example of the sky map of probable locations for GW150914.

which can heuristically be thought of as the area between the curves.

Also shown are the chains of the number of wavelets used in each model, and histograms of those chains. These give an idea of the "complexity" of trigger, as discussed in Chapter 5. These can also be helpful to diagnose problems when something is not quite right in the data or the run– as that will often times lead to the number of wavelets running up against the edge of the prior. An example of the model dimension plots are shown in Fig. 6.11.

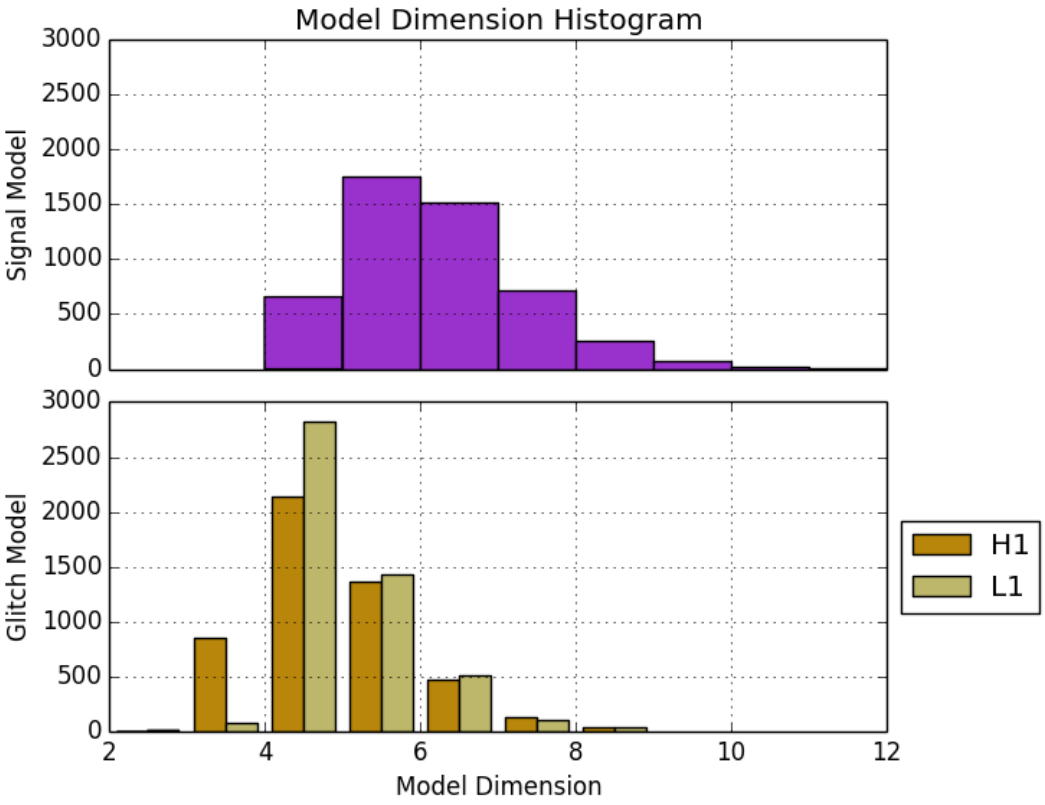

Figure 6.11: Example histograms of the number of wavelets used in the signal model and glitch models for GW150914.

Parameter	Low frequency trigger	High frequency trigger
Time window	1s	1s
Frequency window	32-512Hz	32-2048Hz
Sampling rate	1024Hz	2048Hz
Q range	0.01-40	0.01-40
SNR_* (Eq. 4.4 and 4.11)	5	5

Table 6.1: Parameters used in the *BayesWave* pipeline for the GW150914 significance estimation [13].

Search Results

Now that we have discussed what went into setting up and running the cWB+BW pipeline for real LIGO data, we can look at some of our results. Here we report on the significance results for searches on two different sets of data. First, we report on the *BayesWave* significance of GW150914, the first GW detected by LIGO. Then we report on the all-sky search for GWs over all of the O1 data.

GW150914

On September 14, 2015 cWB identified a potential GW and uploaded the event to GraceDB. Soon after, some of the searches targeting binary black holes identified the same event and added their own events to GraceDB. This event ultimately earned the moniker GW150914, and was the first GW detected by LIGO. Here we summarize the *BayesWave* results presented in [14], which reports on the findings of the short duration burst searches for GW150914.

After the initial GW150914 trigger, the two LIGO detectors (H1 and L1) were

held in their same configuration for approximately a month. In this time, 16 days worth of coincident data was recorded. Using on the order of a million time slides, cWB created a background data set of 67,400 years. cWB analyzed all of this data, then divided the triggers into three search classes based on the characteristics of the trigger.

The first search class is designed to mitigate the effects of a common class of glitch known as a "blip glitch" [10, 14]. A blip glitch is a glitch which is highly localized in time, has no distinct evolution in frequency, and is sometimes quite loud. These blip glitches occur in both detectors and have no known cause. Because of the relative simplicity and loudness of these glitches they have been a concern for burst searches in the first two Observing runs. cWB identified blip glitches by finding triggers whose spread in frequency is less than 5 Hz. All of these triggers were placed into the search class $C1$ bin. cWB created another search bin specifically for triggers which have frequency increasing with time– a characteristic of GWs from compact binaries. Any triggers which displayed a chirping behavior were placed into the search class $C3$. Any remaining triggers that are neither blip glitches nor chirping are placed into the search class $C2$.

To remain as general a search as possible, *BayesWave* put all of cWB's triggers into a single search bin and followed up all classes. As a cut on ρ [2], *BayesWave* followed up all background (and zero-lag) triggers above $\rho > 7$ and $c_c > 0.7$. This resulted in about 2000 triggers being analyzed by *BayesWave*.

The parameters and prior ranges used in this search are described in Table 6.1. We note that we found we had to lower the lower bound of the prior range on the quality factor Q of the wavelets from the $Q = 3$ proposed in [44]. This change was implemented in order to account for blip glitches. Because blip glitches are so

[2] In [14] the detection statistic η_c is used, which is $\sqrt{2}\rho$.

localized in time, we can think of them as having a very low Q factor– lower than the original prior limit of 3. Since *BayesWave* could not reproduce the signal with a single wavelet, it would pile on a number of wavelets trying to reproduce a low Q sine-Gaussian. As we saw in Chapter 5, $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ scales with the number of wavelets used so blip glitches would return very high Bayes factors. Extending the prior range on Q means *BayesWave* uses fewer wavelets, giving us a more accurate representation of the background

The results of *BayesWave*'s search for GW150914 are shown in Fig. 6.12. *BayesWave* found GW150914 with a Bayes factor of $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} = 49.4 \pm 0.8$, giving a FAR estimate of 1/67400 years. For comparison, the loudest background event in *BayesWave*'s background was $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} = 53.1 \pm 3.4$ and was a blip glitch. Though the new prior range on Q helped *BayesWave* better distinguish between signals and blip glitches, this particular glitch class remains the biggest source of noise in our background.

As a comparison, cWB's estimated FAR for GW150914 is 1/22500 years in the $C3$ class (to which GW150914 belongs). This is taking into account a trials factor of three for the three different search classes. Combining the $C2$ and $C3$ search classes gives a FAR estimate of 1/8400 years (with an applied trials factor of 2) [14].

O1 Short Duration All Sky Search

At the conclusion of LIGO's first observing run, we also performed a short-duration burst analysis over the entirety of the run. This analysis is done to search for any other GWs that might have not been found in the initial online search, as well as to set upper limits on non-binary black hole GW events in the observable range. The full report of this search (called the O1 short duration all-sky search) can be found in [16], and *BayesWave* results will be summarized below.

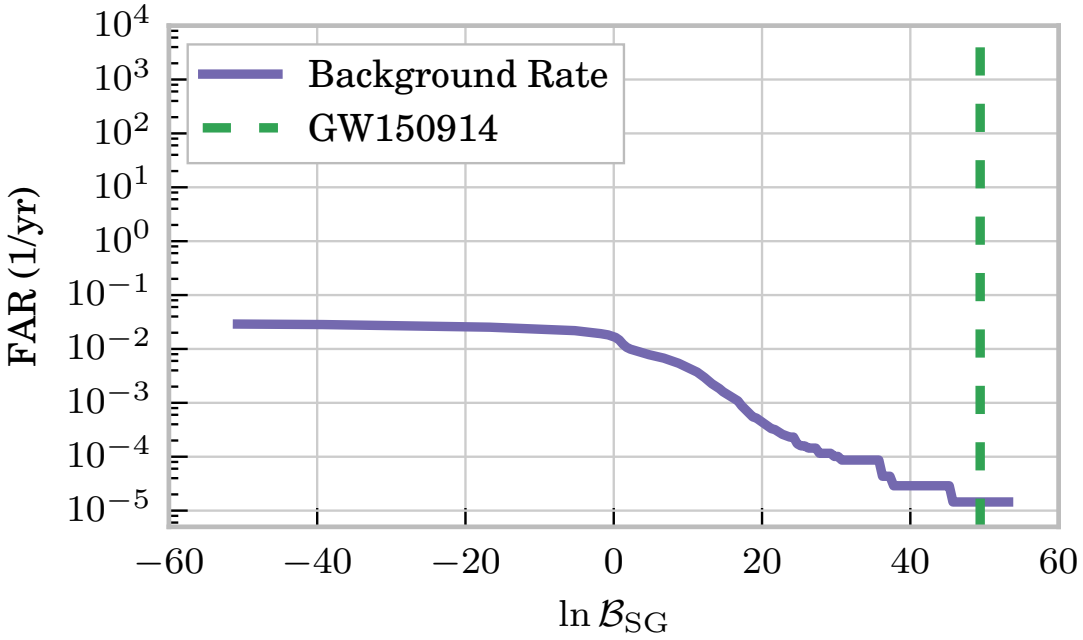

Figure 6.12: Background distribution of *BayesWave* triggers from data analyzed around GW150914. Figure from [14].

Parameter	Low frequency trigger	High frequency trigger
Time window	1s	1s
Frequency window	32-512Hz	32-1024Hz
Sampling rate	1024Hz	2048Hz
Q range	0.01-40	0.01-40
SNR_* (Eq. 4.4 and 4.11)	5	5

Table 6.2: Parameters used in the *BayesWave* pipeline for the O1 All-sky search [16].

The O1 all-sky search consisted of 48 days worth of analyzable coincident time between the two LIGO detectors (after applying data quality vetoes). For cWB, and thus cWB+BW, the amount of time analyzed actually amounted to 44 days as cWB requires at least 600 seconds of continuous data and thus some data was unanalyzable. Just as was done for the GW150914 analysis, the time-slide method was used to produce a background data set from the coincident data. For the all-sky search, we selected a FAR threshold of 1/100 years as the detection threshold– this corresponds roughly a 3σ detection level. In order to confidently reach this FAR level, we require approximately 1000 years worth of background data.

For the cWB+BW followup, the ρ [3] threshold selected was $\rho > 7$. Again *BayesWave* followed up triggers contained in all of cWB's search classes $C1$, $C2$, and $C3$. cWB also split its search into a low and high frequency bin; *BayesWave* only followed up triggers in the low frequency bin. In the all-sky search, the only zero-lag cWB trigger with above the $\rho > 7$ threshold was GW150914. In the background *BayesWave* followed up on about 700 events. The run parameters used for this search were mostly the same as those used in the GW150914 search, but for completeness are described in Table 6.2.

Results of the *BayesWave* all-sky search background are shown in Fig. 6.13. The loudest event in *BayesWave*'s all-sky background has a Bayes factor of about $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} = 20.35 \pm 3.61$, which is much lower than the $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}} = 49.4 \pm 0.8$ found for GW150914. Once again, the loudest event in our background was a blip glitch.

Upper Limits Since the all-sky search for GWs did not turn up any new signals, we proceeded to measure the sensitivity of our searches, and to place upper limits on non-BBH GW signals.

[3] Again [16] uses the detection statistic $\eta_c = \sqrt{2}\rho$.

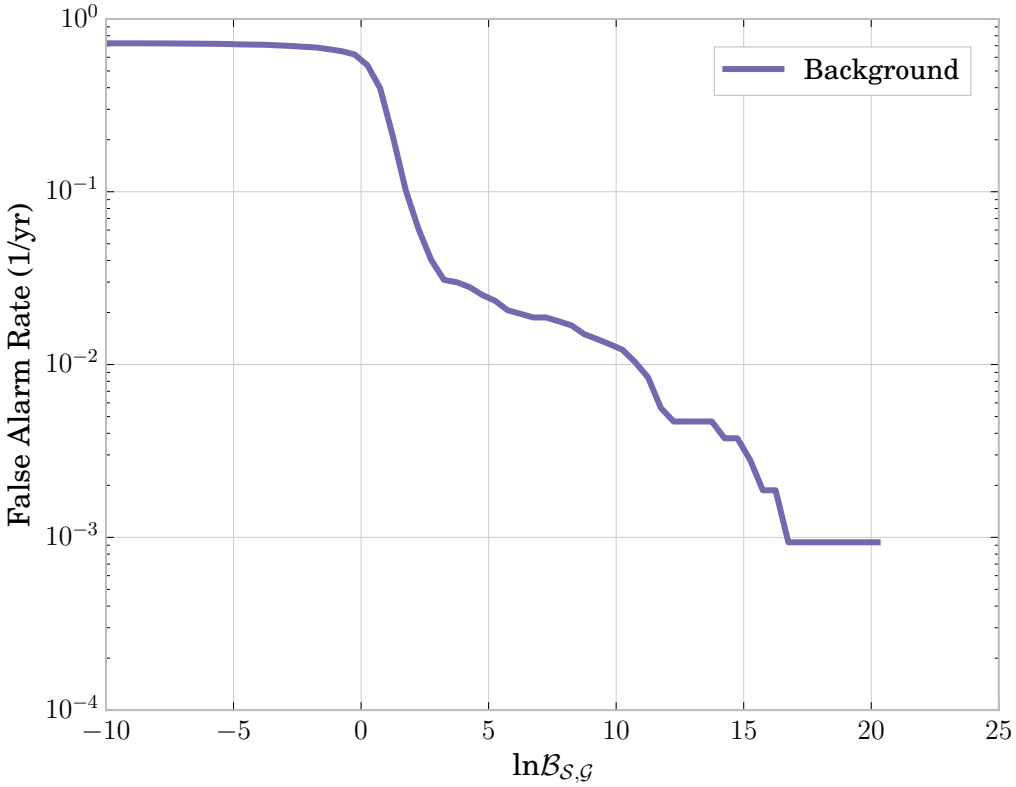

Figure 6.13: Background distribution of *BayesWave* triggers from the O1 short duration all-sky search. Figure from [16].

To measure the sensitivity of our searches for generic transient GW signals, we injected into O1 data the standard simulated ad-hoc signals (sine-Gaussians, white noise bursts) described earlier in Appendix B, along with purely Gaussian waveforms. The details of the waveforms used can be found in Table 6.3, which has been reproduced from [16]. The simulated signals were injected in accordance with a distance probability distribution

$$p(r) = r + \frac{A}{r}, \tag{6.7}$$

where A is a constant. This distribution assures that the the simulated signals are injected such that they cover a wide range of amplitudes, expressed in terms of the root-mean-squared of the amplitude ($h_{\rm rss}$) in the detectors, while giving a sufficient number injections at both low and high $h_{\rm rss}$values.

BayesWave followed up cWB triggers of the simulated signals that passed the ρ threshold described earlier. *BayesWave* further down selected the number of injections in ran on by 1/100 in order to conserve computational resources.

For all pipelines (cWB, oLIB, cWB+BW) the $h_{\rm rss}$value at which 50% detection efficiency (defined by a FAR less than 1/100 years) is achieved is reported in Table 6.3. Note that for *BayesWave* there are a number of waveform families that either did not achieve 50% detection efficiency (denoted by "N/A"), or were not analyzed by *BayesWave* at all (denoted by "-"). The waveforms that did not achieve the target detection efficiency can be explained by *BayesWave*'s prior range on Q for each wavelet. *BayesWave*'s prior only reaches up to $Q = 40$, thus presenting problems when searching for signals with higher Q values. The waveforms which *BayesWave* did not analyze at all fall into cWB's high frequency bin, and thus were not followed up by *BayesWave*.

Morphology	cWB	oLIB	BW
Gaussian pulses			
$\tau = 0.1$ ms	34	N/A	N/A
$\tau = 2.5$ ms	33	7.4	N/A
sine-Gaussian wavelets			
$f_0 = 70$ Hz, $Q = 100$	24	N/A	N/A
$f_0 = 153$ Hz, $Q = 8.9$	1.6	1.7	5.4
$f_0 = 235$ Hz, $Q = 100$	14	1.9	N/A
$f_0 = 554$ Hz, $Q = 8.9$	2.6	2.7	3.6
$f_0 = 849$ Hz, $Q = 3$	27	3.3	5.4
$f_0 = 1615$ Hz, $Q = 100$	5.5	-	-
$f_0 = 2000$ Hz, $Q = 3$	8.7	-	-
$f_0 = 2477$ Hz, $Q = 8.9$	11	-	-
$f_0 = 3067$ Hz, $Q = 3$	15	-	-
White-Noise Bursts			
$f_{low} = 100$ Hz, $\Delta f = 100$ Hz, $\tau = 0.1$ s	2.0	N/A	3.0
$f_{low} = 250$ Hz, $\Delta f = 100$ Hz, $\tau = 0.1$ s	2.2	N/A	9.2

Table 6.3: The $h_{\rm rss}$ values, in units of $10^{-22}{\rm Hz}^{-1/2}$, at which 50% detection efficiency is achieved at a FAR of 1 in 100 yr for each of the algorithms, as a function of the injected signal morphologies. "N/A" denotes that 50% detection efficiency was not achieved. "-" denotes the waveform was not analyzed by oLIB and BW because its characteristic frequency is higher than 1024 Hz. Table and caption reproduced from [16].

In [16] we also reported, for each pipeline, the minimum amount of energy emitted in GWs for a particular waveform morphology to be detected within a given search volume at 50% detection efficiency. The energy emitted can be roughly estimated as

$$E_{\rm GW} = \frac{\pi^2 c^3}{G} r_0^2 f_0^2 h_0^2 \tag{6.8}$$

where f_0 is the central frequency of the GW, h_0 is the average $h_{\rm rss}$, and r_0 is some chosen fiducial distance. This equation also assumes isotropic emission of GWs. We chose $r_0 = 10$kpc as the fiducial distance, but this could easily be rescaled to any

desired distance. The results of this energy limit are shown in Fig. 6.14. We see that the energy upper limits are more dependent on the detection algorithm rather than the particular waveform morphology. This is due to particular characteristics of each pipeline, such as prior ranges and search bin classifications. This also illustrates the benefit of having multiple pipelines search for unmodeled GWs– ensuring we cover as wide of a parameter space as possible. Finally, we note that compared to similar studies [8], our limits shown in Fig. 6.14 are roughly an order of magnitude stronger.

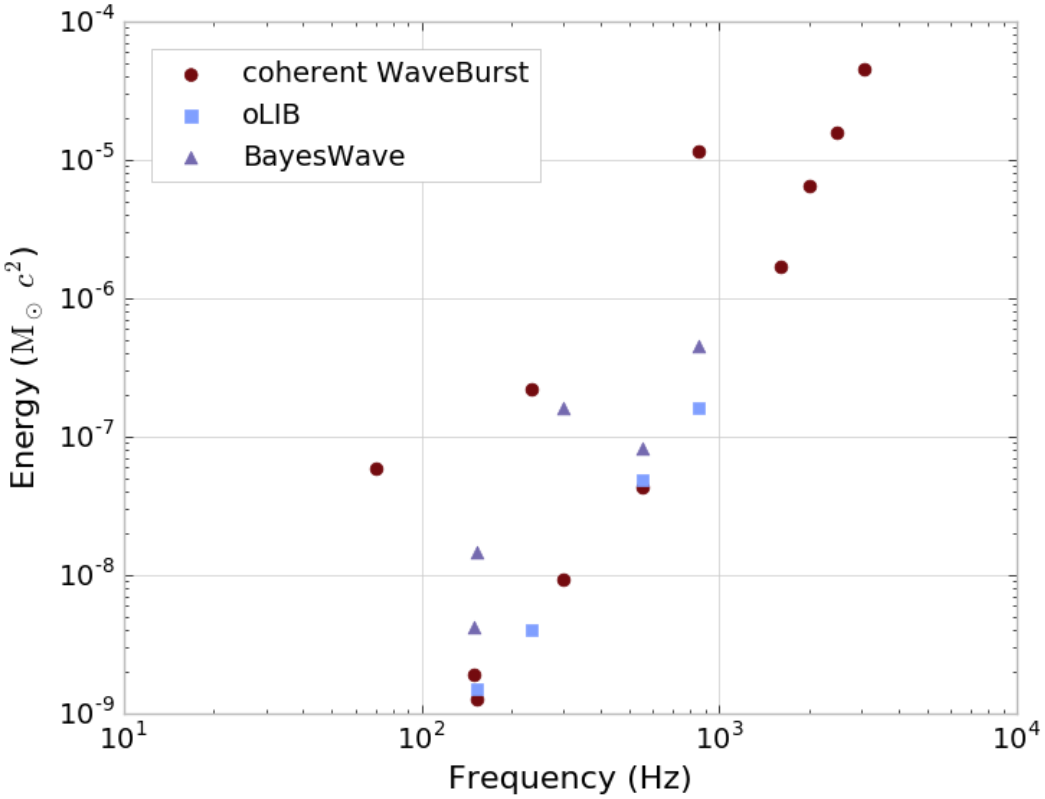

Figure 6.14: GW emission energy, in solar masses, at 50% detection efficiency for standard-candle sources emitting at 10 kpc for the non-GA waveforms listed in Table 6.3. Figure and caption from [16].

In addition to the ad-hoc waveforms, for the all-sky search we also used a set of BBH injections to test the pipelines' sensitive distances to these types of system.

We did this as an illustration of why the burst pipelines detected GW150914 with such high confidence, but did not detect the second GW in O1: GW151226 [12]. The heuristic explanation for this is that GW151226 is a lower mass event ($\sim 20M_{\odot}$) and a distance comparable to GW150914, which means there is less power emitted through GWs, and that power is spread out over a longer period of time. Burst searches are optimal for events that are localized in time, and so not as well suited to lower mass compact binary events. To show this a bit more rigorously, we used the SEOBNRv2 waveform generator [100] to produce a set of injected signals with a range of masses, distributed isotropically in orientation and sky location. The total masses of the systems were generated uniformly between 10 and 50 $M_{\odot}$. These masses are the redshifted mass of the system in the detector frame. The dimensionless spin vectors are distributed uniformly between 0 and 0.99. The mass ratios of the systems are three fixed values of 0.25, 0.5, and 1.0.

We then found the sensitive luminosity radii [85] of each pipeline to these BBH signals. The sensitive radius is the radius of a sphere with volume given by

$$V = \int 4\pi r^2 \epsilon(r) dr \tag{6.9}$$

where $\epsilon(r)$ is the detection efficiency averaged over sky location at a radius R. Then the sensitive radius R is simply

$$R = \left(\frac{3V}{4\pi}\right)^{1/3}. \tag{6.10}$$

The sensitive luminosity radii as a function of total mass is shown in Fig. 6.15. These radii are shown for each mass ratio (q) and spin range. These ranges serve as a general idea of what we expect, as the orientation or sky position of a system can greatly affect its detectability. However we can still see that a GW150914-like systems

which has a mass ratio of about unity, total mass $\sim 70M_{\odot}$, and is at a distance of $\sim$ 400Mpc falls into our sensitive luminosity radius. However for GW151226-like system that has a total mass of $\sim 20M_{\odot}$, we are barely sensitive up to 200 Mpc, which is much closer that the predicted GW151226 distance. This simple analysis helps explain why the burst searches did not detect GW151226, but still detected GW150914.

O2

LIGO's second observing run concluded in August of 2017. A similar report of search results to those from the O1 all sky search is currently in progress.

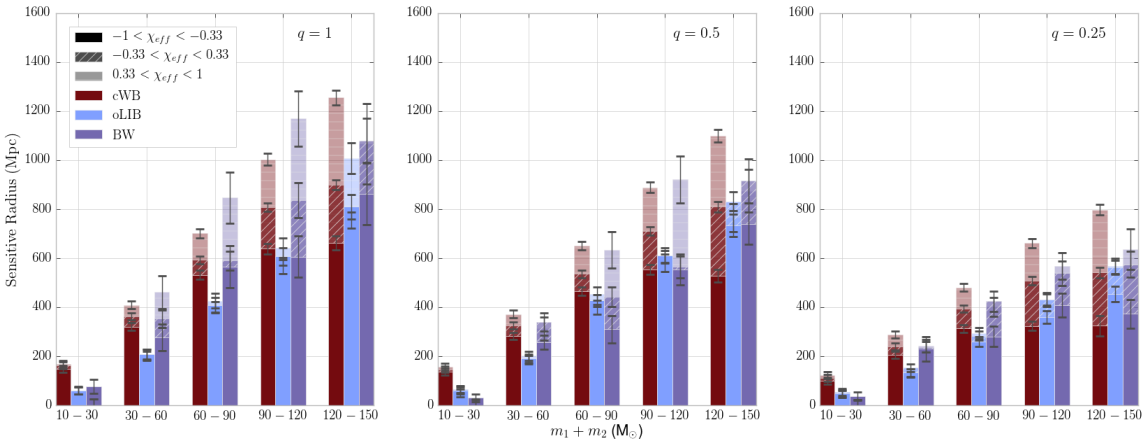

Figure 6.15: The sensitive luminosity radii for BBH signals for various mass and spin configurations. Figure from [16].

SIGNAL CHARACTERIZATION

In the previous chapter we demonstrated *BayesWave*'s ability to detect real GWs from astrophysical signals, and reject instrumental glitches. However this is only a small part of what *BayesWave* can really do. One of *BayesWave*'s other strengths is its ability to reconstruct GW waveforms with minimal assumptions. In this chapter we will look at of *BayesWave*'s powerful ability to reconstruct waveforms in the context of LIGO's first observational run, and what sorts of things we can do with these waveform reconstructions.

As described in Chapter 4, at each iteration in the chain *BayesWave* produces a waveform estimate from the number of wavelets being used and their parameters, along with the external parameters that allow the waveform to be projected onto the individual detectors' responses. While any fair draw of a waveform from this chain may provide information into the GW signal in our data, the real insight comes from looking at the whole collection of waveforms in the chain– our waveform *posterior distribution*. From this posterior we can construct a median waveform, along with credible intervals on the waveform. This feature sets *BayesWave* apart from other unmodeled waveform reconstruction methods, which typically only return a point estimate and must rely on extra analyses to provide error bars for their waveform reconstructions.

A shining example of *BayesWave*'s full reconstruction ability is shown in Fig. 7.1. This figure shows the waveform reconstructions in Hanford (top) and Livingston (bottom). The color scale represents the density of points, so the best way of thinking about these plots is as a histogram of $h(t)$ points at each discrete time step.

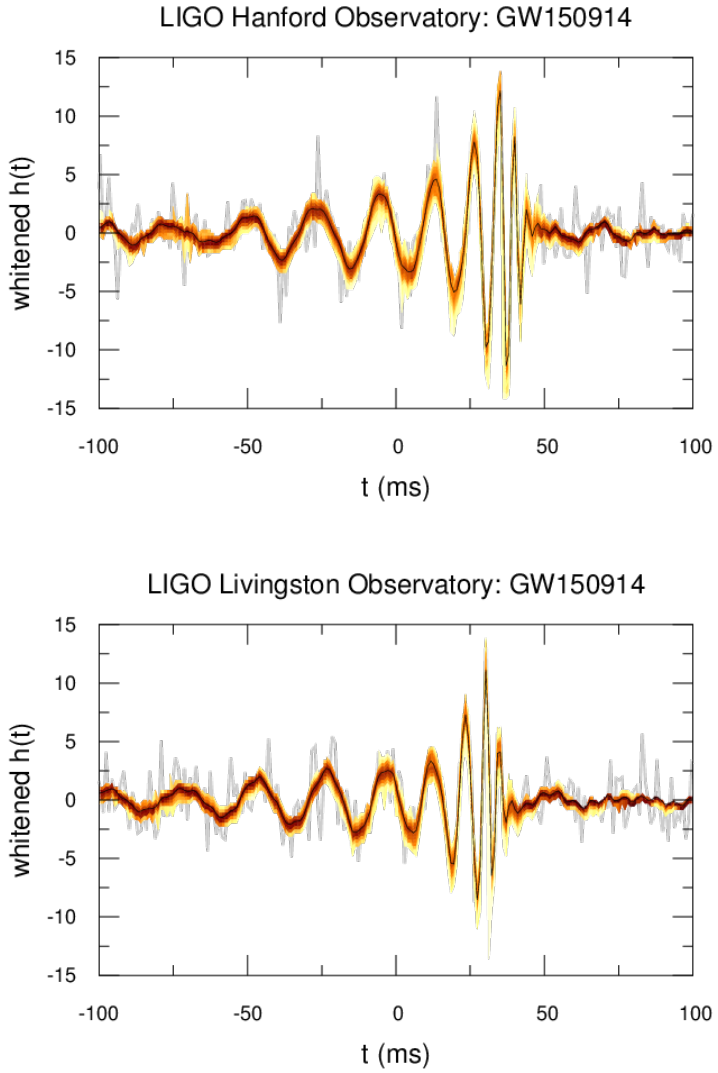

Figure 7.1: Posterior distributions for *BayesWave*'s reconstruction of GW150914 for the Hanford detector (top) and Livingston detector (bottom). The whitened data is shown in light gray. Figure courtesy of T. Littenberg.

Waveform Reconstruction Comparisons

Here we will compare the *BayesWave* waveform reconstructions to reconstructions using full templates from general relativity.

In LIGO and Virgo data analysis pipelines that specifically target compact binaries, once a trigger of sufficiently low FAR has been identified by one of the searches, full parameter estimation methods are implemented in order to get accurate estimates on the physical parameters of the system. Similar to the approach used by *BayesWave*, the full parameter estimation approach subtracts a waveform model from detector data and checks for consistency with the noise model. But rather than using a sum of wavelets as the model like *BayesWave* does, the PE approach uses waveforms from full general relativity templates, and the parameters are the physical parameters of the system. Complementary stochastic sampling methods are used to calculate the posteriors (as well as the evidence) including MCMC methods [93] and nested sampling [104]. At the end of the analysis we get a posterior distribution on the physical parameters of the system. These parameter distributions can in turn be used to form posterior distributions on the waveforms themselves, which we can compare to the *BayesWave* waveform posteriors.

This study was done for the GW150914 waveform distributions. These results can be seen highlighted in [15], and the primary figure of merit is shown in Fig. 7.2. We can see that the two waveforms match very closely. The 90% credible interval (CI) for the BBH modeled waveforms are narrower than those of the *BayesWave* reconstruction, but this is unsurprising as the priors on what the BBH waveform can look like are much tighter.

We can quantify the similarity between these two waveform reconstruction methods by calculating the *match*, as described in Eq 9.8. We find that the

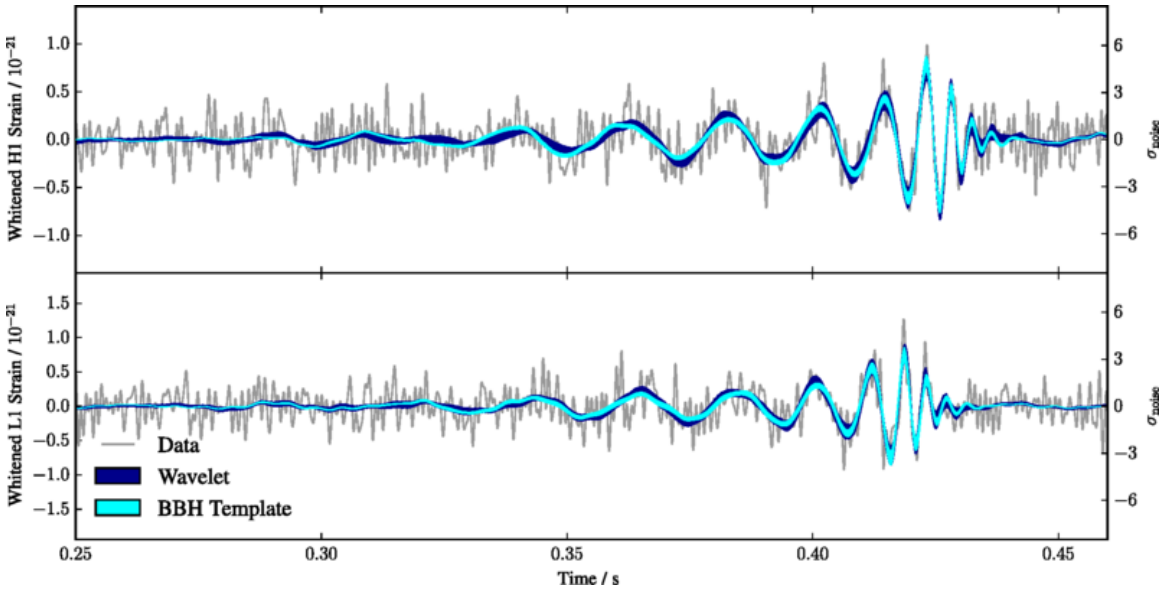

Figure 7.2: A comparison of the 90% CI for the modeled and unmodeled waveform reconstructions for GW150914, overlaid on whitened detector data. Figure from [15].

match between the two waveform reconstructions is $94^{+2}_{-3}\%$ [15]. It is important to note that we do not actually expect a 100% match between the two waveform reconstructions. The templated reconstructions assume a full waveform, and thus will tend to reconstruct more of the waveform in earlier (and potentially later) stages where *BayesWave* does not add wavelets. Further, we don't even expect the matches between waveform draws from the same analysis to achieve a 100% match. For a model of D dimensions, we can estimate the expected match as

$$E[M] = 1 - \frac{D-1}{2\mathrm{SNR}^2}. \tag{7.1}$$

Using Eq. 7.1 we can devise a rough estimate of what the expected match between draws from *BayesWave*'s waveform posterior will be. The typical number of wavelets used in the *BayesWave* reconstruction of GW150914 is $N_W = 6$ so including the 5 parameters per wavelet and the 4 extrinsic parameters the dimensionality is $D = 34$.

The recovered SNR is SNR = 24, and thus Eq. 7.1 gives us an expected match of $E[M] \simeq 0.94$. We test this by drawing pairs of waveforms from the *BayesWave*'s GW150914 posterior and calculating their matches. The results are in Fig 7.3, and the expected median match from the distribution is approximately 93%. So while this is a bit lower than predicted, it is generally consistent.

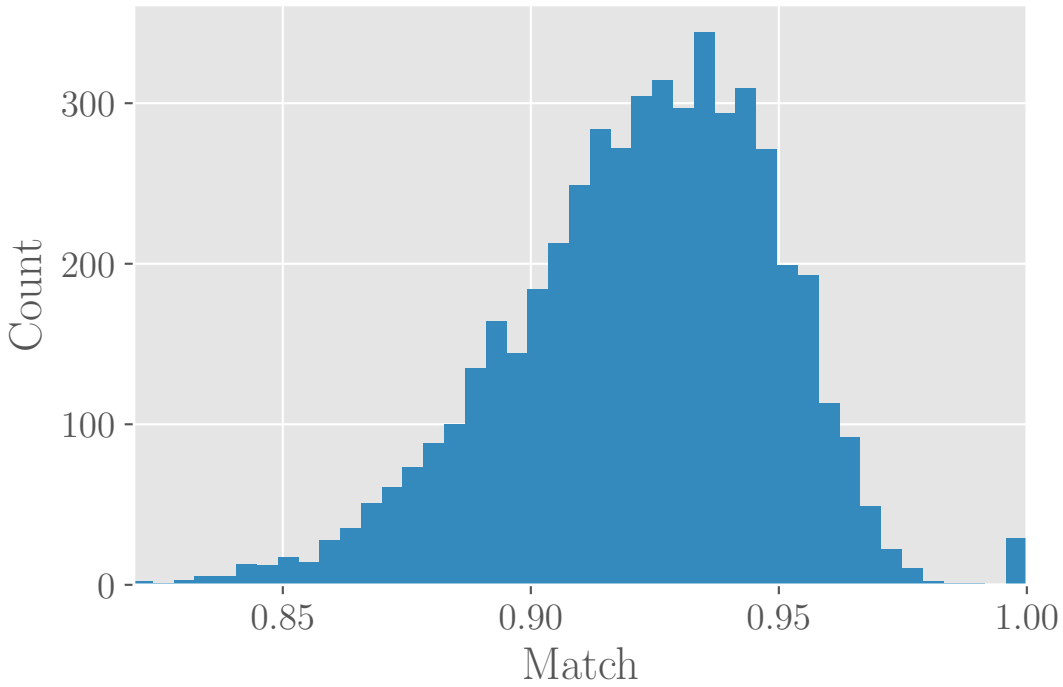

Figure 7.3: Matches between random draws from *BayesWave*'s waveform posterior from GW150914.

Another way to interpret whether the match between the *BayesWave* and templated reconstructions are to be expected is to use simulated BBH signals. In [14], we used a set of BBH injections to test the burst searches sensitivities to BBH mergers. Using this simulated data set we can also study what the typical matches between the injected and recovered (by *BayesWave*) signal is. The data set used included binary black holes mergers generated with the SEOBNRv2 waveform model [100]. The

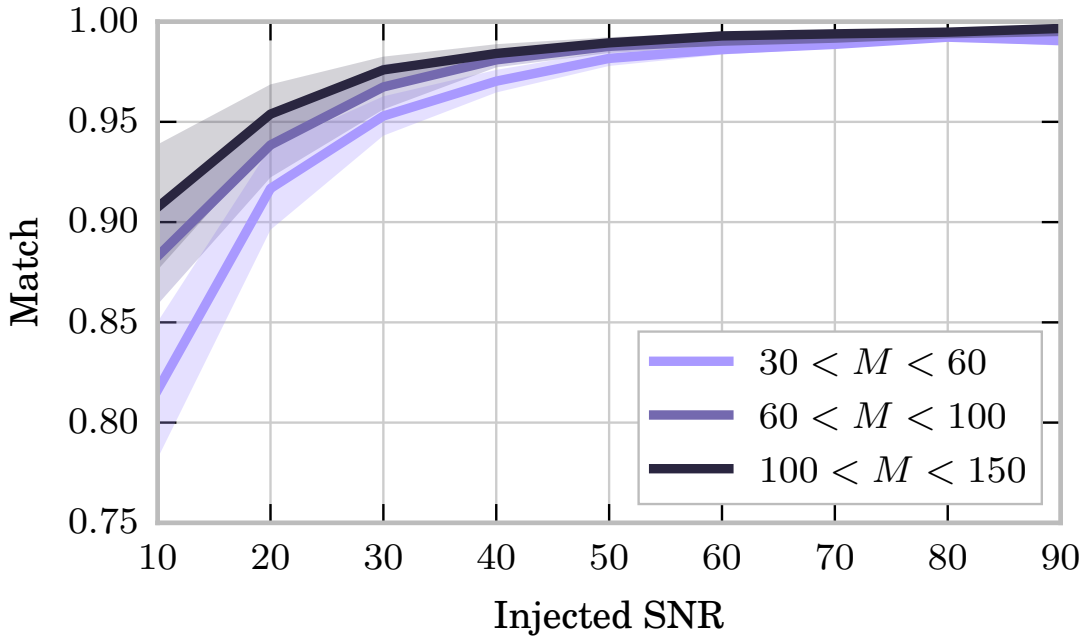

Figure 7.4: Calculated matches with 1σ error bars for a range of BBH injections. Figure from [15].

systems were distributed uniformly in the sky and oriented isotropically, at distances within 3.4Gpc. The total masses were between $30M_{\odot}$ and $150M_{\odot}$, at fixed mass ratios of 0.25, 0.5, and 1. The mass limits were chosen to be about a factor of two the estimated mass limit of GW150914 both on the lower and upper end. Finally, the dimensionless spin parameters were assumed to be aligned with the direction of the angular momentum of the binary system, with spin magnitudes drawn uniformly from a range of zero to one. As usual *BayesWave* followed up on the triggers produced by cWB for this data set.

Fig. 7.4 shows the match between the *BayesWave* reconstruction and the injected waveform as a function of SNR for three different mass ranges. The first feature to note is that the match increases with increased SNR. This is expected as the louder a signal is, the more of the fine details of the signal morphology will become resolvable in the waveform reconstruction. To put it in more Bayesian language, higher SNR means that the improvement gained in the likelihood when adding a new wavelet to map out the signal features is great enough to overcome the Occam penalty incurred for adding in dimensions. The second thing to note in Fig. 7.4 is that BBH signals with higher total masses tend to have higher matches. For higher mass binary systems, more of the GW power is concentrated in the merger stage, rather than earlier in the inspiral. This means the signal is more localized in time-frequency space, and starts to look a bit more like a sine-Gaussian signal. Fewer wavelets are required to reproduce such a signal, and thus we see higher matches for higher mass systems.

Finally we can use Fig. 7.4 to check that the match between the *BayesWave* and templated reconstructions for GW150914 are consistent with injections. The individual masses of GW150914 are estimated as $36^{+5}_{-4}M_{\odot}$ and $29^{+4}_{-4}M_{\odot}$, placing it in the $60-100M_{\odot}$ mass bin. The recovered SNR is about 24, so a quick by-eye check of Fig. 7.4 gives a predicted match of 95%, which is consistent with the 94^{+2}_{-3}% quoted

earlier.

One more further check between the consistency between *BayesWave* waveform reconstructions and actual signals in the detector data is to use *hardware injections*. Hardware injections are method of testing LIGO pipeline performances by physically displacing the test masses to mimic a GW signal. During the analysis of GW150914 a number of hardware injections were performed by injecting BBH signals with parameters drawn from the posterior distributions of the GW150914 parameters [32]. The burst pipelines picked up most of these injected signals, including the cWB+BW pipeline. We calculated the matches between the *BayesWave* recovered waveform and the injected waveform. We found that for the 28 GW150914-like injections *BayesWave* reconstructed, the median match was 94%. Again, this is quite consistent with the match between the *BayesWave* and full parameter estimation reconstructions of GW150914.

Overall we have strong confidence that the modeled and unmodeled reconstructions of the first GW detection are consistent. In LIGO's first detection of GWs, this comparison between the modeled and unmodeled waveform reconstructions served as an important sanity check. As we had never detected GWs from merging black holes, we had strong priors but no proof that our templates were correct. Using the *BayesWave* unmodeled reconstruction as a point of comparison gave us confidence that the waveform in the data truly matched the BBH predictions. The *BayesWave* reconstruction has now become a standard inclusion in GW detection figures of merit including GW150914 as we just discussed [13], but also GW170104 [18], and GW170814 [20].

Other Waveform Metrics

The consistency between waveform reconstruction methods is an important and natural test to perform for a GW detection. But we can think of other ways to test the consistency between these different detection methods. One way to do this is to use information about the signal morphology to perform some coarse parameter estimation without the use of full templates.

While the $h(t)$ waveform reconstruction contains all the information about a GW signal, it may not be in the most useful form. In Chapter 6 we discussed some waveform moments that can be used to characterize a signal. These waveform moments only depend on the waveform reconstruction, which means they can be calculated without assuming any underlying astrophysical model. We may, however, be able to use some of these moments to help constrain some of the physical parameters of the GW source. Here we describe how we did that for GW150914 (as seen in [14]). For BBH signals like this one we do indeed have very good models to constrain the system parameters, but it serves as a road map for how we may approach this sort of problem if we were to detect a signal we do not yet have templates for.

We generalized these moments a bit to give us *network moments*– one measurement that incorporates data from all of the detectors. This is done replacing $\rho(f)$ with

$$\rho_{\rm net} = \frac{\sum_i^{N_{\rm IFO}} \tilde{h}_i(f)^2}{\int \sum_i^{N_{\rm IFO}} \tilde{h}_i(f)^2 df}. \tag{7.2}$$

We can make predictions for these waveform moments by calculating the moments for a range of BBH signals with various total masses and mass ratios, and then compare these to the posterior distribution of moments from *BayesWave*. We tested this using the central frequency and bandwidth moments for GW150914 and the moments from various simulated waveforms. Fig. 7.5 shows the distribution of

BayesWave's recovered moments, with a grid showing the expected moments from various signals overlaid. We see that the estimated total mass and mass ratio ($71^{+5}_{-4} M_{\odot}$ and $0.82^{+0.17}_{-0.20}$ respectively) fall well within the 90% credible interval of the *BayesWave* posterior distribution.

As an extra check we performed this same test using the 29 hardware injections mentioned earlier, and found that 50% of the time the central frequency and bandwidth of the injected signals fell in the 50% credible interval of the posterior distribution, and 89% of the time they fell in the 90% credible interval. This is a solid indicator that our analysis is consistent[1].

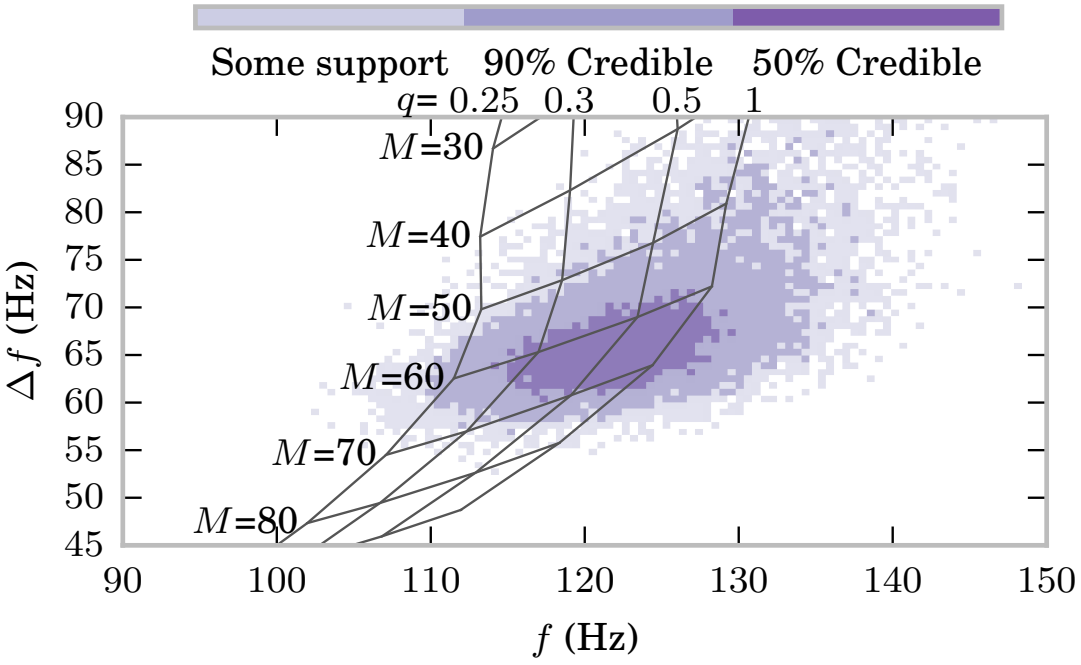

Figure 7.5: Distribution of waveform moments (central frequency and bandwidth) for the *BayesWave* reconstruction of GW150914. Overlaid is a grid of expected moments for various simulated signals. Figure from [14].

A further study of how consistently *BayesWave* can reconstruct these waveform

[1]Or one can think of it as passing the diagonal p-p plot test

moments for a variety of signals (both BBH and ad-hoc) can be found in [30]. This work is being further extended to use frequency moments to place upper limits on the characteristic mass of and distance to a GW source [31].

Frequency as a Function of Time

One final waveform check we implemented is a look at the frequency evolution as a function time for the recovered waveform.

Binary black hole systems have a characteristic "chirping" pattern of increasing frequency. This tell-tale chirp can sometimes be hard to see when just looking at the $h(t)$ waveform reconstruction. We developed a simple method for deriving the $f(t)$ track from a waveform reconstruction to more carefully examine the frequency evolution behavior. To do this, we look at the zero-crossings of the waveform. Fig. 7.6 shows a simple example "waveform" which in this case is a simple sine wave $\sin(2\pi t)$, with the times of the zero crossings marked by $t_{0...3}$. The frequency at any given time step t_i as

$$f(t_i) = \frac{1}{t_{i+1} - t_{i-1}}. \tag{7.3}$$

So for our example waveform in Fig. 7.6, $f(t_1) = \frac{1}{t_2 - t_0} = \frac{1}{1.5-0.5} = 1$ Hz, as expected. We implemented this $f(t)$ track finder into *BayesWave* postprocessing, from which we can calculate a median $f(t)$ of the reconstructed waveform along with credible intervals. Shown in Fig. 7.7 is the *BayesWave* reconstruction $f(t)$ for GW150914 in both LIGO detectors. *BayesWave* is able to reconstruct the signal from about $t = -0.1$ s to just after $t = 0$ s relative to the GPS trigger time. In Fig. 7.7 we can see the clear characteristic chirping pattern that we expect from a BBH signal. This figure also shows some of the limitations of our simple approach to reconstructing the frequency: we can get strange effects in the regions where

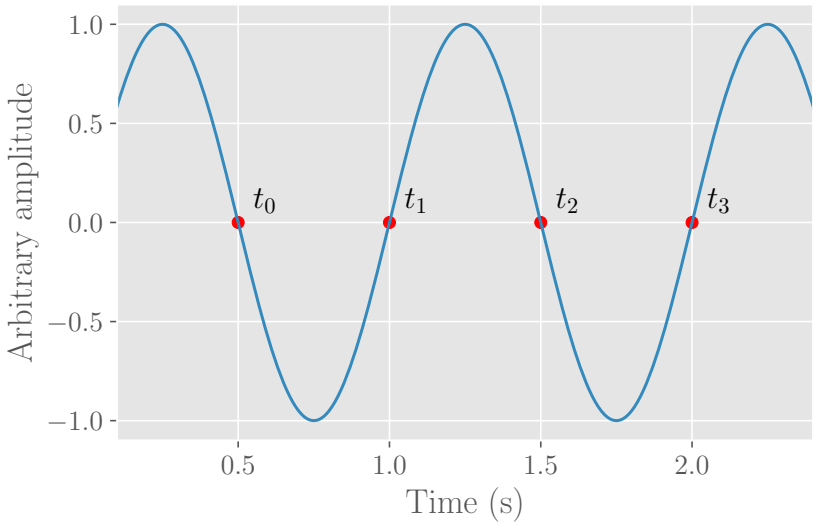

Figure 7.6: An example "waveform" for calculating an $f(t)$ track. In this case the wave is described by $\sin(2\pi t)$.

BayesWave is not reconstructing the signal well– for example the apparent spikes in Fig. 7.7 outside of the chirping region. So while we must be careful looking at these $f(t)$ tracks, they can still provide an informative illustration of the frequency evolution of the signal.

In the next chapter, we look at how we can use this chirping $f(t)$ track to develop a prior for *BayesWave* to improve upon the algorithm.

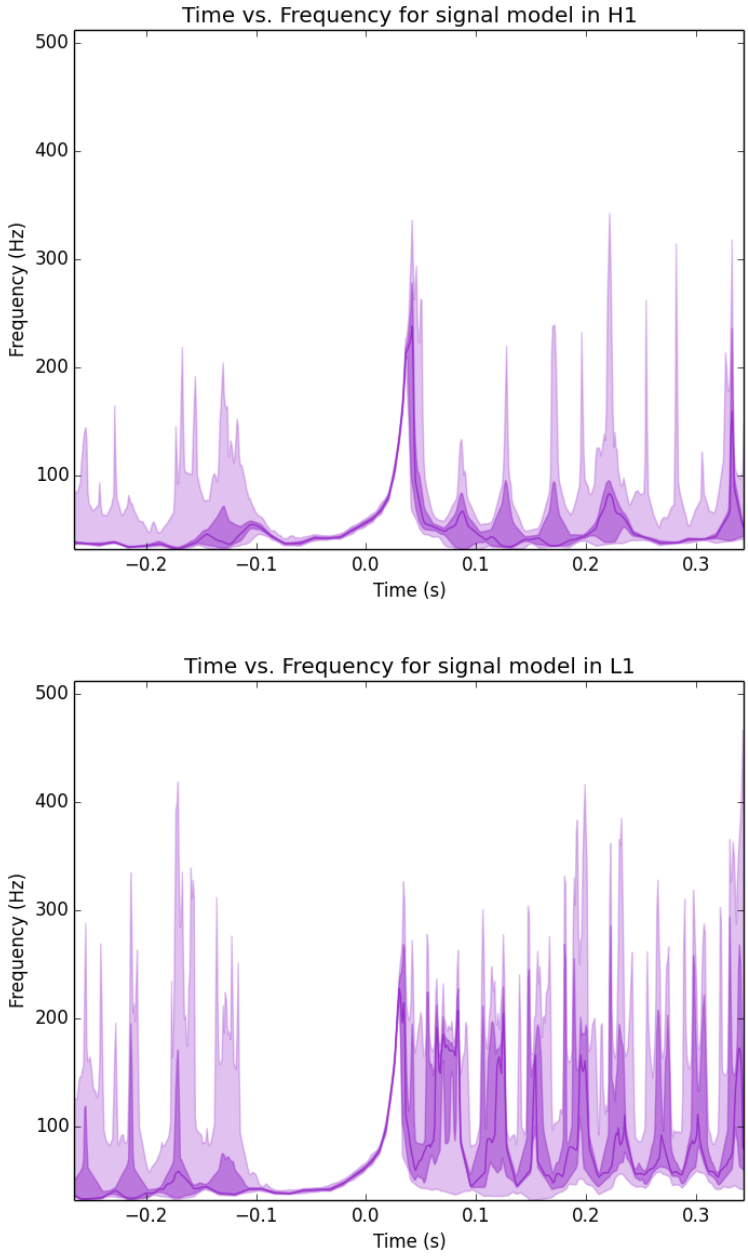

Figure 7.7: The $f(t)$ track from *BayesWave*'s reconstruction of GW150914 in Hanford (top) and Livingston (bottom). Shown is the median $f(t)$ track and the 50% and 90% credible intervals.

F(T) PRIOR

Modeled and Unmodeled Searches

LIGO's searches for transient GW signals is sometimes thought of as two distinct categories: modeled searches for which well studied templates exist, and unmodeled (or "burst") searches which are expected to catch any other signals. In reality it is more useful to think of search possibilities as a continuum from completely modeled to completely unmodeled. While there exist very good models for compact binary coalescences of roughly equal mass and non-extremal spin, there remain regions of parameter space that remain not as well modeled. There also exist predicted GW sources for which we have some idea of the form the signal will take, but not enough information to construct full templates. For example, the remnant of a binary neutron star (NS) merger is predicted to produce a GW signal, but this signal is highly dependent on the as-of-yet unknown NS equation of state and a trove of microphysics that nobody has fully included in numerical simulations yet.

While we have so far been thinking about GW searches in terms purely modeled or purely unmodeled, realistically searches lie on a continuum between those two categories. In this Chapter we will look at a particular application of a search and reconstruction method that lies in this in between area, by incorporating some prior knowledge of expected signals, but without relying on the full use of templates.

Morphology Priors

In the development of *BayesWave* two different priors on t_0 and f_0, the central time and frequency of a wavelet, were tested. The first is simply a uniform prior on the time-frequency space. This prior says that when proposing to add a new wavelet,

there is equal probability of placing it anywhere in the time-frequency volume being analyzed. This version of the prior was ultimately what was used in the *BayesWave* follow-up pipeline in O1.

Another proposed prior is a so-called clustering prior. This prior uses the assumption that any real GW signal (or even detector glitches) will be localized in time-frequency space. This means that new wavelets will have a higher probability of being placed next to existing wavelets. This clustering prior probability is explicitly given by

$$p(t_0, f_0) = \frac{1}{2\pi\sigma_f\sigma_t(\alpha^2 - \beta^2)} \left(e^{-(\Delta f_0^2/\sigma_f^2 + \Delta t_0^2/\sigma_t^2)/\alpha^2} - e^{-(\Delta f_0^2/\sigma_f^2 + \Delta t_0^2/\sigma_t^2)/\beta^2} \right) \tag{8.1}$$

where α gives the width of the distribution, and β gives the size of the hollowed out region around the particular wavelet. Details of this prior can be found in [44]

The clustering prior has been shown to enhance the capabilities of *BayesWave*, namely by increasing the match between the injected and recovered waveforms. The clustering prior is based off of an assumption that can apply to nearly any GW signal– namely that it will be localized in time-frequency space[1]. However, we can introduce a new class of priors which take into account the types of GW sources we expect to see in LIGO.

In the run up to aLIGO's first run, merging binary systems of compact objects (NSs, BHs) were the most promising sources of GWs. And as is well known now, the first two Observing runs did indeed result in the detections of several BBH systems and one BNS system. These events were detected with great confidence by both

[1]One may pose as a counterpoint to this argument GWs from eccentric binary systems, in which GW radiation is emitted during pericenter passage [75]. In this though the power is spread out over several bursts, the power of each individual burst over the timescale of a *BayesWave* analysis is localized in time-frequency space. Modifying *BayesWave* to search specifically for eccentric bursts could be an interesting investigation in the future.

templated searches and unmodeled searches.

Compact binary mergers follow a specific and well defined track in time-frequency space. We can use this information to put a prior on where in time-frequency space *BayesWave* places wavelets, as we expect the central times and frequencies of the wavelets used by *BayesWave* to generally follow the frequency path swept out by a GW signal from a compact binary merger. Fig. 8.1 is an example of this, and shows how the wavelets from one draw from the chain generally follow the predicted $f(t)$ evolution of the signal. Note that for this example and all examples to follow in this section we use a BBH system of component masses $30M_{\odot}$-$40M_{\odot}$ and spins 0.3 and 0.2 at an SNR of 25. By incorporating this information into a prior on t_0 and f_0 we can potentially pick up a signal more readily, and improve our waveform reconstruction.

These time-frequency tracks however depend on the physical parameters of the binary system, specifically the masses and spins of the system. While this may at first appear to be a complication, the dependency on system parameters in fact turns out to be beneficial. By setting the physical parameters of the system as parameters of the prior distribution– or *hyperparameters*– we can perform parameter estimation without the use of full templates. Therefore the advantage of using a prior based on physical setup of the system is twofold: we can increase our search sensitivity, and also return an estimate on the parameters of that system.

Functional form of $f(t)$

Before using the $f(t)$ track to construct a prior, we must first form a method of calculating the $f(t)$ track itself. To do this we will look at how GW waveforms are commonly constructed. One of the standard methods is using a phenomenological model for the inspiral and coalescence of a binary black hole system.

This phenomenological method, described fully in [95], matches waveforms from post-Newtonian (PN) inspiral models with results from numerical relativity (NR) simulations for late inspiral and merger.

The PN waveforms for early inspiral are based off of an expansion in the orbital velocity v, and is accurate when v is much less than the speed of light [33]. In the late inspiral and merger phase of the binary evolution, the orbital speed increases and the PN approximation breaks down. After this breakdown, the most accurate waveform models come from NR simulations of the Einstein field equations [28, 35, 89]. The phenomenological model takes advantage of both of these waveform simulations by using PN for early inspiral and matching the two waveforms models at the last few cycles before merger where the PN approximation starts to break down.

The $f(t)$ track we use as the starting point for our prior uses the PhenomC [23,95] waveform. We first construct $f(t)$ using a purely PN formalism, going up to 3.5PN order and including spins.

First we write the energy $\mathcal{E}$ of the binary system as an expansion in the parameter x, which is related to the orbital frequency of the system by $x = (M\omega)^{2/3}$:

$$\mathcal{E}(x) = \frac{-x\eta}{2} \sum_{k=0}^{6} e_k x^{k/2}, \tag{8.2}$$

where η is the symmetric mass ratio of the binary system. We also use the flux of the system[2]:

$$\mathcal{F}(x) = \frac{32\eta^2 x^5}{2} \sum_{k=0}^{7} f_k x^{k/2}. \tag{8.3}$$

As the rate of energy loss is equal to the flux, we can formulate an expression

[2]We note that this expression is not actually a pure polynomial. The 3PN ($k = 0$) coefficient actually has a $\log(x)$ term in it. See [95].

for the evolution of the orbital frequency over time [33, 38, 74]

$$\frac{dx}{dt} = \frac{-\mathcal{F}(x)}{d\mathcal{E}(x)/dt}. \tag{8.4}$$

By inserting Eqs. 8.2 and 8.3 into Eq. 8.4, yields an expression for dx/dt:

$$\frac{dx}{dt} = \frac{64\eta x^5}{5} \sum_{k=0}^{7} a_k x^{k/2}. \tag{8.5}$$

Integrating this equation and replacing x with its expression for frequency gives the 3.5PN expression for $f(t)$. These full coefficients can be found in Appendix D.

While the first attempts at this prior sought to use only the PN approximation, we found that this was insufficient. As t approaches the time of coalescence t_c, the $f(t)$ track from the pure PN approximation differs significantly from the $f(t)$ track found from the full PhenomC waveform. This leads to a significant biasing in the recovered parameters. So in addition to the PN $f(t)$ approximation used for the inspiral, we needed to devise an $f(t)$ expression that uses the analytical expression that was derived from the NR waveforms. By using the Stationary Phase Approximation (SPA) we obtained an expression for $f(t)$ (technically $t(f)$) in terms of the coefficients found from NR matching. Details of this calculation can be found in Appendix A.

Prior Details

Now that we have an expression for the frequency evolution of a compact binary in time, we wish to turn that into a prior. In order to do this, we take our $f(t)$ track and add a bit of uncertainty to it by making a Gaussian type distribution around a central $f(t)$ track. The form of this prior at any given time and frequency (t, f) is

$$p(t, f) = A \left(e^{-\Delta s_1^2/\alpha^2} + e^{-\Delta s_2^2/\alpha^2} \right), \tag{8.6}$$

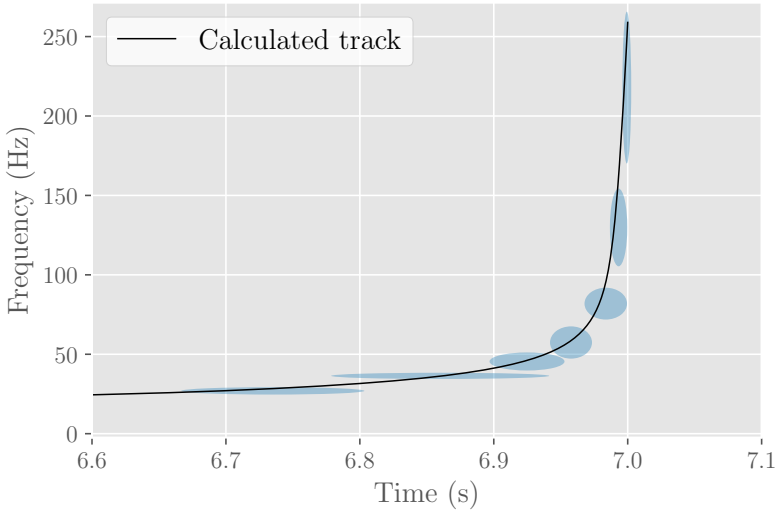

Figure 8.1: An example of a set of wavelets used by *BayesWave* from one chain draw (blue), overlaid on the calculated $f(t)$ track. The example system used is a $30M_{\odot}$-$40M_{\odot}$ mass system with spins 0.3 and 0.2 at an SNR of 25.

where A is the overall normalization constant, α is the overall width, and Δs_1 (Δs_2) give the the distance away from the $f(t)$ track in frequency (time). More explicitly,

$$\begin{aligned} \Delta s_1^2 &= \frac{Q^2}{4f_*^2}(f - f_*)^2 \\ \Delta s_2^2 &= \frac{4\pi^2 f^2}{Q^2}(t - t_*)^2 \end{aligned} \tag{8.7}$$

where f_* (t_*) is the $f(t)$ ($t(f)$) that lies on the track, and Q is the quality factor. A graphical representation of these quantities is in Fig. 8.2.

While so far we have been expressing this prior as a prior in time-frequency space, recall that the full $f(t)$ track expansion is determined by the physical parameters of the system. Our prior can more suggestively be written as $p(t, f|\vec{\lambda})$, where $\vec{\lambda} = (m_1, m_2, \chi_1, \chi_2, t_c)$ represents all of the parameters of the binary system that uniquely determine the track. These parameters of the prior are called *hyperparameters*, and

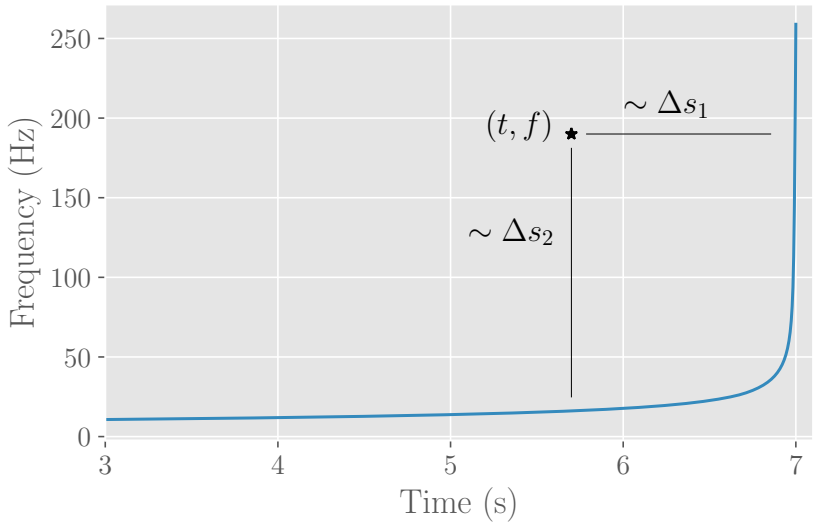

Figure 8.2: Illustrative example of how the prior in Eq. 8.6 is derived from an $f(t)$ track.

the priors on the hyperparameters are *hyperpriors*.

A straightforward example of a set hyperparameters is a simple normal distribution. Say we want to measure some parameter x which we *a priori* believe to be described by a normal distribution,

$$p(x|\mu, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x-\mu)^2/2\sigma^2}. \tag{8.8}$$

Here μ and σ are the expectation value and standard deviation, respectively, of the system, and are also the hyperparameters of the prior distribution.

In our $f(t)$ track prior, we will not keep our predicted $f(t)$ fixed to one particular BBH system, but allow our hyperparameters to vary. Throughout the MCMC, the likelihood, which is dependent on the data, should tend to "pull" the prior towards the true $f(t)$ track, while the prior will keep wavelets close to that track thus improving the waveform reconstruction. The distributions of the hyperparameters will also

provide parameter estimation. Thus we will be able to provide estimates of the masses and spins of the BHs or NSs *without the use of full waveform templates*. While we do not expect the posterior distributions of the parameters with the $f(t)$ prior approach to have quite the same accuracy as the fully modeled approach used by LIGO, this template-less approach to parameter estimation is a novel approach in LIGO data analysis. In addition, these parameter PDFs come almost as a free bonus in the normal course of running *BayesWave*.

Normalization The final piece of our prior to put together is the overall normalization. As the prior is a probability distribution, it is imperative that the sum total of probabilities at all points in the time-frequency volume being analyzed is equal to one. Since our prior is not a normal Gaussian distribution, the normalization is non-trivial. Additionally, because our prior is dependent on the hyperparameters and therefore potentially different at every step in the chain, it is not practical to numerically calculate the overall normalization constant at every step.

The easiest solution we found was to use a two part prior for normalization. To do this we devise some specific time t_1 before which we use a prior based only on the Δs_2 metric, and after which we use a prior based only on the Δs_1 metric. This crossover time is determined by when the widths of the two parts of the priors are equal. Equating the expressions in Eq. 8.7, we get that this occurs when

$$\frac{1}{f^2}\frac{df}{dt} = \frac{4\pi}{Q^2}. \tag{8.9}$$

This specific crossover point is found using a simple root-finding routine. In certain instances this two-part prior approach leads to slight discontinuities in the prior, as can be seen in some of the example priors in Fig. 8.3. However as long as it is normalized, a prior does not necessarily need to be continuous, and so this

discontinuity does not lead to a major issue.

Prior Uncertainty The parameter α in Eqns. 8.7 set the over width– or "uncertainty" of our distributions. A larger α leads to a wider prior and thus more uncertainty. The choice of alpha depends somewhat on what is trying to be accomplished from the use of the prior, and how much confidence we have in the $f(t)$ track used as the base of the prior. A set of example prior distributions with various α's is shown in Fig. 8.3.

One argument for using a wider prior is if we have some reason to believe the base $f(t)$ track is not totally correct. For example, before LIGO's first detection of GW150914, we might have wanted to be ready for any sort of surprise that might come in the waveform. In this case a wider prior might be preferred in order to capture any possible signal. Another example is a binary system with precession. While the basis of our $f(t)$ prior is based on a binary system that includes spin, it does not include the possibility of precession. Again a wider prior would be beneficial in this case to capture a potentially precessing system.

An argument for smaller uncertainties is more precise parameter estimation. A narrow prior should restrict the region of parameter space explored, and thus give tighter bounds on the posterior distributions for our hyperparameters. A limitation of this however is that any biases in parameter recovery will be amplified, and give incorrect estimates for the masses and spins of the system. Details of this limitation are explain in further detail later.

Results

As mentioned earlier there are two main results we seek to test with the $f(t)$ prior. The first is an increase in sensitivity and waveform reconstruction. The second is an estimate of the physical parameters of the system. Here we summarize where

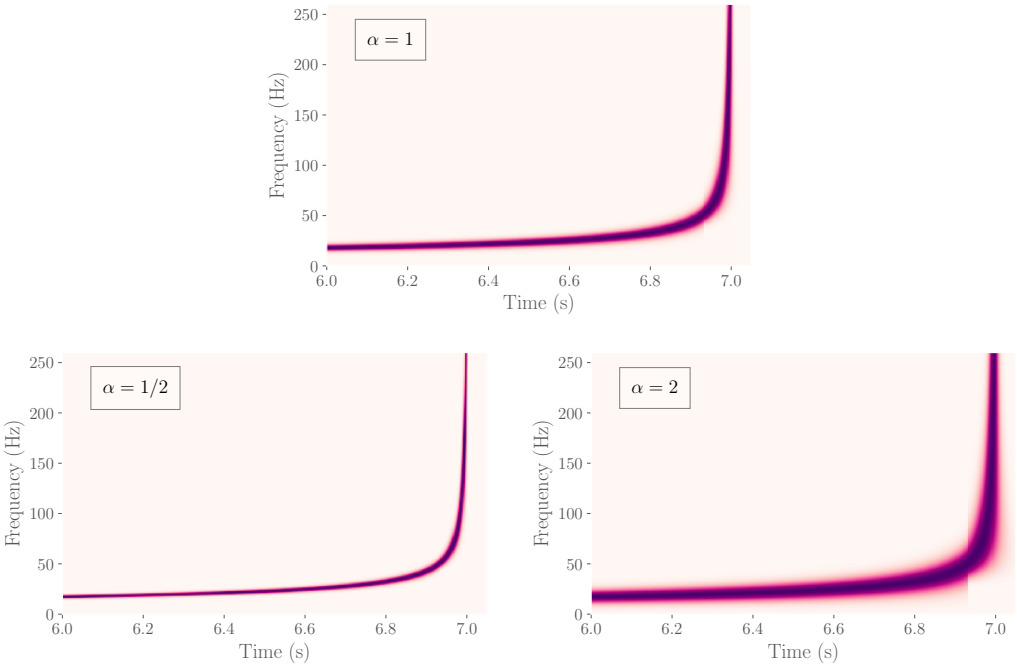

Figure 8.3: $f(t)$ prior for the 30-40$M_\odot$ system at a few different uncertainty levels (α's). The slight discontinuity visible is from the different calculation methods used to assure normalizability.

we succeed and where we fall short of meeting these goals.

Waveform Reconstruction In order to test improvements in waveform reconstruction when using the $f(t)$ prior, we look at the match between the injected waveform and the waveform recovered by *BayesWave*. The match is define as

$$M = \frac{(h|\bar{h})}{\sqrt{(h|h)(\bar{h}|\bar{h})}} \tag{8.10}$$

with h the recovered waveform, and $\bar{h}$ the injected waveform. The match ranges from -1 for a perfectly out of phase signal to 1 for identical signals.

It is important to have good matches between waveforms so that we reconstruct all our signals as accurately as possible. Additionally, the loss in detection goes like $1 - M^2$ so mismatched waveforms can actually lead to losses in detection. In particular, we seek to improve our waveform reconstructions in the lower SNR region as much as possible, as this is where burst detection and reconstruction methods have traditionally struggled the most.

The approach to testing this is to inject a set of BBH waveforms generated with PhenomC into simulated noise, and try to recover these signals using *BayesWave* with and without the $f(t)$ prior. The set of BBH injections we used are over a range of SNR in order to investigate what regions of parameter space the $f(t)$ prior preforms well in. The results of these runs are shown in Fig. 8.4. These results clearly show that using the $f(t)$ prior leads to significant improvement in waveform reconstruction. At all injected SNR values the average match is higher when using the $f(t)$ prior as opposed to the flat prior. This difference holds even for lower SNR events. As stated earlier, this is the region where unmodeled searches typically struggle the most, and thus where improvement is most promising.

The matches for both versions of the prior both show higher matches at higher

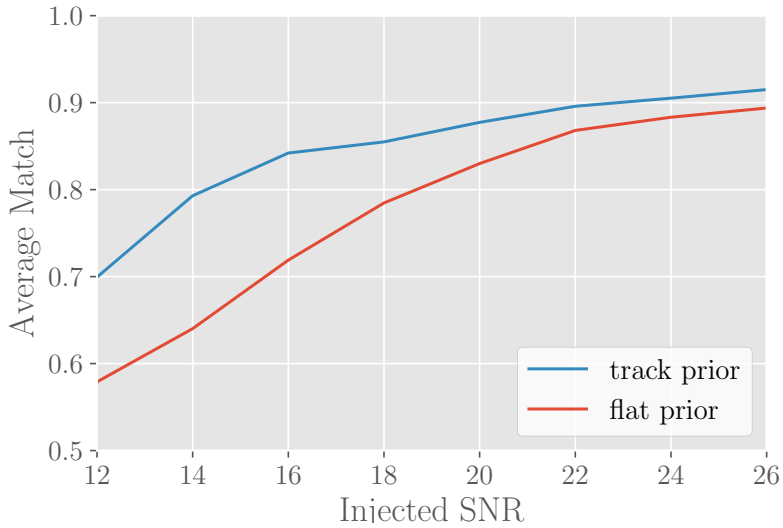

Figure 8.4: Average waveform match for a range of SNR with the $f(t)$ prior turned on and off, using the same BBH system as the rest of this section. The $f(t)$ prior has clear improvements for the Match, particularly at lower SNR.

SNR's. This is due to the fact that at lower SNR's the signal is harder to distinguish from the noise, and the improvement in the likelihood when adding a new wavelet does not justify the the penalty incurred by adding a more dimensions with new wavelets.

Overall, using this prior based off of an expected time-frequency evolution for a BBH system shows great improvement for waveform reconstruction. This is not only promising in terms of maximizing our ability for waveform reconstruction, but serves as an example as to how a prior like this may benefit searches for other kinds of signals. Beyond compact binaries, there are a number of predicted GW sources that are not as well modeled– a good example of these are supernovae (SNe) signals. For SNe signals there are some simulations that give a general idea of what the time-frequency evolution will look like. Using a prior like the one described in this chapter could be very beneficial for a signal like a SNe where we have some information, but

not enough for a fully templated search.

Parameter Estimation Beyond improving our search and waveform reconstruction abilities, one of the main goals of this parameterized prior is to do some level of parameter estimation. As stated earlier, parameter estimation without templates is a novel approach. And though we do not expect this method to replace templated methods, it could still serve as a complementary and preliminary approach.

In order to test the parameter estimation we used the same example BBH system used for the rest of this chapter, injected with SNR = 25. We tested three different widths (α parameter) of the $f(t)$ prior to examine what effect that had on parameter recovery. For our hyperpriors on the physical parameters of the system, we used uniform component masses of $m_i \in [10, 100]M_\odot$, spin parameters uniform over $\chi_i \in [-1, 1]$, and t_c uniform over the time being analyzed.

Fig. 8.5 shows the posterior distributions of the parameters of the $f(t)$ prior, along with the true (injected) parameter. The parameters we have chosen to look at are the chirp mass $\mathcal{M}_c = \frac{(m_1 m_2)^{3/5}}{(m_1+m_2)^{1/5}}$ (which is a commonly used parameter in GW data analysis because it is well constrained), total mass $M = m_1 + m_2$, time of coalescence t_c, and dimensionless spin parameter $\chi = \frac{m_1\chi_1}{M} + \frac{m_2\chi_2}{M}$. These are quantities commonly looked at during parameter estimation investigations. We can easily see that the true parameter value lies within the posterior distributions. This shows that we are successful in recovering our parameters.

There are a couple features to note in these posterior distributions. First, the α parameter chosen has a strong influence on spread of our PDFs. A more conservative choice of α unsurprisingly leads to a tighter posterior distribution. We note though that an infinitesimally wide $f(t)$ prior would not actually lead to infinitesimally wide posterior distributions. As discussed in Chapter 4, the wavelets themselves

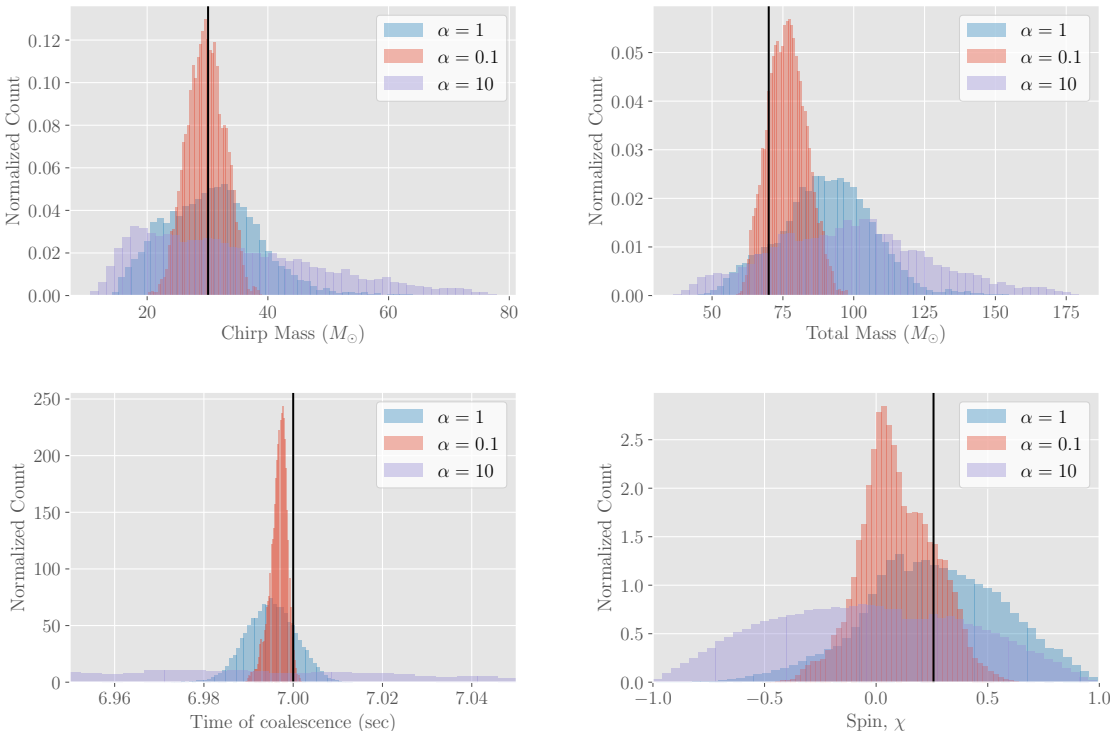

Figure 8.5: Posterior distributions for the intrinsic parameters $\mathcal{M}_c$ (top left), M (top right), t_c (bottom left), and χ (bottom right). Three different values of α, the uncertainty parameter in the prior, are shown. A smaller uncertainty produces a narrower posterior distribution.

have widths in time and frequency that are constrained by the Heisenberg-Gabor uncertainty principle. So at a certain point, we will be limited by the wavelet uncertainty, instead of our prior width.

Besides this wavelet width limitation, we notice that there is another limit on our parameter estimation efforts. In Fig. 8.5, we see that for the narrow prior choice, the posteriors tend to peak away from the injected value. This happens even in the case where we injected a very high SNR signal (e.g. SNR $\simeq$ 100), a situation in which we would expect to recover our parameters very well. To investigate why this happens we look at where the the central times and frequencies of the wavelets chosen by *BayesWave* lie in relation to the time-frequency track from the injected signal. An example of this is shown in Fig. 8.6. We see that the wavelet centers tend to actually fall a bit below the true time-frequency track. What this means is that while the *sum* of the wavelets may accurately reproduce the true time-frequency track, the individual wavelet t_0's and f_0's themselves will not reproduce the true track. Thus, our parameter estimations will be slightly biased.

While we have shown here that we can in fact use a hyperprior to perform parameter estimation with a burst analysis, there are limitations to the accuracy of this method. Nonetheless, we have shown that our hyperprior helps increase the match between the injected and recovered waveforms, which will help with GW detection.

Next we will look at how changing the frame of *BayesWave* could aid in detecting signals like BBHs.

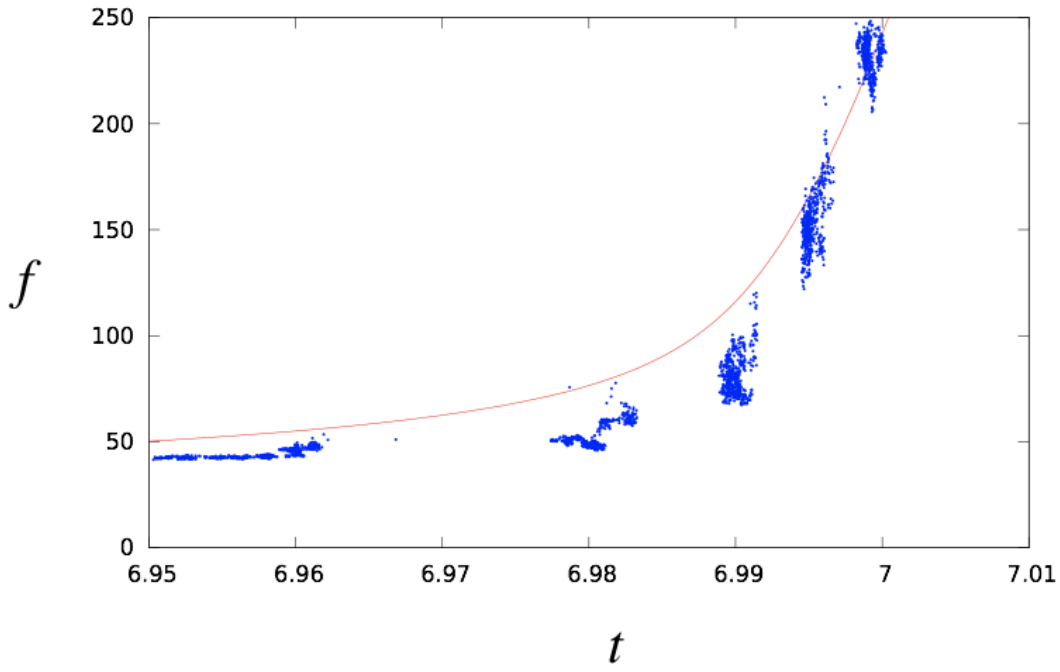

Figure 8.6: The central times and frequencies from draws from the *BayesWave* chain (blue dots), compared to the true $f(t)$ track. We use the same masses and spins as the examples in the rest of this chapter, but for this example we look at an extreme case where SNR = 120.

CHIRPLETS

Introduction

In the previous Chapter, we looked at how we can improve the waveform reconstruction capabilities of *BayesWave* by changing the prior on the wavelet placement. There are other possible approaches to improving *BayesWave*'s sensitivity and to potentially target certain classes of signals. One path of improvement is to change the basis of reconstruction used by the RJMCMC.

As discussed earlier, the *BayesWave* algorithm currently uses as its reconstruction basis sine-Gaussian wavelets. We have seen that these wavelet's variable shape in time-frequency space allow them to reconstruct GW events with good accuracy. However, there may be bases that could perform even better, either across the board or specifically for certain classes of GW signals.

In this Chapter we introduce a new *BayesWave* reconstruction basis of "chirplets"– wavelets with linear frequency evolution [79]. This new basis has even more flexibility than the sine-Gaussian wavelets, while still making minimal assumptions about the form of the GW signal. We show that for events with lower signal-to-noise ratios this more flexible basis improves waveform reconstruction, and uses fewer overall parameters to do so.

Chirplets Frame

A defining characteristic of *BayesWave* is that the number of wavelets is a parameter in the model, with the natural parsimony of Bayesian model selection favoring representations with the smallest dimensionality that can adequately describe the data. In the case of *BayesWave*, the dimensionality of the model (the number

of wavelets used) is explored using a trans-dimensional Markov Chain Monte Carlo algorithm. Additional wavelets are added only if they provide a sufficiently large boost to the likelihood. A natural extension to consider is if there's a way to either better model the data with a comparable number of parameters, or use fewer parameters to fit the data just as well. One possible way to do this is to reconstruct the data in a different way, such as using a different wavelet representation.

The Morlet-Gabor sine-Gaussian wavelets currently used by *BayesWave* form an over-complete basis, or frame, that can reconstruct any possible signal [65]. They have a simple analytic representation in the Fourier domain, making it easy to include time shifts in the detector response, and allowing for efficient calculation of the likelihood function. But there may be other frames that are able to reconstruct signals more efficiently.

In choosing a new wavelet frame for *BayesWave*, we consider what types of gravitational-wave signals we might detect. Many astrophysical sources of GWs have instantaneous frequency evolution, most notably mergers of compact binary objects. Because of this, we might expect that using a basis function that itself includes frequency evolution could better reconstruct GW signals. The simplest way to incorporate changing frequency into *BayesWave* is to add a linear frequency evolution to the Morlet-Gabor wavelets, producing a function known as a chirplet [79].

In the time domain, chirplets can be expressed as

$$\Psi(t; A, f_0, \dot{f}_0, Q, t_0, \phi_0) = Ae^{-\Delta t^2/\tau^2} \cos(2\pi f_0 \Delta t + \pi \dot{f}_0 \Delta t^2 + \phi_0) \tag{9.1}$$

where $\tau = Q/(2\pi f_0)$ and $\Delta t = t - t_0$. Here $\dot{f}_0$ represents the frequency evolution and is the linear frequency derivative at time $t = t_0$. In the limit that $\dot{f}_0 = 0$, this expression reduces to the expression for Morlet-Gabor wavelets. Chirplets can have

either $\dot{f}_0 > 0$ (chirping), or $\dot{f}_0 < 0$ (anti-chirping). Time domain plots of a chirplet and a wavelet are shown in Figure 9.1.

The same characteristics that make Morlet-Gabor wavelets a good frame are also true for chirplets: they are continuous, occupy minimal time-frequency volume, and can be expressed analytically in the Fourier domain, though with a slightly more complicated expression:

$$\begin{aligned}\Psi(f; A, f_0, \dot{f}_0, Q, t_0, \phi_0) =& \frac{A\sqrt{\pi}\tau}{2(1+\pi^2\beta^2)^{1/4}} e^{-\frac{\pi^2\tau^2\Delta f^2}{1+\pi^2\beta^2}} e^{-2\pi i f t_0} (e^{i(\phi_0+\delta-\pi^3\beta\tau^2\Delta f^2)/(1+\pi^2\beta^2)} \\ &+ e^{-Q^2 f/f_0} e^{-i(\phi_0+\delta-\pi^3\beta^2\Delta f^2)/(1+\pi^2\beta^2)})\end{aligned} \tag{9.2}$$

where $\Delta f = f - f_0$, $\delta = \frac{1}{2}\arctan(\pi\dot{f}_0\tau^2)$, and we have introduced the dimensionless parameter $\beta = \dot{f}_0\tau^2$. For the remainder of this paper we will us β as our chirp parameter.

In time-frequency space, wavelets can be represented by ellipses whose principle axes are aligned with the time and frequency axes. Similarly, chirplets can be represented by a tilted ellipse. The equation for the ellipse is

$$(1+\pi^2\beta^2)x^2 + \pi^2 y^2 - 2\pi^2\beta xy = 1 \tag{9.3}$$

where we have introduced the dimensionless variables $x = \Delta t/\tau$ and $y = \tau\Delta f$. In terms of these new coordinates, the ellipse is tilted with respect to the time axis by the angle

$$\theta = \frac{1}{2}\arctan\left(\frac{2\pi^2\beta}{\pi^2(1-\beta^2)-1}\right) \tag{9.4}$$

The ellipse has area $1 + \mathcal{O}(\beta^4)$. A spectrogram of a variety of chirplets is shown in Figure 9.2, confirming that in time-frequency space chirplets appear as ellipses with

a variety of orientations.

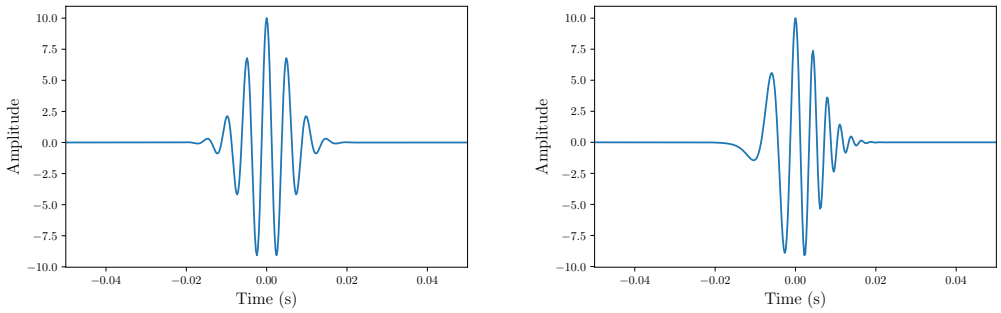

Figure 9.1: Examples of a wavelet (left) and chirplet (right) in the time domain. For both examples $f_0 = 200$, $t_0 = 0$, and $Q = 10$. In the chirplet example $\beta = 0.8$

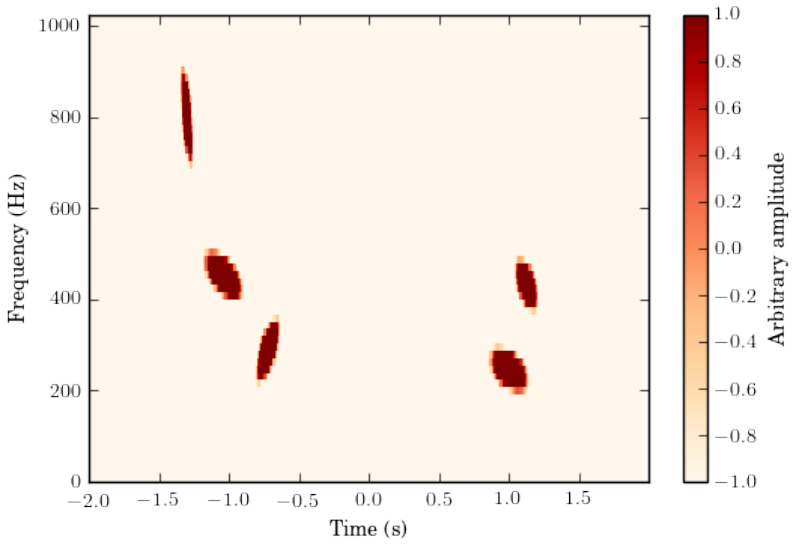

Figure 9.2: Spectrogram of chirplets with a range of central times, frequencies, Q and β.

Methods

Bayesian inference requires the specification of a likelihood and prior, and a method to compute the posterior distribution and model evidence. Since the

replacement of sine-Gaussian wavelets by chirplets only adds one new parameter to the frame functions, the implementation is almost identical to the original *BayesWave* algorithm [44]. In our study we started with the version of *BayesWave* used in the second advanced LIGO observation run, which differs from the original in the choice of priors, and in some of the proposal distributions used to evolve the MCMC algorithm.

Priors

We use uniform priors on Q ($Q \in [0.01, 40]$) and ϕ_0 ($\phi_0 \in [0, 2\pi]$). We also use a uniform prior on f_0 and t_0 over the time-frequency volume being analyzed.

The prior on the amplitudes of the individual wavelets (chirplets) is given as a prior on the SNR of the individual frame functions. For an individual wavelet or chirplet, the SNR is estimated as:

$$\mathrm{SNR} \simeq \frac{A\sqrt{Q}}{\sqrt{2\sqrt{2\pi} f_0 S_n(f_0)}}. \tag{9.5}$$

The prior on the SNR for wavelets in signal model is the same as described in Refs. [44, 73]:

$$p(\mathrm{SNR}) = \frac{3\mathrm{SNR}}{4\mathrm{SNR}_*^2 \left(1 + \frac{\mathrm{SNR}}{4\mathrm{SNR}_*}\right)^5} \tag{9.6}$$

For the glitch model, the prior on the SNR is

$$p(\mathrm{SNR}) = \frac{\mathrm{SNR}}{2\mathrm{SNR}_*^2 \left(1 + \frac{\mathrm{SNR}}{2\mathrm{SNR}_*}\right)^3} \tag{9.7}$$

where SNR_* is the SNR at which the distributions peak, empirically chosen to be $\mathrm{SNR}_* = 5$.

For chirplets, a natural prior range on the frequency evolution is when the tilt of the ellipse in time-frequency space reaches 45 degrees. From Eq. 9.4 we find this

occurs when $\beta = \pm\sqrt{1 - 1/\pi^2} \approx \pm 0.95$. We adopt a uniform prior on β in the range $\beta \in [-\sqrt{1 - 1/\pi^2}, \sqrt{1 - 1/\pi^2}]$.

The version of *BayesWave* used in the second advanced LIGO observation run uses a prior on the number of frame functions that was motivated by the distributions found in the first observation run. Here we adopt the simpler choice that the prior on the number of frame functions is uniform in $N_w \in [0, 20]$.

Simulated Data

To test the performance of the chirplet frame, we will look at how faithfully simulated GW signals can be reconstructed, and the number of frame functions used in the reconstruction. Our test data set consists of the of binary black hole merger signals, and unpolarized white noise bursts.

The binary black hole data set is a system of two $50M_\odot$ back holes with the waveform generated using the Effective One Body approximation [34] over a range of SNRs. We choose binary black holes as a test waveform in part because these are examples of waveforms we know have frequency evolution and thus are somewhere we believe a frame with frequency evolution could be beneficial. We also now know that GWs from black hole systems are detectable by LIGO, and we can likely expect more of these signals in the future. In addition to the standard BBH waveforms, we also tested BBH waveforms that have been time reversed, so that the frequency *decreases* over time. This set is used to demonstrate that the chirplet frame is good for general signals with time-frequency evolution, and is not specifically targeting BBH signals.

The second class of waveforms used are unpolarized white noise bursts (WNBs). These waveforms serve as a good test for the chirplet frame because they contain complicated frequency structure, which does not evolve smoothly like the BBH signals. These signals can sometimes present challenges for *BayesWave* because they

are unpolarized, whereas the current implementation of *BayesWave* assumes that the signals are elliptically polarized. We will see that this mis-modeling throws-off our estimates of the fidelity of the reconstruction as a function of signal-to-noise ratio.

In previous studies of unmodeled searches [16], sine-Gaussian waveforms (SGs) have also been used as test cases. However, we have already seen that though *BayesWave* can reconstruct SGs well, we are relatively insensitive to them in a search. This is a natural result of the fact that the signal-to-glitch Bayes factor scales with the number of wavelets used. For a sine-Gaussian signal we expect and indeed see that *BayesWave* typically uses only one wavelet to reconstruct SGs, so the Bayes factor scales only with SNR, making it more difficult to distinguish between signals and glitches. As wavelets are chirplets in the limit that $\dot{f}_0 \to 0$, we see the same behavior from chirplets and so do not consider SGs here.

Results

The two metrics we will look at are the number of frame functions used (N), and the match between waveforms, which as is defined as

$$M = \frac{(h|\bar{h})}{\sqrt{(h|h)(\bar{h}|\bar{h})}} \tag{9.8}$$

where $\bar{h}$ is the injected signal, and h is the recovered signal. The match ranges from -1 for a perfectly out of phase signal to 1 for a perfect match.

Dimensionality

The results for the average (mean) number of frame functions used for the BBH injection set and the WNB injection set are shown in Figure 9.3. The difference is more apparent for the BBH injections, but we see that in general fewer chirplets are

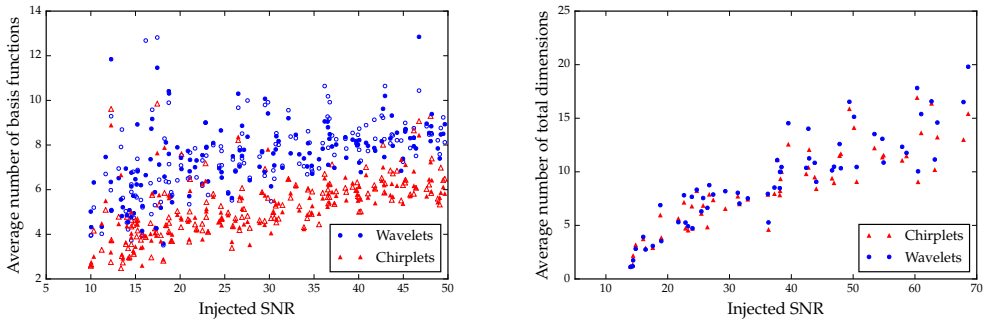

Figure 9.3: The injected SNR vs. the average number of frame functions (either chirplets or wavelets) used to reconstruct the signal for $50M_{\odot}$-$50M_{\odot}$ BBH injected into simulated noise. The filled in markers represent regular BBH injections, and hollow marker represent time-reversed BBH injections.

used than wavelets. This is as predicted– the extra parameter, $\dot{f}_0$ in the chirplet frame allows for fewer frame functions to be used in the reconstruction. This implies that the extra flexibility of chirplets may make them preferable for waveform reconstruction, particularly at low SNRs. A heuristic example of this can be seen in Fig. 9.4, where we see how the chirplets frequency evolution allows them to more closely follow the frequency evolution of the BBH signal.

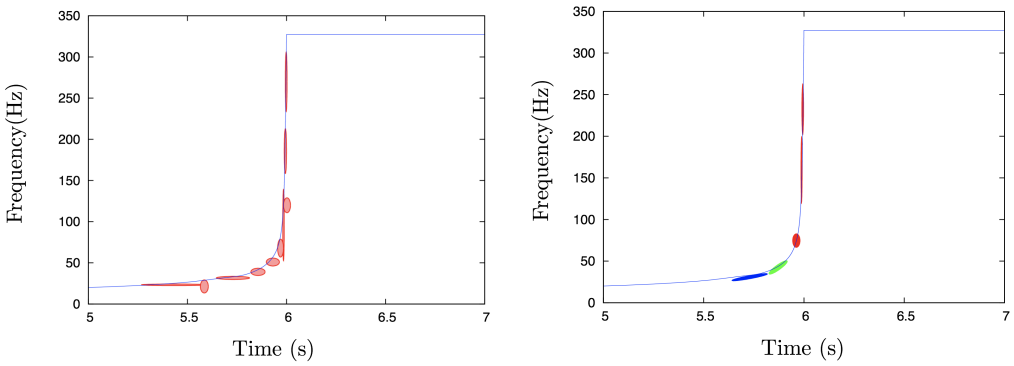

Figure 9.4: An example of the wavelet frame (left) and chirplet frame (right) in action. In this case we used a simulated BBH signal with component masses of $29M_{\odot}$ and $30M_{\odot}$ at an SNR of 35. The solid line is the predicted $f(t)$ track (as calculated in Chapter) and the colored ellipses are the wavelets or chirplets being used by *BayesWave* at one draw from the chain.

We also see that, as shown in Ref. [60,73], the number of wavelets used is roughly linearly dependent on the SNR of the injected signal. In Ref. [73] this dependence was written as $N \approx 1 + \gamma \mathrm{SNR}$, but here we generalize this expression to:

$$N \approx \alpha + \gamma \mathrm{SNR}, \tag{9.9}$$

with the constants α and γ being determined by the waveform morphology. In practice this expression is only valid for sufficient large SNRs, otherwise the number of frame functions used drops rapidly to one (the minimum allowed number of frame functions in *BayesWave*'s signal model).

Using the results from the BBH injections, we preform a simple linear fit to find α and γ for the wavelet and chirplet runs. In both cases the slopes are very similar: $\gamma_{chirp} = 0.065$, $\gamma_{wave} = 0.066$. The starting number of frame functions though varies significantly: $\alpha_{wave} = 5.6$, and $\alpha_{chirp} = 3.3$. So while the number of frame functions used increases at a similar rate for both chirplets and wavelets, chirplets use reliably fewer frame functions.

For the WNB injections, we see that while for higher SNR injections slightly more wavelets tend to be used than chirplets, the difference is not nearly as striking as for BBH injections. Again with a simple linear fit we see $\gamma_{chirp} = 0.21$, $\gamma_{wave} = 0.23$, $\alpha_{chirp} = 0.76$, and $\alpha_{wave} = 0.53$, giving very similar slopes and starting points for both frame functions.

Match

As predicted, the chirplet frame generally uses fewer frame functions. The real test though is the fidelity of the reconstruction. To test how well the injected signal is recovered, we look at the match. Fig. 9.5 shows the mean match between the injected and recovered waveforms for a set of two $50M_{\odot}$ BHs in simulated aLIGO noise (left)

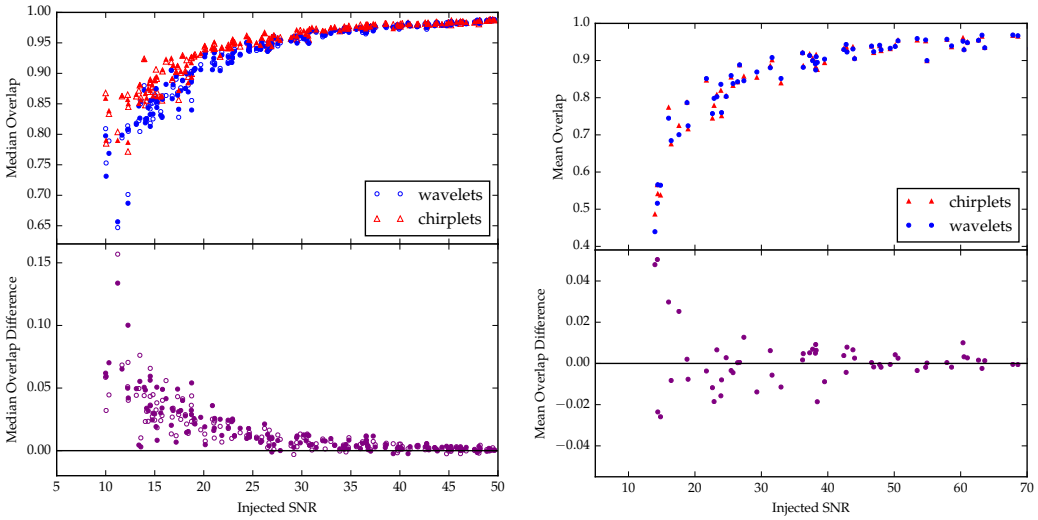

Figure 9.5: *Left*: The injected SNR vs. the network median match between the injected signal and reconstructed waveform for $50M_{\odot}$-$50M_{\odot}$ BBH injected into simulated noise. *Right*: The difference between the matches $\Delta M = M_{\rm chirplet} - M_{\rm wavelet}$. Again the filled in markers represent regular BBH injections, and hollow marker represent time-reversed BBH injections.

and a set of WNBs (right) for a range of SNRs using either chirplets or wavelets as the frame function.

In the BBH case, for SNRs above about 25, the matches of the two different methods are comparable. However at lower SNRs, we see that chirplets outperform wavelets, giving consistently higher matches. This is important because lower SNR events are a region of parameter space where we are more at risk of losing detection. We can also see in one particular example of a chirplet and wavelet reconstruction of a time-reversed BBH signal in Fig. 9.6. We can see here an example of how the chirplet frame manages to fit earlier and later parts of the signal.

For the WNB injections, we see that the two frame functions perform about equally as well. Previous injection studies with *BayesWave* have shown that WNBs can be difficult to to reconstruct. One reason is that the WNBs are unpolarized, while *BayesWave* assumes an elliptical polarization. WNBs also just tend to have

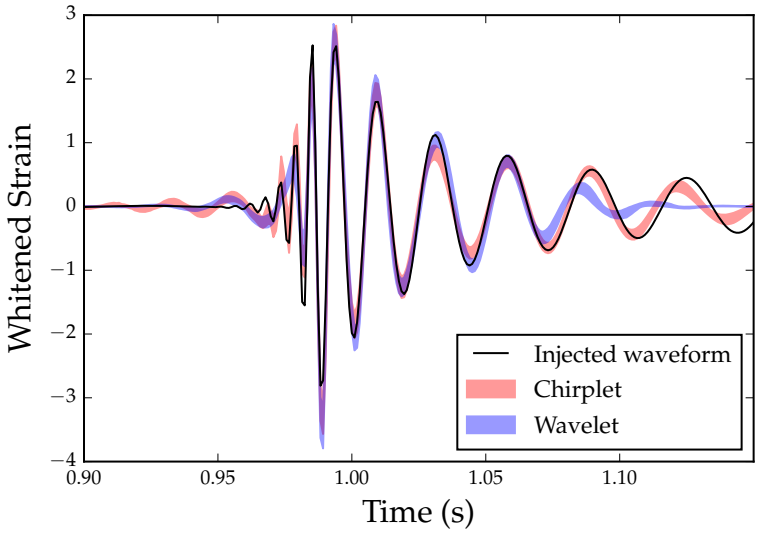

Figure 9.6: Example of waveform reconstruction for a time reversed 50-50$M_\odot$ BBH system. The red (blue) band shows the 50% credible interval of the reconstructed waveform using the chirplet (wavelet) frame. Both bases closely match the injected waveform (black) well in the higher power region, but chirplets are able to more accurately reconstruct the waveform later in the regions with less power. This event was injected with a network SNR of 10.25. The median network match for the chirplet frame is 0.91, and for the wavelet frame it is 0.87.

very complicated frequency evolution– an example of the frequency evolution of a WNB is shown in Figure 9.7. We can see that for WNBs the frequency as a function of time changes rapidly from increasing to decreasing and back again. Because the chirplets we use have only linearly increasing or decreasing frequency, the chirplet frame is still unable to recover all the fine details of signal. Thus we expect chirplets will provide the most benefit for signals with smooth time-frequency evolution.

We can also study what we theoretically would expect that matches to be for these injections. For the match given in Eq. 9.8, we assume that the injected waveform $\bar{h}$ is dependent on parameters $\bar{\lambda}^i$, and the recovered waveform has parameters λ^i. In the high SNR limit, the recovered and true injected parameters should be relatively consistent, or $\Delta\lambda^i = \bar{\lambda}^i - \lambda^i$ is small. Note that in this context the parameters are

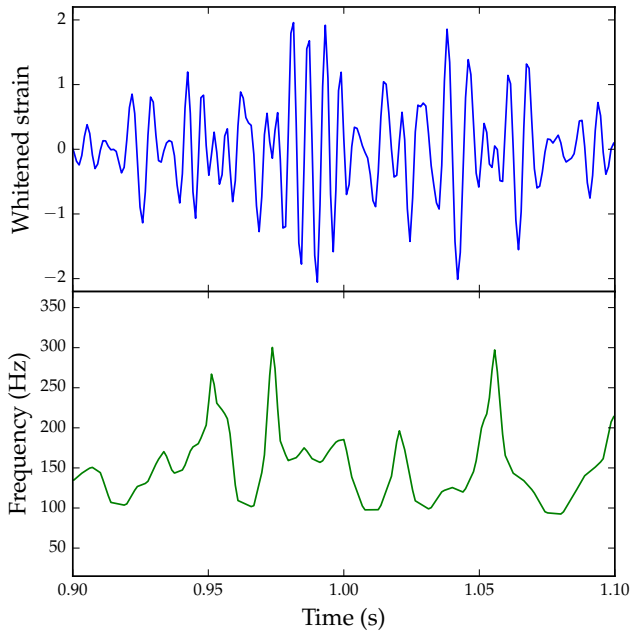

Figure 9.7: An example WNB waveform. The whitened strain (top) and frequency as a function of time (bottom) show that WNBs are complicated signals with no well defined frequency evolution.

those of the wavelet/chirplet representation, and not, for example, the masses and spins of of the black holes. We can approximate the recovered waveform as:

$$h = \bar{h} + h_{,i}\Delta\lambda^i \tag{9.10}$$

and $\Delta\lambda^i$ approximately follows the normal distribution:

$$p(\Delta\lambda^i) = \sqrt{\det(\Gamma/2\pi)}e^{-\Gamma_{ij}\Delta\lambda^i\Delta\lambda^j} \tag{9.11}$$

where $\Gamma_{ij} = (h_{,i}\,|h_{,j}\,)$ is the Fisher information matrix. We can expand our expression

for the match, Eq. 9.8 then to be [86]

$$M = 1 - \frac{1}{2}\Delta\lambda^i \Delta\lambda^j \left(\frac{(h_{,i}\,|h_{,j}\,)}{(h|h)} - \frac{(h|h_{,i}\,)(h|h_{,j}\,)}{(h|h)^2} \right). \tag{9.12}$$

Recognizing that the expected value of $\Delta\lambda^i\Delta\lambda^j$ is $E[\Delta\lambda^i\Delta\lambda^j] \approx \Gamma^{-1}_{ij}$ [103], we find the expected match is:

$$E[M] \approx 1 - \frac{D-1}{2\mathrm{SNR}^2}. \tag{9.13}$$

where D is the dimension of the model. Note that this derivation assumes a templated search, and so should be thought of as more of a "rule of thumb" for this analysis. Using the scaling for the number of wavelets in Eq. 9.9, we have $D = N_p(\alpha+\gamma\mathrm{SNR})+4$, where N_p is 5 for wavelets, and 6 for chirplets, and an additional 4 common extrinsic parameters (sky location, ellipticity and polarization angle). The full expression for the predicted match is then:

$$E[M] \approx 1 - \frac{N_p(\alpha+\gamma\mathrm{SNR})+3}{2\mathrm{SNR}^2}. \tag{9.14}$$

Figure 9.8 shows again the average match for the injected binary black hole signals, with the match predicted by Eq. 9.14. The recovered matches for the BBH injections follow the predicted match relatively well, however the recovered matches for the WNB injections are lower than the analytical prediction due to the signal model (polarized) not matching the simulated signals (un-polarized).

Bayes Factors

The *BayesWave* algorithm currently considers three distinct models: ($\mathcal{S}$) GW Signals + Gaussian noise, ($\mathcal{G}$) Noise transients (Glitches) + Gaussian noise, ($\mathcal{N}$) Gaussian noise and computes evidence ratios, or Bayes factors, between the models.

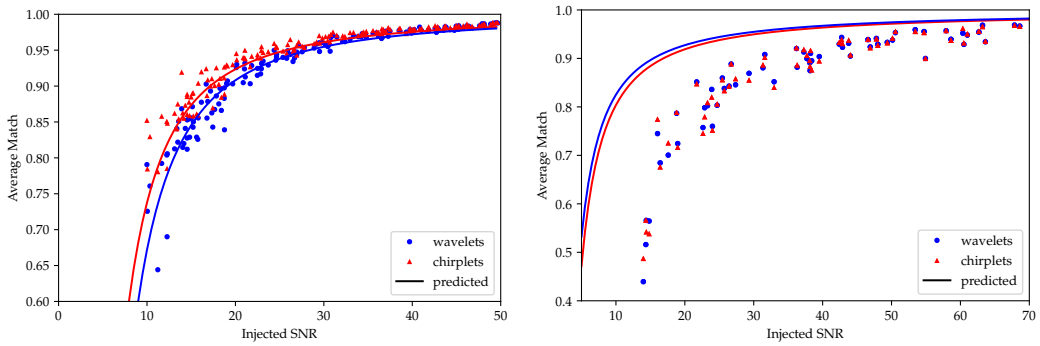

Figure 9.8: The expected match as predicted by Eq. 9.14 plotted with the actual match for the set of BBH injections (left) and WNB injections (right). The BBH injections generally follow the predicted match vs. SNR scaling, but the prediction tends to overestimate the match for WNB injections.

Here we investigate how the choice of frame impacts the Bayes factors between the models.

Figure 9.9 shows $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$ (left) and $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ (right) recovered using chirplets and wavelets for simulated binary black hole signals. We see that both bases return very similar $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$, with the chirplet frame giving just slightly higher Bayes factors. This is unsurprising since $\ln \mathcal{B}_{\mathcal{S},\mathcal{N}}$ scales with the recovered SNR, and chirplets are able to recover more SNR due to their ability to recover signals with higher fidelity.

The signal-to-glitch Bayes factors $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ show the opposite behavior, with the wavelet frame providing better separation between signals and glitches than the chirplet frame. This seemingly paradoxical result is due to the chirplet frame providing higher fidelity reconstructions using less parameters for *both* signals *and* glitches. Moreover, since the glitch model sees the signal in the individual detectors, which has lower signal-to-noise than the network response seen by the signal model, and since the chirplets outperform wavelets mostly at low SNR, the chirplet model boosts the evidence for the glitch model more than it boosts the evidence for the signal model, resulting in lower $\ln \mathcal{B}_{\mathcal{S},\mathcal{G}}$ than for the wavelet model. From the perspective of a

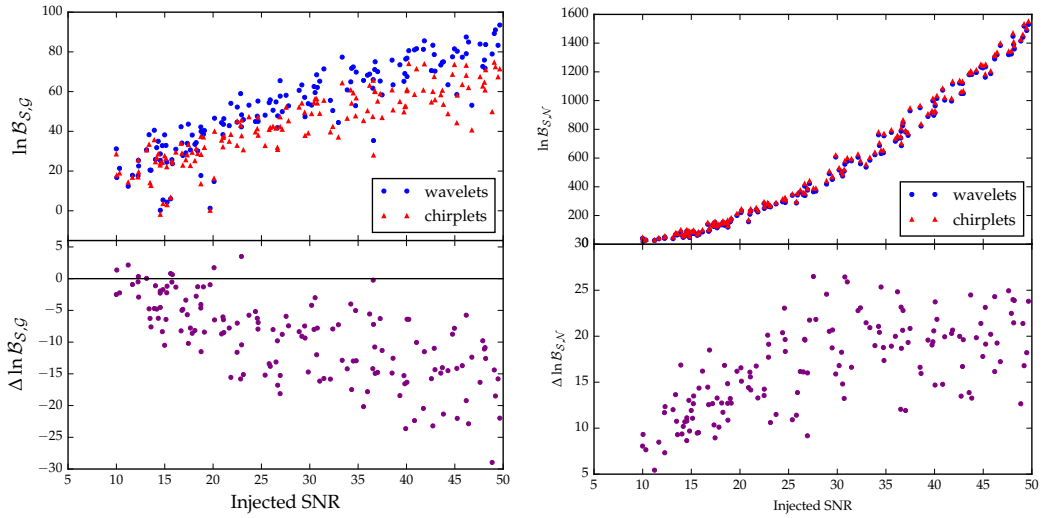

Figure 9.9: The log signal-to-glitch (left) and signal-to-noise (right) Bayes factors for a set of $50-50M_{\odot}$ binary black holes using the chirplet and wavelet bases. The lower panel shows the difference in the log Bayes factors between the chirplet and wavelet frames.

search, where the goal is to separate signals from instrument noise, the wavelet frame outperforms the chirplet frame despite not doing as well at reconstructing signals. The same behavior was also seen when using the "clustering prior" [44], which leads to higher matches, especially at low SNR, but worse separation between signals and glitches.

The reduction in the signal-to-glitch Bayes factor has prevented the clustering prior and the chirplet frame from being used in the current LIGO/Virgo analyses, despite the improvements they offer for signal and glitch reconstruction. We were forced to make this unsatisfactory choice because of the limitations in the specification of models. Going forward we plan to implement new models that do a much better job of separating signals and glitches, and that will not penalize models that do a better job of fitting low signal-to-noise features. One option is to modify the glitch model to be anti-coincident between detectors. This can be done by introducing a

prior that disfavors placing wavelets at frequencies and times that are occupied by wavelets in the glitch models for the other detectors.

Discussion

We have found that added flexibility offered by chirplet frame functions can reduce the overall model dimension, despite adding an additional parameter to each frame function, and improve waveform reconstruction, particularly at low SNRs. Limitations in the model selection approach that is currently used by *BayesWave* to that is distinguish between signals and glitches has so-far prevented the adoption of chirplets, but these limitations will soon be resolved. Ideally *BayesWave* should utilize a wide range of frame elements, including different types of wavelets and chirplets, and perhaps reduced-order-quadrature black holem basis elements. The optimal mix could then be dynamically selected via the trans-dimensional MCMC algorithm, hueing closer to our mantra "model everything and let the data sort it out".

CONCLUSIONS

Summary

The main aim of this Thesis was to maximize our abilities to confidently detect and characterize short-duration GW bursts from any possible source in LIGO data.

We began by introducing the Bayesian inference framework used in GW data analysis, and in particular the *BayesWave* algorithm which can confidently distinguish between GWs and noise, as well as provide accurate reconstructions of GW events. We then examined the evidence calculations in *BayesWave* to gain a deeper understanding of what the Bayes factors really tell us about the data we are analyzing. We found that the the log signal-to-glitch Bayes factor scales with the dimensionality of the model employed by the RJMCMC. This is a novel feature that sets *BayesWave* apart from other unmodeled searches in that our detection statistic scales with the *complexity* of the signal, rather than purely with SNR. This means our search is less affected by loud glitches.

We then showed how actually implementing *BayesWave* as a followup to triggers from the cWB pipeline could help increase our confidence in GW signals. This pipeline was implemented into LIGO in the first two observing runs, and we described in detail how this pipeline was set up and operated in O1 and O2. Of course, the first observing run was marked by the momentous discovery of GW150914, the first ever direct detection of a GW. The *BayesWave* search found this event with high confidence, complementing the targeted searches for compact binaries. We reported on the *BayesWave* results for this event, as well as the results for the rest of the O1 search.

In addition to the significance of GW150914, *BayesWave* looked at the full waveform reconstruction of this event and various other metrics of characterizing

the signal. The waveform reconstruction showed remarkable agreement with the template based reconstruction, further boosting our confidence in this detection. We also looked at how we could perform some nominal parameter estimation with the unmodeled reconstructions which can serve as a model to how we would approach such a problem if we did not yet have full waveform templates.

Finally we looked at developments to the *BayesWave* algorithm to better reconstruct GW signals, as well as to target certain sources without having to rely on a full template. First we implemented a prior that uses the predicted frequency evolution of a binary system in the wavelet placement. This served to increase the matches between the injected and recovered waveforms, and important improvement as we want to capture as many details as possible when reconstructing the waveform. This prior also has the added advantage of being a parameterized prior, and so we can use the distributions of the hyperparameters to perform parameter estimation without templates. We then looked at the benefits of changing *BayesWave*'s basis function from sine-Gaussian wavelets to wavelets with a frequency evolution. Though this adds another parameter to the reconstruction basis, it has an increased ability to change shape to fit the data. We saw that this again increased the matches between the injected and recovered waveforms, while simultaneously decreasing the number of basis functions used. However the new basis function did not improve Bayes factors, so this improvement is mainly for waveform reconstruction rather that signal detection.

Future work

Most of this Thesis focused on the signal model reconstructions by *BayesWave*. However, *BayesWave* also produces good reconstructions in the glitch model. This can serve as a powerful tool to help further understand spurious noise sources that pollute LIGO's searches. Recent work used *BayesWave* to reconstruct a large glitch

in one detector that coincided with GW170817, and subtract it from subsequent analyses [21]. Future work aims to make *BayesWave* glitch reconstruction a regular feature. We also hope to use *BayesWave*'s glitch reconstructions in concert with currently machine learning efforts to classify glitches [105].

Further, most of the developments implemented in this Thesis focused on GW signals from compact binaries. While we have detected a number of these types of signals and will detect even more in the future, as LIGO's sensitivity further improves we can expect GWs from other sources. One possible extension of this work is to implement a similar parameterized prior but for sources such as supernovae [53], or post merger signals from binary neutron stars [40]. This we we are ready for a wide variety of signals.

Outlook

We are now truly in the age of gravitational-wave astronomy, and as such must be prepared to find any type of new and exciting GW signal. In this Thesis we showed how well our unmodeled search algorithm performed during the first two observing runs of Advanced LIGO. This gives us confidence that in the future we will be ready for any surprise the universe gives us.

REFERENCES CITED

[1] Gracedb. https://gw-astronomy.org/wiki/LV_EM/TiGraceDB. Accessed: 2018-02-26.

[2] Indian initiative in gravitational-wave observations. http://www.gw-indigo.org/. Accessed: 2018-03-13.

[3] Laser interferometer space antenna. https://lisa.nasa.gov/. Accessed: 2018-03-13.

[4] Lsc algorithm library software packates. http://www.lsc-group.phys.uwm.edu/lal.

[5] Spinning test-particles in general relativity. i. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 209(1097):248–258, 1951.

[6] Dynamics of extended bodies in general relativity. i. momentum and angular momentum. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 314(1519):499–527, 1970.

[7] J. Aasi et al. Advanced ligo. *Classical and Quantum Gravity*, 32(7):074001, 2015.

[8] J. Abadie et al. All-sky search for gravitational-wave bursts in the second joint ligo-virgo run. *Phys. Rev. D*, 85:122007, Jun 2012.

[9] B. P. Abbott and all. Gw150914: The advanced ligo detectors in the era of first discoveries. *Phys. Rev. Lett.*, 116:131103, Mar 2016.

[10] B. P. Abbott et al. Characterization of transient noise in advanced ligo relevant to gravitational wave signal gw150914. *Classical and Quantum Gravity*, 33(13):134001, 2016.

[11] B. P. Abbott et al. Gw150914: First results from the search for binary black hole coalescence with advanced ligo. *Phys. Rev. D*, 93:122003, Jun 2016.

[12] B. P. Abbott et al. Gw151226: Observation of gravitational waves from a 22-solar-mass binary black hole coalescence. *Phys. Rev. Lett.*, 116:241103, Jun 2016.

[13] B. P. Abbott et al. Observation of gravitational waves from a binary black hole merger. *Phys. Rev. Lett.*, 116:061102, Feb 2016.

[14] B. P. Abbott et al. Observing gravitational-wave transient gw150914 with minimal assumptions. *Phys. Rev. D*, 93:122004, Jun 2016.

[15] B. P. Abbott et al. Properties of the binary black hole merger gw150914. *Phys. Rev. Lett.*, 116:241102, Jun 2016.

[16] B. P. Abbott et al. All-sky search for short gravitational-wave bursts in the first advanced ligo run. *Phys. Rev. D*, 95:042003, Feb 2017.

[17] B. P. Abbott et al. Calibration of the advanced ligo detectors for the discovery of the binary black-hole merger gw150914. *Phys. Rev. D*, 95:062003, Mar 2017.

[18] B. P. Abbott et al. Gw170104: Observation of a 50-solar-mass binary black hole coalescence at redshift 0.2. *Phys. Rev. Lett.*, 118:221101, Jun 2017.

[19] B. P. Abbott et al. Gw170608: Observation of a 19 solar-mass binary black hole coalescence. *The Astrophysical Journal Letters*, 851(2):L35, 2017.

[20] B. P. Abbott et al. Gw170814: A three-detector observation of gravitational waves from a binary black hole coalescence. *Phys. Rev. Lett.*, 119:141101, Oct 2017.

[21] B. P. Abbott et al. Gw170817: Observation of gravitational waves from a binary neutron star inspiral. *Phys. Rev. Lett.*, 119:161101, Oct 2017.

[22] T. Accadia et al. Virgo: a laser interferometer to detect gravitational waves. *Journal of Instrumentation*, 7(03):P03012, 2012.

[23] P. Ajith, M. Hannam, S. Husa, Y. Chen, B. Brügmann, N. Dorband, D. Müller, F. Ohme, D. Pollney, C. Reisswig, L. Santamaría, and J. Seiler. Inspiral-merger-ringdown waveforms for black-hole binaries with nonprecessing spins. *Phys. Rev. Lett.*, 106:241101, Jun 2011.

[24] T. Akutsu et al. The status of KAGRA underground cryogenic gravitational wave telescope. In *15th International Conference on Topics in Astroparticle and Underground Physics (TAUP 2017) Sudbury, Ontario, Canada, July 24-28, 2017*, 2017.

[25] A. Ali, D. S. Ahmad, M. Nawaz, S. Ullah, and M. Aqeel. Bayesian inference on gravitational waves. 11:645, 12 2015.

[26] W. Anderson et al. Beam pattern response functions and times of arrival for earthbound interferometer. 2001.

[27] M. Armano et al. Sub-Femto- g Free Fall for Space-Based Gravitational Wave Observatories: LISA Pathfinder Results. *Phys. Rev. Lett.*, 116(23):231101, 2016.

[28] J. G. Baker, J. Centrella, D.-I. Choi, M. Koppitz, and J. van Meter. Gravitational-wave extraction from an inspiraling configuration of merging black holes. *Phys. Rev. Lett.*, 96:111102, Mar 2006.

[29] P. T. Baker. *Distinguishing Signal from Noise: new techniques for gravitational wave data analysis*. PhD thesis, Montana State University, 2013.

[30] B. Becsy, P. Raffai, N. J. Cornish, R. Essick, J. Kanner, E. Katsavounidis, T. B. Littenberg, M. Millhouse, and S. Vitale. Parameter estimation for gravitational-wave bursts with the bayeswave pipeline. *The Astrophysical Journal*, 839(1):15, 2017.

[31] B. Becsy, P. Raffai, T. B. Littenberg, M. Millhouse, K. Gill, and M. Szczepanczyk. Interpreting a gravitational-wave burst: Constraints on the source without astrophysical models. *In prep*, 2018.

[32] C. Biwer, D. Barker, J. C. Batch, J. Betzwieser, R. P. Fisher, E. Goetz, S. Kandhasamy, S. Karki, J. S. Kissel, A. P. Lundgren, D. M. Macleod, A. Mullavey, K. Riles, J. G. Rollins, K. A. Thorne, E. Thrane, T. D. Abbott, B. Allen, D. A. Brown, P. Charlton, S. G. Crowder, P. Fritschel, J. B. Kanner, M. Landry, C. Lazzaro, M. Millhouse, M. Pitkin, R. L. Savage, P. Shawhan, D. H. Shoemaker, J. R. Smith, L. Sun, J. Veitch, S. Vitale, A. J. Weinstein, N. Cornish, R. C. Essick, M. Fays, E. Katsavounidis, J. Lange, T. B. Littenberg, R. Lynch, P. M. Meyers, F. Pannarale, R. Prix, R. O'Shaughnessy, and D. Sigg. Validating gravitational-wave detections: The advanced ligo hardware injection system. *Phys. Rev. D*, 95:062002, Mar 2017.

[33] L. Blanchet. Gravitational radiation from post-newtonian sources and inspiralling compact binaries. *Living Reviews in Relativity*, 17(1):2, Feb 2014.

[34] A. Buonanno, Y. Pan, J. G. Baker, J. Centrella, B. J. Kelly, S. T. McWilliams, and J. R. van Meter. Approaching faithful templates for nonspinning binary black holes using the effective-one-body approach. *Phys. Rev. D*, 76:104049, Nov 2007.

[35] M. Campanelli, C. O. Lousto, P. Marronetti, and Y. Zlochower. Accurate evolutions of orbiting black-hole binaries without excision. *Phys. Rev. Lett.*, 96:111101, Mar 2006.

[36] K. Cannon, R. Cariou, A. Chapman, M. Crispin-Ortuzar, N. Fotopoulos, M. Frei, C. Hanna, E. Kara, D. Keppel, L. Liao, S. Privitera, A. Searle, L. Singer, and A. Weinstein. Toward early-warning detection of gravitational waves from compact binary coalescence. *The Astrophysical Journal*, 748(2):136, 2012.

[37] S. M. Carroll. *Spacetime and geometry: An introduction to general relativity*. 2004.

[38] S. Chandrasekhar and F. P. Esposito. The 2.5-POST-NEWTONIAN Equations of Hydrodynamics and Radiation Reaction in General Relativity. *"Astrophys. J."*, 160:153, Apr. 1970.

[39] S. K. Chatterji. *The search for gravitational wave bursts in data from the second LIGO science run*. PhD thesis, Massachusetts Institute of Technology, 2005.

[40] K. Chatziioannou, J. A. Clark, A. Bauswein, M. Millhouse, T. B. Littenberg, and N. Cornish. Inferring the post-merger gravitational wave emission from binary neutron star coalescences. *Phys. Rev. D*, 96:124035, Dec 2017.

[41] N. Cornish.

[42] N. Cornish. Fast fisher matrices and lazy likelihoods. 07 2010.

[43] N. Cornish, P. Baker, and T. Littenberg. The BayesWave Algorithm for Detecting and Characterizing Gravitational Wave Burst Signals. In *APS Meeting Abstracts*, page X12.009, Apr. 2011.

[44] N. J. Cornish and T. B. Littenberg. Bayeswave: Bayesian inference for gravitational wave bursts and instrument glitches. *Classical and Quantum Gravity*, 32(13):135012, 2015.

[45] N. J. Cornish and J. Romano. A unified approach to gravitational wave data analysis using bayesian hierarchical modeling. *In prep*, 2018.

[46] T. Dal Canton, A. H. Nitz, A. P. Lundgren, A. B. Nielsen, D. A. Brown, T. Dent, I. W. Harry, B. Krishnan, A. J. Miller, K. Wette, K. Wiesner, and J. L. Willis. Implementing a search for aligned-spin neutron star-black hole systems with advanced ground based gravitational wave detectors. *Phys. Rev. D*, 90:082004, Oct 2014.

[47] M. Drago. *SEARCH FOR TRANSIENT GRAVITATIONAL WAVE SIGNALS WITH UNKNOWN WAVEFORM IN THE LIGO VIRGO NETWORK OF INTEFEROMETRIC DETECTORS USING A FULLY COHERENT ALGORITHM*. PhD thesis, Università degli Studi di Padova, 2010.

[48] F. W. Dyson, A. S. Eddington, and C. Davidson. A Determination of the Deflection of Light by the Sun's Gravitational Field, from Observations Made at the Total Eclipse of May 29, 1919. *Philosophical Transactions of the Royal Society of London Series A*, 220:291–333, 1920.

[49] A. Einstein. Die feldgleichungen der gravitation. *Sitzungsberichte der Königlich Preußischen Akademie der Wissenschaften (Berlin), Seite 844-847.*, 1915.

[50] A. Einstein, L. Infeld, and B. Hoffmann. The gravitational equations and the problem of motion. *Annals of Mathematics*, 39(1):65–100, 1938.

[51] W. Farr. Sky area. `https://github.com/farr/skyarea`.

[52] E. E. Flanagan and S. A. Hughes. The Basics of gravitational wave theory. *New J. Phys.*, 7:204, 2005.

[53] K. Gill et al. Enhancing the sensitivity of searches for gravitational waves from core-collapse supernovae with a bayesian classification of candidate events. *In prep*, 2018.

[54] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759, 2005.

[55] S. E. Gralla and R. M. Wald. Derivation of gravitational self-force.

[56] P. J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[57] W. K. Hastings. Monte carlo sampling methods using markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[58] G. Hobbs, A. Archibald, Z. Arzoumanian, D. Backer, M. Bailes, N. D. R. Bhat, M. Burgay, S. Burke-Spolaor, D. Champion, I. Cognard, W. Coles, J. Cordes, P. Demorest, G. Desvignes, R. D. Ferdman, L. Finn, P. Freire, M. Gonzalez, J. Hessels, A. Hotan, G. Janssen, F. Jenet, A. Jessner, C. Jordan, V. Kaspi, M. Kramer, V. Kondratiev, J. Lazio, K. Lazaridis, K. J. Lee, Y. Levin, A. Lommen, D. Lorimer, R. Lynch, A. Lyne, R. Manchester, M. McLaughlin, D. Nice, S. Oslowski, M. Pilia, A. Possenti, M. Purver, S. Ransom, J. Reynolds, S. Sanidas, J. Sarkissian, A. Sesana, R. Shannon, X. Siemens, I. Stairs, B. Stappers, D. Stinebring, G. Theureau, R. van Haasteren, W. van Straten, J. P. W. Verbiest, D. R. B. Yardley, and X. P. You. The International Pulsar Timing Array project: using pulsars as a gravitational wave detector. *Classical and Quantum Gravity*, 27(8):084013, Apr. 2010.

[59] R. A. Hulse and J. H. Taylor. Discovery of a pulsar in a binary system. *Astrophys. J.*, 195:L51–L53, 1975.

[60] J. B. Kanner, T. B. Littenberg, N. Cornish, M. Millhouse, E. Xhakaj, F. Salemi, M. Drago, G. Vedovato, and S. Klimenko. Leveraging waveform complexity for confident detection of gravitational waves. *Phys. Rev. D*, 93:022002, Jan 2016.

[61] R. P. Kerr. Gravitational field of a spinning mass as an example of algebraically special metrics. *Phys. Rev. Lett.*, 11:237–238, Sep 1963.

[62] J. Kissel. *CALIBRATING AND IMPROVING THE SENSITIVITY OF THE LIGO DETECTORS*. PhD thesis, Louisiana State University, 2010.

[63] S. Klimenko, G. Vedovato, M. Drago, F. Salemi, V. Tiwari, G. A. Prodi, C. Lazzaro, K. Ackley, S. Tiwari, C. F. Da Silva, and G. Mitselmakher. Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors. *Phys. Rev. D*, 93:042004, Feb 2016.

[64] S. Klimenko, I. Yakushin, A. Mercer, and G. Mitselmakher. A coherent method for detection of gravitational wave bursts. *Classical and Quantum Gravity*, 25(11):114029, 2008.

[65] J. Kovacevic and A. Chebira. An introduction to frames. *Foundations and Trends in Signal Processing*, 2:1–94, 2008.

[66] I. Kowalska-Leszczynska, M.-A. Bizouard, T. Bulik, N. Christensen, M. Coughlin, M. Gokowski, J. Kubisz, A. Kulak, J. Mlynarczyk, F. Robinet, and M. Rohde. Globally coherent short duration magnetic field transients and their effect on ground based gravitational-wave detectors. *Classical and Quantum Gravity*, 34(7):074002, 2017.

[67] L. Laboratory. Ligo's interferometer. `https://www.ligo.caltech.edu/page/ligos-ifo`. Accessed: 2018-04-10.

[68] N. Lartillot and H. Philippe. Computing bayes factors using thermodynamic integration. *Systematic Biology*, 55(2):195–207, 2006.

[69] D. V. Lindley and A. F. M. Smith. Bayes estimates for the linear model. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(1):1–41, 1972.

[70] T. Littenberg. *A COMPREHENSIVE BAYESIAN APPROACH TO GRAVITATIONAL WAVE ASTRONOMY*. PhD thesis, Montana State University, 2009.

[71] T. B. Littenberg and N. J. Cornish. Separating gravitational wave signals from instrument artifacts. *Phys. Rev. D*, 82:103007, Nov 2010.

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

[73] T. B. Littenberg, J. B. Kanner, N. J. Cornish, and M. Millhouse. Enabling high confidence detections of gravitational-wave bursts. *Phys. Rev. D*, 94:044050, Aug 2016.

[74] H. A. Lorentz and J. Droste. *Collected Papers: Volume V*, chapter The Motion of a System of Bodies under the Influence of their Mutual Attraction, According to Einstein's Theory, pages 330–355. Springer Netherlands, Dordrecht, 1937.

[75] N. Loutrel and N. Yunes. Eccentric gravitational wave bursts in the post-newtonian formalism. *Classical and Quantum Gravity*, 34(13):135011, 2017.

[76] R. Lynch, S. Vitale, R. Essick, E. Katsavounidis, and F. Robinet. Information-theoretic approach to the gravitational-wave burst detection problem. *Phys. Rev. D*, 95:104046, May 2017.

[77] D. J. C. MacKay. Comparison of approximate methods for handling hyperparameters. *Neural Computation*, 11(5):1035–1068, 1999.

[78] M. Maggiore. *Gravitational Waves. Vol. 1: Theory and Experiments*. Oxford Master Series in Physics. Oxford University Press, 2007.

[79] S. Mann and S. Haykin. The chirplet transform: A generalization of Gabor's logon transform. *Vision Interface '91*, pages 205–212, June 3-7 1991. ISSN 0843-803X.

[80] D. V. Martynov, E. D. Hall, B. P. Abbott, R. Abbott, T. D. Abbott, C. Adams, R. X. Adhikari, R. A. Anderson, S. B. Anderson, K. Arai, M. A. Arain, S. M. Aston, L. Austin, S. W. Ballmer, M. Barbet, D. Barker, B. Barr, L. Barsotti, J. Bartlett, M. A. Barton, I. Bartos, J. C. Batch, A. S. Bell, I. Belopolski, J. Bergman, J. Betzwieser, G. Billingsley, J. Birch, S. Biscans, C. Biwer, E. Black, C. D. Blair, C. Bogan, R. Bork, D. O. Bridges, A. F. Brooks, C. Celerier, G. Ciani, F. Clara, D. Cook, S. T. Countryman, M. J. Cowart, D. C. Coyne, A. Cumming, L. Cunningham, M. Damjanic, R. Dannenberg, K. Danzmann, C. F. D. S. Costa, E. J. Daw, D. DeBra, R. T. DeRosa, R. DeSalvo, K. L. Dooley, S. Doravari, J. C. Driggers, S. E. Dwyer, A. Effler, T. Etzel, M. Evans, T. M. Evans, M. Factourovich, H. Fair, D. Feldbaum, R. P. Fisher, S. Foley, M. Frede, P. Fritschel, V. V. Frolov, P. Fulda, M. Fyffe, V. Galdi, J. A. Giaime, K. D. Giardina, J. R. Gleason, R. Goetz, S. Gras, C. Gray, R. J. S. Greenhalgh, H. Grote, C. J. Guido, K. E. Gushwa, E. K. Gustafson, R. Gustafson, G. Hammond, J. Hanks, J. Hanson, T. Hardwick, G. M. Harry, J. Heefner, M. C. Heintze, A. W. Heptonstall, D. Hoak, J. Hough, A. Ivanov, K. Izumi, M. Jacobson, E. James, R. Jones, S. Kandhasamy, S. Karki, M. Kasprzack, S. Kaufer, K. Kawabe, W. Kells, N. Kijbunchoo, E. J. King, P. J. King, D. L. Kinzel, J. S. Kissel, K. Kokeyama, W. Z. Korth, G. Kuehn, P. Kwee, M. Landry, B. Lantz, A. Le Roux, B. M. Levine, J. B. Lewis, V. Lhuillier, N. A. Lockerbie, M. Lormand, M. J. Lubinski, A. P. Lundgren, T. MacDonald, M. MacInnis, D. M. Macleod, M. Mageswaran, K. Mailand, S. Márka, Z. Márka, A. S. Markosyan, E. Maros, I. W. Martin, R. M. Martin, J. N. Marx, K. Mason, T. J. Massinger, F. Matichard, N. Mavalvala, R. McCarthy, D. E. McClelland, S. McCormick, G. McIntyre, J. McIver, E. L. Merilh, M. S. Meyer, P. M. Meyers, J. Miller, R. Mittleman, G. Moreno, C. L. Mueller, G. Mueller, A. Mullavey, J. Munch, L. K. Nuttall, J. Oberling, J. O'Dell, P. Oppermann, R. J. Oram, B. O'Reilly, C. Osthelder, D. J. Ottaway, H. Overmier, J. R. Palamos, H. R. Paris, W. Parker, Z. Patrick, A. Pele, S. Penn, M. Phelps, M. Pickenpack, V. Pierro, I. Pinto, J. Poeld,

M. Principe, L. Prokhorov, O. Puncken, V. Quetschke, E. A. Quintero, F. J. Raab, H. Radkins, P. Raffai, C. R. Ramet, C. M. Reed, S. Reid, D. H. Reitze, N. A. Robertson, J. G. Rollins, V. J. Roma, J. H. Romie, S. Rowan, K. Ryan, T. Sadecki, E. J. Sanchez, V. Sandberg, V. Sannibale, R. L. Savage, R. M. S. Schofield, B. Schultz, P. Schwinberg, D. Sellers, A. Sevigny, D. A. Shaddock, Z. Shao, B. Shapiro, P. Shawhan, D. H. Shoemaker, D. Sigg, B. J. J. Slagmolen, J. R. Smith, M. R. Smith, N. D. Smith-Lefebvre, B. Sorazu, A. Staley, A. J. Stein, A. Stochino, K. A. Strain, R. Taylor, M. Thomas, P. Thomas, K. A. Thorne, E. Thrane, C. I. Torrie, G. Traylor, G. Vajente, G. Valdes, A. A. van Veggel, M. Vargas, A. Vecchio, P. J. Veitch, K. Venkateswara, T. Vo, C. Vorvick, S. J. Waldman, M. Walker, R. L. Ward, J. Warner, B. Weaver, R. Weiss, T. Welborn, P. Weßels, C. Wilkinson, P. A. Willems, L. Williams, B. Willke, L. Winkelmann, C. C. Wipf, J. Worden, G. Wu, H. Yamamoto, C. C. Yancey, H. Yu, L. Zhang, M. E. Zucker, and J. Zweizig. Sensitivity of the advanced ligo detectors at the beginning of gravitational wave astronomy. *Phys. Rev. D*, 93:112004, Jun 2016.

[81] C. Messick, K. Blackburn, P. Brady, P. Brockill, K. Cannon, R. Cariou, S. Caudill, S. J. Chamberlin, J. D. E. Creighton, R. Everett, C. Hanna, D. Keppel, R. N. Lang, T. G. F. Li, D. Meacher, A. Nielsen, C. Pankow, S. Privitera, H. Qi, S. Sachdev, L. Sadeghian, L. Singer, E. G. Thomas, L. Wade, M. Wade, A. Weinstein, and K. Wiesner. Analysis framework for the prompt discovery of compact binary mergers in gravitational-wave data. *Phys. Rev. D*, 95:042001, Feb 2017.

[82] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[83] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[84] A. A. Michelson and E. W. Morley. On the relative motion of the Earth and the luminiferous ether. *American Journal of Science*, 34:333–345, Nov. 1887.

[85] S. Mohapatra, L. Cadonati, S. Caudill, J. Clark, C. Hanna, S. Klimenko, C. Pankow, R. Vaulin, G. Vedovato, and S. Vitale. Sensitivity comparison of searches for binary black hole coalescences with ground-based gravitational-wave detectors. *Phys. Rev. D*, 90:022001, Jul 2014.

[86] B. J. Owen and B. S. Sathyaprakash. Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement. *Phys. Rev. D*, 60:022002, Jun 1999.

[87] Y. Pan, A. Buonanno, M. Boyle, L. T. Buchman, L. E. Kidder, H. P. Pfeiffer, and M. A. Scheel. Inspiral-merger-ringdown multipolar waveforms of nonspinning black-hole binaries using the effective-one-body formalism. *Phys. Rev. D*, 84:124052, Dec 2011.

[88] C. Pankow. *SEARCH FOR GRAVITATIONAL WAVES FROM INTERMEDIATE MASS BLACK HOLE BINARIES*. PhD thesis, UNIVERSITY OF FLORIDA, 2011.

[89] F. Pretorius. Evolution of binary black-hole spacetimes. *Phys. Rev. Lett.*, 95:121101, Sep 2005.

[90] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[91] J. D. Romano and N. J. Cornish. Detection methods for stochastic gravitational-wave backgrounds: a unified treatment. *Living Reviews in Relativity*, 20(1):2, Apr 2017.

[92] C. Röver. *Bayesian inference on astrophysical binary inspirals based on gravitational-wave measurements*. PhD thesis, The University of Auckland, 2007.

[93] C. Rover, R. Meyer, and N. Christensen. Bayesian inference on compact binary inspiral gravitational radiation signals in interferometric data. *Classical and Quantum Gravity*, 23(15):4895, 2006.

[94] L. Sampson. *TESTING GENERAL RELATIVITY WITH GRAVITATIONAL WAVES*. PhD thesis, Montana State University, 2014.

[95] L. Santamaría, F. Ohme, P. Ajith, B. Brügmann, N. Dorband, M. Hannam, S. Husa, P. Mösta, D. Pollney, C. Reisswig, E. L. Robinson, J. Seiler, and B. Krishnan. Matching post-newtonian and numerical relativity waveforms: Systematic errors and a new phenomenological model for nonprecessing black hole binaries. *Phys. Rev. D*, 82:064016, Sep 2010.

[96] K. Schwarzschild. On the gravitational field of a mass point according to Einstein's theory. *Sitzungsber. Preuss. Akad. Wiss. Berlin (Math. Phys.)*, 1916:189–196, 1916.

[97] X. Siemens, B. Allen, J. Creighton, M. Hewitson, and M. Landry. Making h (t) for ligo. *Classical and Quantum Gravity*, 21(20):S1723, 2004.

[98] J. Skilling. Skilling, j.: Nested sampling for general bayesian computation. bayesian anal. 1(4), 833-860. 1:833–860, 12 2006.

[99] R. H. Swendsen and J.-S. Wang. Replica monte carlo simulation of spin-glasses. *Phys. Rev. Lett.*, 57:2607–2609, Nov 1986.

[100] A. Taracchini, A. Buonanno, Y. Pan, T. Hinderer, M. Boyle, D. A. Hemberger, L. E. Kidder, G. Lovelace, A. H. Mroué, H. P. Pfeiffer, M. A. Scheel, B. Szilágyi, N. W. Taylor, and A. Zenginoglu. Effective-one-body model for black-hole binaries with generic mass ratios and spins. *Phys. Rev. D*, 89:061502, Mar 2014.

[101] S. R. Taylor, M. Vallisneri, J. A. Ellis, C. M. F. Mingarelli, T. J. W. Lazio, and R. van Haasteren. Are we there yet? time to detection of nanohertz gravitational waves based on pulsar-timing array limits. *The Astrophysical Journal Letters*, 819(1):L6, 2016.

[102] S. A. Usman, A. H. Nitz, I. W. Harry, C. M. Biwer, D. A. Brown, M. Cabero, C. D. Capano, T. D. Canton, T. Dent, S. Fairhurst, M. S. Kehl, D. Keppel, B. Krishnan, A. Lenon, A. Lundgren, A. B. Nielsen, L. P. Pekowsky, H. P. Pfeiffer, P. R. Saulson, M. West, and J. L. Willis. The pycbc search for gravitational waves from compact binary coalescence. *Classical and Quantum Gravity*, 33(21):215004, 2016.

[103] M. Vallisneri. Use and abuse of the fisher information matrix in the assessment of gravitational-wave parameter-estimation prospects. *Phys. Rev. D*, 77:042001, Feb 2008.

[104] J. Veitch and A. Vecchio. Bayesian coherent analysis of in-spiral gravitational wave signals with a detector network. *Phys. Rev. D*, 81:062003, Mar 2010.

[105] M. Zevin, S. Coughlin, S. Bahaadini, E. Besler, N. Rohani, S. Allen, M. Cabero, K. Crowston, A. K. Katsaggelos, S. L. Larson, T. K. Lee, C. Lintott, T. B. Littenberg, A. Lundgren, C. sterlund, J. R. Smith, L. Trouille, and V. Kalogera. Gravity spy: integrating advanced ligo detector characterization, machine learning, and citizen science. *Classical and Quantum Gravity*, 34(6):064003, 2017.

APPENDICES

APPENDIX A

STATIONARY PHASE APPROXIMATION INVERSION

In this appendix we outline the calculation used to develop the $f(t)$ track from a PhenomC waveform that we employed in Chapter 8.

The GW phase in the frequency domain can be written as

$$\psi(f) = 2\pi \int^{f/2} dF \left(2 - \frac{f}{F}\right) \frac{F}{\dot{F}(F)} \tag{A.1}$$

With F being the orbital frequency and f the Fourier frequency. We write $\dot{F}^{-1}$ as power series, $\dot{F}^{-1} = \sum_k \alpha_k F^{a_k}$, so Eq. A.1 becomes

$$\begin{aligned} \psi(f) \quad &= 4\pi \int^{f/2} \sum_k \alpha_k F^{a_k+1} dF - 2\pi f \int^{f/2} \sum_k \alpha_k F^{a_k} dF & \text{(A.2)} \\ &= 4\pi \sum_k \left(\frac{f}{2}\right)^{a_k+2} \alpha_k \left(\frac{1}{a_k+2} - \frac{1}{a_k+1}\right). & \text{(A.3)} \end{aligned}$$

The PhenomC waveform generator has the coefficients in the form [95]

$$\psi(f) = \sum_i \beta_i f^{b_i}, \tag{A.4}$$

so by matching coefficients and powers, we get the relations

$$\begin{aligned} b_k \quad &= a_k + 2 & \text{(A.5)} \\ \beta_k \quad &= \alpha_k 2^{-(a_k+2)} \left(\frac{1}{a_k+2} - \frac{1}{a_k+1}\right) & \text{(A.6)} \end{aligned}$$

$$t(f) = \int^{f/2} dF \sum_k \alpha_k F^{a_k} \equiv \sum_k \gamma_k f^{c_k} \tag{A.7}$$

Integrating then gives the relations:

$$a_k + 1 = c_k \tag{A.8}$$

$$\frac{\alpha_k}{(a_k+1)2^{a_k+1}} = \gamma_k, \tag{A.9}$$

which can be used with the relations given in Eqs. A.5 and A.6 to give our final $f(t)$ expression (which in this case is actually an expression for $t(f)$).

APPENDIX B

AD-HOC WAVEFORMS

In the development and testing of LIGO data analysis pipelines, it is imperative to test their performances with simulated signals. For pipelines that specifically look for compact binary coalescences, the natural set of test injections are simulated binary black hole or neutron star waveforms made from our templates. Unmodeled searches, which are sensitive to a wider range of signals than the CBC searches can also pick up these sorts of signals, and so they are commonly used to test the short-duration burst pipelines as well. But given that we want our burst pipelines to be ready to confidently detect and reconstruct and possible signal. It is then important to develop simulated signals that can in some sense mimic GW signals from "unknown" sources. Two waveforms typically used are sine-Gaussian waveforms, and white noise bursts. We used both these waveforms in numerous tests of the *BayesWave* algorithm and cWB+BW pipeline, so here we give an overview of how those signals are constructed.

Sine-Gaussian Waveforms

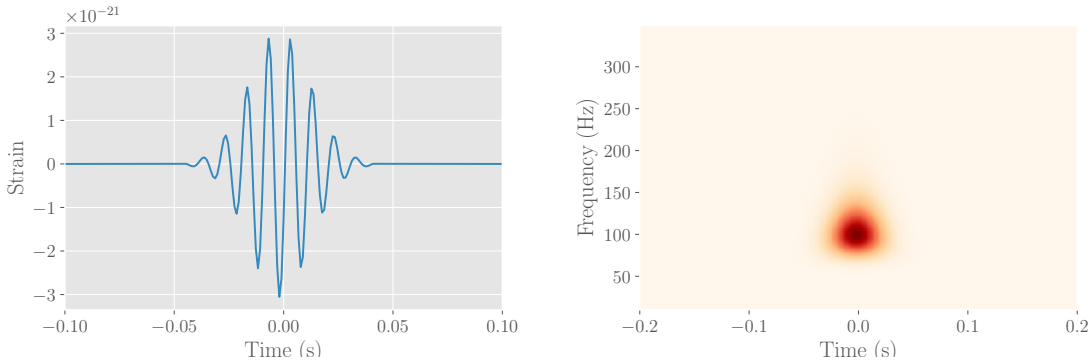

Figure B.1: Example of a SG waveform in the time domain (left) and in a time-frequency representation using a Q-scan as described in Appendix C. For this example, the SG waveform has a central frequency of $f_0 = 100$ Hz, and quality factor $Q = 9$.

The sine-Gaussian waveforms (SGs) are exactly as their name suggests: sine waves in a Gaussian envelope. These can be described mathematically via their h_+

and $h_\times$ strains [8]:

$$h_+(t) = e^{-t^2/\tau^2} \sin(2\pi f_0 t)$$
$$h_\times(t) = e^{-t^2/\tau^2} \cos(2\pi f_0 t) \tag{B.1}$$

where $\tau = Q/(\sqrt{2}\pi f_0)$, and f_0 is the central frequency and Q is the usual quality factor. SGs can be thought of as a very generic waveform with well defined frequency content. Furthermore, heavier stellar mass black hole binaries can be roughly approximated as SGs, and so these signals are similar to what we could expect to see in nature.

White Noise Bursts

White noise burst waveforms (WNBs) are an excess of Gaussian noise, limited in time and frequency extent. These simulated waveforms are also unpolarized. These waveforms don't have an analytic form, but can be characterized by their extent in frequency Δf, the lower bound on their frequency f_{low} and extent in time τ.

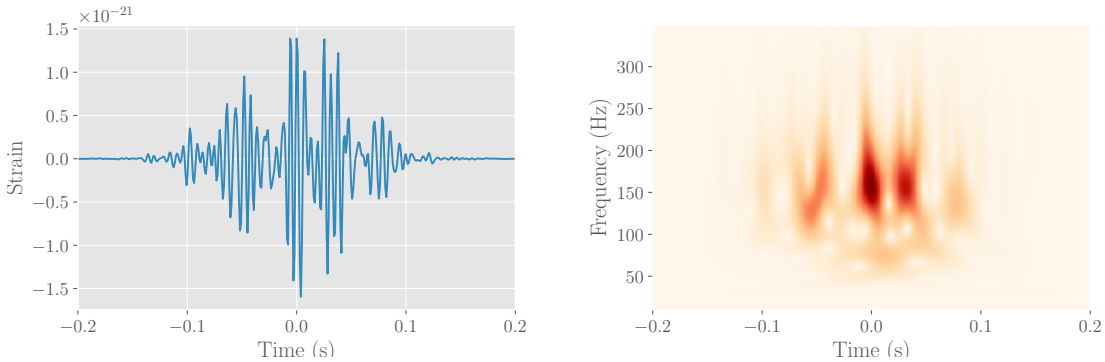

Figure B.2: Example of a WNB waveform in the time domain (left) and in a time-frequency representation using a Q-scan as described in Appendix C. For this example, the WNB waveform has lower frequency $f_{\text{low}} = 100$ Hz, bandwidth $\Delta f = 100$ Hz, and duration $\tau = 0.1$ s.

APPENDIX C

WAVELET TRANSFORMS AND Q-SCANS

For almost any set of data there are a number of ways to look at the data. First, there is as a time series which shows how the measured value changes over time. We can also look at the data in the Fourier (or frequency) domain, where one can see how much of the signal is at each frequency. The time and frequency representations of a data set are related through a Fourier transform:

$$\tilde{x}(f) = \int_{-\infty}^{\infty} x(t) e^{-2\pi i f t} dt \tag{C.1}$$

$$x(t) = \int_{-\infty}^{\infty} \tilde{x}(f) e^{2\pi i f t} df \tag{C.2}$$

where $x(t)$ is the data in the time domain, and $\tilde{x}(f)$ is the data in the frequency domain. The limitation of time series data is that there is no frequency resolution, and similarly for series of frequency domain data we have no time resolution. A method to capture some of the features of both the time *and* frequency data is to use a wavelet transformation. A wavelet transformation lies somewhere between frequency-only and time-only representations of the data, and is similar to Eq. C.1 but using a different kernel function:

$$X(t_0, f_0, Q) = \int_{-\infty}^{\infty} x(t) w(t - t_0, f_0, Q) e^{-2\pi i f t} dt \tag{C.3}$$

where the function $w(t-t_0, f, Q)$ is some time domain function center on time t_0 with a quality factor Q that describes its extent in time which can be thought of as the "resolution" of the transformation. The wavelet transformation gives an estimate of the power in time-frequency space around t_0 and f_0, the space of which is determine by the wavelet function $w(t-t_0, f_0, Q)$. An in depth discussion of wavelet transformations can be found in [39].

One choice of wavelet kernel functions is a sine-Gaussian wavelet, as described

in Eq. 4.1. In the *BayesWave* analysis we use this wavelet transformation to make Q-scans, also called spectrograms. This is applying the wavelet transformation at a variety of central frequencies and times, and can be done at a range of Q's to produce different resolution Q-scans. As described in Chapters 4 and 6, these Q-scans are used as proposal distributions in the MCMC algorithm, as well as a tool to visualize where the power in time-frequency space is for our data and reconstructed waveforms.

APPENDIX D

POST-NEWTONIAN EXPANSION OF F(T)

The full expression for $f(t)$ for a binary system of total mass M, symmetric mass ratio η, spins $\chi 1$ and $\chi 2$, and a time of coalescence of t_c in the 3.5PN expansion including spins is given by

$$f(t) = \frac{\theta^{-3/8}}{8\pi M} \sum_{k=0}^{7} \alpha_k \theta^{-k/8} \tag{D.1}$$

where $\theta = \frac{\eta}{5M}(t_c - t)$. The coefficients are:

$$\alpha_0 = 1 \tag{D.2}$$

$$\alpha_1 = 0 \tag{D.3}$$

$$\alpha_2 = \frac{11\eta}{32} + \frac{743}{2688} \tag{D.4}$$

$$\alpha_3 = -\frac{19\eta\chi 1}{80} - \frac{19\eta\chi 2}{80} + \frac{113\chi}{160} - \frac{3\pi}{10} \tag{D.5}$$

$$\alpha_4 = \frac{371\eta^2}{2048} + \frac{3\eta\chi 1\chi 2}{512} + \frac{56975\eta}{258048} - \frac{15\chi^2}{64} + \frac{1855099}{14450688} \tag{D.6}$$

$$\begin{aligned}\alpha_5 = &-\frac{17\eta^2\chi 1}{128} - \frac{17\eta^2\chi 2}{128} - \frac{9}{128}\eta\chi\chi 1\chi 2 + \frac{13\eta\chi}{128} - \\ &\frac{1285\eta\chi 1}{1152} - \frac{1285\eta\chi 2}{1152} + \frac{13\pi\eta}{256} + \frac{3\chi^3}{128} + \frac{147101\chi}{64512} - \frac{7729\pi}{21504}\end{aligned} \tag{D.7}$$

$$\begin{aligned}\alpha_6 = &\frac{107\gamma\mathrm{E}}{280} + \frac{235925\eta^3}{1769472} + \frac{361\eta^2\chi 1^2}{38400} + \frac{23929\eta^2\chi 1\chi 2}{1228800} + \frac{361\eta^2\chi 2^2}{38400} - \\ &\frac{30913\eta^2}{1835008} - \frac{55\eta\chi^2}{2048} - \frac{2147\eta\chi\chi 1}{38400} - \frac{2147\eta\chi\chi 2}{38400} + \frac{743\eta\chi 1\chi 2}{1376256} + \\ &\frac{19\pi\eta\chi 1}{800} + \frac{19\pi\eta\chi 2}{800} - \frac{451\pi^2\eta}{2048} + \frac{25302017977\eta}{4161798144} + \frac{88219\chi^2}{1433600} - \frac{113\pi\chi}{1600} + \frac{53\pi^2}{200} - \\ &\frac{720817631400877}{288412611379200} - \frac{107\log(\theta/256)}{2240}\end{aligned} \tag{D.8}$$

$$\alpha_7 = \frac{31841\eta^3\chi1}{92160} + \frac{31841\eta^3\chi2}{92160} - \frac{645\eta^2\chi\chi1\chi2}{2048} - \frac{138941\eta^2\chi}{92160} - \frac{299\eta^2\chi1^2\chi2}{20480} - \frac{299\eta^2\chi1\chi2^2}{20480} - \frac{5181145\eta^2\chi1}{3096576} - \frac{5181145\eta^2\chi2}{3096576} + \frac{141769\pi\eta^2}{1290240} + \frac{215\eta\chi^3}{2048} + \frac{185}{512}\eta\chi^2\chi1 + \frac{185}{512}\eta\chi^2\chi2 - \frac{41127\eta\chi\chi1\chi2}{143360} + \frac{8424467\eta\chi}{6193152} - \frac{63\pi\eta\chi1\chi2}{2560} - \frac{367335473\eta\chi1}{92897280} - \frac{367335473\eta\chi2}{92897280} - \frac{97765\pi\eta}{258048} - \frac{131707\chi^3}{172032} + \frac{45\pi\chi^2}{64} + \frac{5386538891\chi}{650280960} - \frac{188516689\pi}{433520640} \tag{D.9}$$