Analysis Framework for the Prompt Discovery of Compact Binary Mergers in Gravitational-wave Data

Cody Messick,[1,a] Kent Blackburn,[2] Patrick Brady,[3] Patrick Brockill,[3] Kipp Cannon,[4] Sarah Caudill,[3] Sydney J. Chamberlin,[1] Jolien D. E. Creighton,[3] Ryan Everett,[1] Chad Hanna,[1] Ryan N. Lang,[3] Tjonnie G. F. Li,[5] Duncan Meacher,[1] Chris Pankow,[6] Stephen Privitera,[7] Hong Qi,[3] Surabhi Sachdev,[2] Laleh Sadeghian,[3] B. Sathyaprakash,[8] Leo Singer,[9] E. Gareth Thomas,[10] Leslie Wade,[11] Madeline Wade,[11] and Alan Weinstein[2]

[1]*Department of Physics, The Pennsylvania State University, University Park, PA 16802, USA*
[2]*LIGO Laboratory, California Institute of Technology, MS 100-36, Pasadena, California 91125, USA*
[3]*Leonard E. Parker Center for Gravitation, Cosmology, and Astrophysics,*
University of Wisconsin-Milwaukee, Milwaukee, WI 53201, USA
[4]*Canadian Institute for Theoretical Astrophysics, 60 St. George Street,*
University of Toronto, Toronto, Ontario, M5S 3H8, Canada
[5]*Department of Physics, The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong, China*
[6]*Center for Interdisciplinary Exploration and Research in Astrophysics (CIERA) and Department of Physics and Astronomy,*
Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, USA
[7]*Albert-Einstein-Institut, Max-Planck-Institut für Gravitationsphysik, D-14476 Potsdam-Golm, Germany*
[8]*School of Physics and Astronomy, Cardiff University, 5 The Parade, Cardiff, CF24 3AA, UK*
[9]*NASA/Goddard Space Flight Center, Greenbelt, MD 20771, USA*
[10]*University of Birmingham, Birmingham, B15 2TT, UK*
[11]*Department of Physics, Hayes Hall, Kenyon College, Gambier, Ohio 43022, USA*
(Dated: April 18, 2016)

We describe a stream-based analysis pipeline to detect gravitational waves from the merger of binary neutron stars, binary black holes, and neutron-star–black-hole binaries within ~ 1 minute of the arrival of the merger signal at Earth. Such low-latency detection is crucial for the prompt response by electromagnetic facilities in order to observe any fading electromagnetic counterparts that might be produced by mergers involving at least one neutron star. Even for systems expected not to produce counterparts, analysis of the data in low-latency for their GW signals is useful for deciding when not to point telescopes, and as feedback to observatory operations. The pipeline also operates in an offline mode, in which it incorporates more refined information about data quality and employs acausal methods that are inapplicable to the online mode.

I. INTRODUCTION

The field of gravitational-wave astronomy has come to life in a spectacular way, with the first detection of gravitational waves on September 14, 2015 [1] by the two detectors of the Laser Interferometer Gravitational-wave Observatory (LIGO) [2]. These detectors are currently undergoing further commissioning and will reach design sensitivity in the next few years. Additionally, they will be joined by a network of gravitational-wave observatories that include Advanced Virgo [3], KAGRA [4], and a third LIGO observatory in India [5]. We expect this network to bring more observations of binary black hole mergers [6], as well as binary neutron star (BNS) and neutron-star–black-hole (NSBH) mergers [7].

As we enter the era of gravitational-wave astronomy, the need for low-latency analyses becomes critical. Gravitational waves from BNS and NSBH mergers are expected to be paired with electromagnetic emission and neutrinos [8–10]. Gravitational-wave-triggered electromagnetic observations may lead to the detection of prompt short gamma-ray bursts and high-energy neutrinos within seconds, followed by X-ray, optical, and radio afterglows days to years later. Multimessenger observations will aid in our understanding of astrophysical processes and increase our search sensitivity [8, 11]. Additionally, even in the absence of a counterpart, the rapid identification of gravitational waves has a number of benefits. Low-latency detection allows us to provide feedback to commissioners when search sensitivity drops unexpectedly, helping to return the detector to its nominal state [12]. Furthermore, upon identification of a candidate, we can submit timely requests to minimize detector changes in order to gather enough data to reliably estimate the search background and perform followup calibration measurements.

In this work, we present the GstLAL-based inspiral pipeline, a gravitational-wave search pipeline based on the GstLAL library [13], and derived from GStreamer [14] and the LIGO Algorithm Library [15]. The pipeline can operate in a low-latency mode to ascertain whether a gravitational-wave signal is present in data, provide point estimates for the binary parameters, and estimate event significance. The pipeline can also operate in an "offline" configuration that can be used to process archival gravitational-wave data with additional background statistics and data quality information.

The GstLAL-based inspiral pipeline expands on the parameter space covered by previous low-latency

[a] Cody.Messick@ligo.org

searches [16–19]. In addition, it extends many of the techniques used in prior searches for compact binary coalescences [20, 21] to operate in a fully parallel, stream-based mode that allows for the identification of candidate gravitational-wave events within seconds of recording the data. The key differences include: (1) time-domain [22] rather than frequency-domain [23] matched filtering, (2) time-domain rather than frequency-domain [24] signal consistency tests to reject non-stationary noise transients, (3) a multidimensional likelihood ratio ranking statistic to robustly identify gravitational-wave candidates in a way that automatically adjusts to the properties of the noise [25], and (4) a background estimation technique that relies on tracking noise distributions to allow rapid evaluation of significance of identified candidates [26].

This paper is organized as follows: In Sec. II, we discuss inputs to the low-latency and offline analyses, the online acquisition of data, measurement of the power spectral density (PSD), and whitening and conditioning of the data for matched filtering. We also present the basic offline and low-latency workflows in Fig. 1 and Fig. 2, respectively. In Sec. III, we discuss the matched-filter algorithm and our procedure for producing a list of ranked candidate events. In Sec. IV, we explain the significance calculation for identified candidate events and the procedure for responding to significant events via alerts to our observing partners. Differences between the offline and low-latency operation modes will be highlighted when relevant.

II. MATCHED FILTERING INPUT

Matched filtering algorithms for compact binary mergers have traditionally filtered the data $d(t)$ against a set of complex template waveforms $\{h_i^c(t)\}$ in the frequency domain using the relation

$$z_i(t) = x_i(t) + iy_i(t) = 4 \int_0^\infty \mathrm{d}f \frac{\widetilde{h}_i^{c*}(f)\widetilde{d}(f)}{S_n(f)} e^{2\pi i f t}, \quad (1)$$

where $z_i(t)$ is the complex signal-to-noise ratio (SNR) using the ith template, $x_i(t)$ is the matched filter response to a gravitational wave signal with orbital coalescence phase ϕ_0 (the real part of the template in the time domain), $y_i(t)$ is the matched filter response to the same signal with orbital coalescence phase $\phi_0 + \pi/4$ (the imaginary part of the template in the time domain), and $S_n(f)$ is the single-sided noise power spectral density (PSD). The templates are normalized such that

$$1 = 4 \int_0^\infty \mathrm{d}f \frac{\left|\widetilde{h}_i^c(f)\right|^2}{S_n(f)}. \quad (2)$$

Defining the SNR, $\rho(t)$, as the modulus of the complex SNR, $z(t)$, allows one to search efficiently over the unknown coalescence phase, while its expression in the
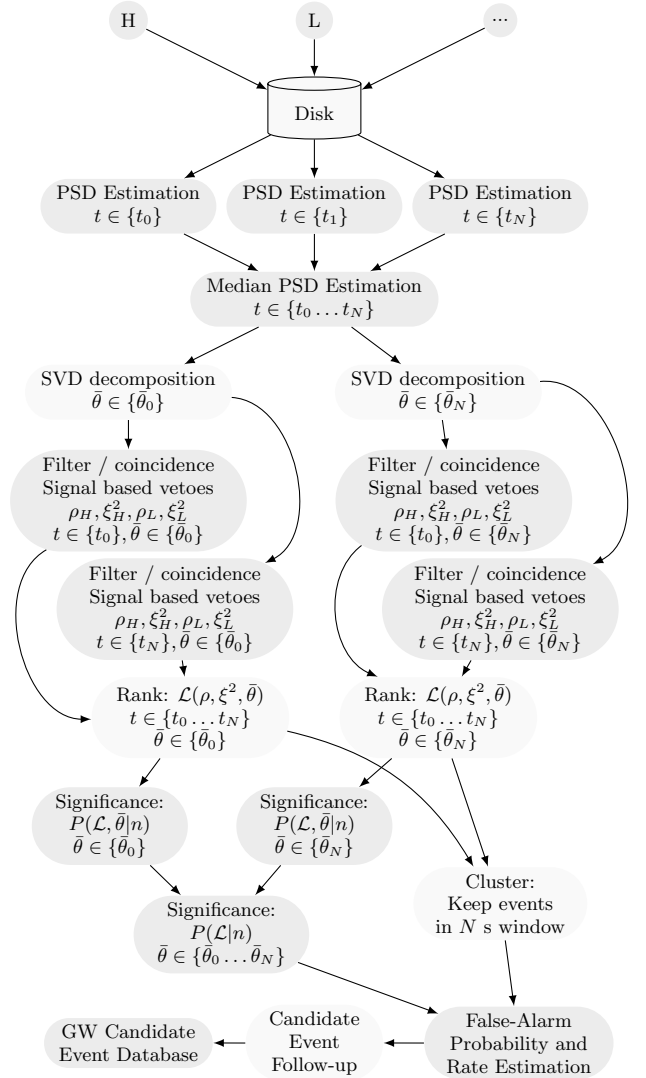

FIG. 1. Diagram of the offline search mode of the Gst-LAL based inspiral pipeline. First, data is transferred from each observatory (H,L,...) to a central computing cluster (Sec. II A). Next, data is read from disk and the PSD is estimated (Sec. II B) in chunks of time $t_0, t_1, \ldots, t_N$ for each observatory. The median over the entire analysis time of each observatory PSD estimate is computed. The input template bank, which is generated upstream of the analysis, is split into regions of similar parameters $\bar{\theta}_0, \bar{\theta}_1, \ldots, \bar{\theta}_N$ (Sec. II D) and then decomposed into a set of orthonormal filters weighted by the median PSD for each observatory. The data is filtered to produce a series of triggers characterized by SNR, ρ, signal consistency check, ξ^2 (Secs. III A, III B, and III C), and coalescence time. Coincident triggers between detectors are identified and promoted to the status of events (Sec. III D). Events are ranked according to their relative probability of arising from signal versus noise (Sec. III E). The data is then reduced to the most highly ranked event in 8 second windows (Sec. III F). In parallel, triggers not found in coincidence are used to construct the probability of obtaining a given event from noise, $P(\Lambda|n)$. Finally, the event significance and False-Alarm Rate are estimated (Sec. IV A).
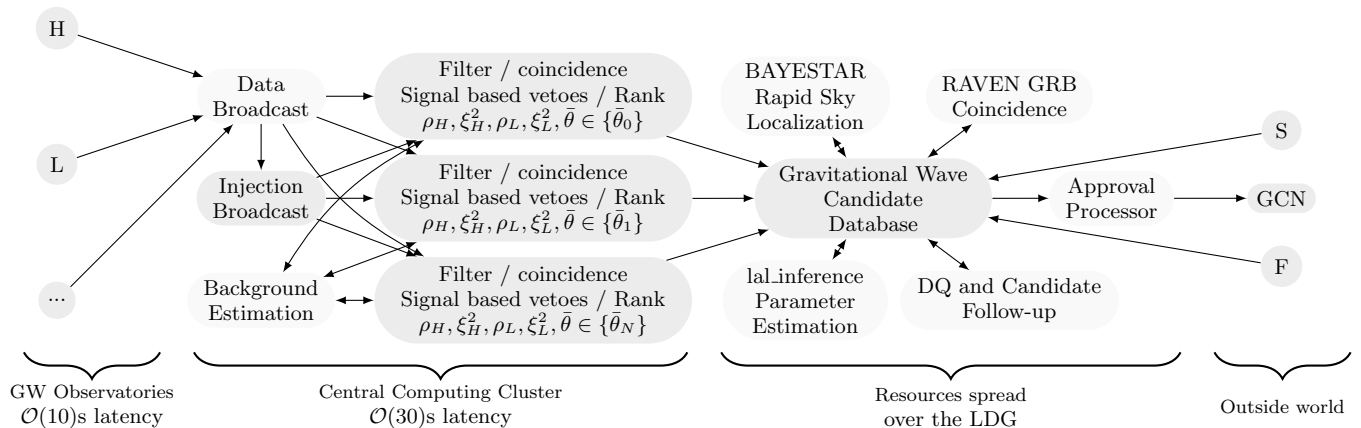
FIG. 2. Diagram of the low-latency search mode of the GstLAL based inspiral pipeline. First, data is received over a network connection from each observatory to a data broadcaster in a central computing facility. The data is then broadcast over the entire cluster with an efficient multicast protocol. The online analysis uses precomputed bank decompositions for each observatory from reference PSDs as input to jobs that combine the filtering, vetoing, coincidence, ranking, and significance estimation steps from the offline pipeline. The online analysis work-flow can no longer be described as a directed acyclic graph, and in fact, data from each filtering job is exchanged bi-directionally and asynchronously to a process that constantly evaluates the global background estimates for the entire analysis. Events that are identified by any one filtering job, which pass a predetermined threshold, are sent to the gravitational wave candidate database within a matter of seconds of the data being recorded at the observatories.

frequency domain allows one to efficiently implement matched filtering using FFT routines.

The GstLAL-based inspiral pipeline, however, performs matched filtering in the time domain with real templates $\{h_i(t)\}$. The matched filter output is thus the real-valued $x_i(t)$ instead of the complex-valued $z_i(t)$. We can recast Eq. (1) in the time domain using the convolution theorem, which gives

$$x_i(t) = 2 \int_{-\infty}^{\infty} \mathrm{d}f \frac{\widetilde{h}_i^*(f)\widetilde{d}(f)}{S_n(|f|)} e^{2\pi i f t} \tag{3a}$$

$$= 2 \int_{-\infty}^{\infty} \mathrm{d}\tau \, \hat{h}_i(\tau)\hat{d}(t+\tau), \tag{3b}$$

where

$$\hat{d}(\tau) = \int_{-\infty}^{\infty} \mathrm{d}f \frac{\widetilde{d}(f)}{\sqrt{S_n(|f|)}} e^{2\pi i f \tau} \tag{4}$$

is the whitened data; the whitened template $\hat{h}_i(\tau)$ is defined similarly. As a consequence of using real templates, Eq. (3) returns the matched filter response to a single coalescence phase, while Eq. (1) returns the response to two phases. Thus, Eq. (3) must be evaluated a second time using the template corresponding to the $\pi/4$-shifted phase in order to compute the SNR. To account for using twice as many templates, the template index, i, used on real templates is related to the index used on complex templates via

$$h_{2i}(t) = \mathrm{Re}\left[h_i^c(t)\right], \tag{5a}$$

$$h_{2i+1}(t) = \mathrm{Im}\left[h_i^c(t)\right]. \tag{5b}$$

A different template normalization is also used, specifically

$$1 = 4 \int_0^{\infty} \mathrm{d}f \frac{\left|\widetilde{h}_i(f)\right|^2}{S_n(f)}. \tag{6}$$

In this section, we discuss the inputs to the time-domain matched filtering calculation expressed in Eq. (3). We begin by discussing the low-latency distribution of the data itself. We then describe our method for estimating the PSD, which we use to construct the whitened data $\hat{d}(t)$ and whitened templates $\hat{h}_i(t)$. In describing our construction of the whitened data stream, we also describe the removal of loud noise transients and dealing with data dropouts in the low-latency broadcast. Finally, we describe the construction of the whitened template filters, which involves a number of computational enhancements to reduce the cost of filtering in the time domain.

A. Data Acquisition

Gravitational wave strain data acquired at the LIGO sites is digitized at a sample rate of 16384 Hz and bundled into IGWD frames, a custom LIGO file format described in Ref. [27], on a four second cadence. Although advanced LIGO (aLIGO) monitors a host of auxiliary environmental and instrumental control system channels that are not directly used to construct the strain data, information about the state of the instrument and data quality are distilled from these channels into a single state vector channel that indicates instrument health and data

quality. The four second frames containing gravitational wave strain and the state vector channels are delivered for low-latency processing at computing clusters across the LIGO Data Grid within ~ 12 seconds of the data being acquired.

Searches for compact binary coalescences require using hundreds or thousands of compute nodes in parallel to process all the possible template waveforms. Low-latency data must be made available to all of these nodes as soon as it arrives, thus an efficient multicast protocol is used to broadcast the data in low-latency to the entire cluster. The nature of the low-latency transmission incurs some small data loss within the tolerances acceptable to the pipeline, with efforts underway to reduce these losses.

B. PSD

Abstractly, we define the (one-sided) noise power spectral density $S_n(f)$ as

$$\langle \widetilde{n}(f)\widetilde{n}^*(f')\rangle = \frac{1}{2}S_n(f)\delta(f - f'), f > 0 \qquad (7)$$

where $\langle \cdots \rangle$ denotes an ensemble average over realizations of the detector noise $n(t)$, which is assumed to be stationary and Gaussian. In practice, we cannot use Eq. (7) to calculate the PSD for a variety of reasons. To begin with, our knowledge of the detector noise comes exclusively from the observed data, which may contain signal in addition to noise. Furthermore, real data may contain brief departures from stationarity (commonly called "glitches"), which we do not want to contribute to the PSD estimate. Finally, the PSD can drift slowly over time scales shorter than the duration of a typical detector lock segment, and we want to track these changes. For low-latency applications, we also require a PSD estimate that converges quickly using only data in the past, so that we obtain an accurate estimate of the PSD as soon as possible after the data begin to flow. In this subsection, we discuss the PSD estimation algorithm and how the result is used to whiten the data and template bank. We also present the results of a study done on the convergence of an estimated PSD to its known spectrum.

1. Estimation and Whitening

We use a median and a running geometric mean to meet these requirements for each analyzed segment of data. The median estimate operates on medium time scales and is robust against shorter time-scale fluctuations in the noise, while the running geometric mean tracks longer time-scale changes in the PSD, averaging the PSD estimates with the most recent estimates weighted more strongly. The time scales of the median and geometric mean are set, respectively, by the tunable parameters n_{med} and n_{avg}.

At the time of writing, the typical values used are $n_{\mathrm{med}} = 7$ and $n_{\mathrm{avg}} = 64$.

The PSD calculation begins by partitioning the strain time series into blocks of length N points with each block overlapping the previous by $N/2 + Z$ points, where N and Z are even-valued integers. Each block of data, denoted $d_j[k]$, is windowed and Fourier transformed,

$$\widetilde{d}_j[\ell] = \frac{\Delta t}{\sigma_w} \sum_{k=0}^{N-1} d_j[k]w[k]e^{-2\pi i\ell k/N}, \qquad (8)$$

where Δt is the time sample step,

$$w[k] = \begin{cases} 0, & 0 \leq k < Z \\ \sin^2 \frac{\pi(k-Z)}{N-2Z}, & Z \leq k < N - Z \\ 0, & N - Z \leq k \leq N - 1 \end{cases} \qquad (9)$$

is a zero-padded Hann window function, and

$$\sigma_w = \sqrt{\frac{\sum_{k=0}^{N-1} w[k]^2}{N}}, \qquad (10)$$

is the standard deviation of the window function. The zero-padded Hann window is defined such that the sequence of overlapping window functions sum to unity everywhere. The squared magnitude of Eq. (8) is the instantaneous PSD, which has a frequency resolution of $\Delta f = \frac{1}{N\Delta t}$.

The median of the most recent n_{med} instances of the instantaneous PSD, $S_j^{\mathrm{med}}[\ell]$, is determined for each frequency bin ℓ. Mathematically,

$$S_j^{\mathrm{med}}[\ell] = \mathrm{median}\{ \ 2\Delta f|\widetilde{d}_k[\ell]|^2 \ \}_{k=j-n_{\mathrm{med}}}^{k=j}. \qquad (11)$$

The median is relatively insensitive to short time-scale fluctuations, which must persist for a time scale set by $\frac{1}{2}n_{\mathrm{med}}(N/2 - Z)$ in order to affect the median.

The logarithm of the running geometric mean of median estimated PSDs, $\log S_j[\ell]$, is computed from one part $\log S_j^{\mathrm{med}}[\ell]$ and $(n_{\mathrm{avg}} - 1)$ parts $\log S_{j-1}[\ell]$. Mathematically,

$$S_j[\ell] = \exp\left[\frac{n_{\mathrm{avg}} - 1}{n_{\mathrm{avg}}} \log S_{j-1}[\ell] + \frac{1}{n_{\mathrm{avg}}} \log \frac{S_j^{\mathrm{med}}[\ell]}{\beta} \right]. \qquad (12)$$

Changes to the PSD must persist for $\sim n_{\mathrm{avg}}(N/2 - Z)$ sample points or longer to be fully accounted for by Eq. (12).

To whiten the data and the templates, the geometric mean is converted into an arithmetic mean. Assuming the noise is a stationary, Gaussian process allows us to assume that the measured frequency bins of the estimated PSD are χ^2-distributed random variables. The arithmetic mean of a χ^2-distributed random variable is equal to the geometric mean multiplied by $\exp(\gamma)$, where γ is Euler's constant. If the noise assumptions are violated, then the true arithmetic mean of the spectrum

will differ from the measured spectrum by some unknown factor. This estimated arithmetic mean is referred to as $S_n(f)$ in the continuum limit (see e.g. Eq. 7).

The low-latency operating mode must whiten the data and update the running geometric mean of the PSD at the same time. The whitening process is done after the running geometric mean has been updated and is performed by dividing each frequency bin of Eq. 8 by the square root of the corresponding frequency bin in the estimated arithmetic mean of the PSD. Mathematically,

$$\widetilde{\hat{d}}_j[\ell] = \frac{\widetilde{d}_j[\ell]}{\sqrt{S_j[\ell] \exp(\gamma)}}, \tag{13}$$

$$\hat{d}_j[k] = 2\Delta t \sqrt{\sum_{m=0}^{N-1} w[m]^2 \Delta f} \sum_{\ell=0}^{N-1} \widetilde{\hat{d}}_j[\ell] e^{2\pi i \ell k/N}. \tag{14}$$

The extra terms in the inverse Fourier transform are necessary for unity variance. The low-latency analysis typically uses $N = 8f_s$ and $Z = 2f_s$, where $f_s = 1/\Delta t$ is the sampling frequency, giving $1/4$ Hz frequency resolution. This introduces four seconds of latency into the analysis.

Unlike the low-latency case, the offline analysis begins with a known list of data segments. The PSD of each segment is estimated using $N = 32f_s$ and $Z = 0$; the final result of the running average is written to disk and referred to as a 'reference PSD.' The median of the reference PSDs is used to whiten the template bank before matched filtering. However, the data segments are whitened in a procedure similar to the low-latency analysis, using $N = 32f_s$ and $Z = 8f_s$. The only procedural difference between the offline and low-latency whitening steps is that the offline analysis seeds the running average with the segment's reference PSD.

2. Convergence

For low-latency applications, we require a PSD estimate that converges quickly using only data in the past, so that we obtain an accurate estimate of the PSD as soon as possible after the data begin to flow. To quantify the convergence, we create noise with a known power spectrum and compute a quantity that is proportional to the expected SNR for a given PSD in the absence of noise (commonly referred to as the 'optimal SNR'). In the stationary phase approximation, binary waveforms in the frequency domain, $h(f)$, are proportional to $f^{-7/6}$ [28], thus

$$\rho \propto \int_{f_1}^{f_2} df \frac{f^{-7/3}}{S_n(f)}. \tag{15}$$

We choose $f_1 = 10$ Hz and $f_2 = 2048$ Hz. Specifically, we compare the quantity computed using the *measured* spectrum $S_n(f)$, which we denote simply as ρ, to the SNR computed from the *known* spectrum $\hat{S}_n(f)$, which

we denote as $\hat{\rho}$. Figure 3 shows the fractional change of ρ with respect to $\hat{\rho}$ as a function of time,

$$\frac{\delta\rho}{\rho}(t) = \frac{\rho(t) - \hat{\rho}}{\rho(t)}. \tag{16}$$

We find that convergence happens quickly relative to the length of the data. The approximation of the true PSD does not affect the measured SNR after tens of seconds.
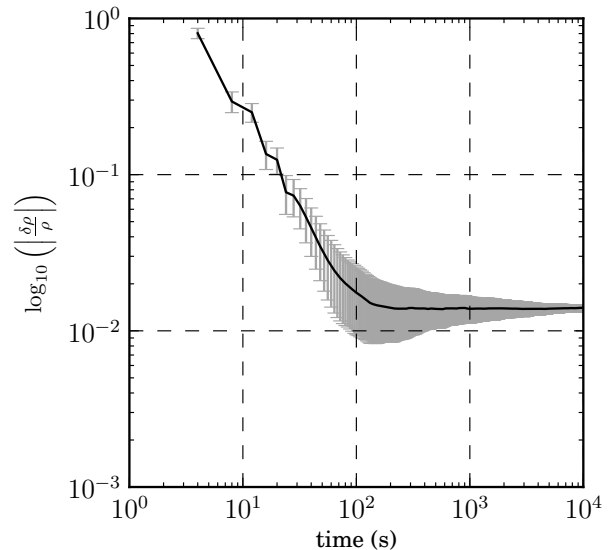

FIG. 3. PSD convergence properties. Estimating the PSD is a critical part of ensuring that events are detected and assigned the appropriate significance. This figure illustrates the convergence properties of the PSD estimation in terms of the impact on SNR. Within 20 s the PSD will have an $\mathcal{O}(10\%)$ impact on SNR, and within 200 s the impact drops to $\mathcal{O}(1\%)$, where it remains.

C. Data Conditioning

Matched filtering is optimal under the condition that the noise, $n(t)$, is both stationary and Gaussian. Although non-stationarity on long time scales can be handled by tracking the PSD, short noise transients, commonly referred to as "glitches," can cause high-SNR matched filter outputs that mimic signal detections. Glitches are handled by either removing them from the data or using signal consistency checks to vet the matched filter output. Sec. III C provides more detail on the latter.

The whitening filter in the frequency domain can be approximated by a rectangle window function of width $\mathcal{O}(100)$ Hz, thus the filter in the time domain can be approximated by the rectangle function's Fourier transform, the sinc function.

The GstLAL based inspiral pipeline removes glitches from the data in two ways. In some cases, the matched

filter outputs of glitches have considerably higher amplitude than any expected output from a compact binary signal and can thus be safely removed from the data through a process called gating. Once the data has been whitened, it has unit variance. If a momentary excursion greater than some number of standard deviations, σ, is observed in the whitened data, then the gating process zeros the excursion in the whitened data with a 0.25 second padding on each side. An example of this is shown in Fig. 4. When gating the strain data, care must be taken to choose a threshold that will not discard real gravitational wave signals. The threshold is chosen by testing with simulated gravitational wave signals.

In many cases, auxiliary information is available through environmental and instrumental monitors that can ascertain times of clear coupling between local transient noise sources [29, 30]; in cases where data quality is known to be poor, vetoes are applied after the strain data is whitened. Since whitened data is, by definition, uncorrelated between adjacent samples for stationary Gaussian processes, vetoes are applied by simply replacing the whitened data during vetoed times with zeros.

Large noise transients occasionally remain in the absence of clear instrumental or environmental coupling.

In some cases, the matched filter outputs of these glitches have considerably higher amplitude than any expected output from a compact binary signal and can thus be safely removed from the data through a process called gating. Once the data has been whitened, it has unit variance. If a momentary excursion greater than some number of standard deviations, σ, is observed in the whitened data, then the gating process zeros the excursion in the whitened data with a 0.25 second padding on each side. An example of this is shown in Fig. 4. When gating the strain data, care must be taken to choose a threshold that will not discard real gravitational wave signals. The threshold is chosen by testing with simulated gravitational wave signals.

D. Template Bank Decomposition

In order to cover the binary parameter space, we filter the data against a bank of template signals. As the true binary parameter space is continuous, actual signals may not exactly match any one template from the bank; such signals incur a loss of SNR. The parameters of the templates in the bank are chosen to minimize this loss of SNR using as few templates as possible [31–33]. Techniques for efficiently covering the binary parameter space with templates have been extensively developed [34–40]. We assume here that such a template bank has already been constructed, and describe how the bank is decomposed to more efficiently filter the data.

The standard methods for template bank construction naturally lead to banks of highly redundant templates. In the frequency-domain, filtering directly with the physical templates has the advantage of admitting
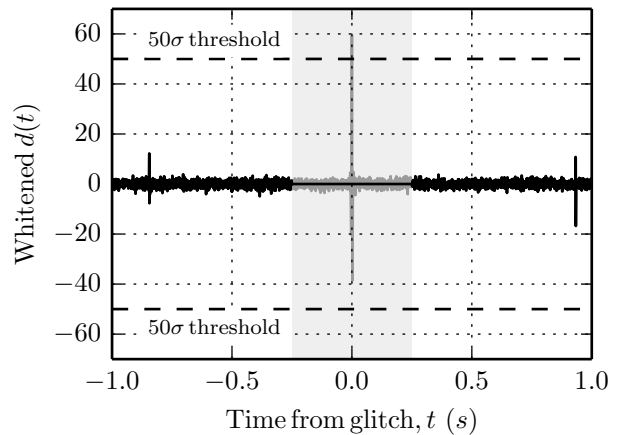

FIG. 4. Data conditioning. In this two second block of LIGO S6 data, three noise transients ("glitches") are visible. The glitch at time zero surpassed the threshold of 50 standard deviations (σ), triggering the gate to veto a ±0.25 second window around the glitch by replacing the data with zeros (black). The gray trace shows what the data looked like prior to gating.

computationally-efficient searches over the unknown signal coalescence phase and time; this advantage is lost in the time-domain. The GstLAL-based inspiral pipeline therefore does not directly filter the data against the physical template waveforms themselves. Rather, it employs the LLOID method [22] (see also Sec. III A), which combines singular value decomposition (SVD) [41–43] with near-critical sampling to construct a reduced set of orthonormal filters with far fewer samples.

In order to prepare the templates for the LLOID decomposition, the template bank is first split into partially-overlapping "split-banks" of templates with similar time-frequency evolution based on the template parameters, as depicted in Fig. 5. Templates corresponding to binary black hole systems with circular orbits and component spins parallel to the orbital angular moment can be characterized by the component masses m_i and the dimensionless spin parameters $\chi_i = \vec{S}_i \cdot \hat{L}/m_i^2$ for $i = 1, 2$, where $\vec{S}_i$ are the spin vectors and $\hat{L}$ is the orbital angular momentum unit vector. Circularized binary neutron star templates with aligned spins can also be characterized by m_i and χ_i, however not as accurately due to neutron-star specific effects such as tidal disruption. The templates for these systems are binned in a two-dimensional space, first by an effective spin parameter χ_{eff},

$$\chi_{\text{eff}} \equiv \frac{m_1\chi_1 + m_2\chi_2}{m_1 + m_2}, \qquad (17)$$

and then by chirp mass $\mathcal{M}$,

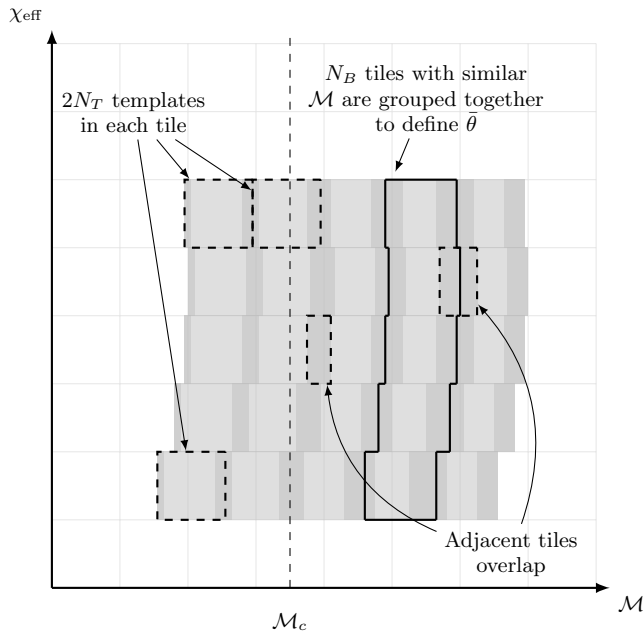$$\mathcal{M} = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}}. \qquad (18)$$

FIG. 5. An illustration of how the physical parameter space is tiled into regions in which the LLOID decomposition is done. The physical parameter space is projected onto the $\mathcal{M}, \chi$ plane. Tiles of equal template number, $2N_T$, are constructed and overlapped in the $\mathcal{M}$ direction by $\mathcal{O}(10\%)$. Above a specified chirp mass, $\mathcal{M}_c$, waveforms that use the full inspiral-merger-ringdown description are used. Below $\mathcal{M}_c$, waveforms that model only the inspiral phase are used. Tiles of similar chirp mass are then grouped together to define a one-dimensional family of similar parameters, $\bar{\theta}$, used in the evaluation of the likelihood-ratio ranking statistic (Sec. III E).

$2N_T$ real templates are placed in each split-bank, where N_T is typically $\mathcal{O}(100)$. The factor of 2 is a result of using two orthogonal real-valued templates in place of 1 complex-valued template (Sec. II). The input templates in adjacent $\mathcal{M}$ bins are overlapped in order to mitigate boundary effects from the SVD. Overlapping regions are clipped after reconstruction such that the output has no redundant template waveforms[1]. The waveforms in each split-bank are then individually whitened, using reference PSDs as described in Sec. II B, and decomposed via the LLOID method, as described below and in Fig. 6.

Each split-bank is divided into various time slices, which allows us to efficiently sample different regions of our waveforms with the appropriate Nyquist frequency instead of over-sampling the low-frequency regions of the waveform with the sampling frequencies required for the high-frequency regions. The SVD is then performed on each time slice of each split-bank, and we retain only the

---

[1] Split-banks that contain the lowest and highest $\mathcal{M}$ templates in a given χ bin are padded with duplicate templates from within the split-bank in order to keep the clipping uniform between split-banks.
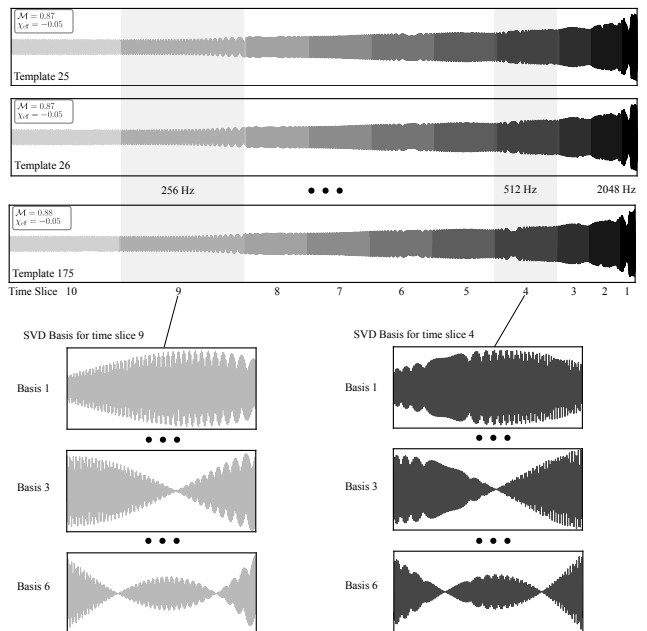

FIG. 6. An example of the LLOID decomposition [22]. In this example, $N_T = 195$ binary inspiral waveforms (390 including the two possible phases) with a chirp mass between 0.87 and 0.88 are first "whitened" by dividing them by a realistic noise amplitude spectral density from Advanced LIGO. The line features in the spectrum account for the amplitude modulation of the waveforms. The waveforms were decomposed into 30 time slices at sample rates ranging between 128 Hz and 2048 Hz (only the last 10 slices are shown). A basis filter set from the waveforms in each time slice was constructed using the SVD [41]. Only 6-10 basis waveforms per slice were needed to reconstruct both phases of the 195 input waveforms to an accuracy of better than 99.9%.

most important basis waveforms returned by the SVD algorithm such that the match between the original template and the one we get from the reconstruction of the retained basis waveforms is more than a decided tolerance [22].

In addition to being used for the LLOID decomposition, split-banks are binned by the lowest chirp mass in each split-bank to construct bins of similar templates. These are referred to as $\bar{\theta}$ bins, and define a binning of the likelihood-ratio detection statistic defined in Sec. III E.

III. EVENT IDENTIFICATION STAGE

Borrowing the language commonly used in particle experiments, the GstLAL-based inspiral pipeline identifies "triggers" from individual interferometer data streams. Triggers which arrive in coincidence are elevated to the "event" classification and ranked by the likelihood-ratio ranking statistic. In this section, we discuss how a list of triggers is generated by the matched-filtering algorithm and how coincidences are identified and ranked as

events.

A. Matched filtering and the LLOID method

As discussed in Sec. II D, groups of templates are partitioned into time slices as part of the LLOID decomposition. Specifically, any split-bank $\mathbf{H}$ can be written as a collection of matrices $\mathbf{H}^s$,

$$\mathbf{H} = \{\mathbf{H}^s\}, \tag{19}$$

where each $\mathbf{H}^s$ contains time-slice s of all $2N_T$ templates in the split-bank,

$$\mathbf{H}^s = \{\hat{h}_i^s(t) : i \in [0, 2N_T - 1]\}. \tag{20}$$

The index s is chosen to be largest at the start of the template waveforms, decreasing until $s = 0$ for the last slice (as seen in Fig. 6). Each slice of the split-bank, $\mathbf{H}^s$, is decomposed via the SVD to provide basis functions u. These basis functions can be used to reconstruct any $\hat{h}_i$ to a predetermined tolerance, i.e.,

$$\hat{h}_i^s(t) \approx \sum_{\nu=0}^{N-1} v_{i\nu}^s \sigma_\nu^s u_\nu^s(t), \tag{21}$$

where $\mathbf{u}^s = \{u_\nu^s(t)\}$ is a matrix comprised of N basis vectors, $\mathbf{v}^s = \{v_{i\nu}^s\}$ is a reconstruction matrix, and $\vec{\sigma} = \{\sigma_\nu^s\}$ is a vector of singular values whose magnitudes are directly proportional to how important a corresponding basis vector is to the reconstruction process [41]. Now instead of evaluating Eq. (3b) $2N_T$ times for each slice of $2N_T$ templates, we can evaluate

$$U_\nu^s(t) = 2 \int_{-\infty}^{\infty} \mathrm{d}\tau \, u_\nu^s(t)\hat{d}^s(t + \tau) \tag{22}$$

$N < 2N_T$ times for each slice, where $\hat{d}^s(t)$ is sampled at the same rate as $u_\nu^s(t)$. The matched filter output time series is calculated for each time slice $\mathbf{H}^s$, then up-sampled via sinc interpolation and added to the output of other time slices (in order of decreasing s) to obtain the output of Eq. (3b). The matched filter output accumulated up through slice s is defined recursively for each template in a given split-bank as

$$x_i^s(t) = \overbrace{\left(H^\uparrow x_i^{s+1}\right)(t)}^{\text{Previous } x_i} + \underbrace{\sum_{\nu=0}^{N-1} v_{i\nu}^s \sigma_\nu^s U_\nu^s(t)}_{\text{Current } x_i}, \tag{23}$$

where $H^\uparrow$ acts on a time series sampled at f_{s+1} and up-samples it to f_s. Recall that the GstLAL-based inspiral pipeline uses two real-valued templates in place of one complex-valued template (Sec. II), thus the computed SNR is the quadrature sum of matched filter outputs

from waveforms which differ only in coalescence phase by $\pi/4$,

$$\rho_j(t) = \sqrt{x_{2j}(t)^2 + x_{2j+1}(t)^2}, \, j \in [0, N_T - 1]. \tag{24}$$

Note that there are half as many SNRs as there are templates, which is a result of using real-valued templates in place of complex-valued templates (Sec. II). Evaluating Eq. (23) saves a factor of 10^4 in computational cost over a direct time domain convolution of the template waveforms.

B. Triggers

The raw SNR time series, typically sampled at 2 kHz, is discretized into "triggers" before being stored. The discretization is done by maximizing the SNR over time in one-second windows and recording the peak if it crosses a predetermined threshold. With the typical threshold of SNR = 4, it is probable to have at least one trigger in every one-second interval for every template for every detector dataset analyzed. Although the number of triggers can easily be hundreds of thousands per second, storing them is a marked improvement over storing the raw SNR time series, which is over three orders of magnitude more voluminous. However, we do not discard the raw SNR time series information immediately, because it is needed for the next stage of the pipeline (Sec. III C). For each trigger, we record the parameters of the template, the trigger time, the SNR, and the coalescence phase. The trigger time is computed via sub-sample interpolation to nanosecond precision; while low SNR triggers suffer from poor timing resolution, high SNR triggers can be resolved to better resolution than that of the sample rate [8, 44, 45]. Triggers are identified in parallel across each template in a given $\bar{\theta}$ bin.

C. Signal-based vetoes

Detector data often contain glitches that are not removed during the data conditioning stage (Sec. II C). Therefore, ranking triggers solely by SNR is not sufficient to separate noise from transient signals. Fortunately, we can exploit consistency checks to improve our ability to discriminate spurious glitches from true gravitational-wave events. Requiring multiple-detector coincidence (Sec. III D) is one powerful check, but here we discuss a separate check on waveform consistency for a single detector's matched-filter output. This waveform consistency check determines how similar the SNR time series of the data is to the SNR time series expected from a real signal.

Under the assumption that the signal in the data exactly matches the matched-filter template up to a constant, it is possible to predict the local matched-filter SNR by computing the template autocorrelation function

and scaling it to the known SNR. However, the known SNR is a result of the matched-filter response from two identical but out of phase templates, thus instead of scaling the autocorrelation function to the SNR, a complex SNR series is constructed from the two matched-filter outputs,

$$z_j(t) = x_{2j}(t) + i x_{2j+1}(t), \ j \in [0, N_T - 1]. \qquad (25)$$

These are compared to the complex autocorrelation function,

$$R_j(t) = \int_{-\infty}^{\infty} df \frac{|\widetilde{h}_{2j}(f)|^2 + |\widetilde{h}_{2j+1}(f)|^2}{S_n(|f|)} e^{2\pi i f t}, \qquad (26)$$

where $t = 0$ is chosen to be the peak time, t_p. By convention, each *real* template is normalized such that its autocorrelation is $\frac{1}{2}$ at the peak time, thus $R_j(0) = 1$. We compute a signal consistency test value, ξ^2, as a function of time given the complex SNR time series $z_j(t)$, a trigger's peak complex SNR $z_j(0)$, and the autocorrelation function time series $R_j(t)$ as

$$\xi_j^2(t) = |z_j(t) - z_j(0) R_j(t)|^2. \qquad (27)$$

If the gravitational wave strain data contain only noise (i.e., $\tilde{d}(f) = \tilde{n}(f)$), then (see Appendix A for derivation)

$$\langle \xi_j^2(t) \rangle = 2 - 2|R_j(t)|^2. \qquad (28)$$

In practice, a value of ξ^2 is computed for each trigger by integrating $\xi^2(t)$ in a window of time around the trigger and normalizing it using Eq. (28). The integral takes the form

$$\xi_j^2 = \frac{\int_{-\delta t}^{\delta t} dt |z_j(t) - z_j(0) R_j(t)|^2}{\int_{-\delta t}^{\delta t} dt (2 - 2|R_j(t)|^2)}, \qquad (29)$$

where δt is a tunable parameter that defines the size of the window around the peak time over which to perform the integration. Typically, δt is calculated in terms of an odd-valued autocorrelation length (ACL), specified as a number of samples such that $\delta t = (\text{ACL} - 1)\Delta t_s / 2$, where $\Delta t_s = f_s^{-1}$ is the sampling time step. A suitable value for ACL was found to be 351 samples when filtering is conducted at a 2048 Hz sample rate, resulting in $\delta t \sim 85.4$ ms; this value was found using by Monte Carlo simulations in real data.

Figure 7 plots the SNR and scaled autocorrelation for a template that recovered a simulated signal in initial LIGO data. Subtracting the measured SNR time series from the predicted series shown in this figure is what is done in (29) on a trigger-by-trigger basis.

We note that ξ^2 differs from the traditional time-frequency χ^2 test in [24], and it is not in fact a χ^2-distributed number in Gaussian noise. However, the statistics of the ξ^2 test are recorded for both noise and simulated signals and can therefore be used in the likelihood-ratio test described in Sec. III E.
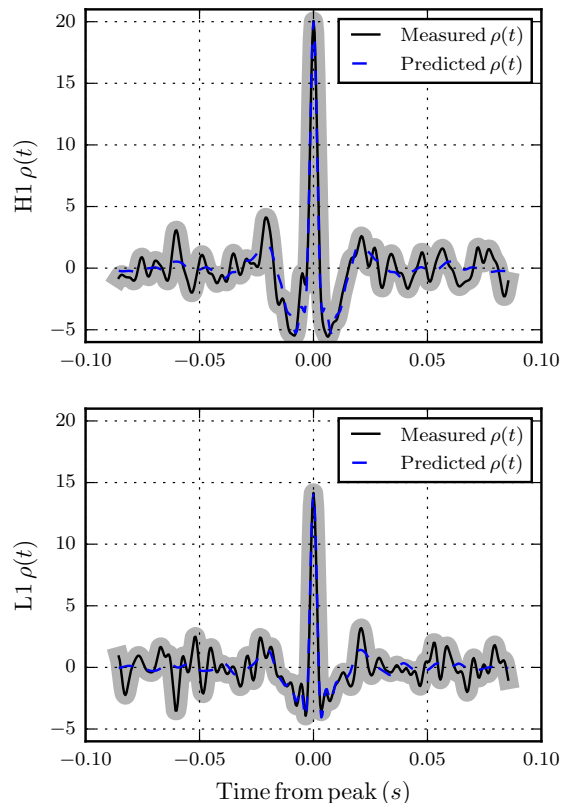

FIG. 7. Ingredients in the auto-correlation-based least-squares test as described in (27). The two panels show the SNR time series near a simulated signal in initial LIGO data (black) along with the predicted SNR computed from the template autocorrelation. Subtracting these two time series and integrating their squared magnitude provides a signal consistency test, ξ^2, at the time of a given trigger that can be used to reject non-stationary noise transients.

D. Coincidence

Demanding that two triggers are found in temporal coincidence between the LIGO sites is a powerful technique to suppress the background of the search. For a single detector trigger, we define the time of an event to coincide with the peak of its SNR time series. Given a trigger in one detector, we check for corresponding triggers in the other detector within an appropriate time window, which takes into account the maximum gravitational-wave travel time between detectors and statistical fluctuations in the measured event time due to detector noise. For the two LIGO detectors, the time window is typically ± 15 ms. We further require that the mass and spin template parameters are the same for the two triggers. This exact match requirement potentially results in a small loss of SNR for real signals, since the loudest trigger in each detector will in general not have the exact

same template parameters due to independent noise in the detectors. However, taking into account such fluctuations requires detailed knowledge of the metric on the signal manifold [46], which may not be easily available. Furthermore, the exact match restriction suppresses the noise and drastically simplifies the pipeline.

E. Event Ranking

Each trigger from each detector has independently computed ρ, ξ^2, and t_p values. After coincidences are formed, it is necessary to rank the coincident events from least likely to be a signal to most likely to be a signal and to assign a significance to each. The GstLAL-based inspiral pipeline uses the likelihood-ratio statistic described in [25] to rank coincident events by their SNR, ξ^2, the instantaneous sensitivity of each detector (expressed as the horizon distance, $\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}$), and the detectors involved in the coincidence (expressed as the set $\{\mathrm{H1}, \mathrm{L1}\}$). For the case where only the Advanced LIGO observatories H1 and L1 are participating, the likelihood ratio of an event found in coincidence is defined as

$$
\begin{aligned}
\mathcal{L}\left(\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}, \{\mathrm{H1}, \mathrm{L1}\}, \rho_{\mathrm{H1}}, \xi_{\mathrm{H1}}^2, \rho_{\mathrm{L1}}, \xi_{\mathrm{L1}}^2, \bar{\theta}\right) &= \mathcal{L}\left(\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}, \{\mathrm{H1}, \mathrm{L1}\}, \rho_{\mathrm{H1}}, \xi_{\mathrm{H1}}^2, \rho_{\mathrm{L1}}, \xi_{\mathrm{L1}}^2 \mid \bar{\theta}\right) \mathcal{L}\left(\bar{\theta}\right) \\
&= \frac{P\left(\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}, \{\mathrm{H1}, \mathrm{L1}\}, \rho_{\mathrm{H1}}, \xi_{\mathrm{H1}}^2, \rho_{\mathrm{L1}}, \xi_{\mathrm{L1}}^2 \mid \bar{\theta}, \mathrm{signal}\right)}{P\left(\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}, \{\mathrm{H1}, \mathrm{L1}\}, \rho_{\mathrm{H1}}, \xi_{\mathrm{H1}}^2, \rho_{\mathrm{L1}}, \xi_{\mathrm{L1}}^2 \mid \bar{\theta}, \mathrm{noise}\right)} \mathcal{L}\left(\bar{\theta}\right), \quad (30)
\end{aligned}
$$

where $\bar{\theta}$ is a label corresponding to the template bank bin being matched-filtered (Sec. II D). The numerator and denominator are factored into products of several terms in [25], assuming that the noise distributions for each interferometer are independent of each other and that the autocorrelation ξ^2 statistics are dominated by detector noise and hence are statistically independent random variables in the case that a signal is present in the data. The computation of each term in the factored numerator and denominator is discussed in detail in [25]; in this paper, we will give only a short summary of the denominator. The denominator is factored such that

$$
P\left(\{D_{\mathrm{H1}}, D_{\mathrm{L1}}\}, \{\mathrm{H1}, \mathrm{L1}\}, \rho_{\mathrm{H1}}, \xi_{\mathrm{H1}}^2, \rho_{\mathrm{L1}}, \xi_{\mathrm{L1}}^2 \mid \bar{\theta}, \mathrm{noise}\right)
$$
$$
\propto \prod_{\mathrm{inst} \in \{\mathrm{H1, L1}\}} P(\rho_{\mathrm{inst}}, \xi_{\mathrm{inst}}^2 \mid \bar{\theta}, \mathrm{noise}). \quad (31)
$$

The detection statistics ρ and ξ^2 from non-coincident triggers are used to populate histograms for each detector, which are then normalized and smoothed by a Gaussian smoothing kernel to approximate $P(\rho_{\mathrm{inst}}, \xi_{\mathrm{inst}}^2 \mid \bar{\theta}, \mathrm{noise})$. Running in the low-latency operation mode requires a burn-in period until the analysis collects enough non-coincident triggers to construct an accurate estimate of the (ρ, ξ^2) PDFs. Neither operation mode tracks time dependence of these PDFs, instead the PDFs are constructed from cumulative histograms. Future work may be add time dependence.

Rather than collecting non-coincident (ρ, ξ^2) statistics from individual templates, we group like templates together to avoid the computational cost and complexity of tracking each template separately. Furthermore, it has been observed that groups of like templates produce

similar PDFs, thus coarse graining the parameter space allows one to approximate these PDFs for collections of templates. Therefore, the $\bar{\theta}$ in the likelihood ratio is a label that identifies a specific template bank bin. Exactly how templates are grouped together into background bins is left as a tuning decision for the user, but typically $\mathcal{O}(1000)$ templates from each detector are grouped together.

An example of the ρ and ξ^2 distributions is shown in Fig. 8. The plots were made from an analysis of one week of data from initial LIGO's sixth science run [47, 48], beginning September 14, 2010, at 23:58:48 UTC. The start and stop times of the analysis were chosen to include the blind injection performed on September 16, 2010, at 06:42:23 UTC, often referred to as the "Big Dog." The warm colormap corresponds to the noise distribution. The cool colormap corresponds to a distribution generated by adding the coincident triggers to the single detector triggers before smoothing and normalizing. The cool-colormap distribution was then masked to only show regions which deviate from the background estimate, and the location of the Big Dog parameters is indicated by a black cross. The only region which differs from the estimated background is the region where the Big Dog resides.

Examples of two other likelihood-ratio components from the Big Dog analysis are shown in Fig. 9. The top plot shows the joint SNR PDF, which is used in the numerator of the likelihood ratio to enforce amplitude consistency [25]; the bottom plot shows the signal hypothesis model of the (ρ, ξ^2) plane. The semi-analytic models used to generate these plots are described in [25].
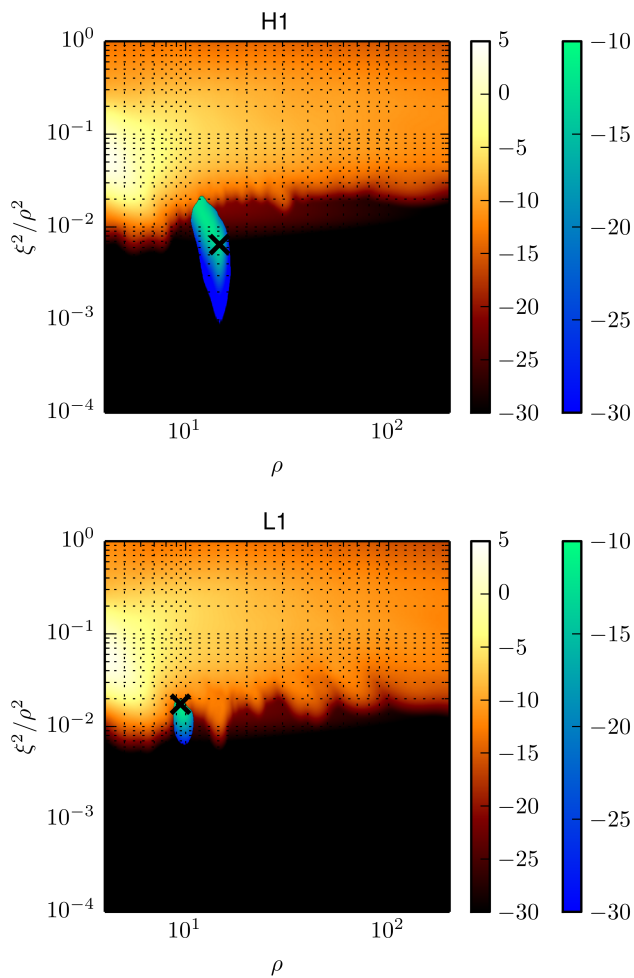
FIG. 8. PDFs used in the likelihood ratio calculation, generated by histogramming, smoothing, and normalizing triggers. The plots shown are from an analysis of S6 data beginning at September 14, 2010, at 23:58:48 UTC and ending at September 21, 2010, at 23:58:48 UTC, which includes the blind injection known as the "Big Dog." The warm colormap corresponds to the marginalized background model generated from non-coincident triggers only; the cool-colormap region was computed by adding coincident triggers to the histograms before smoothing and normalizing and then masking the regions consistent with the warm-colormap model. The location of the Big Dog in the (ρ, ξ^2) plane is marked with a black X.
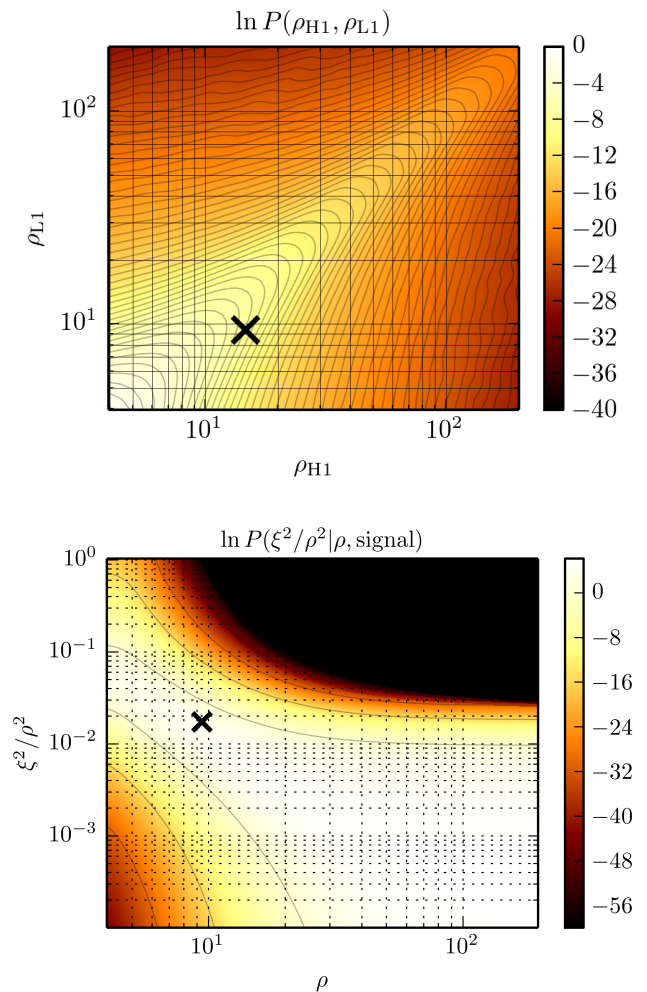

FIG. 9. Instances of two of the distributions included in the calculation of the likelihood-ratio numerator. Top: The joint SNR PDF used to enforce amplitude consistency across observatories. Bottom: The (ρ, ξ^2) signal distribution used in the numerator of the likelihood ratio. These plots were generated from an analysis of S6 data beginning at September 14, 2010, at 23:58:48 UTC and ending at September 21, 2010, at 23:58:48 UTC, which includes the blind injection known as the "Big Dog." The location of the Big Dog parameters are marked with a black X.

likelihood-ratio event in each $\bar{\theta}$ bin within a ± 1-second window.

F. Event Clustering

Signals can produce several high-likelihood events at the same time in different templates; we wish to ensure that we only consider the most likely event associated with a signal. In the offline analysis, we use a clustering algorithm that picks out the maximum likelihood-ratio event globally across the input template bank within a ± 4-second window. The online analysis does not do global event clustering in order to not incur any latency. Instead, the online analysis keeps the maximum

IV. EVENT PROCESSING AND SENSITIVITY ESTIMATION

The result of the pipeline components described in Sec. III is a list of events ranked from most to least likely to be a gravitational-wave signal. In this section, we discuss how the significance is estimated, the procedure in the case that a sufficiently significant candidate is identified, and how simulated waveforms are used to infer

knowledge of analysis sensitivity to gravitational waves.

A. Event Significance Estimation

Most events found in coincidence are noise. Therefore, it is important to ascertain the probability that noise would produce an event with a log likelihood ratio at least as large as the one under consideration, which is called the false-alarm probability (FAP). The FAP has conventionally been evaluated by performing *time slides*, where a set of time shifts that are much larger than the gravitational-wave travel time (tens of milliseconds) between gravitational-wave detectors is introduced into one or more datasets and the coincidence and event-ranking procedure is repeated in the same way as it is done without the time shifts. However, the bookkeeping of time slides is more complicated [49, 50] than is desirable for a low-latency search mode.

Instead of performing time slides, the GstLAL-based inspiral pipeline calculates the FAP directly from probability density functions of the parameters used in the likelihood-ratio calculation. These probability density functions are obtained from triggers that are not found in coincidence and thus are taken to be noise-like [26]. The procedure to calculate $P\left(\ln \mathcal{L} \mid \bar{\theta}, \text{noise}\right)$ for each $\bar{\theta}$ bin is discussed in detail in [26], and the procedure to marginalize over $\bar{\theta}$ to calculate $P\left(\ln \mathcal{L} \mid \text{noise}\right)$ is discussed in [25, 26]. The marginalized PDF is used to compute the complementary cumulative distribution function (CCDF),

$$P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid \text{noise}\right) = \int_{\ln \mathcal{L}^*}^{\infty} \mathrm{d} \ln \mathcal{L} \, P\left(\ln \mathcal{L} \mid \text{noise}\right). \tag{32}$$

The FAP that we seek describes the probability that a population of M independent coincident noise-like events, which are the output of the search, contains $k \geq 1$ events with a log-likelihood ratio greater than or equal to some threshold $\ln \mathcal{L}^*$. This probability can be thought of as a trials factor that corrects the single event FAP and can be written as the complement of the binomial distribution [26],

$$\begin{aligned} \text{FAP} &= P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid k \geq 1, \text{noise}, M\right) \\ &= 1 - \left(1 - P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid k \geq 1, \text{noise}, 1\right)\right)^M, \end{aligned} \tag{33}$$

where $P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid k \geq 1, \text{noise}, 1\right)$ is the probability of measuring a nonzero number of events, k, with a likelihood ratio greater than or equal to a threshold $\ln \mathcal{L}^*$ from the noise after measuring only 1 event.

In order to calculate an event's significance during an experiment of undetermined length, such as the low-latency processing of data during a science run, it is convenient to express the significance in terms of false-alarm rate (FAR). Since time does not explicitly enter the FAP distribution, we assume that event generation follows Poisson statistics and that the single event FAP is the complement of the Poisson distribution with mean event rate $P\left(\ln \mathcal{L}^* \geq \ln \mathcal{L} \mid \text{noise}\right)$,

$$\begin{aligned} P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid k \geq 1, \text{noise}, 1\right) = \\ 1 - \exp\left[-P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid \text{noise}\right)\right]. \end{aligned} \tag{34}$$

The Poisson assumption asserts the ensemble average of the background is the mean event rate, in units of events/analysis. This admits a simple calculation of the FAR, by dividing the estimated mean event rate by the live-time of the experiment, which has units seconds/analysis:

$$\text{FAR} = \frac{P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid \text{noise}\right)}{T}. \tag{35}$$

The time used in the calculation of the FAR is the total elapsed observing time regardless of instrument state for the low-latency configuration and the total time where at least two detectors are operating for the offline analysis operation. The offline configuration definition is typically what has been used in the field; however, the online definition leads to predictable false alarm rates for sharing low-latency events to external observing partners. An alternative derivation of the mean rate, $\lambda\left(\ln \mathcal{L}\right)$, is given in Appendix A of [26]. Readers should be careful to note that the $P\left(\mathcal{L} \geq \mathcal{L}^* \mid \text{noise}\right)$ in [26] is what we have referred to as $P\left(\ln \mathcal{L} \geq \ln \mathcal{L}^* \mid k \geq 1, \text{noise}, 1\right)$.

The procedure to estimate the background distribution described thus far does not account for the clustering described in Sec. III F. Events with low $\ln \mathcal{L}$ are more common than those with high $\ln \mathcal{L}$, thus the clustering process removes low $\ln \mathcal{L}$ events preferentially. The normalization of the background model is determined by the observed events above a log-likelihood ratio threshold chosen to be safely out of the region affected by clustering. This is acceptable because low $\ln \mathcal{L}$ events are, by construction, the least likely to contain a signal. Consequently, we only consider events well above this threshold as viable candidates. Work is currently underway to create a background model that accounts for clustering.

A plot of the significance results from the Big Dog run discussed in Sec. III E is shown in Fig. 10. The Big Dog was found with a FAP of 5.4×10^{-9} (5.8σ), which corresponded to a FAR of 1.1×10^{-14} Hz (1 per $\sim 2.7 \times 10^{6}$ years); Table I lists the recovered parameters of the Big Dog.

B. Generating Alerts

When operating in a low-latency analysis configuration, one of the primary goals of the GstLAL-based inspiral pipeline is to identify candidate events and upload them to the Gravitational-wave Candidate Event Database (GraceDB [51]) as quickly as possible in order to issue alerts to observing partners [11].
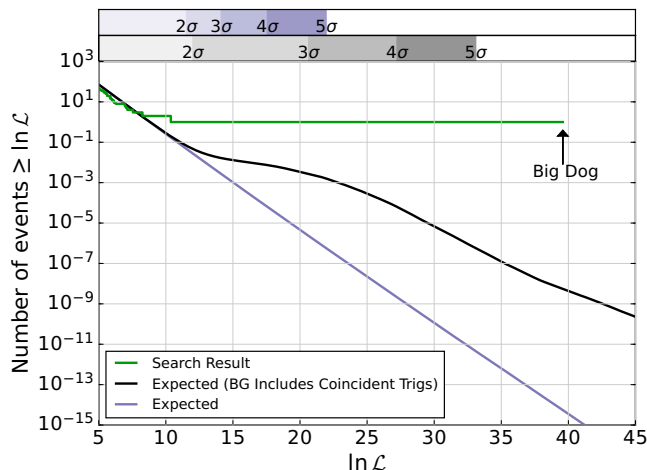
FIG. 10. Number of observed events as a function of log likelihood ratio in an analysis of S6 data beginning on September 14, 2010, at 23:58:48 UTC and ending at September 21, 2010, at 23:58:48 UTC. The Big Dog injection, found with a false alarm probability of 5.4×10^{-9} (5.8σ), is marked on the observed distribution (green). The black line represents the predicted number of events when observed events are included in the background model, while the blue line is the predicted number when the observed events are not included in the background model.

FAP	FAR (Hz)	$\log \mathcal{L}$	ρ_H	ξ_H^2	ρ_L	ξ_L^2	$\mathcal{M}$ (M$_\odot$)
5.8σ	1.1×10^{-14}	39.6	14.7	1.4	9.4	1.5	4.7

TABLE I. The result of the analysis of S6 data beginning on September 14, 2010, at 23:58:48 UTC and ending on September 21, 2010, at 23:58:48 UTC. Only the parameters found for the recovered Big Dog injection are shown. The Big Dog was the most significant event found in this analysis period.

Events that pass a given FAR threshold are identified within ~ 1 minute of the gravitational-wave signals arriving at Earth. The basic parameters of the event are transmitted to GraceDB, including the GPS time of the event, the SNR and ξ^2 values for the triggers in each detector, and the parameters of the best-fit template (for example, mass and spin values, the significance estimate, etc.). Furthermore, the instantaneous estimate of the PSD is uploaded to GraceDB, as well as the histogram data used in computing the FAP.

An event upload automatically initiates several automated and human followup activities to aid rapid communication with observing partners [52]. First, a rapid sky localization routine known as BAYESTAR [8, 45, 53] uses the event information and the PSD to estimate the event's sky position within minutes. At the same time, deeper parameter estimation analysis begins in order to provide updated position reconstruction, as well as the full posterior probability distributions of the binary parameters [54], on a timescale that ranges from hours to days.

In addition to parameter estimation, data-quality information is also mined to provide rapid feedback to analysts. Time-frequency spectrograms are automatically generated to indicate the stationarity of noise near an event [55]. Furthermore, low-latency mining of LIGO's auxiliary channels provide additional information about the state of the detector and environment when an alert is first generated [15, 56, 57]

The suitability of the low-latency pipeline for generating data for external alerts has been studied extensively in [8, 53]

C. Software injections

Simulated gravitational waveforms known as "software injections" are used to assess the pipeline response to real gravitational-wave signals. The LIGO strain data is duplicated and simulated compact binary waveforms are digitally added to the duplicated data streams. In low latency, the new data with software injections added are broadcast to the LIGO data grid in parallel to the normal dataset so that a simultaneous run can measure the instantaneous sensitivity of the low-latency analysis to compact binary sources. In the offline mode, strain data is read from disk, software injections are added, and the new data is written back to disk before the offline inspiral pipeline processes the data.

Injections are considered 'found' if a coincident event with the correct template parameters is found with a FAR ≤ 30 d at the time of the injection, and 'missed' otherwise. The volume of space the pipeline is sensitive to, V, is approximated as a sphere and computed via

$$V = 4\pi \int_0^\infty \mathrm{d}r \epsilon(r) r^2, \qquad (36)$$

where $\epsilon(r)$ is an efficiency parameter given by the ratio of found to total injections modeled to be a distance r away. We define our estimated range, the average furthest distance a signal can originate from and still be detected, as

$$R = \left(\frac{3V}{4\pi} \right)^{1/3}. \qquad (37)$$

It is important to note the range depends on parameters of the compact binary system. For example, the range for a $1.4 - 1.4$M$_\odot$ binary neutron star system will be different than that of a $10 - 10$M$_\odot$ binary black hole system, thus different injection sets must be used to determine the pipeline's sensitivity to different regions of the compact binary parameter space. Eq. (37) is compared to the analytically computed SenseMon range, an estimate of pipeline sensitivity calculated from the PSD [58]. Comparing the sensitivity estimated from the PSD to the sensitivity estimated from injections provides additional confidence in the sensitivity estimates.

Typically, injections are added at a much higher rate than the expected gravitational wave signal rate. However, their cadence is chosen such that they do not bias the PSD estimate described in Sec. II B. In practice, injections are typically added about once per minute so that it is possible to evaluate the average response to certain signal types over the entire experiment duration.

V. CONCLUSION

The GstLAL-based inspiral pipeline is a new stream-based pipeline that allows for time-domain compact binary searches capable of identifying and uploading candidate gravitational-wave signals within seconds. This provides rapid feedback to the gravitational-wave detector control rooms and enables prompt event alerts for electromagnetic followup by observing partners. The analysis techniques were designed for second- and third-generation gravitational-wave detectors and have been demonstrated to be applicable even to the computationally challenging case of the future Einstein Telescope [59].

GstLAL and all related software is available for public use and licensed under the GPL [13].

VI. ACKNOWLEDGEMENTS

Appendix A: Derivation of $\langle \xi^2(t) \rangle_{\text{noise}}$

Expanding Eq. (27) and taking the ensemble average, we find

$$
\begin{aligned}
\langle \xi_j^2(t) \rangle &= \langle |z_j(t) - z_j(0) R_j(t)|^2 \rangle, \\
&= \langle |z_j(t)|^2 \rangle - 2\text{Re} \left[\langle z_j^*(t) z_j(0) \rangle R_j(t) \right] \\
&\quad + \langle |z_j(0)|^2 \rangle |R_j(t)|^2.
\end{aligned} \tag{A1}
$$

Starting with Eq. (3a),

$$
\begin{aligned}
\langle |z_j(t)|^2 \rangle &= \left\langle \left| 2 \int_{-\infty}^{\infty} df \frac{\widetilde{n}(f)(\widetilde{h}_{2j}^*(f) + i\widetilde{h}_{2j+1}^*(f))}{S_n(f)} e^{2\pi i t f} \right|^2 \right\rangle, \\
&= 4 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} df_1 df_2 \Bigg(\langle \widetilde{n}(f_1) \widetilde{n}(f_2) \rangle e^{2\pi i t (f_1 - f_2)} \\
&\quad \frac{(\widetilde{h}_{2j}^*(f_1) + i\widetilde{h}_{2j+1}^*(f))(\widetilde{h}_{2j}(f_2) - i\widetilde{h}_{2j+1}(f_2))}{S_n(|f_1|) S_n(|f_2|)} \Bigg), \\
&= 2 \int_{-\infty}^{\infty} df \frac{|\widetilde{h}_{2j}(f)|^2 + |\widetilde{h}_{2j+1}(f)|^2}{S_n(|f|)}, \\
\langle |z_j(t)|^2 \rangle &= \langle |z_j(0)|^2 \rangle = 2. \tag{A2}
\end{aligned}
$$

where Eq. (7) was used in the last step. Computing $\langle z_j^*(t) z_j(0) \rangle$ follows the same steps, except the $z_j(0)$ term does have a complex exponential to cancel the complex exponential accompanying $z_j^*(t)$, thus

$$
\langle z_j^*(t) z_j(0) \rangle = 2 R_j^*(t), \tag{A3}
$$

$$
\langle \xi_j^2(t) \rangle = 2 - 2|R_j(t)|^2 \tag{A4}
$$

[1] B. Abbott, R. Abbott, T. Abbott, M. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari, *et al.*, Physical Review Letters **116**, 061102 (2016).

[2] J. Aasi, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari, *et al.*, Classical and quantum gravity **32**, 074001 (2015).

[3] F. Acernese, M. Agathos, K. Agatsuma, D. Aisa, N. Allemandou, A. Allocca, J. Amarni, P. Astone, G. Balestri, G. Ballardin, *et al.*, Classical and Quantum Gravity **32**, 024001 (2015).

[4] Y. Aso, Y. Michimura, K. Somiya, M. Ando, O. Miyakawa, T. Sekiguchi, D. Tatsumi, and H. Yamamoto, Physical Review D **88**, 043007 (2013).

[5] B. Iyer *et al.*, "LIGO-India, Proposal of the Consortium for Indian Initiative in Gravita tional-wave Observations (IndIGO)," (2011), LIGO-DCC-M1100296.

[6] LIGO RATES PAPER, TO BE ADDED BEFORE SUBMISSION, In review.

[7] J. Abadie, B. Abbott, R. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, R. Adhikari, P. Ajith, B. Allen, *et al.*, Classical and Quantum Gravity **27**, 173001 (2010).

[8] L. P. Singer, L. R. Price, B. Farr, A. L. Urban, C. Pankow, S. Vitale, J. Veitch, W. M. Farr, C. Hanna, K. Cannon, *et al.*, The Astrophysical Journal **795**, 105 (2014).

[9] B. P. Abbott *et al.* (LIGO Scientific, Virgo), (2016), arXiv:1602.08492 [astro-ph.HE].

[10] S. Adrián-Martínez *et al.* (ANTARES, IceCube, LIGO Scientific, Virgo), (2016), arXiv:1602.05411 [astro-ph.HE].

[11] "Identification and follow up of electromagnetic counterparts of gravitational wave candidate events," `http://www.ligo.org/scientists/GWEMalerts.php`, accessed: 2016-01-15.

[12] B. P. Abbott *et al.* (LIGO Scientific, Virgo), (2016), arXiv:1602.03844 [gr-qc].

[13] "Gstlal," `https://www.lsc-group.phys.uwm.edu/daswg/projects/gstlal.html` (), accessed: 2015-07-01.

[14] "Gstreamer," `https://gstreamer.freedesktop.org` ().

[15] "Lalsuite," `https://www.lsc-group.phys.uwm.edu/daswg/projects/lalsuite.html`, accessed: 2015-07-01.

[16] J. Abadie, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, R. Adhikari,

C. Affeldt, *et al.*, Astronomy & Astrophysics **541**, A155 (2012).

[17] S. Klimenko *et al.*, Classical and Quantum Gravity **25**, 114029 (2008).

[18] R. Lynch, S. Vitale, R. Essick, E. Katsavounidis, and F. Robinet, (2015), arXiv:1511.05955 [gr-qc].

[19] B. P. Abbott *et al.* (Virgo, LIGO Scientific), (2016), arXiv:1602.03843 [gr-qc].

[20] D. Buskulic, L. S. Collaboration, V. Collaboration, *et al.*, Classical and Quantum Gravity **27**, 194013 (2010).

[21] S. Babak, R. Biswas, P. Brady, D. A. Brown, K. Cannon, C. D. Capano, J. H. Clayton, T. Cokelaer, J. D. Creighton, T. Dent, *et al.*, Physical Review D **87**, 024033 (2013).

[22] K. Cannon, R. Cariou, A. Chapman, M. Crispin-Ortuzar, N. Fotopoulos, M. Frei, C. Hanna, E. Kara, D. Keppel, L. Liao, *et al.*, The Astrophysical Journal **748**, 136 (2012).

[23] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. Creighton, Physical Review D **85**, 122006 (2012).

[24] B. Allen, Physical Review D **71**, 062001 (2005).

[25] K. Cannon, C. Hanna, and J. Peoples, arXiv preprint arXiv:1504.04632 (2015).

[26] K. Cannon, C. Hanna, and D. Keppel, Physical Review D **88**, 024025 (2013).

[27] S. Anderson *et al.*, "Specification of a Common Data Frame Format for Interferometric Gravitational Wave Detectors," (2009), LIGO-DCC-T970130.

[28] S. Droz, D. J. Knapp, E. Poisson, and B. J. Owen, Physical Review D **59**, 124016 (1999).

[29] J. Slutsky, L. Blackburn, D. Brown, L. Cadonati, J. Cain, M. Cavaglia, S. Chatterji, N. Christensen, M. Coughlin, S. Desai, *et al.*, Classical and Quantum Gravity **27**, 165023 (2010).

[30] N. Christensen, L. S. Collaboration, V. Collaboration, *et al.*, Classical and Quantum Gravity **27**, 194010 (2010).

[31] B. J. Owen, Physical Review D **53**, 6749 (1996).

[32] B. J. Owen and B. Sathyaprakash, Physical Review D **60**, 022002 (1999).

[33] T. A. Apostolatos, Physical Review D **52**, 605 (1995).

[34] T. Cokelaer, Phys. Rev. **D76**, 102004 (2007), arXiv:0706.4437 [gr-qc].

[35] B. Abbott, R. Abbott, R. Adhikari, J. Agresti, P. Ajith, B. Allen, R. Amin, S. Anderson, W. Anderson, M. Arain, *et al.*, Physical Review D **78**, 042002 (2008).

[36] I. W. Harry, A. H. Nitz, D. A. Brown, A. P. Lundgren, E. Ochsner, and D. Keppel, Physical Review D **89**, 024010 (2014).

[37] S. Babak, Classical and Quantum Gravity **25**, 195011 (2008).

[38] I. W. Harry, B. Allen, and B. Sathyaprakash, Physical Review D **80**, 104014 (2009).

[39] G. M. Manca and M. Vallisneri, Physical Review D **81**, 024004 (2010).

[40] S. Privitera, S. R. Mohapatra, P. Ajith, K. Cannon, N. Fotopoulos, M. A. Frei, C. Hanna, A. J. Weinstein, and J. T. Whelan, Physical Review D **89**, 024003 (2014).

[41] K. Cannon, A. Chapman, C. Hanna, D. Keppel, A. C. Searle, and A. J. Weinstein, Physical Review D **82**, 044025 (2010).

[42] K. Cannon, C. Hanna, and D. Keppel, Physical Review D **84**, 084003 (2011).

[43] K. Cannon, C. Hanna, and D. Keppel, Physical Review D **85**, 081504 (2012).

[44] S. Fairhurst, New Journal of Physics **11**, 123006 (2009).

[45] L. P. Singer and L. R. Price, Physical Review D **93**, 024013 (2016).

[46] C. Robinson, B. Sathyaprakash, and A. S. Sengupta, Physical Review D **78**, 062002 (2008).

[47] J. Abadie, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, R. Adhikari, C. Affeldt, *et al.*, Physical Review D **85**, 082002 (2012).

[48] J. Abadie, B. Abbott, R. Abbott, T. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, R. Adhikari, C. Affeldt, *et al.*, The Astrophysical Journal **760**, 12 (2012).

[49] D. G. Keppel, *Signatures and dynamics of compact binary coalescences and a search in LIGOs S5 data*, Ph.D. thesis, California Institute of Technology (2009).

[50] C. D. Capano, *Searching for gravitational waves from compact binary coalescence using LIGO and virgo data*, Ph.D. thesis, Syracuse University (2011).

[51] "Gravitational wave candidate event database," https://www.lsc-group.phys.uwm.edu/daswg/projects/gracedb.html, accessed: 2016-01-15.

[52] https://www.lsc-group.phys.uwm.edu/daswg/projects/lvalert.html, accessed: 2016-01-15.

[53] C. P. Berry, I. Mandel, H. Middleton, L. P. Singer, A. L. Urban, A. Vecchio, S. Vitale, K. Cannon, B. Farr, W. M. Farr, *et al.*, arXiv preprint arXiv:1411.6934 (2014).

[54] J. Veitch, V. Raymond, B. Farr, W. Farr, P. Graff, S. Vitale, B. Aylott, K. Blackburn, N. Christensen, M. Coughlin, *et al.*, Physical Review D **91**, 042003 (2015).

[55] "Ligo data viewer," https://www.lsc-group.phys.uwm.edu/daswg/projects/ligodv.html, accessed: 2016-01-15.

[56] R. Biswas, L. Blackburn, J. Cao, R. Essick, K. A. Hodge, E. Katsavounidis, K. Kim, Y.-M. Kim, E.-O. Le Bigot, C.-H. Lee, *et al.*, Physical Review D **88**, 062003 (2013).

[57] R. Essick, L. Blackburn, and E. Katsavounidis, Classical and Quantum Gravity **30**, 155010 (2013).

[58] J. Abadie, B. Abbott, R. Abbott, M. Abernathy, T. Accadia, F. Acernese, C. Adams, R. Adhikari, P. Ajith, B. Allen, *et al.*, arXiv preprint arXiv:1003.2481 (2010).

[59] D. Meacher, K. Cannon, C. Hanna, T. Regimbau, and B. Sathyaprakash, Physical Review D **93**, 024018 (2016).