

# Hierarchical search for compact binary coalescences in the Advanced LIGO's first two observing runs

Kanchan Soni\*

*Inter-University Centre for Astronomy and Astrophysics, Pune 411007, India*

Bhooshan Uday Gadre†

*Max Planck Institute for Gravitational Physics (Albert Einstein Institute), D-14476 Potsdam, Germany*

Sanjit Mitra‡

*Inter-University Centre for Astronomy and Astrophysics, Pune 411007, India*

Sanjeev Dhurandhar§

*Inter-University Centre for Astronomy and Astrophysics, Pune 411007, India*

Detection of many compact binary coalescences (CBCs) is one of the primary goals of the present and future ground-based gravitational-wave (GW) detectors. While increasing the detectors' sensitivities will be crucial in achieving this, efficient data analysis strategies can play a vital role. With given computational power in hand, efficient data analysis techniques can expand the size and dimensionality of the parameter space to search for a variety of GW sources. Matched filtering based analyses that depend on modeled signals to produce adequate signal-to-noise ratios for signal detection may miss them if the parameter space is too restrained. Specifically, the CBC search is currently limited to non-precessing binaries only, where the spins of the components are either aligned or anti-aligned to the orbital angular momentum. A hierarchical search for CBCs is thus well motivated. The first stage of this search is performed by matched filtering coarsely sampled data with a coarse template bank to look for candidate events. These candidates are then followed up for a finer search around the vicinity of an event's parameter space found in the first stage. Performing such a search leads to enormous savings in the computational cost without losing sensitivity. Here we report the first successful implementation of the hierarchical search as a PyCBC-based production pipeline to perform a complete analysis of LIGO observing runs. With this, we analyze Advanced LIGO's first and second observing run data. We recover all the events detected by the PyCBC (flat) search in the first GW catalog, GWTC-1, published by the LIGO-Virgo collaboration, with nearly the same significance using a scaled background. In the analysis, we get an impressive factor of 20 speed-up in computation compared to the flat search. With a standard injection study, we show that the sensitivity of the hierarchical search remains comparable to the flat search within the error bars.

## I. INTRODUCTION

Gravitational-wave (GW) astronomy began with the detection of GW signal from a binary black hole (BBH) merger, GW150914 [1], using the Advanced LIGO [2, 3] observatories. With the latest advancements in the sensitivities of detectors and search techniques like cWB [4], GstLAL [5], PyCBC [6], LIGO-Virgo (LV) collaboration detected GW signals from ten BBHs and one binary neutron star (BNS) coalescence in the first two, O1 and O2, observing runs [7]. During this period, several independent searches [8–10] over publicly available data detected a few additional BBH events. A paradigm shift in the number of detections occurred with the improvement in the sensitivities of Advanced LIGO [11] and Advanced Virgo [12] detectors in the third observing run.

At present, the LV collaboration has detected 39 GW events [13] including GW190425 [14], the second BNS event, GW190412 [15] and GW190814 [16] the first two highly asymmetric CBCs that emit a significant amount of gravitational radiation beyond the quadrupole moment, and GW190521 [17], the first binary merging to form an intermediate-mass black hole.

Matched filtering [18], a primary and most sensitive algorithm, is used to detect signals that can be well modeled. Since the GW signals from merging binaries in circular orbits can be modeled using their intrinsic<sup>1</sup> and extrinsic<sup>2</sup> parameters, the matched filtering method is employed for their detection. The method involves correlating an interferometer's output, time-series data, with the modeled waveforms (*templates*) for each detector in

\* kanchasoni@iucaa.in

† bhooshan.gadre@aei.mpg.de

‡ sanjit@iucaa.in

§ sanjeev@iucaa.in

<sup>1</sup> Component masses ( $m_1, m_2$ ) and individual spins ( $\vec{s}_1, \vec{s}_2$ ) vectors of the coalescing binary.

<sup>2</sup> Sky location ( $\zeta, \phi$ ), luminosity distance ( $d_L$ ), orbital inclination ( $\iota$ ), polarization angle ( $\psi$ ), and time and phase of coalescence ( $t_c, \phi_c$ ) of the coalescing binary with respect to the detector.

the network. If a GW signal is present in the detector’s output, the correlation results in a peak (*trigger*) in the signal-to-noise ratio (SNR) corresponding to the best matching template. Since the prior knowledge of the source parameters, like its component masses, spins, and location in the sky, remains unknown to the observers, the search is required to be performed over a wide range of several source parameters using a “bank of templates”. The templates in the bank are closely placed to ensure that the search does not miss any signal. Since the data contain non-Gaussian noise, a coincidence search over the time of arrival, phase, and other source parameters is performed between different detectors to eliminate the false triggers. The resultant candidates obtained are then assigned significance based on the noise background estimated by counting coincident triggers for unphysical time delays.

The above procedure for detecting GW signals from CBCs is followed by the search pipelines like GstLAL [5, 19, 20], MBTA [21, 22], PyCBC [6, 8, 9, 23], and SPIIR [24]. These pipelines perform a *one-step search*<sup>3</sup> for the non-precessing coalescing binaries in circular orbits.

One of the challenges that template-based search pipelines face is the high computational cost of matched filtering, typically a year’s worth of data over  $\sim 10^5$  templates. Since this process, especially in the PyCBC (or flat) search, involves Fast Fourier Transform (FFT) of the product of a uniformly sampled time-series data and a template, the number of floating-point operations scales as  $N \log_2 N$ , where  $N$  is the number of data points. These operations repeat over  $\sim 10^5$  templates, even in the restricted parameter space of non-precessing binaries, which amounts to sizable computational cost. The cost further increases when a search is envisaged for precessing binaries where the orbital plane precesses due to the misalignment of spins with the orbital angular momentum. In such cases, the number of templates and the matched filtering operations increases at least tenfold [25], making the search unfeasible with the current capabilities.

With the advancements in current detectors and upcoming new detectors, e.g., KAGRA [26] and LIGO-India [27], the CBC detection rate is bound to increase, and finer details of the detected sources would be sought to unravel their exact dynamics, their formation, and evolution scenarios. However, this would significantly increase the volume of the search parameter space. The increment in volume would happen in two ways; first, the number of parameters (dimension of the parameter space) of different CBC sources would increase, and secondly, their ranges may increase. Nevertheless, a comprehensive search of this type is important because one

would like to capture the non-trivial dynamics of interesting astrophysical sources. We, therefore, argue that to facilitate this quest, a matched filter-based search pipeline needs speeding up by orders of magnitude.

One way to speed up the search is by performing matched filtering *hierarchically using multiple banks of varying densities*. The idea of performing matched filtering in hierarchical steps was formally introduced in Mohanty and Dhurandhar [28], where hierarchy was performed over the chirp mass of binaries using Newtonian waveforms. This work was later extended to the post-Newtonian waveforms [29], where hierarchy was performed over the component masses of a binary system. A further improvement was realized by reducing the sampling rate in the first stage of the hierarchy. In the recent work of Gadre *et al.* [30] hierarchy was performed over all the three intrinsic parameters, including the effective spin of the binary. This algorithm had used multi-detector coincidence analysis and had provided an order of magnitude speed-up compared to the flat analysis.

In this paper, we revisit the hierarchical search formulated in Gadre *et al.* [30], and for the *first time, implement it as a working PyCBC-based pipeline* to analyze the data from an entire LIGO observing run. We describe an efficient two-stage hierarchical search pipeline to search for GW signals from CBCs in the two detectors. This pipeline improves the search sensitivity by incorporating better detection statistics for the single-detector and coincident triggers, as used by the flat analysis in Abbott *et al.* [13]. For this work, we construct two template banks—*coarse* and *neighbourhood (nbhd)* bank, to target GW signals from non-precessing coalescing BBH, BNS, and neutron star-black holes (NSBH) sources that have quasi-circular orbits. Using the former bank in the first stage and a dynamical subset of the latter bank in the second stage of the hierarchical search, we test the potency of the pipeline by applying it to the data from the first two observing runs of Advanced LIGO. Our pipeline recovers all the GW events observed by the flat search from the first gravitational wave catalog (GWTC-1) [7].

In our work, we assign the significance to the detected events using a *scaled background* [30], constructed by scaling the background obtained in the first stage by time-sliding the filtered output across detectors using the coarse bank. To justify the accuracy of this background, we perform simulations that involve the injection of the GW signals into the data and compare their recoveries with the hierarchical and flat search separately. Furthermore, we compare the sensitivities of the two searches through ‘volume-time’ (*VT*) ratio curves. Lastly, we conclude our findings from the injection study by discussing the two searches’ computational cost.

The paper is organised as follows:

- In Section II we state the prerequisites and describe the search methodology for the hierarchical search. The section segregates into sub-parts. In II A, we describe the generation of template banks. Section II B elaborates on the matched filtering process

---

<sup>3</sup> Search involving match filtering data, sampled at a fixed rate using a bank of templates

and the selection criteria for the generated triggers in two stages. The strategy to collect coincident triggers is described in II C. The final step in the pipeline is to assign significance to the coincident candidates. We describe this process in II D.

- In Section III, we implement the hierarchical search pipeline over the first two observing runs of Advanced LIGO and present our findings.
- In Section IV, we compare the sensitivities of the hierarchical search with flat search. We also discuss the computational advantages of the former search with the latter.
- In Section V, we summarize our findings and point out the directions of future research.

## II. PREREQUISITES AND SEARCH METHODOLOGY

The idea of the hierarchical search is straightforward; the flat search algorithm is divided into two stages, *Stage-1* and *Stage-2*, such that the number of matched filter operations reduce successively. Stage-1 ensures matched filtering of the data sampled at the lower sampling rate (512 Hz) using a sparsely sampled bank called the coarse bank. Having fewer templates in a coarse bank significantly reduces the contribution to the computational cost of matched filtering. Further reduction in computation is achieved by sampling the data at a lower sampling rate than the rate at the Nyquist frequency. The coarse bank can reduce the SNRs for a good fraction of events because of the sparsely placed templates. To compensate for the loss in SNRs, we identify triggers in each detector above *coarse thresholds*, set at lower values than those used in the flat search. We then perform a coincidence test on these identified triggers, using an optimal detection statistics and obtain the zero-lag (or foreground) candidates. The foreground candidates are then followed up in stage-2 to ascertain whether they are signals or false alarms.

In stage-2, we again perform matched filtering over the data segments that contain the followed-up foreground candidates. These data segments are sampled at a flat search sampling rate (2048 Hz) and filtered using a dynamic union of nbhds of mismatch extending up to 0.75 around the followed-up stage-1 trigger templates. We refer to this union of nbhds triggered from stage-1 as the *stage-2 bank* in this paper. The triggers generated for each detector in this stage are identified above *fine thresholds*, equal to the thresholds set for SNRs in the flat search. These triggers are then subjected to the coincidence test before generating the final list of foreground candidates.

Our two-stage hierarchical search pipeline is described through the flowchart in Fig. 1.

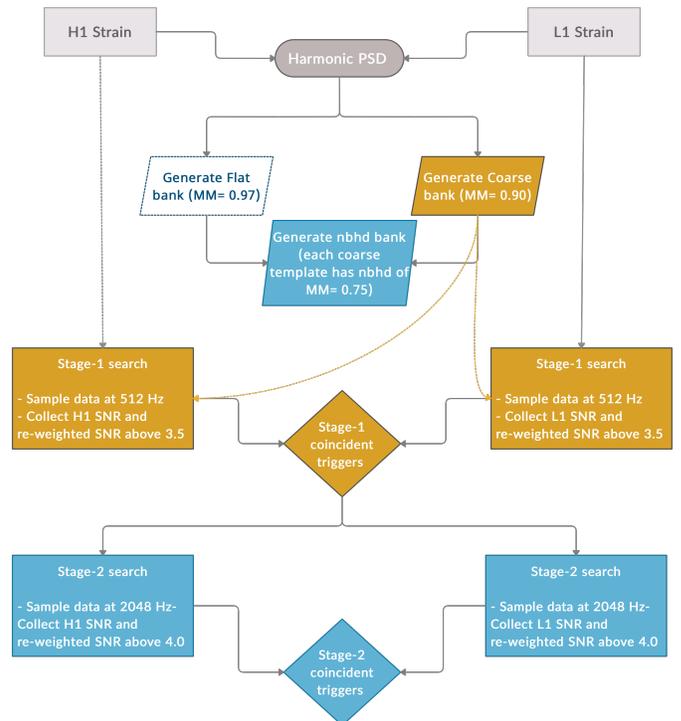


FIG. 1. A flowchart depicting the working of a two-stage hierarchical search pipeline. The choice of color describes the stage: yellow for stage-1, and blue for stage-2. The first step generates the harmonic power spectral density (PSD) using the data strains from the two detectors. The generated PSD is used to create flat (in dotted box) and coarse banks. Using these two banks, a nbhd bank is thus constructed. Stage-1 search begins with matched filtering of the data strains from the two detectors using the coarse bank. The triggers are identified above SNRs and re-weighted SNRs of 3.5. Next, a coincidence test is made to collect foreground candidates (in diamond box) over which stage-2 matched filtering is again performed. Stage-2 search uses a subset of nbhd bank, stage-2 bank, for filtering data and generates single-detector triggers above SNR and re-weighted of 4. The selected triggers then undergo the coincidence test to obtain foreground candidates of the second stage.

### A. Template banks

One of the most crucial steps in a template-based search is to adequately grid up the parameter space. A pragmatic approach suggests densely populating the search space to minimize the loss in SNR. However, such dense placement of templates increases the computational cost and limits the volume and dimensionality of the parameter space that can be covered, given a fixed amount of computation power. Generally, the templates are placed such that the match does not fall below a certain minimum value called the minimal match (MM). For instance, if MM is chosen as 0.97, it means that the expected SNR for a signal does not fall more than 3% ( $1 - MM = 0.03$ ), corresponding to a loss of  $\sim 10\%$

( $\approx 1 - \text{MM}^3$ ) in the astrophysical events.

The ‘match’ ( $\mathcal{M}$ ) between two normalised templates is their scalar product. A scalar product of two data trains  $x(t)$  and  $y(t)$  is defined as:

$$(x, y) = 4 \mathcal{R} \left\{ \int_{f_{low}}^{f_{high}} \frac{\tilde{x}(f)\tilde{y}^*(f)}{S_n(f)} df \right\}, \quad (1)$$

where  $\mathcal{R}$  denotes the real part of a complex quantity and the frequencies  $f_{low}$  to  $f_{high}$  represent the sensitive band of the detector. The tilde represents Fourier transform of the quantity in question, e.g.,  $\tilde{x}(f)$  is the Fourier transform of  $x(t)$  and is given by:

$$\tilde{x}(f) = \int_{-\infty}^{\infty} x(t)e^{-2\pi ift} dt. \quad (2)$$

The one-sided power spectral density (PSD),  $S_n(f)$ , of the detector’s noise is defined via the equation:

$$\langle \tilde{n}(f)\tilde{n}(f') \rangle = \frac{1}{2} S_n(f)\delta(f - f'), \quad (3)$$

when the data contains only the noise  $n(t)$ . Here, angular brackets denote the ensemble average of the noise realizations.

Consider two neighboring templates  $h(\vec{\theta})$  and  $h(\vec{\theta} + \Delta\vec{\theta})$  given by parameters  $\vec{\theta}$  and  $(\vec{\theta} + \Delta\vec{\theta})$ , where  $\vec{\theta} = \{m_1, m_2, s_{1z}, s_{2z}\}$ . If the template are normalised, that is,  $\|h(\vec{\theta})\| = \|h(\vec{\theta} + \Delta\vec{\theta})\| = 1$ , then the match is given by,

$$\mathcal{M}(\vec{\theta}, \Delta\vec{\theta}) \equiv (h(\vec{\theta}), h(\vec{\theta} + \Delta\vec{\theta})). \quad (4)$$

Geometrically, the match is the cosine of the angle between the two normalised templates.

Assuming a slowly varying metric  $g_{mn}(\vec{\theta})$  around the targeted templates  $h(\vec{\theta})$  and  $h(\vec{\theta} + \Delta\vec{\theta})$ , we Taylor expand the match to the lowest order of  $\Delta\theta$  as:

$$(h(\vec{\theta}), h(\vec{\theta} + \Delta\vec{\theta})) \approx 1 - g_{mn}(\vec{\theta})\Delta\theta^m\Delta\theta^n. \quad (5)$$

where template space-time metric is defined as:

$$g_{mn} = -\frac{1}{2} \left( h(\vec{\theta}), \frac{\partial^2 h}{\partial\theta^m\partial\theta^n}(\vec{\theta}) \right). \quad (6)$$

Therefore by varying the source parameters  $\vec{\theta}$  and calculating the metric  $g_{mn}$ , templates could be effectually placed in the bank. However, generally, the metric does not have a closed-form form for aligned-spin waveforms having inspiral, merger, and ringdown (IMR) phases for a wide range of source parameters, e.g., SEOBNR [31]. Moreover, in some cases, metric changes rapidly across the parameter space, making the sphere-covering problem [32] highly non-trivial. Therefore, techniques like stochastic placement [33] are used to construct the bank, where a direct match is computed between templates for varying source parameters. This technique efficiently places

the templates in a bank. However, if the volume of the parameter space (as defined via the metric) is large, then the template bank also becomes large and consequently increases the computational cost for bank generation. In such a case, techniques like hybrid geometric-random placements [34, 35] efficiently generate a full non-precessing bank.

The density of templates in a bank relies on the noise PSD of the detectors in the network. Since the search pipelines use a common template bank for each detector, it is necessary to compute the time-averaged noise PSD for each detector for a bank’s construction. For estimating this effective PSD, the PSDs for the detectors are combined as a harmonic mean [36–38].

In this work, we construct a coarse and nbhd bank for targeting GW signals from non-precessing sources with quasi-circular orbits, using Advanced LIGO-Virgo noise PSD as used in GWTC-2 [13]. The density of templates in these two banks varies depending on the mismatch value used for their construction. The banks are constructed over the parameter space described in GWTC-1 [7]. We describe the construction of banks for the parameter ranges (see Table II A) and their effectualness in the following subsections.

Bank	MM	$M_{tot}(M_{\odot})$	$\chi_{BH}$	$\chi_{NS}$	$f_{min}$ (Hz)
Coarse	0.90	3–500	-0.998–0.998	-0.05–0.05	15
Flat & nbhd	0.97	3–500	-0.998–0.998	-0.05–0.05	15

TABLE I. Table summarizing minimal match values and ranges of the source parameters for the coarse, nbhd, and flat banks. The  $\chi_{BH}$  and  $\chi_{NS}$  are the dimensionless effective spins for a black hole and a neutron star, respectively.

### 1. Coarse bank

We construct a coarse bank with a mismatch of 10% (or  $\text{MM} = 0.9$ ) following Gadre *et al.* [30], using the hybrid geometric-random method [34, 35]. The templates in the bank are generated at a minimum frequency of 15 Hz. We discard the templates with a duration of less than 0.15 seconds to avoid artifacts in the matched filtering steps. The bank is designed to search for non-precessing CBC sources with the total mass ( $M_{tot}$ ) of the binary in the range  $[3 M_{\odot}, 500 M_{\odot}]$ . We restrict the secondary mass ( $m_2$ ) of the binary components  $[1 M_{\odot}, 120 M_{\odot}]$ . The ranges for individual dimensionless spins of the binaries comprising a black hole ( $\chi_{BH}$ ) and a neutron star ( $\chi_{NS}$ ) are provided in Table II A. Thus, we construct an effectual non-precessing bank (see Fig. 2) consisting of 85080 templates.

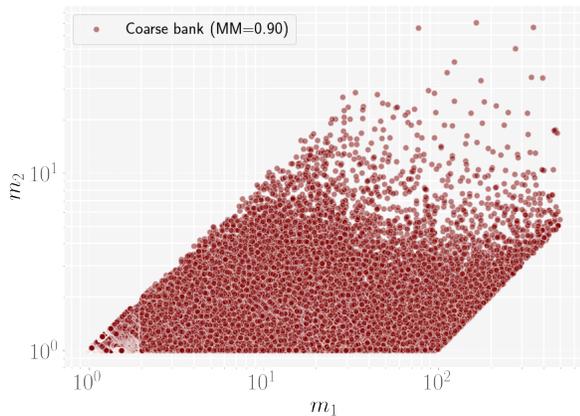


FIG. 2. The distribution of coarse bank templates in  $m_1 - m_2$  mass-plane. Each point (or a template) in the plane has a match  $\mathcal{M} \geq 0.90$  with the neighbouring template.

## 2. Neighbourhood bank

A nbhd of a stage-1 coincident trigger template is the region in parameter space where mismatch with neighboring templates can be up to 25%, as described in section IIIB2 of *Gadre et al.* [30]. To sample these nbhds, we use a pre-generated flat bank with  $MM = 0.97$  covering a similar search parameter space (see Table II A). We include flat bank's templates having  $MM \geq MM_{\text{nbhd}} \equiv 0.75$  with the trigger template. We calculate nbhds for all the coarse templates. This pre-computed bank with assigned nbhds is referred to as a nbhd bank, and a dynamic subset of it is a stage-2 bank. The stage-2 bank is dynamic because it changes depending on the noise realisation.

To identify nbhds of coarse templates, we adopt the following strategy. For coarse templates with  $M_{\text{tot}} > 12 M_{\odot}$ , we perform an exact match calculation with all the flat bank templates. For templates with  $M_{\text{tot}} < 12 M_{\odot}$  [39], we first shortlist a set of templates that may be able to satisfy the nbhd criteria. For that, we define a minimal match ellipsoid with  $MM = MM_{\text{nbhd}}$  in the following way: Consider a coarse template of  $M_{\text{tot}} < 12 M_{\odot}$  for which nbhd has to be calculated. We first construct a minimal-match ellipsoid centered at this template in a coordinate system where the metric varies slowly over the parameter space, i.e., the metric is almost constant, and the signal manifold is almost flat. Therefore we choose chirp-time coordinates  $\{\tau_0, \tau_3, \tau_{3s}\}$ , collectively labeled as  $\tau^\alpha$ . These coordinates are given by scaling  $\{\theta_0, \theta_3, \theta_{3s}\}$ , described in the Ref. [34], with  $(2\pi f_o)^{-1}$  at  $f_o = 20$  Hz. In these coordinates, we estimate the metric components using *TaylorF2RedSpin* [39] waveform model.

Following [34], we diagonalise the metric by an orthogonal transformation  $\mathcal{O}$  and obtain the eigenvalues  $\gamma_\alpha$  with new coordinates  $\xi^\alpha = \mathcal{O}_\beta^\alpha \tau^\beta$ . The metric in these coordinates

is in a diagonal form and is given by:

$$ds^2 = \sum_{\alpha=1}^3 \gamma_\alpha (d\xi^\alpha)^2. \quad (7)$$

This is just a principal axis transformation to an orthogonal basis. Along the eigen-directions, the lengths of the semi-axes ( $r_\alpha(MM)$ ) of the ellipsoid for a given value of  $MM$  are given by:

$$r_\alpha(MM) = \sqrt{\frac{1 - MM}{\gamma_\alpha}}. \quad (8)$$

As  $MM$  reduces from its maximum value of unity, the ellipsoid increases in size. In the  $\xi^\alpha$  coordinates, let the coarse and flat templates be labelled by  $\xi_0^\alpha$  and  $\xi^\alpha$ , respectively. Let  $\Delta\xi^\alpha = \xi^\alpha - \xi_0^\alpha$ , and define the distance  $d(\xi^\alpha, \xi_0^\alpha)$  by the equation:

$$d^2(\xi^\alpha, \xi_0^\alpha) = \sum_{\alpha=1}^3 \gamma_\alpha (\Delta\xi^\alpha)^2. \quad (9)$$

Then the relation  $d(\xi^\alpha, \xi_0^\alpha) \leq \sqrt{1 - MM_{\text{nbhd}}}$  defines the ellipsoid in  $\xi^\alpha$  coordinates. We use this ellipsoid to guide our selection of flat templates. Note that the metric approximation is extrapolated beyond its validity regime, and so the ellipsoid is only a crude estimate of the nbhd. In any case, since we have made a generous choice of  $MM = MM_{\text{nbhd}}$ , we do not expect to miss out on any signals. We choose templates accordingly in this region and compute the match between a flat template and a given coarse template. If the match is above the stipulated  $MM_{\text{nbhd}}$ , we retain the template in the nbhd. *Thus, the final list of templates in the nbhd is obtained by the actual computation of the match between coarse and fine templates inside the ellipsoid.* In general, we find that a single nbhd around a coarse template contains  $\sim 100 - 150$  templates (see Fig. 3).

## B. Matched-filter

The model-dependent search for GWs from CBCs using templates in the LIGO-Virgo data exploits the matched filtering [18] technique rigorously. This technique correlates discretely sampled time-series data ( $s(t)$ ) with the normalized templates ( $h(\vec{\theta})$ ) for the source parameters ( $\vec{\theta}$ ) within the detectors' sensitive band. The correlation generates matched filter SNR ( $\rho$ ) maximized over phase, which is defined as:

$$\rho(t; \vec{\theta}) \equiv \left| \left( s(t), (1+i)h(\vec{\theta}) \right) \right|. \quad (10)$$

Generally, the data obtained from the detectors is non-stationary and non-Gaussian due to the presence of high-power noise transients. Pre-processing steps involving data-quality checks and application of vetoes flag most

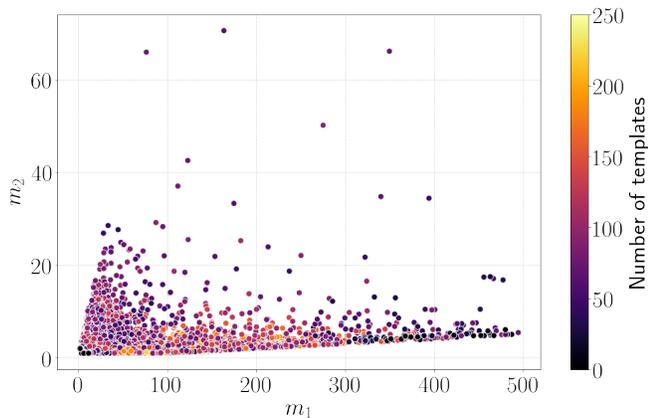


FIG. 3. The distribution of nbhd bank templates in  $m_1 - m_2$  mass-plane. The color bar represents the number of templates in the nbhd of each coarse template with masses  $m_1$  and  $m_2$ . The density of templates in each nbhd is large for lighter-mass binaries. The nbhd shrinks for high-mass coarse templates. Typically, there are  $\sim 100 - 150$  flat templates in a nbhd.

of the artifacts present in the data [40]. Nevertheless, the short-duration glitches or long-duration correlations, as described in Venumadhav *et al.* [10], still remain in it. Matched filtering over these noise transients often leads to high SNRs. These high-SNR triggers generated are removed from the standard search pipelines by nullifying noise contributions in the time series data via gating [6]. We, therefore, apply a similar gating priority to the matched filtering for each detector to remove the non-stationary transients from the data strain in our analysis.

Matched filtering the data produces several triggers with varying SNRs for each template in the bank. These triggers are first clustered within a time window of 0.5 seconds to retain only the ones with high SNRs [41]. In the second step, the SNRs of triggers due to loud noise artifacts are suppressed using signal consistency tests like the standard chi-square ( $\chi_r^2$ ) [6, 42], and sine-Gaussian ( $\chi_{sg}^2$ ) [43].

Like in flat search, the trigger SNRs ( $\rho$ ) generated in both the stages of the hierarchical search are down-weighted with their reduced chi-square values using  $\chi_r^2$  [6, 42] veto defined as:

$$\tilde{\rho} = \begin{cases} \frac{\rho}{[(1+(\chi_r^2)^3)/2]^{1/6}} & \text{if } \chi_r^2 > 1, \\ \rho & \text{otherwise.} \end{cases} \quad (11)$$

Usually,  $\chi_r^2$  veto is ineffective in the region where signals are too short. In such cases, the short-duration templates ring with ‘blip’ glitches present in the data. Therefore, we further down-weight  $\tilde{\rho}$  for the templates with  $M_{tot} > 30 M_\odot$  using  $\chi_{sg}^2$  [43] veto defined as:

$$\hat{\rho} = \begin{cases} \tilde{\rho} (\chi_{r,sg}^2)^{-1/2} & \text{if } \chi_{r,sg}^2 > 6, \\ \tilde{\rho} & \text{otherwise.} \end{cases} \quad (12)$$

In each stage of the hierarchical search, the triggers that surpass the two tests above certain values of  $\rho$  and  $\tilde{\rho}$  (see

Sec. III) are then used for estimating the coincidences and significance of the real GW signals. Finally, the coincidences are obtained based on the optimal detection statistics, which we elaborate on in the following section.

### C. Ranking statistics

A pair of triggers from the two detectors is coincident if it simultaneously occurs within the light travel time between the detectors and is recovered with identical template parameters. The coincidence is evaluated based on optimal detection or ranking statistics ( $\Lambda_{opt}$ ), which is defined as the ratio of the likelihood for data containing signal to the likelihood for data having noise [44]. These likelihoods are the functions of the template parameters ( $\vec{\theta}$ ) and  $\hat{\rho}$ ,  $\chi_r^2$ .

In the recent works [7–9, 23], the optimal detection statistics were improvised by taking the ratio of coincident event rate densities due to signal ( $p(\vec{\kappa}|S)$ ) and noise ( $p(\vec{\kappa}|N)$ ). Therefore, for an unknown coincident with template parameters  $\vec{\kappa} = \{\hat{\rho}_H, \hat{\rho}_L, \chi_H^2, \chi_L^2, \delta t, \delta\phi, \vec{\theta}\}$  where  $\delta t$ ,  $\delta\phi$  is the time and phase difference in between two detectors,  $\Lambda_{opt}$  is given as:

$$\Lambda_{opt} = \frac{p(\vec{\kappa}|S)}{p(\vec{\kappa}|N)} \equiv \frac{p(\vec{\kappa}|S)}{r_{\vec{\theta}}^{HL} p(\vec{\theta}, \delta t, \delta\phi|N)}. \quad (13)$$

For the statistics, we expect  $p(\delta t, \delta\phi|N)$  to be uniform over ( $\vec{\theta}, \delta t, \delta\phi$ ) [44], thereby it is marginalized and treated as a constant. If we assume the noise to be uncorrelated between detectors, then we can safely write  $p(\vec{\kappa}|N)$  ( $\approx r_{\vec{\theta}}^{HL}$ ) as a product of single-detector noise rate densities [23] given by:

$$r_{\vec{\theta}}^{HL} = 2 \tau_{HL} (r_{\vec{\theta},H}(\hat{\rho}_H) r_{\vec{\theta},L}(\hat{\rho}_L)), \quad (14)$$

where,  $\tau_{HL}$  is the allowed time window for a coincidence of trigger in twin LIGO detectors.

Thus, by estimating  $r_{\vec{\theta}}^{HL}$  and  $p(\vec{\kappa}|S)$  through accurate modeling [23, 43], we can obtain  $\Lambda_{opt}$  for the coincident triggers in the flat and hierarchical search stages. We describe the modeling procedure in the following sections.

#### 1. Signal model

To model  $p(\vec{\kappa}|S)$ , we require the probable astrophysical distribution of the binary sources that Advanced LIGO detectors can detect. In reality, the exact distribution is unknown to the observers. Nevertheless, the source population can be approximated as uniform in volume and isotropic in the sky location and orientation of the binary. Assuming these distributions for sources, we can estimate how their detection parameters like signal amplitudes, time, and phase differences vary with respect to the pair of the LIGO detectors.

To begin with the calculation for  $p(\vec{\kappa}|S)$ , Monte Carlo simulations are performed for each coincident trigger with parameter  $\vec{\kappa}$  [45]. A histogram is generated in this process and used as a look-up table for the distribution. We use the above recipe to generate  $p(\vec{\kappa}|S)$  in flat and both stages of the hierarchical search.

## 2. Noise model for flat and stage-1

To obtain  $r_{\vec{\theta}}^{HL}$ , we first estimate single-detector noise rate densities ( $r_{\vec{\theta},d}$ ). To calculate this quantity, we model the tail of the trigger distribution for each detector ( $d$ ) and template with a falling exponential function [23] as:

$$r_{\vec{\theta},d}(\hat{\rho}_d, N) = \mu(\vec{\theta}) p(\hat{\rho}_d|\vec{\theta}, N), \quad (15)$$

given,

$$p(\hat{\rho}_d|\vec{\theta}, N) = \begin{cases} \alpha(\vec{\theta}) \exp[-\alpha(\vec{\theta})(\hat{\rho}_d - \hat{\rho}_{th,d})] & \text{if } \hat{\rho}_d > \hat{\rho}_{th,d}, \\ 0 & \text{otherwise,} \end{cases} \quad (16)$$

where  $\mu(\vec{\theta})$  and  $\alpha(\vec{\theta})$  denotes trigger count above the threshold ( $\hat{\rho}_{th,d}$ ) and exponential decay rate, respectively.

The fit parameter  $\alpha(\vec{\theta})$  is obtained by maximum logarithmic likelihood fitting method. For discrete samples of  $\hat{\rho}_d$  of  $k^{th}$  trigger, we maximize:

$$\ln p(\hat{\rho}_d|\alpha, n) = n \ln \alpha - \alpha \sum_k^n (\hat{\rho}_{k,d} - \hat{\rho}_{th,d}), \quad (17)$$

at a fixed  $\hat{\rho}_{th,d}$  ( $\equiv 6$ ) to obtain  $\alpha_{max} = (\bar{\hat{\rho}}_d - \hat{\rho}_{th,d})^{-1}$ . Here,  $\bar{\hat{\rho}}_d$  is the mean of  $\hat{\rho}_d$  and the variance ( $\sigma_d$ ) in fit parameter is given by  $1/\sqrt{n}$ , where  $n$  denotes the number of triggers generated for a particular template.

In the flat and stage-1 search, we calculate  $\alpha_{max}$  and  $n$  for each coarse template. Generally, not all the templates have a sufficient number of triggers above 6 to fit the trigger distribution's exponential tail. In such cases, the low number of triggers gives a high variance to the fit parameter values. To avoid such problems, we take the moving average of the fit parameters [23]. We also smooth the counts above the threshold by taking mean over nearby templates with similar values of effective spin ( $\chi_{eff}$ ), template duration, and symmetric mass ratio ( $\eta$ ).

## Noise model for Stage-2

In principle, the procedure for obtaining single-detector noise rate densities described previously can be applied in stage-2. However, it cannot be implemented, as this stage possesses insufficient triggers above  $\hat{\rho}_{th,d}$  to obtain meaningful fit parameters. The reason is, we follow only foreground candidates from stage-1 that have  $\Lambda_{opt} \geq 7$ . Matched filtering over these followed-up triggers utilizes fewer nbhds and corresponding templates to

generate fewer triggers. Having an inadequate number of triggers for a template can give a large variance in the values fit parameters, leading to overestimating single-detector noise rates. We, therefore, *do not* explicitly calculate the fit parameters in stage-2. Instead, we reuse the fit-values of the ‘closest’ coarse template to the stage-2 trigger template. The ‘closeness’ relies on the highest match value between the coarse and stage-2 bank templates. This approximation is justified as the noise trigger rate fits represent the region's response of the template parameter space to the strain data.

To verify the applicability of the above procedure, we perform a flat and hierarchical search on 14 days and obtain fit parameters. Figure 4 compares the fit parameters obtained in both the searches. The scatter points in the diagonal signifies that the values are comparable for the two searches in both the detectors. A few templates in the Handford (H1) detector show low  $\alpha$  indicating small fluctuations in their values due to noise. These small fluctuations can appear at different periods of observational time. However, these variations in  $\alpha$  negligibly affect the modeling of single-detector noise rate density, as can be seen later in IV.

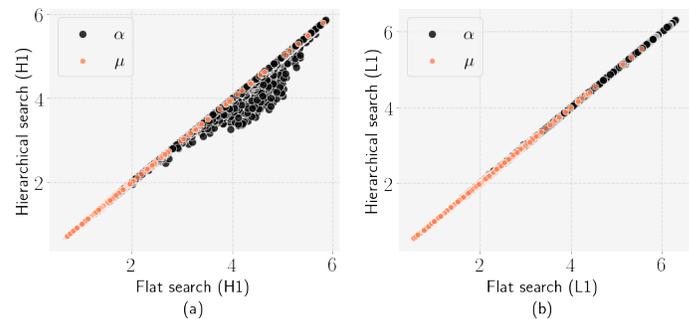


FIG. 4. Comparison plot for the fit coefficients,  $\alpha$ , and  $\mu$ , obtained from a flat and hierarchical search for (a) Handford (H1) and (b) Livingston (L1) detectors.

## D. Assigning significance

The significance of any event is evaluated based on their FAR estimate above a fixed statistic  $\Lambda^*$  as:

$$\text{FAR}(\Lambda^*) = \int d^n \vec{\kappa} r_{\vec{\kappa}}^{HL} \Theta(\Lambda_{opt}(\vec{\kappa}) - \Lambda^*), \quad (18)$$

where  $r_{\vec{\kappa}}^{HL} \equiv r_{\vec{\theta}}^{HL}$  by construction. FAR signifies the odds of finding a non-astrophysical coincident candidate with a similar or higher  $\Lambda_{opt}$  (see Eq. (13)) in the observing period. FAR is estimated in the flat and stage-1 search with respect to a noise background constructed by time-sliding data by a minimum of 15 milliseconds across the detectors. Such a procedure omits all the possibilities to have a coincidence due to a real GW signal. At each time-shift,  $\Lambda_{opt}$  is re-computed to rank the

candidates above a certain threshold ( $\Lambda^*$ ). Performing several time shifts generates many plausible candidates and unnecessarily adds to the background computation cost. To overcome this problem, the computation of coincidences is optimized. At first, a clustering over time is performed such that candidates with the highest statistic, falling within 10 seconds, are kept. In the next step, we selectively choose the candidates with all or few time-slides falling in the ranking statistic value's bin. For instance, we collect the candidates with all possible time-slides for ranking statistics greater than 9 but only select time slides of 30 seconds for those with a statistic value between 8 and 8.5.

In principle, a similar strategy can be implemented to assign FARs to the detected candidates in stage-2 of the hierarchical search. However, the background constructed by time-sliding stage-2 triggers across detectors using a stage-2 bank can lead to biases in the FAR estimates of detected candidates, as shown in Gadre *et al.* [30]. Therefore we avoid such biases by constructing an approximate background that would mimic a background constructed in the flat search. As proposed in Gadre *et al.* [30], we construct a scaled stage-1 background for assigning significance to the final list of coincident triggers. First, we construct stage-1 background by performing several time-slides of stage-1 foreground triggers across the detectors as done in a flat search. We then scale this background by a factor equal to the ratio of the number of templates times the sampling frequency used in a flat search to that used in stage-1. This number is close to the computational gain and comes out to be 20.1.

To justify our argument on mimicking a flat background, we compare the foreground and background obtained from a flat and hierarchical search performed over 14 days of data around the first BBH, GW150914 [1] event. In Fig. 5, the foregrounds due to noise candidates match their respective backgrounds for both the searches. We observe that the noise background is higher in the lower ranking statistic region than that of flat. This observation is expected as the scaling factor linearly increases the number density of noise triggers in a particular ranking statistic bin. We also notice that the scaled stage-1 background roughly matches the flat background above ranking statistic value 8. Therefore, the reliability of the FARs will be limited to the ranking statistic value  $\gtrsim 8$ . This method reasonably mimics the flat search background and produces approximate FARs, which may provide reasonable reliability to expand the size and dimensionality of the parameter space, without which we may miss out on some of the interesting sources.

### III. SEARCH FOR CBC IN O1 AND O2 DATA

We search for CBCs using the two-stage hierarchical search [46] over the data from the first (O1) and second (O2) observing runs of twin LIGO detectors. We use 21.39 days of coincident data from O1 and 31.4 days from

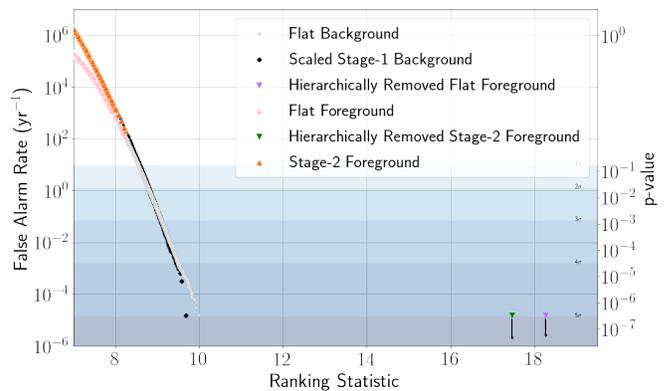


FIG. 5. Plot showing FAR vs. ranking statistic curves for the foreground candidates (foreground) and the time-shifted candidates (background) from a flat and hierarchical search. The foreground (in a triangle) overlays the background (in a circle) in each search. The loudest event, GW150814, is hierarchically removed from the background in both searches. Note that the scaled stage-1 background (in black) roughly matches the flat background (black) above ranking statistic value 8.

O2.

The periods of poor quality data are marked and removed from the analysis using data-quality flags, Category-1 (CAT-1) and Category-2 (CAT-2) [40]. CAT-1 vetoes remove the times during which at least one of the key components of a detector was not operational in the nominal configuration due to critical issues. The duration over which excessive noise is observed due to instrumental artifacts is marked and removed by CAT-2 flags.

As described in Sec. II B, the data undergo pre-processing before entering the matched filtering step. In both stages, we use 512 seconds of overlapping data segments for matched-filter computation. We pad data segments with zeros 144 seconds at the beginning and 16 seconds in the end to avoid the artifacts generated from the FFT algorithm. Once the data segment is prepared, we perform a hierarchical search in two stages.

We begin the search by matched filtering data segments sampled at 512 Hz with a coarse bank, as described in Sec. II A 1 and obtain a list of stage-1 triggers above coarse thresholds on  $\rho$  and  $\tilde{\rho}$ . For triggers with  $\rho > 3.5$  that pass the  $\chi_r^2$  test with  $\tilde{\rho} > 3.5$ , gets further re-weighted by  $\chi_{sg}^2$  veto. The choice of coarse thresholds for stage-1 search may seem arbitrary. However, we tested out different values for  $\rho$  and  $\tilde{\rho}$  and found that setting both values at 3.5 gives the optimal computational cost of handling bulk triggers.

The surviving single-detector triggers then undergo a coincidence test (see Sec. II C) to obtain foreground candidates. These foreground candidates are then followed up in stage-2.

The stage-2, or hierarchical search, begins with matched filtering data segments sampled at 2048 Hz that

contains foreground candidates with  $\Lambda_{opt} > 7$  [30] from stage-1. These segments are filtered using a stage-2 bank constructed from the dynamic union of nbhds around each followed-up trigger template (see Sec. II A 2). The SNRs generated in this process are then re-weighted with fine thresholds on  $\rho$  and  $\tilde{\rho}$  of 4. The resultant triggers are then subjected to a coincidence test, as described in Sec. II C, to obtain the second stage’s foreground candidates.

The final step in the search involves assigning significance to the potential foreground candidates ( $\Lambda_{opt} > 8$ ) obtained in stage-2. We assign FARs to these candidates using a scaled stage-1 background, as described in Sec. II D. Based on this background, we present the results from the analysis in the next section.

We report the recovery of all ten confirmed GW events with FAR below 1 per year in stage-2 of the hierarchical search. These events were previously detected by the flat analysis in GWTC-1. Although the detection statistics used in both the stages of hierarchical search are more recent than those used in the flat analysis of GWTC-1, we still detect these events with nearly similar detection confidence levels in stage-2 but with a computational gain in the matched filtering by a factor of  $\sim 20$ . A comparison of the recovered events’ FARs and network SNRs ( $\hat{\rho}_T \equiv \sqrt{\hat{\rho}_H^2 + \hat{\rho}_L^2}$ ) from the flat search in GWTC-1 and both the stages of the hierarchical search, is given in Table II.

In our analysis, we recover the loudest events—GW150914, GW151226, GW170104, GW170608, GW170814, GW170817, and GW170823, with comparable FARs in both the stages of the hierarchical search. However, the network SNRs of these events improve in stage-2. The remaining events, like GW151012, GW170729, and GW170809, see improvements in their FARs and network SNRs in the stage-2 search. This result signifies that conducting a stage-2 search over the potential stage-1 foreground candidates is essential for improving the SNRs and the significance of the candidates.

#### IV. COMPARISON WITH THE FLAT SEARCH

The sensitivity of any search pipeline depends on the number of astrophysical signals detected at a given FAR threshold. Thus, in general, the FAR estimates of the detected signals can primarily give insight into its efficiency. However, to accurately assess the search pipeline’s sensitivity, simulated injection studies are generally found beneficial.

In this part, we highlight the computational advantages of hierarchical search over flat search while comparing their search sensitivities using similar detection statistics, as defined in Sec. II C.

#### A. Comparison of Sensitivities

For a population of binary mergers, uniformly distributed over comoving volume ( $V$ ), one can compute the sensitive reach of the detectors in terms of the volume covered in the given observable time. Suppose that a binary’s merger rate is defined by  $\mu_m$ , then the number of detection that one can make above a certain FAR threshold in  $T_{obs}$  observation time is the product of volume, time, and merger rate ( $\mu_m \langle VT \rangle$ ) [47]. The sensitive volume-time ( $\langle VT \rangle$ ) over here is defined as:

$$\langle VT \rangle_{\{\vec{\theta}\}} = T_{obs} \int_0^\infty p(z|\{\vec{\theta}\}) \frac{dV}{dz} \frac{1}{(1+z)} dz, \quad (19)$$

where,  $p(z|\{\vec{\theta}\})$  is the probability of recovering a signal with parameters  $\vec{\theta}$  at a redshift  $z$ .  $VT$  in Eq. (19) is computed by Monte Carlo integration for the detected sources. If we assume  $\mu_m$  to be constant for all the search pipelines, the ratio of  $VT$  can be exploited to compare the sensitivities of any two search pipelines [7, 23].

To compute the  $VT$  ratio, we inject simulated GW signals from non-precessing CBCs into the real data. For BNS-like sources, we use `SpinTaylorT4` [48], and for BBH and NSBH like sources we use `SEOBNRv4_opt` [31] waveform models. Since we lack prior information on the binary merger population, we distribute the signals obtained from the models uniformly over the chirp distance between 50 Mpc and 400 Mpc. We uniformly distribute the logarithms of component mass of BBH and NSBH-like sources in the ranges  $[2 M_\odot, 5 M_\odot]$  and  $[3.5 M_\odot, 100 M_\odot]$ , respectively. For BNS injections, we uniformly distribute the component masses between  $2 M_\odot$  and  $5 M_\odot$ .

In our study, we inject 12203 simulated BNS, and 16271 BBH and NSBH signals in five days of coincident data in O1 from September 12, 2015, to September 26, 2015. We analyze the data using the flat and hierarchical search pipelines separately. The matched filtering and coincidence studies in the hierarchical search are carried out as per Sec. III. In the case of flat search, we perform matched filtering over data segments sampled at 2048 Hz and identify triggers with  $\rho$  and  $\tilde{\rho}$  above 4 in each detector. With the appropriate clustering in time as defined in II B, we run a coincidence test over the collected single-detector triggers. Here, triggers observed within 15 milliseconds of a time window in two detectors are identified and ranked according to their statistic values (see Sec. II C).

The foreground candidates obtained in both the searches are assigned FARs based on their respective noise backgrounds using similar ranking statistics described in Sec. II C. In the flat search, we estimate the background by time-sliding triggers across the detectors. Here, each trigger is shifted by 0.1 seconds in time, and then again, the statistic is estimated. A time slide of 0.1 seconds can generate a large number of triggers. Therefore, we first cluster the candidates within a time window

Sl. no.	Event	UTC	Flat			Hierarchical					
			FAR (yr <sup>-1</sup> )	$\hat{\rho}_T$	$\mathcal{M}$ (M <sub>⊙</sub> )	Stage-1			Stage-2		
						FAR (yr <sup>-1</sup> )	$\hat{\rho}_T$	$\mathcal{M}$ (M <sub>⊙</sub> )	FAR (yr <sup>-1</sup> )	$\hat{\rho}_T$	$\mathcal{M}$ (M <sub>⊙</sub> )
1	GW150914	09:50:45.4	$\leq 1.53 \times 10^{-5}$	23.6	32.75	$1.526 \times 10^{-5}$	23.30	29.71	$1.526 \times 10^{-5}$	24.0	31.96
2	GW151012	09:54:43.4	0.17	9.5	18.47	0.422	8.98	18.68	0.059	9.8	18.3
3	GW151226	03:38:53.6	$\leq 1.70 \times 10^{-5}$	13.1	9.7	$1.692 \times 10^{-5}$	11.91	9.89	$1.692 \times 10^{-5}$	13.1	9.72
4	GW170104	10:11:58.6	$\leq 1.39 \times 10^{-5}$	13.0	20.19	$1.374 \times 10^{-5}$	12.26	18.37	$1.374 \times 10^{-5}$	12.9	29.17
5	GW170608	02:01:16.5	$\leq 3.09 \times 10^{-4}$	15.4	8.61	$3.08 \times 10^{-4}$	8.90	8.65	$3.08 \times 10^{-4}$	14.83	9.03
6	GW170729	18:56:29.3	1.36	9.8	40.27	1.689	9.36	54.41	0.054	10.6	47.5
7	GW170809	08:28:21.8	$1.45 \times 10^{-4}$	12.2	23.53	0.561	11.32	29.71	0.0017	12.1	23.65
8	GW170814	10:30:43.5	$\leq 1.25 \times 10^{-5}$	16.3	25.2	$1.253 \times 10^{-5}$	16.04	25.09	$1.253 \times 10^{-5}$	17.2	26.58
9	GW170817	12:41:04.4	$\leq 1.25 \times 10^{-5}$	30.9	1.2	$2.506 \times 10^{-5}$	28.67	1.2	$1.253 \times 10^{-5}$	31.5	1.2
10	GW170823	13:13:58.5	$\leq 3.29 \times 10^{-5}$	11.1	23.61	$3.301 \times 10^{-5}$	11.29	32.32	$3.301 \times 10^{-5}$	11.1	46.85

TABLE II. List of GW events detected via hierarchical search. The events are arranged in the ascending order of their event time. We report and compare detected event’s FAR, network SNR ( $\hat{\rho}_T$ ), and chirp mass ( $\mathcal{M}$ ) in the stage-1 and stage-2 search with the events reported by the flat search in Abbott *et al.* [7]. We see an improvement in the FAR and network SNR values for the events with network SNR varying between 9 to 12 from stage-1 to stage-2.

of 10 seconds and select candidates with time shifts with a statistic greater than pre-defined cutoffs. In the case of hierarchical search, we assign FARs to the detected candidates after scaling the stage-1 background, as described in Sec. IID. The recovered candidates via clustering over statistic values are then sorted with respect to their FARs. A highly ranked candidate with a FAR value below 1 per year [7] and falling within 1 second of merger time is marked as a detected injection.

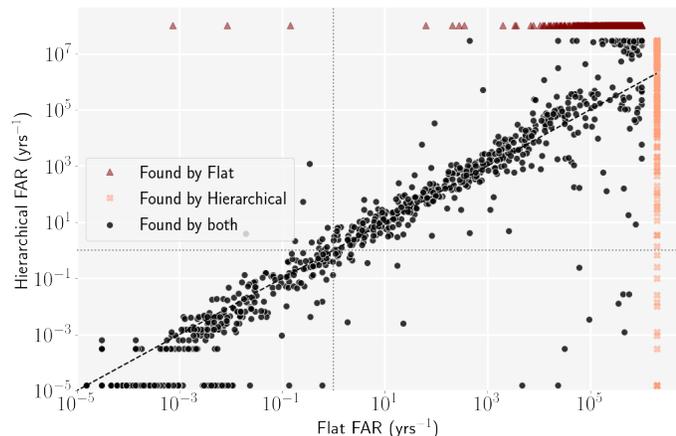


FIG. 6. Scatter plot of FARs for injections recovered from the hierarchical search vs. FARs found by the flat search. The injections that are found by both searches are represented by circular points (in black). The concentration of these points near the diagonal implies that the estimated FAR by both the searches are reasonably close. The other markers show the injections that are found by one search pipeline.

In Fig 6 we compare FARs of the recovered injections from the hierarchical and flat search. We see that most of the injections are recovered with comparable FARs be-

tween both the searches. This can be viewed from the high density of scattered points lying near the diagonal of the plot. Some of the events are recovered by either of the searches. However, these standalone events are seen with low astrophysical significance. Notice that three injections are recovered with higher false alarms by only the flat search. A follow-up study showed that these injections were made at very low SNRs and were likely recovered due to coincidence with noise fluctuation around the injection time. We miss these injections in the hierarchical search because we failed to recover them in the stage-1 search.

The FAR comparison in Fig 6 shows that both flat and hierarchical search performs similarly for loud CBC injection. However, the sensitivity towards detecting fainter injections varies for both searches. This conclusion is further supported by the  $VT$  comparison in Fig 7. The  $VT$  ratio comparison plot between stage-1 search and flat search in the top panel of Fig. 7, shows a sharp decline in the sensitivity across all the chirp mass, and IFAR<sup>4</sup> bins. This result is expected as the loss in matched filter SNRs is bound to happen in stage-1 due to low sampling rates and the use of a coarse bank. However, performing nbhd search on the potential foreground candidates from stage-1 retains the overall sensitivity of the search pipeline, which can be viewed in the bottom panel of Fig. 7. In this plot, we see that the sensitivity of hierarchical search remains consistent with the flat search with  $VT$  ratio varying between a factor of  $1 \pm 1.042$  and  $1 \pm 0.08$  for IFAR of 10 years depending on the chirp mass bins.

<sup>4</sup> Inverse False Alarm Rate (IFAR =  $\frac{1}{\text{FAR}}$ )

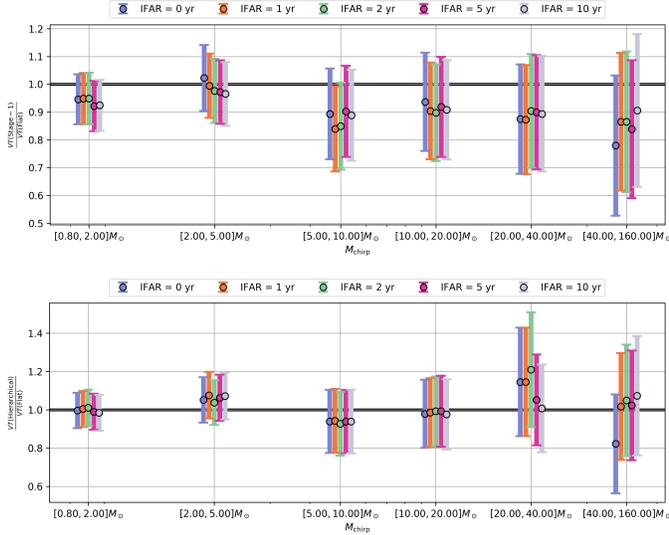


FIG. 7. Comparison of Volume  $\times$  time (VT) ratio of (a) stage-1, and (b) stage-2 (hierarchical) with flat search. The sensitivity of stage-1 search drops for higher chirp mass bins across all IFAR bins in (a). In (b), the sensitivity of hierarchical search improves, maintaining the overall sensitivity of hierarchical search comparable to flat, across entire chirp mass and IFAR bins.

## B. Comparison of computational efficiencies

Now we estimate the computational cost of matched filtering for the flat and hierarchical search.

The computational cost of matched filtering relies on the number of FFT operations performed on a segment using a bank of templates. As defined previously, FFT operations scale as  $N \log_2 N$ . In the case of flat search, we filter a data segment sampled at 2048 Hz with the entire flat bank. If the segment is of length 512 seconds, then  $N$  in the flat search is  $512 \times 2048$ , and the number of matched filter operations is  $512 \times 2048 \times 428725 \times \log_2(512 \times 2048)$ , where 428725 represents the number of templates in the flat bank.

In the case of hierarchical search, the total number of matched filter operations is the sum of the number of FFTs performed in stage-1 and stage-2. Since in stage-1 search, we matched filter a data segment sampled at 512 Hz using the coarse bank containing 85080 templates, the number of matched filter operations comes  $512 \times 512 \times 85080 \log_2(512 \times 512)$ . If the same segment gets followed up for the stage-2 search, then the number of matched filter operations reduces due to fewer templates in a stage-2 bank. The number of templates in this bank can vary for each segment and detector, as can be seen from Fig. 8. Thus, we compute a total number of FFT operations for all the segments for the flat and combined stages of the hierarchical search. To estimate the over gain in the computational speed, we take the ratio of the computed FFT operations for the flat to the

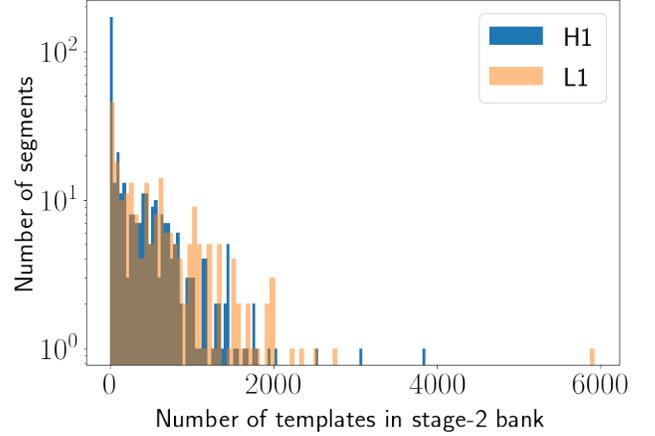


FIG. 8. Histogram depicting the number of templates in stage-2 bank generated for each segment.

hierarchical search.

We first define the following quantities:

Quantity	Description
$N_{\text{seg}}$	Total number of data segments in two detectors
$t_{\text{seg}}$	Duration of each segment
$f_{\text{flat}}$	Sampling rate for flat and stage-2 search
$f_{\text{coarse}}$	Sampling rate for stage-1
$N_{\text{temp}}^{\text{flat}}$	Number of templates in the flat bank
$N_{\text{temp}}^{\text{stage1}}$	Number of templates in the stage-1 bank
$N_{\text{temp}}^{\text{stage2}}$	Total number of templates for <i>all</i> the segments used in the stage-2 search

Let,

$$O_{\text{flat}} = k f_{\text{flat}} t_{\text{seg}} \log_2(f_{\text{flat}} t_{\text{seg}}),$$

$$O_{\text{coarse}} = k f_{\text{coarse}} t_{\text{seg}} \log_2(f_{\text{coarse}} t_{\text{seg}}), \quad (20)$$

where,  $O_{\text{flat}}$  and  $O_{\text{coarse}}$  are the number of floating-point operations required for performing a FFT for a segment at the flat and coarse sampling rates, respectively.  $k$  is a factor of few which cancels out from the numerator and denominator. Thus the gain is given by:

$$\text{gain} \approx \frac{N_{\text{seg}} N_{\text{temp}}^{\text{flat}} O_{\text{flat}}}{N_{\text{seg}} N_{\text{temp}}^{\text{stage1}} O_{\text{coarse}} + N_{\text{temp}}^{\text{stage2}} O_{\text{flat}}}. \quad (21)$$

While the number of templates in flat search and stage-1 are fixed for all segments, it varies for stage-2 as only specific triggers are followed up with the stage-2 bank. The total area of the histograms for the two detectors together shown in Fig. 8 provides us with  $N_{\text{temp}}^{\text{stage2}}$ . Since  $N_{\text{temp}}^{\text{stage2}}$  is much smaller than  $N_{\text{seg}} N_{\text{temp}}^{\text{stage1}}$ , the computation in stage-1 dominates the cost, so the stage-2 cost does not affect the gain.

Substituting the numerical values,  $N_{\text{seg}} = 390 [H1] + 225 [L1] = 615$ ,  $t_{\text{seg}} = 512$  secs,  $f_{\text{flat}} = 2048$  Hz,  $f_{\text{coarse}} = 512$  Hz,  $N_{\text{temp}}^{\text{flat}} = 428725$ ,  $N_{\text{temp}}^{\text{stage1}} =$

85080, and  $N_{\text{temp}}^{\text{stage2}} = (132036 [H1] + 132134 [L1]) = 264170$ , the gain becomes 22 for the analysis. We do not expect this number to change significantly for different observing runs. Thus, we conclude that with the present setting, *the hierarchical search provides an overall computational speed up by a factor of  $\sim 20$ .*

## V. CONCLUSION AND DISCUSSION

Efficient searches for GWs originating from CBCs can expand the size and dimensionality of the search parameter space to detect interesting sources with present and future detectors. The hierarchical search is perhaps the most straightforward approach that brings more than one order of magnitude enhancement in the efficiency without compromising the robustness of the search. In this work, we successfully demonstrate the efficiency of hierarchical search by applying the analysis on the first two observing runs of Advanced LIGO. By introducing essential modifications to the previously developed codes, we transform them into a complete analysis pipeline [46]. We improve upon the selection criteria for single detector triggers using chi-square and sine-Gaussian vetoes to re-weight matched filter SNRs. We also implement a new coincident detection statistics in the hierarchical search that utilizes phase and time differences between detectors and detection parameters, significantly reducing false alarms due to noise events. With our pipeline, we recover all the events in the LIGO-Virgo collaboration's official transient catalog, GWTC-1, detected by the standard PyCBC analysis with nearly the same statistical confidence and a whopping factor of 20 computational speed up. This work also demonstrates that hierarchical search is at hand for production analysis of the present and upcoming datasets from ground-based detectors.

Following Gadre *et al.* [30], we estimate the detected candidates' significance by scaling the noise background obtained in stage-1 with a factor close to the speed-up factor. Although the argument on assigning significance to detected candidates using this background may not be so rigorous, our work shows that this prescription works and can be used to draw astrophysical inferences and electromagnetic follow-up of the detected events with sufficient accuracy. While the outcome of this exercise builds enough confidence for application in production

runs that are otherwise prohibitive due to computational cost, we plan to carry out an extensive study focused on accurate background estimation for the hierarchical search.

In our opinion, the hierarchical search pipeline can be used for ambitious searches that are currently deferred due to computational limitations. For instance, a search for binaries with non-aligned spins and sub-solar sources requires an enormous number of templates. With hierarchical search, we can attempt to carry out their search at feasible computation cost without compromising the accuracy of sensitivity of the search. The hierarchical search can also make it possible to make the low latency searches more reliable and closer to offline analysis. Developing a comprehensive offline or a low latency search for such sources is an arduous task ahead, and more sophisticated techniques will have to be brought in, in the coming years. Nevertheless, the hierarchical search is a giant leap forward in this direction that should be exploited.

## ACKNOWLEDGMENTS

The authors thank the computational resources provided by the IUCAA LDG cluster Sarathi, LIGO Laboratory, and supported by National Science Foundation Grants. KS acknowledges support from Soumen Roy in providing code and helps in the generation of banks. The authors are grateful for the useful discussions from Shreejit P. Jadhav at various stages of this work. KS acknowledges technical support for cluster-related issues from Deepak Bankar. The hierarchical search pipeline uses PyCBC version 1.16.13, and is built upon LALSuite [49], NumPy [50], SciPy [51], and Astropy [52]. KS acknowledges the Inter-University Centre of Astronomy and Astrophysics (IUCAA), India, for the fellowship support. BG acknowledges the support of the Max Planck Society. S. M. acknowledges support from the Department of Science and Technology (DST), Ministry of Science and Technology, India, provided under the Swarna Jayanti Fellowships scheme. SVD acknowledges the support of the Senior Scientist Platinum Jubilee Fellowship from NASI, India. This manuscript has been assigned a LIGO document number LIGO-P2100202.

- 
- [1] B. P. Abbott *et al.*, *Phys. Rev. Lett.* **116**, 061102 (2016).  
 [2] B. P. Abbott *et al.*, *Phys. Rev. Lett.* **116**, 131103 (2016).  
 [3] B. P. Abbott *et al.*, *Phys. Rev. D* **93**, 112004 (2015).  
 [4] S. Klimenko, G. Vedovato, M. Drago, F. Salemi, V. Tiwari, G. A. Prodi, C. Lazzaro, K. Ackley, S. Tiwari, C. F. Da Silva, *et al.*, *Phys. Rev. D* **93**, 042004 (2016).  
 [5] C. Messick *et al.*, *Phys. Rev. D* **95**, 042001 (2017).  
 [6] 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, *et al.*, *Astrophys. J.* **33**, 215004 (2016).  
 [7] B. P. Abbott *et al.*, *Phys. Rev. X* **9**, 031040 (2019).  
 [8] A. H. Nitz, C. Capano, A. B. Nielsen, R. Reyes, R. White, D. A. Brown, and B. Krishnan, *Astrophys. J.* **872**, 195 (2019).  
 [9] A. H. Nitz, T. Dent, G. S. Davies, S. Kumar, C. D. Capano, I. Harry, S. Mozzon, L. Nuttall, A. Lundgren, and M. Tápai, *Classical Quantum Gravity* **891**, 123 (2020).

- [10] T. Venumadhav, B. Zackay, J. Roulet, L. Dai, and M. Zaldarriaga, *Phys. Rev. D* **100**, 023011 (2019).
- [11] J. Aasi, B. P. Abbott, R. Abbott, T. Abbott, M. R. Abernathy, K. Ackley, C. Adams, A. T., P. Addesso, *et al.*, *Classical Quantum Gravity* **32**, 074001 (2015).
- [12] F. Acernese, M. Agathos, K. Agatsuma, D. Aisa, N. Allemandou, A. Allocca, J. Amarni, P. Astone, G. Balestri, G. Ballardin, *et al.*, *Classical Quantum Gravity* **32**, 024001 (2015).
- [13] R. Abbott *et al.*, *Phys. Rev. X* **11**, 021053 (2021).
- [14] B. P. Abbott *et al.*, *Astrophys. J.* **892**, L3 (2020).
- [15] B. P. Abbott *et al.*, *Phys. Rev. D* **102**, 043015 (2020).
- [16] R. Abbott *et al.*, *Astrophys. J.* **896**, L44 (2020).
- [17] R. Abbott *et al.*, *Astrophys. J.* **900**, L13 (2020).
- [18] B. S. Sathyaprakash and S. Dhurandhar, *Phys. Rev. D* **44**, 3819 (1991).
- [19] S. Sachdev, S. Caudill, H. Fong, K. L. R. Lo, C. Messick, D. Mukherjee, R. Magee, L. Tsukad, *et al.*, (2019), [arXiv:1901.08580](https://arxiv.org/abs/1901.08580).
- [20] C. Hanna, C. Caudill, C. Messick, A. Reza, S. Sachdev, L. Tsukada, K. Cannon, K. Blackburn, J. D. E. Creighton, H. Fong, *et al.*, *Phys. Rev. D* **101**, 022003 (2020).
- [21] T. Adams, D. Buskulic, V. Germain, G. M. Guidi, F. Marion, M. Montani, B. Mours, F. Piergiovanni, and G. Wang, *Classical Quantum Gravity* **33**, 175012 (2016).
- [22] F. Aubin, F. adn Brighenti, R. Chierici, D. Estevez, G. Greco, G. M. Guidi, J. V., F. Marion, B. Mours, E. Nitoglia, *et al.*, *Classical Quantum Gravity* **38**, 095004 (2021).
- [23] G. S. Davis, T. Dent, M. Tápai, I. Harry, C. McIsaac, and A. H. Nitz, *Phys. Rev. D* **102**, 022004 (2020).
- [24] Q. Chu, *Ph.D. thesis*, The University of Western Australia (2017).
- [25] I. Harry, S. Privitera, A. Bohé, and A. Buonanno, *Phys. Rev. D* **94**, 024012 (2016).
- [26] T. Akutsu, M. Ando, K. Arai, Y. Arai, S. Araki, S. Araya, N. Aritomi, Y. Aso, S. Bae, Y. Bae, *et al.*, (2020), [arXiv:2005.05574](https://arxiv.org/abs/2005.05574).
- [27] B. Iyer *et al.*, *LIGO-India Tech. rep.* (2011).
- [28] S. D. Mohanty and S. V. Dhurandhar, *Phys. Rev. D* **54**, 7108 (1996).
- [29] S. D. Mohanty, *Phys. Rev. D* **57**, 630 (1998).
- [30] B. U. Gadre, S. Mitra, and S. Dhurandhar, *Phys. Rev. D* **99**, 124035 (2019).
- [31] C. Devine, Z. B. Etienne, and S. T. McWilliams, *Classical Quantum Gravity* **33**, 125025 (2016).
- [32] R. Prix, *Classical Quantum Gravity* **24**, S481 (2007).
- [33] I. Harry, B. Allen, and B. S. Sathyaprakash, *Phys. Rev. D* **80**, 104014 (2009).
- [34] S. Roy, A. S. Sengupta, and N. Thakor, *Phys. Rev. D* **95**, 104045 (2017).
- [35] S. Roy, A. S. Sengupta, and P. Ajith, *Phys. Rev. D* **99**, 024048 (2019).
- [36] I. W. Harry and S. Fairhurst, *Phys. Rev. D* **83**, 084002 (2011).
- [37] D. Keppel, *Phys. Rev. D* **87**, 124003 (2013).
- [38] D. Keppel, (2013), [arXiv:1307.4158](https://arxiv.org/abs/1307.4158).
- [39] C. Kalaghatgi, P. Ajith, and K. G. Arun, *Phys. Rev. D* **91**, 124042 (2015).
- [40] B. P. Abbott *et al.*, *Classical Quantum Gravity* **35**, 065010 (2018).
- [41] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. E. Creighton, *Phys. Rev. D* **85**, 122006 (2012).
- [42] B. Allen, J. K. Blackburn, P. R. Brady, J. D. E. Creighton, T. Creighton, S. Droz, A. D. Gillespie, S. A. Hughes, S. Kawamura, T. T. Lyons, *et al.*, *Phys. Rev. Lett.* **83**, 1498 (1999).
- [43] A. H. Nitz, *Classical Quantum Gravity* **35**, 035016 (2018).
- [44] R. Biswas, P. R. Brady, J. B. Castell, K. Cannon, J. Clayton, A. Dietz, N. Fotopoulos, L. M. Goggin, D. Keppel, C. Pankow, *et al.*, *Phys. Rev. D* **85**, 122008 (2012).
- [45] A. H. Nitz, T. Dent, T. D. Canton, S. Fairhurst, and D. A. Brown, *Astrophys. J.* **849**, 118 (2017).
- [46] Hierarchical search pipeline, [https://github.com/Kanchan-05/pycbc/tree/Hierarchical-search/hierarchical\\_search](https://github.com/Kanchan-05/pycbc/tree/Hierarchical-search/hierarchical_search) (2021).
- [47] V. Tiwari, *Classical Quantum Gravity* **35**, 145009 (2018).
- [48] R. Sturani, S. Fischetti, L. Cadonati, G. M. Guidi, J. Healy, D. Shoemaker, and A. Viceré, *J. Phys.* **243**, 012007 (2010).
- [49] Ligo scientific collaboration, ligo algorithm library—lal-suite, free software (gpl) (2018).
- [50] S. v. d. Walt, S. C. Colbert, and G. Varoquaux, *Comput. Sci. Eng.* **13** (2011).
- [51] P. Virtanen *et al.*, *Nat. Methods* **17** (2020).
- [52] A. M. Price-Whelan *et al.*, *Astron. J.* **156** (2018).
- [53] A. S. Sengupta, S. V. Dhurandhar, A. Lazzarini, and T. Prince, *Classical Quantum Gravity* **19**, 1507 (2003).
- [54] B. Allen, *Phys. Rev. D* **100**, 124004 (2019).
- [55] A. H. Nitz, C. D. Capano, S. Kumar, Y. F. Wang, S. Kastha, M. Schäfer, R. Dhurkunde, and M. Cabero, (2021), [arXiv:2105.0915](https://arxiv.org/abs/2105.0915).
- [56] T. Venumadhav, B. Zackay, J. Roulet, L. Dai, and M. Zaldarriaga, *Phys. Rev. D* **101**, 083030 (2020).