PHYSICAL REVIEW D **95,** 104045 (2017)

Hybrid geometric-random template-placement algorithm for gravitational wave searches from compact binary coalescences

Soumen Roy,[*] Anand S. Sengupta,[†] and Nilay Thakor[‡]

Indian Institute of Technology Gandhinagar, Gujarat 382355, India

(Received 23 February 2017; published 30 May 2017)

Astrophysical compact binary systems consisting of neutron stars and black holes are an important class of gravitational wave (GW) sources for advanced LIGO detectors. Accurate theoretical waveform models from the inspiral, merger, and ringdown phases of such systems are used to filter detector data under the template-based matched-filtering paradigm. An efficient grid over the parameter space at a fixed minimal match has a direct impact on the overall time taken by these searches. We present a new hybrid geometric-random template placement algorithm for signals described by parameters of two masses and one spin magnitude. Such template banks could potentially be used in GW searches from binary neutron stars and neutron star–black hole systems. The template placement is robust and is able to automatically accommodate curvature and boundary effects with no fine-tuning. We also compare these banks against vanilla stochastic template banks and show that while both are equally efficient in the fitting-factor sense, the bank sizes are ∼25% larger in the stochastic method. Further, we show that the generation of the proposed hybrid banks can be sped up by nearly an order of magnitude over the stochastic bank. Generic issues related to optimal implementation are discussed in detail. These improvements are expected to directly reduce the computational cost of gravitational wave searches.

DOI: 10.1103/PhysRevD.95.104045

I. INTRODUCTION

The direct detection of gravitational waves (GWs) was vigorously pursued by the LIGO Scientific Collaboration (LSC), consisting of several hundred scientists, which culminated with the discovery of the first gravitational wave event GW150914 [1] in the twin advanced LIGO (advLIGO) [2] detectors. This event was determined to have been caused by the inspiral and merger of a spinning binary black hole system of component masses $(36, 29)\mathrm{M}_{\odot}$ located nearly 1.3 billion light years away from Earth. It was also the first observational evidence for the existence of stellar mass black holes. Subsequently, advLIGO detectors also detected a second event, GW151226 [3], from the inspiral and merger of lighter compact objects. These detections mark the transition to the era of gravitational wave astronomy.

Several other kilometer-scale detectors are being upgraded or are under construction at present around the globe—these include the French-Italian advanced Virgo detector [4] and the Japanese cryogenic KAGRA detector [5,6]. In India, the proposal for the advanced LIGO-India detector [7,8] has been approved and is expected to be built over the next few years. The network of these advanced detectors is expected to improve their overall science potential and herald a new wave in astronomy with the potential to observe the very early Universe and complement information gathered by electromagnetic observations.

Binary black holes and neutron stars are considered one of the most promising sources for the advanced terrestrial detectors. Precise theoretical model waveforms for GWs emitted from the inspiral, merger, and ringdown phases of such compact binary coalescences (CBCs) are now available. These models are parametrized by the system's intrinsic properties, such as the component masses, spin, etc., and extrinsic properties, such as sky location, distance to the source, time at coalescence, etc. Accurate theoretical models allow the use of matched-filtering techniques to search for weak GW signals buried in detector noise. As the signal parameters are not known *a priori*, one filters the data using a set of expected signals spanning the deemed parameter space. Each one of these expected signal corresponds to a single point in the parameter space, and are collectively known as the template bank. Coverage of the full range of search parameters using a finite grid of discrete points leads to an inevitable loss in the signal-to-noise ratio (SNR) which can be controlled by fixing the minimal match of the bank. The latter is often decided by striking a balance between desired detection efficiency and the computational cost of carrying out the search.

Advanced detectors are reaching unprecedented sensitivities at low frequencies. Over the last few years, the development of theoretical spinning waveform models has also reached a mature stage. The combined effect of these factors is that one now needs to search over a significantly larger volume compared to initial LIGO era, in a parameter

[*]soumen.roy@iitgn.ac.in
[†]asengupta@iitgn.ac.in
[‡]thakor.nilaysinh@iitgn.ac.in

space spanning three or more dimensions; this poses new challenges for data analysis, which include devising efficient grid-placement strategies. The practical issue of optimal template placement in matched-filter searches is an important open optimization problem [9,10].

There are at least two approaches to solving the problem of efficient template placement: (a) via deterministically placed points through the tessellations of a geometrical lattice [11,12] or (b) via stochastically placed points by choosing them at random [13] over the parameter space. The geometric method requires the metric over the signal manifold and has been used extensively for CBC searches over 2D mass parameters by arranging the grid points in a hexagonal lattice. A variant of the geometrical placement method for aligned spin CBC systems has been explored [14] for the TaylorF2 [15,16] signal model. The lack of availability of metrics and the intricate fine-tuning required to avoid uncovered regions arising from variations in curvature across the parameter space also make it difficult to generalize the geometric placement methods to higher dimensions.

The stochastic template bank is constructed from random proposals drawn from a uniform distribution over the deemed parameter space that are accepted as a new template point only if the new proposal is far from (in the minimal-match sense) existing templates in the bank. Such banks are easy to implement, robust, and can work even without explicit knowledge of the metric on the signal manifold by using brute-force match calculations. This approach has been demonstrated to be more space efficient than a square lattice in 2D but less efficient than a hexagonal lattice. Such banks can be calculated for higher dimensions as well [17].

An earlier attempt to combine the geometric and stochastic methods [18] by seeding a stochastic bank with a prefabricated geometric bank has been demonstrated to improve the efficiency marginally.

We present a new hybrid algorithm for template placement in 3D parameter space (including two mass components and one reduced-spin magnitude) for gravitational wave searches from compact binary coalescence by combining the efficiency of optimal geometrical placement and the robustness and ease of the stochastic placement algorithm. This geometric-random bank placement method uses a local truncated octahedral lattice to place the templates and requires the metric over the signal manifold.

The paper is organized as follows. In Sec. II we recapitulate the fundamentals of the template-based matched-filtering technique used for gravitational wave searches from CBC and review the definition of the metric on the signal manifold with the aim of setting the notation used in the paper and defining key terms. Numerous papers on template placement strategies have been written over the last few years and many new ideas have emerged [13,14,17,18]: we provide a concise review of these efforts for the benefit of the reader. In particular, we elucidate the stochastic algorithm by casting it in two different ways that are algorithmically equivalent. This sets the stage for Sec. III, where we present a new hybrid geometric-random template placement algorithm and give a detailed explanation of the issues concerning its optimal implementation. We argue that, by construction, this new method cannot be less space efficient than the vanilla stochastic placement method. We also construct an explicit template bank for neutron star–black hole searches using this new algorithm and compare it against a vanilla stochastic template bank, noting the improvement in overall bank size as well as the time taken to generate the bank. The TaylorF2RedSpin signal model [19] was used for this purpose. The template banks presented in this work have been calculated using the aLIGOZeroDetHighPower [20] noise sensitivity curve for advLIGO. The set of parameters used to generate the model template banks are listed in Table I.

In Sec. IV, the template banks generated by the hybrid method are tested and validated against the vanilla stochastic bank. We present the fitting factor results using 50 000 signal injections using TaylorF2RedSpin and TaylorF2 aligned-spin waveform models using software implemented in the LALAPPS package of the LIGO Algorithm Library (LAL) [21] and show that the two banks are nearly identical in efficiency. We also compare the hybrid bank against a A_3^*-lattice-seeded vanilla stochastic bank and show that the former is more efficient.

Finally, in Sec. V we summarize the main results and make some comments related to several key issues related to the new method, indicating a possible way for extending it to higher dimensions.

II. METRIC ON THE SIGNAL MANIFOLD

In this section we shall quickly summarize the basics of the template-based matched-filtering technique used in GW searches from CBC. A basic assumption in matched-filtering-based searches is that the astrophysical GW signal buried in the detector noise is faithfully represented by the signal model used in the search over the range of search parameters.

The signal manifold $\boldsymbol{\lambda}$ is the set of all possible GW signals $h(\vec{\lambda}) \equiv h(t;\vec{\lambda})$ characterized by the parameter vector $\vec{\lambda}$. It is customary to represent the corresponding frequency domain signal as $\tilde{h}(\vec{\lambda}) \equiv \tilde{h}(f;\vec{\lambda})$. The detector output $s(t)$ consists of detector noise $n(t)$ and a possible gravitational wave signal of unknown parameters. The additive noise throws the signal out from this manifold to the space of all possible functions. In order to find the point in signal manifold closest to the detector output $s(t)$, the latter is projected over the signal manifold by calculating the maximum likelihood over $\vec{\lambda}$, which serves as the detection statistic. For additive Gaussian noise, the likelihood Λ is given by [22]

$$\Lambda(\vec{\lambda}) = \exp\left\{ \langle s|h(\vec{\lambda})\rangle - \frac{1}{2}\langle h(\vec{\lambda})|h(\vec{\lambda})\rangle \right\}. \quad (1)$$

The inner product $\langle s|h\rangle$ is the complex cross-correlation [23] between the detector output $s(t)$ and the gravitational wave signal $h(\vec{\lambda})$ weighed inversely by the noise power spectral density of the detector [24],

$$\langle s|h(\vec{\lambda})\rangle_{\Delta t} = 4\mathrm{Re}\int_{f_{\mathrm{low}}}^{f_{\mathrm{high}}} \frac{\tilde{s}^*(f)\tilde{h}(f;\vec{\lambda})}{S_n(f)} e^{-2\pi i f\Delta t} df, \quad (2)$$

where $S_n(f)$ is the one-sided noise-power spectral density defined by $\langle \tilde{n}(f)|\tilde{n}^*(f')\rangle = \frac{1}{2}S_n(|f|)\delta(f-f')$, the asterisk denotes the complex conjugation operator, the frequency range $f_{\mathrm{low}} \le f \le f_{\mathrm{high}}$ marks the effective bandwidth of the detector, and Δt is the time delay between these two signals. The SNR ρ after filtering $s(t)$ is defined as

$$\rho(\Delta t;\vec{\lambda}) = \frac{\langle s|h(\vec{\lambda})\rangle_{\Delta t}}{\sqrt{\langle h(\vec{\lambda})|h(\vec{\lambda})\rangle_{\Delta t=0}}}. \quad (3)$$

Without any loss of generality, we assume that all template waveforms are normalized such that $\langle h(\vec{\lambda})|h(\vec{\lambda})\rangle_{\Delta t=0} = 1$. From Eq. (1) it is clear that this allows us to use the log-likelihood function (or, equivalently, the SNR) maximized over the parameters as the detection statistic.

The log-likelihood function can be maximized over all time lags (Δt) by using fast Fourier transform (FFT)–based convolution as shown in Eq. (2). It can be maximized over other extrinsic parameters analytically [23]. On the other hand, as one cannot maximize the log-likelihood function over intrinsic parameters analytically, a brute-force approach is needed. In this case, a discrete grid of points is placed to cover the intrinsic parameter space. One evaluates the log-likelihood surface at each of these points and the maximum is suitably determined.

The template bank consisting of these discrete set of points on $\boldsymbol{\lambda}$ is constructed using a control parameter $\mathcal{M}_{\mathrm{min}}$, commonly known in GW literature as minimal match. This parameter is chosen such that the minimum overlap of an arbitrary vector in the signal manifold (within the deemed parameter space) and at least one template in the bank never drops below this value. The art of template placement lies in maximizing the intertemplate separation without violating this constraint, with the aim of achieving the smallest bank size. In this way, we can map the template placement to the sphere-covering problem [25,26] with spherical cells of radius equal to $\sqrt{1-\mathcal{M}_{\mathrm{min}}}$. One wants the smallest number of overlapping templates (i.e., metric spheres) to fully cover the space (i.e., leave no holes). The equation of such a spherical cell, centered at $\vec{\lambda} \in \boldsymbol{\lambda}$, is given by

$$\langle h(\vec{\lambda})|h(\vec{\lambda}+\Delta\vec{\lambda})\rangle = \mathcal{M}_{\mathrm{min}}. \quad (4)$$

From the normalization of waveforms, it is clear that $\mathcal{M}_{\mathrm{min}} \le 1$. For high values of this parameter, the lhs can be Taylor-series expanded up to leading order in the small parameter $\Delta\vec{\lambda}$,

$$1 - g_{\mu\nu}\Delta\lambda^\mu\Delta\lambda^\nu = \mathcal{M}_{\mathrm{min}}, \quad (5)$$

where $\Delta\lambda^\mu$ are the components of the vector $\Delta\vec{\lambda}$ and

$$g_{\mu\nu} = -\frac{1}{2}\frac{\partial^2\langle h(\vec{\lambda})|h(\vec{\lambda}+\Delta\vec{\lambda})\rangle}{\partial\Delta\lambda^\mu\partial\Delta\lambda^\nu}\bigg|_{\Delta\vec{\lambda}=0} \quad (6)$$

is the metric over the signal manifold, which is essentially the Fischer information matrix projected on the intrinsic parameter space and calculated using standard covariance matrix method [16,27,28]. The match between two nearby points in the parameter space can be calculated easily using the metric. Equation (5) can be rearranged as $g_{\mu\nu}\Delta\lambda^\mu\Delta\lambda^\nu = (1-\mathcal{M}_{\mathrm{min}})$, and identified to be the equation of an ellipsoid in 3D centered at a point $\vec{\lambda}$. For higher dimensions it represents a hyperellipsoid. We shall refer to this as the minimal-match ellipsoid elsewhere in the paper. In GW searches from CBC sources, template banks are usually constructed at $\mathcal{M}_{\mathrm{min}} \sim 0.97$ [24]. This corresponds to a loss in detection rate of ∼9% assuming uniform distribution of such sources.

A. State of the art in template placement

As mentioned earlier, the geometric and stochastic template placement algorithms are two broad classes of methods used in searches for gravitational wave signals from compact binary coalescences.

Previous searches for GW signals from nonspinning compact binaries in initial LIGO and Virgo [29–31] data have used the metric-based geometrical hexagonal template placement in two dimensions [11,12,23,27,28]. Templates are placed in chirp-time coordinates $\{\tau_0,\tau_3\}$ instead of component masses $\{m_1,m_2\}$, since the templates are almost uniformly spaced in the former. This process starts by initializing a template point and then finding neighboring points in a A_2^* hexagonal lattice, where the center of the hexagons represent the position of individual templates. Hexagonal tiling offers the most efficient space filling in two dimensions, which optimizes the number of template points and in turn reduces the total computational cost of the search. To construct this geometric bank, one requires the semi-analytic metric on the signal manifold to be assumed to be slowly varying over the parameter space. The curvature effects leads to some loss of efficiency in this strategy. This geometric placement was initially demonstrated [11] for the inspiral-only waveforms accurate up to second post-Newtonian (2 PN) order in the stationary phase approximation (SPA). At present, the metric for 3.5 PN SPA waveforms [32] is available, which allows template placement for such waveforms as well.

Geometric template placement in higher-dimensional intrinsic parameter space (e.g., component masses and spins) has several problems. First, the metric may not be available for such signal manifolds and second, unlike the hexagonal packing in 2D, optimal geometrical placement in

higher dimensions with curvature is not well known. It is further complicated by curvature and boundary effects. To mitigate the curvature issues, which lead to rapidly changing metric components over the parameter space, new coordinates are being explored in which the metric is slowly varying [17], but these do not solve the problem completely. Recent studies [14,33] have explored geometrical placement in higher dimensions for aligned-spin binary neutron star (BNS) and neutron star–black hole (NSBH) systems for some specific waveform families. In this method, one constructs a metric on the parameterized coordinates, taken to be the coefficients of the 3.5 PN TaylorF2 expansion of the orbital phase [34] instead of the usual chirp-time coordinates. Since the metric is globally flat in these new coordinates, one can globally transform it into a Euclidean coordinate system. Finally, a principal coordinate analysis facilitates the projection to an effective lower-dimensional parameter space, which can be covered by a grid placed in a hexagonal lattice. For NSBH [33] systems, the parameter space is covered by stacking several two-dimensional hexagonal lattices along the direction of the minor axis. This method is available in the PYCBC software package [35–38].

An alternative approach for template placement is the so-called stochastic method [13] where one starts with an empty template bank (or a set of seed points) to which random points, drawn from a uniform distribution over the deemed parameter space, are appended in an iterative fashion. At every iterative step, a new random point is proposed to be included in the template bank: this proposal is rejected if it happens to lie too close to the points that are already in the existing list, otherwise it is accepted. For each accepted proposal, the rejection rate is determined and the process terminates when this rate exceeds a certain threshold averaged over the last few acceptances. We shall call this the bottom-up approach in building the list of templates by considering random proposals one by one, retaining only the valid ones. This has been encoded in the LAL software suite via the program LALAPPS_CBC_SBANK.

The stochastic-bank-placement algorithm can also be cast in a top-down fashion. In this alternate implementation, one starts with an empty template list $\mathcal{T}$ and a list of very large number of proposals $\mathcal{U}$ distributed uniformly over the deemed parameter space. One picks a random point $r \in \mathcal{U}$ and appends it to $\mathcal{T}$, following which all points from $\mathcal{U}$ that lie within the minimal-match distance from r are removed. The process continues in an iterative manner until all points from $\mathcal{U}$ are exhausted. We call this the top-down approach as the template bank is microfabricated out of a large block of random proposals by paring it down to the desired shape and order.

Both these methods are algorithmically equivalent; however, the top-down approach is more useful in projecting a geometric structure over the stochastic template bank leading to the hybrid geometric-random placement algorithm described in this paper. In Table II we demonstrate that the top-down approach is also faster by a factor of $\gtrsim 2$ over the traditional bottom-up approach, as one is able to eliminate many proposals for a single accepted proposal using efficient computational data structures such as binary search trees (BSTs) [39]. The number of templates generated by both these methods is nearly identical.

The stochastic method is relatively easier to implement and does not require the metric over the signal manifold *per se*. The distance between the proposed point and the elements of the current template bank can be directly calculated by evaluating the match inner product Eq. (2). This brute-force approach for match calculation allows it to be extended easily to higher-dimensional parameter spaces and overcome irregular boundary effects. The disadvantages of the stochastic method include the requirement of high computational time, as several million proposals have to be processed to guarantee adequate coverage, and the fact that it generates substantially more templates than the geometric bank. The computational time can be reduced if we use the metric (if available) for match calculation. However, the intrinsic stochastic nature of the algorithm leads to the inefficiency in grid placement.

Another instance of template placement developed for aligned spin binary black hole (BBH) searches has explored a combination of geometric and stochastic approaches [18,40]. In this method, at first one generates an aligned-spin geometric hexagonal lattice template bank up to some valid range of parameters, which is then used as a "seed" bank for the stochastic placement, thereby accelerating the placement. This method generates ∼5.5% fewer template points than the stochastic method and has been used as part of the uber template bank used in CBC searches in the data from the first observational run of aLIGO [41].

III. A NEW GEOMETRIC RANDOM ALGORITHM FOR TEMPLATE PLACEMENT

In this section we present the metric-based hybrid geometric-random template placement algorithm in three dimensions using a truncated octahedral lattice. Such lattices are the Dirichlet-Voronoi polytope of a body-centered cubic A_3^* lattice [42]. The latter provide optimal coverage for conformally flat spaces where the metric coefficients are constant [26]. It is interesting to note that this is in line with Kelvin's conjecture [43], according to which truncated-octahedron-based space filling is optimal in flat 3D space. A truncated octahedron (TO) is a 14-sided space-filling polyhedron and has the highest volumetric quotient [Eq. (B1)], which makes it suitable for the lattice structure. Other geometric properties are tabulated in Appendix B.

Such a template bank would be applicable for gravitational wave searches from the compact binaries described adequately by their component masses and a single "reduced-spin" parameter χ_r using the TaylorF2RedSpin signal model. This signal model is constructed using a post-Newtonian template family of gravitational waveforms

from inspiraling compact binaries with nonprecessing spins. Here, the spin effects are captured by the parameter χ_r, defined as a mass-weighted, linear combination of individual dimensionless spin magnitudes. The details of the signal model are in Appendix A.

In order to construct the three-dimensional geometric random template bank, we require the metric on the parameter space. The metric g_{ij} for the TaylorF2RedSpin approximant varies rapidly in $\{m, \eta, \chi_r\}$ coordinate system, where m denotes the total mass and η denotes the symmetric mass ratio. In order to enhance the efficiency of the algorithm, one places the templates in dimensionless chirp-time coordinates $\{\theta_0, \theta_3, \theta_{3s}\}$, in which the metric components are slowly varying over the parameters. These new coordinates are defined as

$$\theta_0 = \frac{5}{2^{\frac{1}{3}}}\left(\frac{1}{16\pi f_0 m\eta^{3/5}}\right)^{\frac{5}{3}}$$
$$\theta_3 = \left(\frac{16\pi^5}{25}\frac{\theta_0^2}{\eta^3}\right)^{\frac{1}{5}}$$
$$\theta_{3s} = \frac{48\pi}{113\chi_r\theta_3}. \tag{7}$$

However, it is to be noted that the curvature effects do not vanish completely, due to the above coordinate transformations. As such, Kelvin's conjecture does not hold directly in this nonflat space. In the following sections, we show that the truncated octahedral design can still be used for spaces with slowly varying curvature. This is achieved by merging the stochastic bank placement algorithm with a local A_3^* lattice.

The method outlined in Algorithm 1 proceeds by initializing three lists as follows:

(i) $\mathcal{U}$: a list of uniformly distributed random points sprayed over the deemed parameter space;
(ii) $\mathcal{T}$: an empty list for template points;
(iii) $\mathcal{K}$: an empty temporary list.

Algorithm 1 Geometric-Random Template Placement

```
1: generate 𝒰
2: 𝒦 = []
3: 𝒯 = []
4: while (𝒰) do
5:     𝒦 ← random point p ∈ 𝒰
6:     while 𝒦 do
7:         𝒯 ← append 𝒦[0]
8:         find all possible TO neighbors of 𝒦[0]
9:         𝒦 ← append valid TO neighbors
10:        delete 𝒦[0]
11:    end while
12:    delete all minimal-match neighbors of 𝒯 from 𝒰
13: end while
```

At first $\mathcal{K}$ is initialized or seeded with a random point chosen from $\mathcal{U}$ which is immediately appended to the list of templates $\mathcal{T}$ as its first element. The possible 14 TO neighbors of this initial point are then calculated and appended to $\mathcal{K}$, followed by the removal of the first element. Only those TO neighbors which fall within the parameter space and are farther than $\mathcal{M}_{\rm min}$ from existing points in $\mathcal{T}$ and $\mathcal{K}$ can be appended. This completes one iteration of the inner loop of the algorithm. We continue these steps until $\mathcal{K}$ becomes empty. The latter happens when no new valid TO neighbors are appended in successive iterations of the inner loop of the algorithm, which ultimately exhausts all the points in $\mathcal{K}$. At this stage, all points from $\mathcal{U}$ that are within the minimal-match distance from the elements in $\mathcal{T}$ are deleted, following which $\mathcal{K}$ is reseeded with a new random point from $\mathcal{U}$. Termination of template placement algorithm occurs when $\mathcal{U}$ is exhausted.

The advantage of casting the stochastic template placement algorithm (see Sec. II A) in a top-down fashion is clear when we consider the extreme case where no TO lattice neighbors can be found for any proposal point in $\mathcal{U}$. In this case, it is clear from Algorithm 1 that the geometric-random algorithm naturally falls back to the vanilla stochastic placement algorithm. This brings in the robustness of the latter to the proposed new algorithm. This contingency arises for a small fraction (∼5%–10%) of points in $\mathcal{U}$, due to boundary edge effects and small uncovered patches arising from curvature effects. By construction the geometric-random placement method presented here is more efficient than vanilla stochastic algorithm.

Also note that the TO, whose neighbors are added to $\mathcal{K}$ in lines 8–9 of Algorithm 1, refers to the one inscribed inside the minimal-match ellipsoid given in Eq. (5) above. The gradual change in curvature (changing metric components) leads to changing orientation and size of these ellipsoids, which affects the size and orientation of the inscribed truncated octahedron. This, in turn, affects the coordinates of its 14 neighbors. In this way the placement algorithm automatically responds to the curvature effects. This is equivalent to assuming flat local patches of the signal manifold, which can be optimally covered by A_3^* lattice.

A. Implementation

We now highlight some salient features for efficient implementation of the algorithm.

1. Initializing $\mathcal{U}$ with uniform random points

In order to generate a random list of points over the parameter space, at first we calculate the minimum and maximum possible values of dimensionless chirp times θ_0, θ_3, and θ_{3s} for the range of parameters over which the bank is to be placed. A set of random points are generated in $\{m_1 - m_2\}$ space along the boundaries, marking the constraints for component masses, chirp masses, and mass ratios. Afterwards, a coordinate transformation is taken to the dimensionless chirp-time coordinate system $\{\theta_0 - \theta_3\}$, which allows the calculation of the extremities in these

coordinates. The extreme values of θ_{3s} are evaluated using $|\chi_r| \leq 1$ in Eq. (7). Once these extremities are known, uniform random points are proposed within these values, out of which only those which lie within the specified mass and spin ranges are retained.

The final bank size depends strongly on the initial density of points in $\mathcal{U}$ up to a critical value, beyond which it does not change very much. Empirically, we find this number to be about $\sim 1/10$ of the total volume calculated in the dimensionless coordinates. The latter depends on the metric and varies across the parameter space, due to which we estimate an average cell volume by sampling different points across this space. The total number of cells needed to cover the space is given by the ratio of the total volume divided by the average volume of a cell. A Monte Carlo integration method can be used to estimate the total volume.

2. Deletion of points within the minimal-match ellipsoid

We discuss an efficient way to delete points from $\mathcal{U}$ that lie within minimal-match distance from the template bank points. For a fixed minimal-match value, Eq. (5) represents the surface of an ellipsoid with semiaxis length

$$A_i = \sqrt{\frac{1 - \mathcal{M}_{\rm min}}{e_i}}, \tag{8}$$

where the e_is are eigenvalues of the metric g_{ij} evaluated at the ellipsoid center. The orientation of the ellipsoid is defined by matrix $\mathcal{R}$ composed of eigenvectors of the metric g_{ij}, where $\mathcal{R}_{ij}$ is jth component of the ith eigenvector. To visualize the deletion process, let us consider a cell centered around a template. The points in $\mathcal{U}$ lying inside the cell can be determined by calculating the match between this template with the all other points in $\mathcal{U}$, but such a brute-force approach will not be computationally efficient. We need to find a smaller set of points against which we can identify these neighbors more efficiently. To this end, we can first use a BST [39,44,45] data structure to identify a smaller list of points from $\mathcal{U}$ that lie within a sphere of radius equal to the largest semiaxis of the cell. From this smaller list of points, we then proceed to delete the ones that satisfy $g_{ij}\Delta\lambda^i\Delta\lambda^j \leq 1 - \mathcal{M}_{\rm min}$.

This can be further refined by binning the template points in θ_0 such that each bin contains about ~ 1000 templates. For each such subset, the removal of points from $\mathcal{U}$ proceeds by identifying those points that lie in the same bin (within some acceptable margin in θ_0) and then applying the above strategy to eliminate the points. This refinement is made possible due to the fact that match values decrease monotonically with the difference in parameters and are most sensitive to changes in θ_0. Additional binning in the other two coordinates may possibly further improve the computational efficiency of removing the points from $\mathcal{U}$.

3. Global coordinate transformation

In dimensionless chirp-time coordinates, most of the ellipsoidal cells have a ratio of semiaxes around $1000:10:1$. This implies that the number of points inside this flat and elongated ellipsoid is a small fraction of the total number of points contained in the sphere with radius equal to the semimajor axis. This undermines the efficiency of deleting the points using the technique outlined in Sec. III A 2. We use a conformal coordinate transformation in such a way that one of the cells transforms to a unit sphere. Note that, due to curvature effects, the same transformation when applied to other cells does not guarantee them to change to unit spheres, but mitigates the problem of a highly asymmetric semiaxis ratio to a large extent. We use the eigenvalues and eigenvectors of the metric at a fiducial point to transform the coordinates ($\theta_k \to \xi_k$) as

$$\xi_i = \sum_{j,k=1}^{3} \mathcal{S}_{ij}\mathcal{R}^T_{jk}\theta_k, \tag{9}$$

where $\mathcal{S}_{ij} = \sqrt{e_i}\delta_{ij}$ is a diagonal scaling matrix and the rotational matrix $\mathcal{R}_{ij}$ is constructed from the ith component of the jth eigenvector of the metric calculated at the fiducial point. This is carried out by first binning the templates in θ_0. The transformation in Eq. (9) is carried out using the metric of the center point in each bin.

The metric at all other points transforms as

$$\bar{g}_{ij} = \sum_{p,q,r,t=1}^{3} \mathcal{S}^{-1}_{ip}\mathcal{R}^T_{pq}g_{qr}\mathcal{R}_{rt}\mathcal{S}^{-1}_{tj}. \tag{10}$$

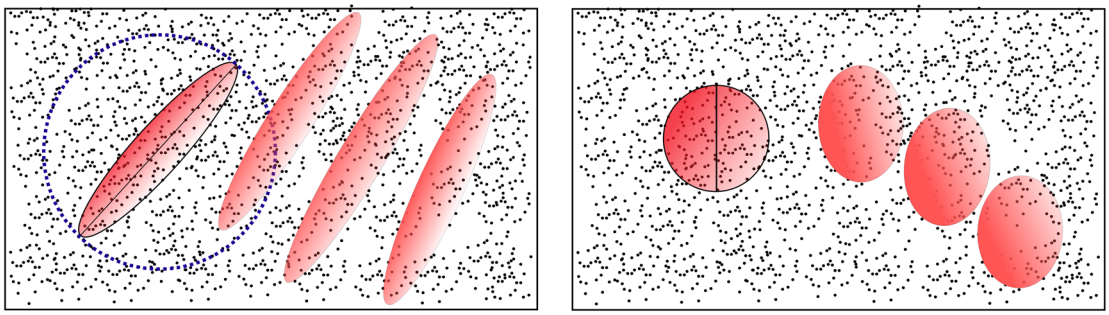
It is trivial to check that the metric at the fiducial points transforms to a 3×3 identity matrix.

This global coordinate transformation Eq. (9) causes other nearby cells to become almost spherical with semiaxes in the ratio $\sim 1:1:1$. In Fig. 1 we show the effect of global coordinate transformation on nearby cells. This is doubly advantageous: not only does the ball-point query volume for the BST searches decrease (leading to fewer points to deal with); the points that lie within the inscribed minimal-match ellipsoid now occupy a large fraction of its volume, leading to increased efficiency in the removal of points.

4. Locally placed truncated octahedral lattice

The template placement problem in 3D can be mapped to the tiling problem such that a minimum number of similar cells is used without leaving any region uncovered. As such, a truncated octahedral lattice becomes a natural choice for this problem.

As stated earlier, the TO shares 14 faces with its neighbors in the lattice. Assuming a TO inscribed in a sphere of radius R, the coordinates of its 14 lattice neighbors N^p_k are available



(a) Before global transformation (b) After global transformation

FIG. 1. An example of global transformation Eq. (9) to speed up the point elimination part using a BST algorithm. Here we construct the global transformer using the metric of an ellipse, which is inscribed in the blue dashed circle, as shown in the left panel. After the transformation the same cell becomes a unit sphere and nearby cells also become more spherical.

in Table IV. Here $k = (1, 2, 3)$ is an index on the coordinates of the pth neighbor. When mapped to the template placement problem, we need to consider TOs inscribed within the minimal-match ellipsoids. In this case, the coordinates of the lattice neighbors $\bar{N}_i^p$ can be calculated using the rotation and scaling matrices,

$$\bar{N}_i^p = \sum_{j,k=1}^{3} \mathcal{R}_{ij}^T \mathcal{S}_{jk} N_k^p. \tag{11}$$

Figure 2 shows the 14 neighbors of a TO inscribed in a elliptical cell.

Because of boundary effects, all 14 TO neighbors of a point need not necessarily be part of the template bank. The following conditions must be checked for:

(i) The point is inside the deemed parameter space and also satisfies $\eta \leq 1/4$. The latter corresponds to the condition that θ_ks can be inverted to yield physical masses.
(ii) The point is not within the minimal-match distance of existing points in $\mathcal{T}$ and $\mathcal{K}$.

Check (b) above ensures that we do not double count the neighbors.

As shown in Algorithm 1, we start from a random point in the parameter space (by seeding $\mathcal{K}$) and tessellate with a local TO lattice. Because of curvature and boundary effects, it is not guaranteed that these tessellations cover the entire parameter space. This is marked by the exhaustion of $\mathcal{K}$ as the placement proceeds. At this stage, we need to reseed $\mathcal{K}$ and continue the process iteratively until all points in $\mathcal{U}$ have been used up.

5. Choice of the initial point and variations in bank size

The template placement algorithm starts from a randomly chosen point in the parameter space by seeding $\mathcal{K}$, which is copied over as the first element of the template bank list $\mathcal{T}$.

One can start from any point in parameter space: in the present work we have started from a "midpoint" corresponding to component masses $m_{1,2} = (m_{1,2}^{\max} + m_{1,2}^{\min})/2$ with individual spins $\chi_{1,2} = (\chi_{1,2}^{\max} + \chi_{1,2}^{\min})/2$. We have checked that starting from other points (e.g., the extremities of the parameter space) also work quite well. This initial choice results in a minor fluctuation of the template bank size, and it is recommended that we start from a point well inside the deemed parameter space where the local variations in curvature are less. This ensures maximum tiling before we need to reseed $\mathcal{K}$.

We demonstrate these fluctuations in template bank size by constructing several template banks for compact binary systems whose bank parameters are given in set II of Table I, starting from different locations in the deemed parameter space. The random seed used to initialize $\mathcal{U}$ was kept the same to eliminate bias. The five different starting points were taken to be the center and extremities of the parameter space, respectively, and as expected different choices of the initial seed point resulted in a slightly different number of templates in the final bank. We generated an average of 111 257 templates with ∼3% fluctuation, which is quite insignificant.

IV. TEST OF THE ALGORITHM

In this section we construct a template bank using our algorithm and demonstrate its validity for CBC searches. We

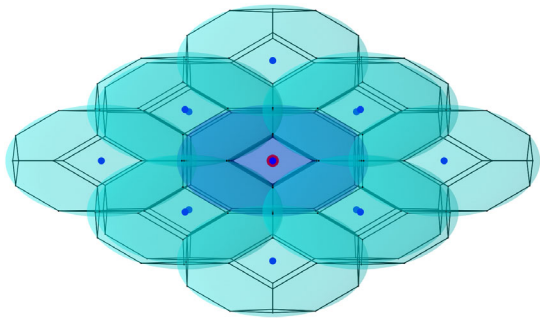


FIG. 2. Lattice neighbors of a truncated octahedron inscribed in a elliptical cell. Here only nine neighbors are visible and the remaining five are on the opposite side.

TABLE I. Parameters used to generate the geometric-random and vanilla stochastic banks. The results for different sizes of $\mathcal{U}$ are summarized in Fig. 4. In set I, the parameter space is chosen by satisfying the NSBH boundary mass; i.e., components with individual mass $\leq 3M_\odot$ are identified as NS with dimensionless spin magnitude in the range ± 0.4.

Bank parameter	Set I	Set II
Waveform model	TaylorF2RedSpin	TaylorF2RedSpin
Noise model	aLIGOZeroDetHighPower	aLIGOZeroDetHighPower
Lower cutoff frequency, $f_{\rm low}$	20 Hz	30 Hz
Higher cutoff frequency, f_{high}	2048 Hz	1024 Hz
Mass of first object, m_1	$[1, 20]M_\odot$	$[3, 15]M_\odot$
Mass of second object, m_2	$[1, 3]M_\odot$	$[1, 3]M_\odot$
Spin of first object, χ_1	$[-0.98, 0.98]$	$[-0.6, 0.6]$
Spin of second object, χ_2	$[-0.4, 0.4]$	$[-0.05, 0.05]$
Size of $\mathcal{U}$	$[1-8]\times 10^7$	1×10^7
Minimal match, $\mathcal{M}_{\rm min}$	0.97	0.97

also compare its performance against a bank generated using the vanilla stochastic method available in the LSC Algorithm Library suite (LALSuite) [21]. We need a metric over the dimensionless chirp coordinates; at present, two such models are available for use, namely, the TaylorF2RedSpin and IMRPhenomB signal models [46]. Our demonstration makes use of the TaylorF2RedSpin model.

A. Construction of the template banks for a reduced-spin binary system

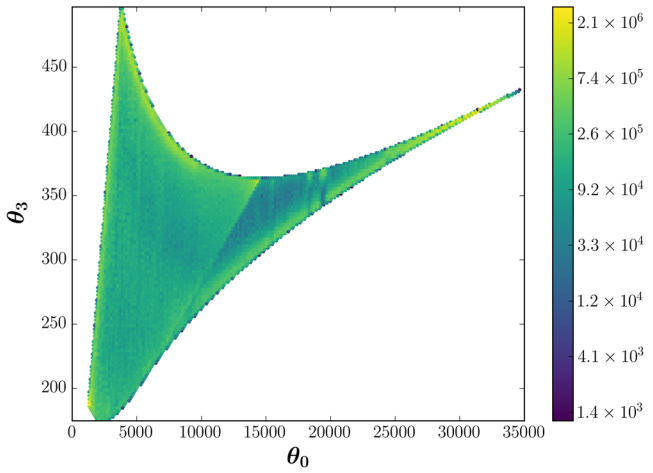
We generate a template bank using the TaylorF2RedSpin metric in $(\theta_0, \theta_3, \theta_{3s})$ parameter space for an aligned-spin compact binary system. The range of parameters is chosen such that the mass of the first object lies between $1M_\odot$ and $20M_\odot$, with dimensionless spin magnitude in the range ± 0.98. The mass of the second object is taken between $1M_\odot$ and $3M_\odot$, with dimensionless spin magnitude in the range ± 0.4. The NSBH boundary mass condition is satisfied; i.e., any object with mass $\leq 3M_\odot$ is considered to be a neutron star with a corresponding limit on spin magnitude. This template bank can be used for BNS and NSBH searches.

We construct several template banks by varying the number of uniformly sprayed random points over the parameter space $\mathcal{U}$. We consider sizes of $\mathcal{U}$ varying between 1×10^7 and 8×10^7 points. In all the cases, the placement proceeds from a point corresponding to individual masses $m_{1,2} = (\frac{21}{2}, \frac{4}{2})M_\odot$ and individual spin magnitudes $\chi_1 = \chi_2 = 0.0$. The full specification of input arguments for template generation are given in set I of Table I.

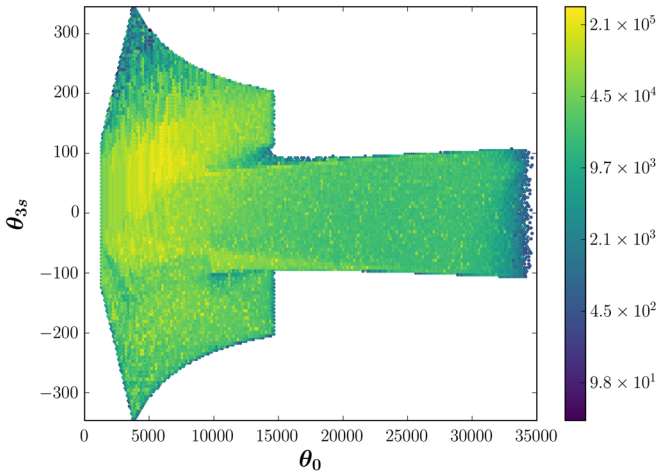
The corresponding final template bank sizes are listed in Table II. In Fig. 3 we show the two-dimensional projections of such a bank along the $\hat{\theta}_{3s}$, $\hat{\theta}_3$, and $\hat{\theta}_0$ directions, respectively. The hat $\hat{(\,)}$ over the symbols denotes the unit normal vector along the specified direction. For comparison, we also generated a template bank using

TABLE II. Summary of various template banks mentioned in this paper. The semianalytic metric g_{ij} for the TaylorF2RedSpin signal model was used in all cases. The vanilla stochastic algorithm can also work by directly calculating matches (instead of using g_{ij}), but in this head-to-head comparison, we have used the semianalytic metric, which speeds up the vanilla stochastic template placement significantly. The usage of the metric is compulsory for geometric-random placement.

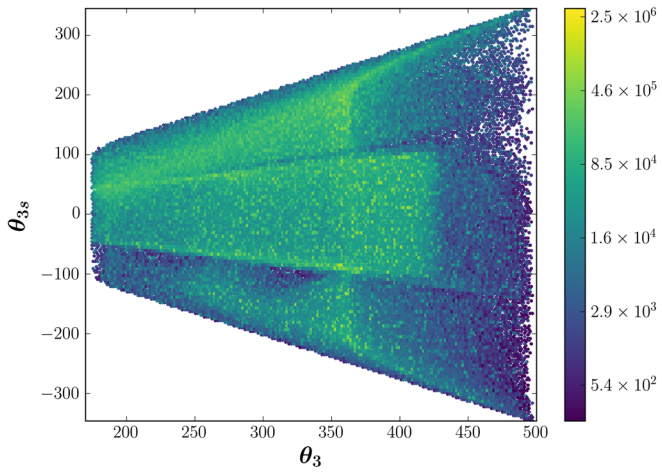
Bank parameters	Placement algorithm	Size of $\mathcal{U}$	Bank size	Execution time (min)	Comments
Set I of Table I	Hybrid construction	1×10^7	694 422	375	25% more templates
	Geometric random	2×10^7	749 705	482	$\times$(8–10) faster
		3×10^7	777 113	616	
		4×10^7	798 269	885	
		5×10^7	812 570	990	
		6×10^7	824 541	1191	
		8×10^7	843 177	1712	
	Vanilla stochastic (LALAPPS_CBC_SBANK)	...	939 787	3666	
Set II of Table I	Hybrid construction	1×10^7	107 547	69	25% more templates
	Geometric random				$\times$11 faster
	Vanilla stochastic (LALAPPS_CBC_SBANK)	...	134 563	762	
	A_3^*-seeded stochastic	...	128 185	...	
	Vanilla stochastic (TOP_DOWN)	...	134 426	320	$\times$2.3 faster



(a) Projection along θ_{3s}



(b) Projection along θ_3



(c) Projection along θ_0

FIG. 3. Area-normalized histograms of the template density in various planes. The template bank was constructed using hybrid geometric random algorithm presented in this paper. Each bin of the histogram was normalized by the square root of the determinant of the metric $\sqrt{|g_{ij}|}$ to ensure equal area. The metric was calculated at the bin center. The boundary effects are clearly seen. We can also see that bank is highly elongated along $\hat{\theta}_0$ direction as compared to both $\hat{\theta}_3$ and $\hat{\theta}_{3s}$ directions.

the vanilla stochastic method for the same parameters listed in set I of Table I. The vanilla stochastic bank placement was terminated at a point when there were 1000 rejected proposals for each accepted proposal averaged over the last 10 acceptances. This stochastic bank was found to contain 939 787 templates, which is ∼25% larger than the geometric-random bank. The computational run time was also recorded; we observed that the geometric-random method took 482 minutes while the vanilla stochastic method took 3666 minutes to execute on a single, unloaded processor, which is ∼8 times faster.

B. Validation of the template banks

We investigate the performance of both the geometric-random and the vanilla stochastic template banks against a set of signal injections from the reduced-spin TaylorF2RedSpin signal model. In this section we summarize the results of this comparison and demonstrate that the two banks are nearly identical in performance.

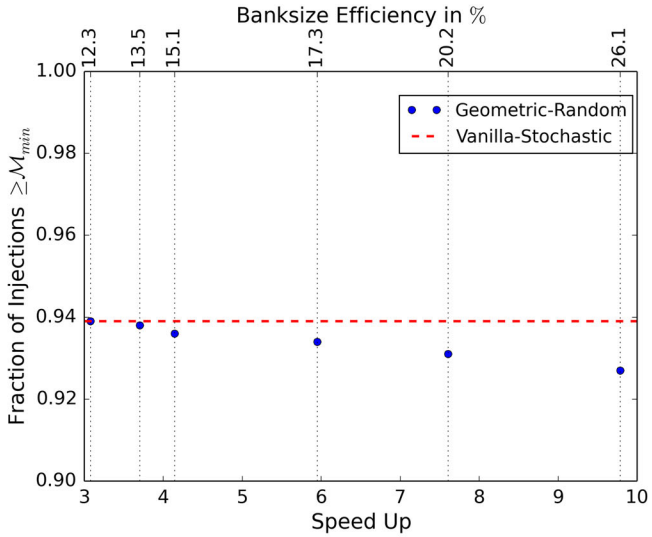
Following Apostolatos [47], the "fitting factor" $\mathcal{FF}(h_a)$ is defined as a measure of the maximum match over the template bank $\mathcal{T}$ for a putative injected signal h_a,

$$\mathcal{FF}(h_a) = \max_{h\in\mathcal{T}} \langle h_a | h \rangle. \tag{12}$$

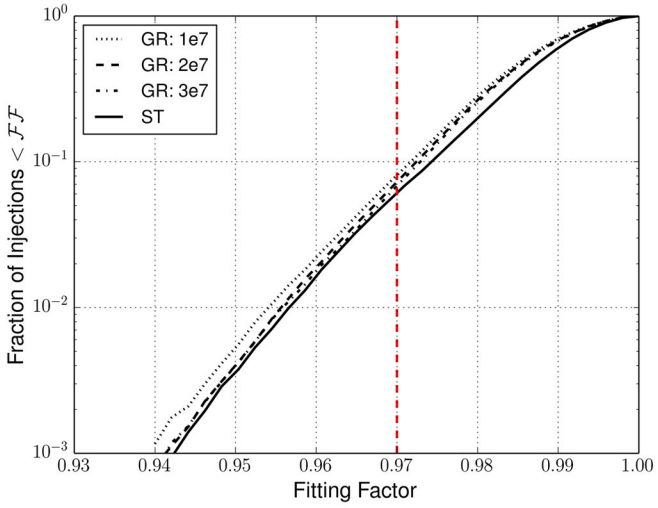
The parameters of the injected signal (chosen from within the deemed parameter space over which the bank is placed) is chosen at random and may not coincide with that of a template point. The mismatch $[1 - \mathcal{FF}(h_a)]$ indicates the fractional loss of optimal SNR. Because differences between true waveforms and waveform models will always reduce the fitting factor, banks are usually tested to achieve fitting factors slightly larger than the

minimal match used in their construction. The minimal match at which the template banks are constructed (0.97 in this case) is somewhat arbitrary, and is arrived by carefully balancing the computational cost of the search against the desired detection efficiency.

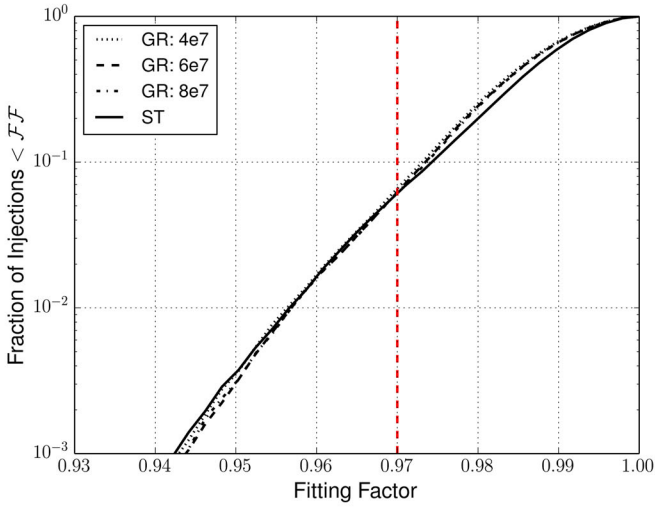
For the banks generated per the parameters in set I of Table I, we injected 50 000 signals from reduced-spin TaylorF2RedSpin waveform model and calculated the fitting factor using Eq. (12). The LALAPPS_CBC_SBANK_SIM program as implemented in LALAPPS package of the LALSuite [21] was used for this calculation. The range of mass and spin parameters of the injected signals were chosen to be the same as that of the bank and were drawn from a uniform distribution in this range. Other extrinsic parameter were drawn from (i) a uniform random distribution over all possible sky locations, (ii) a fixed inclination angle corresponding to edge-on orientation of the plane of the binary, and (iii) a fixed luminosity distance of 1 Mpc. For the banks (corresponding to different initial sizes of $\mathcal{U}$), we found that ∼92%–94% signals were recovered with fitting factor



(a) Comparison of hybrid bank performance

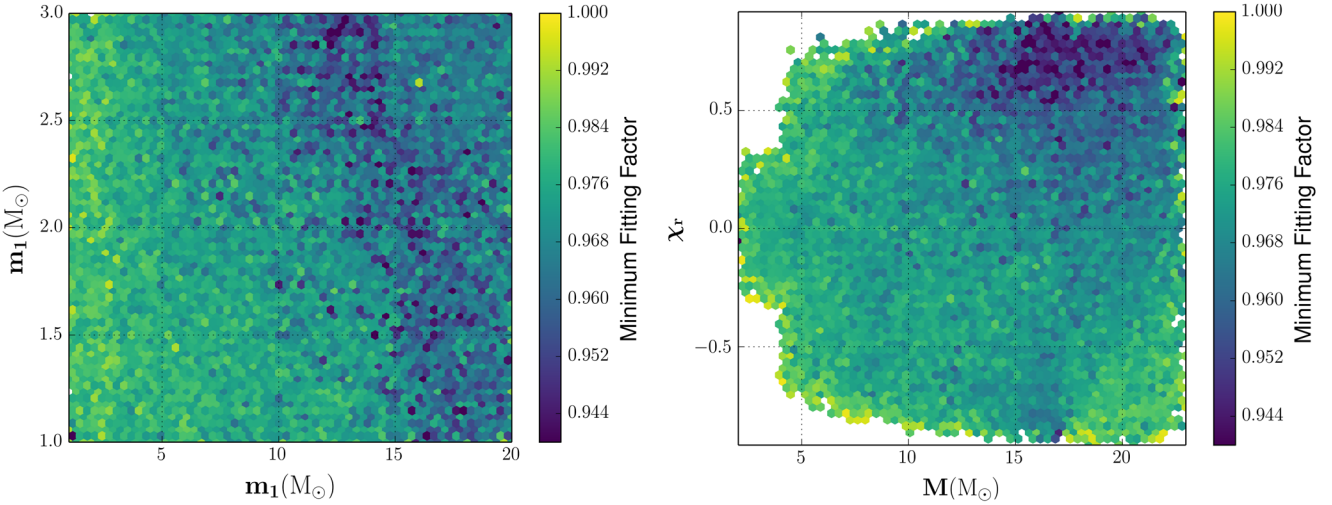


(b) Fitting factor distribution for hybrid bank



(c) Fitting factor distribution for hybrid bank

FIG. 4. Panel (a) shows a comparison of the geometric-random bank and the vanilla stochastic bank constructed over identical parameter ranges is made by plotting fitting factors for 50 000 TaylorF2RedSpin signals. The horizontal line depicts the percentage of such injections for which the fitting factor is above the bank minimal match in the case of vanilla stochastic bank. The solid dots correspond to the percentage for geometric-random banks constructed with different sizes of $\mathcal{U}$ as tabulated in Table II. The bottom horizontal axis measures the computational speed-up of the hybrid geomtric-random placement while the one on top shows the corresponding efficiency in bank size. Panels (b) and (c) show the cumulative distribution of the hybrid bank fitting factors for different initial sizes of $\mathcal{U}$. They have been split into two panels for clarity.



(a) over component masses (b) over total mass and reduced spin parameters

FIG. 5. (a) The minimum fitting factor for the hybrid template bank for a set of injected aligned-spin NSBH and BNS signals as a function of component masses. (b) The minimum fitting factor as a function of the reduced-spin parameter and the total mass. Both the signal and template waveforms are modeled using the TaylorF2RedSpin approximant. We see a region showing poor coverage corresponding to high total masses and high reduced-spin values. This is seen for the stochastic bank as well. One possible reason could be that the TaylorF2RedSpin metric could be doing worse for systems with high masses and high reduced spin.

≥ 0.97 for both the geometric-random bank as well as for the stochastic bank. The results are shown in Fig. 4. The fluctuation at higher significant digits was ignored due to the limited sample size of injections. These results are graphically depicted in Fig. 4 and demonstrate the equivalence of the two banks along with the computational speed-up and corresponding efficiency of template placement using the new method.

Although the injection signal model is identical to the one used to construct the template banks, we still find that the fitting factors fall below the minimal match of the bank for a small fraction of the injections. This alludes to the fact that both template banks have holes in certain regions of the deemed parameter space where neighboring templates do not provide adequate coverage. This arises from curvature effects.

The fitting factor depends on the parameters of the injected signals, e.g., masses, spin components, sky position, and inclination angle of the binary's plane to the line of sight, etc. For nonprecessing signals including only the dominant radiation mode, the fitting factor depends only on the intrinsic parameters. To understand the systematics, it is convenient to represent it as a function of two parameters while averaging out over the remaining ones.

In Figs. 5(a)–(b), we show the histogram of "minimum" fitting factors [33] (for a geometric stochastic template bank) over various combinations of intrinsic parameters of the compact binary system, where both the signals and template waveforms are generated from the TaylorF2RedSpin waveform model. The bank performs well to match the injected signals throughout the parameter space for NSBH and BNS systems, except for those regions where both the total mass and reduced spins are high.

In order to carry out a high-precision test comparing the efficiency of the hybrid bank with that of the vanilla stochastic bank, we need a large number of injections to calculate the fitting factors up to high significant figures. To this end, we construct both the template banks using the TaylorF2RedSpin metric and parameters given in set II of Table I. The geometric random bank was constructed by initializing $\mathcal{U}$ with 1×10^7 uniform random points in dimensionless chirp-time coordinates, whereas the stochastic bank code was set to terminate when the rejection rate reached a value 0.9996 averaged over the last 10 acceptances. The geometric random bank was found to contain 107 547 templates whereas the stochastic bank contained (25% more) 134 563 template points. In this case, the geometric stochastic bank took $\gtrsim$11 times less time than the vanilla stochastic bank on a single unloaded processor.

We quantify the performance of these two template banks by computing fitting factors for two different injection families of aligned spin waveforms, TaylorF2RedSpin and TaylorF2. In both cases, 50 000 injections were made where the intrinsic parameters of the injected waveforms were chosen from set II of Table I and other parameters chosen as earlier. For the case of the geometric-random template bank created using local TO lattice, we found 0.354% of signals were recovered below a fitting factor of 0.97 for TaylorF2RedSpin injections, and 2.892% of signals are recovered below this level for TaylorF2 injections. In the latter case, the injection model is different from the one used to construct the template bank; hence, it is expected that the coverage for TaylorF2 will be less. The corresponding numbers for the vanilla-stochastic bank are found to be 0.514% (TaylorF2RedSpin injections) and 2.408%

(TaylorF2 injections) respectively. From these numbers it is evident that the geometric-stochastic bank is equally efficient as the vanilla-stochastic bank. These results are summarized in Fig. 6.

As mentioned earlier, the A_3^* lattice provides optimal coverage for conformally flat spaces in 3D. One may be tempted therefore to use such a lattice as a seed for stochastic template placement. We have investigated this approach and compared it with the geometric-random and vanilla stochastic algorithms. At first, the deemed parameter space (corresponding to parameters in set II of Table I) was covered by a A_3^* lattice using the metric at a putative point to determine the dimensions of the unit cell. Using a point in this volume for which the unit cell had the smallest dimension, we placed 31 732 A_3^* lattice points to entirely cover the deemed parameter space. Using these as seed points for stochastic placement, the final bank size was found to have 128 185 templates. This is marginally ($\sim$5%) smaller than the vanilla stochastic bank, which has 134 563 templates as reported above. The geometric-random bank outperforms this A_3^*-seeded stochastic bank by a good margin of more than 16%.

As remarked earlier, the vanilla stochastic algorithm can be cast in two different ways. The traditional bottom-up approach has been implementation in the LSC Algorithm Library and has been used in this work for comparison with the geometric-random bank. In order to compare it head to head with the top-down approach, we implemented it in software and ran it for the exact same parameters as given in set II of Table I. As expected, the top-down implementation gives nearly identical bank sizes (0.1% difference in size) but takes less than half the time as the bottom-up LAL implementation. The computational speed comes from the fact that in this implementation one can reject many proposals that lie within the minimal-match ellipsoid centered around a single accepted proposal. Efficient computational data structures like binary search trees are readily available for such queries.

A summary of the various templates banks referred to in the above discussion is available in Table II.

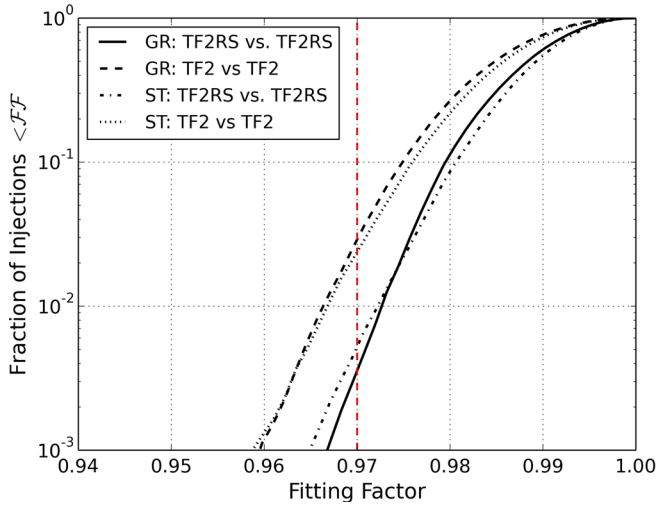


FIG. 6. Fitting factors computed for various sets of aligned-spin signal families against geometric-random (GR) and vanilla stochastic template (ST) banks (see set II of Table I for parameters). The performance of the geometric-random bank when both template and injected signals are generated from the TaylorF2RedSpin (TF2RS) approximant (black solid line) and when both are generated from the TaylorF2 (TF2) approximant (black dashed line) are shown. The performance of the vanilla stochastic bank when both the template and injected signals are generated from the TaylorF2RedSpin approximant (black dashed-dot line) and when both are generated from the TaylorF2 approximant (black dot line) are also shown. In this case, the vanilla stochastic bank has 25% more templates than the geometric random bank and can be placed about $\gtrsim$11 times faster in time.

V. DISCUSSION AND OUTLOOK

Templated matched filtering is the mainstay of gravitational wave detection pipelines. With unprecedented improvement in low-frequency sensitivity of advanced detectors, and the availability of theoretical spinning waveform models, it has become imperative to conduct these searches over increasingly larger volumes in higher-dimensional parameter spaces. For such cases, the stochastic algorithm is used for template placement as it is easily scalable to higher dimensions. However, it is computationally expensive and, by design, not the most space efficient.

This paper introduces a new template placement algorithm in 3D with an attempt to combine the space efficiency of A_3^* lattice with the robustness of a stochastic placement algorithm. Such a template bank can be used in gravitational wave searches from binary neutron stars and neutron star–black hole compact binary systems, where the waveform is described by two mass parameters and a mass-weighted spin magnitude parameter providing coverage for aligned-spin systems.

The truncated octahedron (a Dirichlet-Voronoi polytope of the A_3^* lattice) inscribed within the minimal-match ellipsoid is used as a unit cell for the geometric placement. Such lattices are known to provide optimal coverage for conformally flat 3D spaces. While the signal manifold is not globally flat, one can assume local flat patches and use such cells to cover them optimally. The interface to the stochastic placement algorithm is made by spraying random points over the parameter space, which are removed if found within the minimal-match ellipsoid of any template. We discuss how this merger of methods is able to handle the issues arising out of varying curvature and irregular boundaries. The nuances of its optimal implementation are discussed in detail. In Table II, we show by a direct comparison with a stochastic algorithm that the new method generates significantly fewer templates and is computationally more efficient. We now make a couple of comments related to the new template placement algorithm presented in this paper.

One of the key issues with the geometric template placement algorithms (e.g., geometric hexagonal placement

[11]) is related to the amount of fine-tuning needed in the method to account for curvature effects. For example, in 2D the hexagonal cells used to place templates need to be whittled or pared to a slightly smaller size so that no holes are left due to the changes in relative orientation of neighboring hexagonal cells. This reduces the overall efficiency of placement, leading to an increase in template bank sizes. More importantly, one loses the ability to make generic codes that are usable for different waveform models. In this regard, the geometric-random algorithm proposed here is robust against such fine-tuning by design. The template placement proceeds by first spraying a large number of random points over the parameter space which are later removed if found within the minimal-match ellipsoid of the templates in the bank. Suppose a small portion of the deemed space is left uncovered due to curvature effects; this would lead to some residual points that are not removed from $\mathcal{U}$. As evident from Algorithm 1, these points are revisited in subsequent iterations where a random point (out of the residuals) is added to the template bank, leading to complete coverage.

We would like to point out that the template banks shown in this paper make use of the TaylorF2RedSpin waveform family, which model only the inspiral part of the evolution of the compact objects. They are not ideal for BBH searches directly, where a significant part of the SNR is expected to be contributed from the merger and ringdown phases. However the method developed in this paper is quite general and can be used for placing templates for inspiral-merger-ringdown (IMR) waveform families as well. If analytical metrics for IMR approximants (IMRPhenomD [48,49], SEOBNRv2 [50,51], etc.) were available, the proposed method would be an elegant and efficient solution for covering the entire combined BNS + NSBH + BBH space. This will thereby mesh well with modern LIGO searches, which are carried out over a large range of parameters with combined BNS + NSBH + BBH banks using a mix of inspiral-only and IMR templates.

The assumption that the metric on the signal manifold is slowly varying and is locally flat is crucial for space efficiency. One can imagine a hypothetical case where this assumption does not hold true (e.g., where metric coefficients are random at every point), in which case the geometric-random algorithm will effectively fall back to the top-down version of the vanilla stochastic template placement by design. In other words, the number of templates in the bank from the new method will not exceed the stochastic template bank in the limiting case. We have also shown in a direct comparison that the top-down stochastic bank implementation is computationally more efficient and should be used where the metric is available. Incorporating an intelligent way of spraying the random proposals (instead of drawing them from a uniform distribution) over the parameter space may lead to further optimization of this method.

On the other hand, if the metric was perfectly flat (and given in coordinates where it was constant), the hybrid construction would fall back on a perfect A_3^* lattice. Finally, the geometric-random placement method presented here for 3D is generically scalable to signal manifolds in higher dimensions by using the appropriate $A_n^*(n > 3)$ lattices.

ACKNOWLEDGMENTS

S. R. thanks IIT Gandhinagar for research fellowship and fellow graduate students (Amit Reza, Chakresh Singh, and Md. Yousuf) for useful discussions and help with the manuscript. S. R. also would like to thank Amit Kumar and Niladri Naskar for help in preparing the manuscript. We thank Samarth Vaijanapurkar for useful discussions. The authors thank Duncan Brown for valuable feedback. Last but not least, the authors thank the anonymous referees whose careful reading of the manuscript, detailed queries, and many suggestions have helped to improve the content and presentation of this work.

APPENDIX A: THE TaylorF2RedSpin SIGNAL MODEL FOR GRAVITATIONAL WAVES FROM INSPIRALING COMPACT BINARY COALESCENCE

The TaylorF2 reduced-spin waveform model in frequency domain is given by [19]

$$\tilde{h}(f) = \mathcal{A} f^{-7/6} \exp\left\{-i\left[\Psi(f) - \frac{\pi}{4}\right]\right\}, \tag{A1}$$

where the amplitude $\mathcal{A}$ depends on the component masses, distance to the source, sky position, and orientation of the binary's plane. $\Psi(f)$ is the instantaneous phase, which can be explicitly written as

$$\begin{aligned}\Psi(f) = {} & 2\pi f t_0 + \phi_0 + \frac{3}{128\eta v_f^5}\left\{1 + v_f^2\left[\frac{55\eta}{9} + \frac{3715}{756}\right] + v_f^3[4\beta - 16\pi] + v_f^4\left[\frac{3085\eta^2}{72} + \frac{27145\eta}{504} + \frac{15293365}{508032} - 10\sigma_0\right]\right. \\ & + v_f^5\left[\frac{38645\pi}{756} - \frac{65\pi}{\eta} - \gamma_0\right](3ln(v_f) + 1) + v_f^6\left[-\frac{6848\gamma_E}{21} - \frac{127825\eta^3}{1296} + \frac{76055\eta^2}{1728} + \left(\frac{2255\pi^2}{12} - \frac{15737765635}{3048192}\right)\eta\right. \\ & \left.\left. - \frac{640\pi^2}{3} + \frac{11583231236531}{4694215680} - \frac{6848ln(4v_f)}{21}\right] + v_f^7\left[-\frac{74045\pi\eta^2}{756} + \frac{37815\pi\eta}{1512} + \frac{77096675\pi}{254016}\right]\right\},\end{aligned} \tag{A2}$$

where t_0 is the time of arrival of the signal at the detector marking the epoch at which the instantaneous frequency takes a fiducial value, ϕ_0 is the corresponding phase, $v_f \equiv (\pi m f)^{1/3}$ is the instantaneous velocity, $m = m_1 + m_2$ is the total mass, $\eta = m_1 m_2/(m_1 + m_2)^2$ is the symmetric mass ratio of the binary, and γ_E is the Euler gamma constant.

The spin effects are encoded through β, σ_0, and γ_0 which appear at 1.5 PN, 2 PN, and 2.5 PN phase terms, respectively, and are given by

$$\begin{aligned}\beta &= \frac{113\chi_r}{12}\\ \sigma_0 &= \left(-\frac{12769(4\eta-81)}{16(76\eta-113)^2}\right)\chi_r^2\\ \gamma_0 &= \left(\frac{565(1713\eta^2+135856\eta-146597)}{2268(76\eta-13)}\right)\chi_r,\end{aligned} \tag{A3}$$

where the reduced-spin parameter χ_r is defined as the weighted sum of individual spins χ_1 and χ_2 of the component masses as

$$\chi_r = \frac{1}{2}\left(1-\frac{76}{113\eta}\right)(\chi_1+\chi_2)+\frac{1}{2}\frac{m_1-m_2}{m_1+m_2}(\chi_1-\chi_2). \tag{A4}$$

The individual spins of the components $\chi_{1,2}$ are the projections of their spin vectors $\boldsymbol{S}_{1,2}$ along the Newtonian orbital angular momentum vector $\boldsymbol{L}_N$ and defined as

$$\chi_{1,2} = \frac{\boldsymbol{S}_{1,2}\cdot\hat{\boldsymbol{L}}_N}{2m_{1,2}^2}. \tag{A5}$$

APPENDIX B: SPACE-FILLING TRUNCATED OCTAHEDRON

A polyhedron is a three-dimensional solid that has a finite number of polygon faces. One can fill a 3D space completely without any overlap or gap through the tessellations of space-filling polyhedra. Examples of such space-filling polyhedron include cubes, hexagonal prisms, etc. Solution to such space-filling problems find many practical applications like optimal placement of a network of communication towers [52]. The template placement problem addressed in this paper can also be mapped to an optimal space-filling problem in curved space. The geometric properties of optimal space-filling polyhedra can be understood from the volumetric quotient (Q_v) defined as

$$Q_v = \frac{V}{\frac{4}{3}\pi R^3}, \tag{B1}$$

where V is the volume of polyhedron, R is maximum distance from its center to any vertex, and S is the surface area of the polyhedron. Note that (Q_v) is the inverse of the thickness, a common measure for the efficiency of a given covering. The polyhedra with the highest Q_v is expected to provide the optimum coverage.

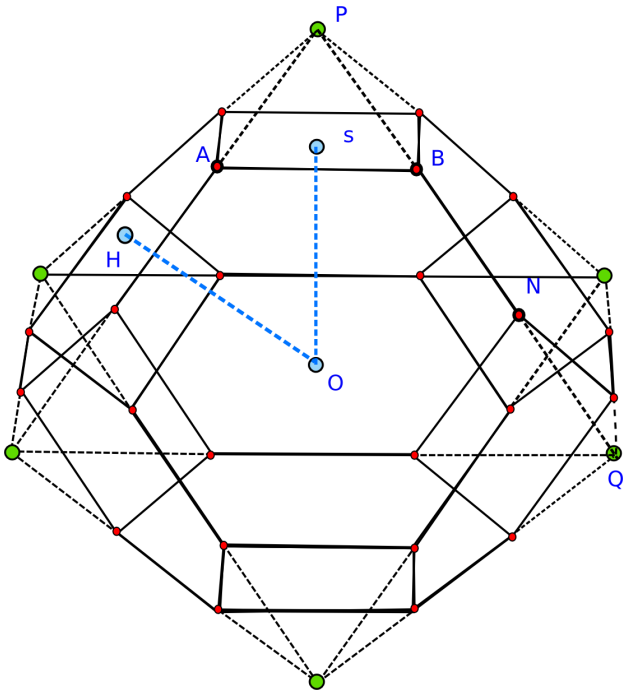


FIG. 7. Truncated octahedron created by truncating six pyramids from the six vertices of two square-base pyramids.

The TO is a 14-faced Archimedean solid, with eight hexagonal faces, six square faces, and 24 vertices. It is generated by joining two regular pyramids upside down and cutting a pyramid from all six vertices in such a way that the length of all the sides generated are equal. Thus, a truncated octahedron of side a can be created by removing six pyramids of side a from an octahedron of side $3a$. Figure 7 shows the TO obtained from two pyramids. The geometric properties of a TO and pyramid are given in Table III.

TABLE III. Geometrical properties of a truncated octahedron and pyramid where the truncated octahedron is constructed using two truncated pyramids.

Name	Value
Side length of pyramid (PQ)	$3a$
Side length of TO (BN)	a
Height of pyramid (OP)	$\frac{3}{\sqrt{2}}a$
Height of TO along square face (OS)	$\sqrt{\frac{3}{2}}a$
Height of TO along hexagonal face (OH)	$\sqrt{2}a$
Height of TO along vertices (ON)	$\sqrt{\frac{5}{2}}a$
Volume of TO (V)	$8\sqrt{2}a^3$
Surface area of TO (S)	$(6+12\sqrt{3})a^2$
Volumetric quotient of TO (Q_v)	≈ 0.683292042

Suppose a three-dimensional volume is covered by the tessellations of TO cells. Each face of such a cell can be shared by another neighboring TO cell. There are two kind of neighboring cells: ones that share the square faces, which we shall call S neighbors, and the others that share the hexagonal faces, which we shall refer to as H neighbors. Each TO has a maximum of six S neighbors and eight H neighbors. The distance between S neighbors is twice the height of the TO along the square face, and that for H neighbors is twice of height of the TO along the hexagonal face. When a TO is inscribed in a sphere of radius R such that the z axis goes through the center of one of the square faces, then the sides of the squares and hexagons a is given by $a = \sqrt{\frac{2}{5}}R$, the distance from the center to each of the S neighbors is equal to $\sqrt{\frac{12}{5}}R$, and the distance to the H neighbors is $\frac{4}{\sqrt{5}}R$. The coordinates of all the 14 neighbors are listed in Table IV.

TABLE IV. These coordinates are the neighborhood positions of a TO when it is inscribed in a sphere with radius R, where the center of a TO is placed at origin and the z axis goes through the center of one of the square faces.

Neighborhood type	Position
S neighborhood	$\left(\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}}\right)$, $\left(-\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}}\right)$,
Square faces	$\left(-\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}}\right)$, $\left(\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}}\right)$, $\left(-\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}}\right)$, $\left(\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}}\right)$, $\left(-\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}}\right)$, $\left(\frac{2R}{\sqrt{5}},-\frac{2R}{\sqrt{5}},\frac{2R}{\sqrt{5}}\right)$
H neighborhood	$\left(0,\frac{4R}{\sqrt{5}},0\right)$, $\left(0,-\frac{4R}{\sqrt{5}},0\right)$, $\left(\frac{4R}{\sqrt{5}},0,0\right)$,
Hexagonal faces	$\left(-\frac{4R}{\sqrt{5}},0,0\right)$, $\left(0,0,\frac{4R}{\sqrt{5}}\right)$, $\left(0,0,-\frac{4R}{\sqrt{5}}\right)$

[1] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. Lett. **116**, 061102 (2016).
[2] G. M. Harry and LIGO Scientific Collaboration, Classical Quantum Gravity **27**, 084006 (2010).
[3] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. Lett. **116**, 241103 (2016).
[4] F. Acernese *et al.*, J. Phys. Conf. Ser. **610**, 012014 (2015).
[5] K. Somiya, Classical Quantum Gravity **29**, 124007 (2012).
[6] Y. Aso, Y. Michimura, K. Somiya, M. Ando, O. Miyakawa, T. Sekiguchi, D. Tatsumi, and H. Yamamoto (KAGRA Collaboration), Phys. Rev. D **88**, 043007 (2013).
[7] B. Iyer, T. Souradeep, C. Unnikrishnan, S. Dhurandhar, S. Raja, and A. Sengupta, LIGO-India Technical Report No. M1100296-v2, 2011, https://dcc.ligo.org/cgi-bin/DocDB/ShowDocument?docid=75988.
[8] C. S. Unnikrishnan, Int. J. Mod. Phys. D **22**, 1341010 (2013).
[9] G. M. Manca and M. Vallisneri, Phys. Rev. D **81**, 024004 (2010).
[10] C. Messenger, R. Prix, and M. A. Papa, Phys. Rev. D **79**, 104017 (2009).
[11] T. Cokelaer, Phys. Rev. D **76**, 102004 (2007).
[12] S. Babak, R. Balasubramanian, D. Churches, T. Cokelaer, and B. S. Sathyaprakash, Classical Quantum Gravity **23**, 5477 (2006).
[13] I. W. Harry, B. Allen, and B. S. Sathyaprakash, Phys. Rev. D **80**, 104014 (2009).
[14] D. A. Brown, I. Harry, A. Lundgren, and A. H. Nitz, Phys. Rev. D **86**, 084017 (2012).
[15] A. Buonanno, B. R. Iyer, E. Ochsner, Y. Pan, and B. S. Sathyaprakash, Phys. Rev. D **80**, 084043 (2009).
[16] E. Poisson and C. M. Will, Phys. Rev. D **52**, 848 (1995).
[17] P. Ajith, N. Fotopoulos, S. Privitera, A. Neunzert, N. Mazumder, and A. J. Weinstein, Phys. Rev. D **89**, 084041 (2014).
[18] C. Capano, I. Harry, S. Privitera, and A. Buonanno, Phys. Rev. D **93**, 124007 (2016).
[19] P. Ajith, Phys. Rev. D **84**, 084037 (2011).
[20] D. Shoemaker (LIGO Scientific Collaboration), LIGO Report No. T0900288-v3, 2010, https://dcc.ligo.org/LIGO-T0900288/public.
[21] LIGO Data Analysis Software Working Group, LALSuite: LSC Algorithm Library Suite, https://www.lsc-group.phys.uwm.edu/daswg/projects/lalsuite.html.
[22] L. S. Finn, Phys. Rev. D **46**, 5236 (1992).
[23] B. S. Sathyaprakash and S. V. Dhurandhar, Phys. Rev. D **44**, 3819 (1991).
[24] S. Babak *et al.*, Phys. Rev. D **87**, 024033 (2013).
[25] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, Grundlehren der mathematischen Wissenschaften Vol. 290 (Springer Science & Business Media, New York, 1999).
[26] R. Prix, Classical Quantum Gravity **24**, S481 (2007).
[27] B. J. Owen, Phys. Rev. D **53**, 6749 (1996).
[28] B. J. Owen and B. S. Sathyaprakash, Phys. Rev. D **60**, 022002 (1999).
[29] B. P. Abbott *et al.* (LIGO Scientific Collaboration), Phys. Rev. D **79**, 122001 (2009).
[30] B. P. Abbott *et al.* (LIGO Scientific Collaboration), Phys. Rev. D **80**, 047101 (2009).
[31] J. Abadie *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. D **82**, 102001 (2010).
[32] D. Keppel, A. P. Lundgren, B. J. Owen, and H. Zhu, Phys. Rev. D **88**, 063002 (2013).
[33] I. W. Harry, A. H. Nitz, D. A. Brown, A. P. Lundgren, E. Ochsner, and D. Keppel, Phys. Rev. D **89**, 024010 (2014).

[34] A. Pai and K. G. Arun, Classical Quantum Gravity **30**, 025011 (2013).
[35] A. Nitz *et al.*, ligo-cbc/pycbc: Er10 production release 2, 2016, https://doi.org/10.5281/zenodo.164940.
[36] S. A. Usman *et al.*, Classical Quantum Gravity **33**, 215004 (2016).
[37] T. Dal Canton *et al.*, Phys. Rev. **D90**, 082004 (2014).
[38] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. E. Creighton, Phys. Rev. D **85**, 122006 (2012).
[39] T. N. Hibbard, J. Assoc. Comput. Mach. **9**, 13 (1962).
[40] S. Privitera, S. R. P. Mohapatra, P. Ajith, K. Cannon, N. Fotopoulos, M. A. Frei, C. Hanna, A. J. Weinstein, and J. T. Whelan, Phys. Rev. D **89**, 024003 (2014).
[41] B. P. Abbott *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. D **93**, 122003 (2016).
[42] A. Schürmann and F. Vallentin, Discrete Comput. Geom. **35**, 73 (2006).
[43] S. W. Thomson, Philos. Mag. **24**, 503 (1887).
[44] A. S. Douglas, Comput. J. **2**, 1 (1959).
[45] P. F. Windley, Comput. J. **3**, 84 (1960).
[46] C. Kalaghatgi, P. Ajith, and K. G. Arun, Phys. Rev. D **91**, 124042 (2015).
[47] T. A. Apostolatos, Phys. Rev. D **52**, 605 (1995).
[48] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Pürrer, X. J. Forteza, and A. Bohé, Phys. Rev. D **93**, 044007 (2016).
[49] S. Husa, S. Khan, M. Hannam, M. Pürrer, F. Ohme, X. J. Forteza, and A. Bohé, Phys. Rev. D **93**, 044006 (2016).
[50] A. Taracchini, A. Buonanno, Y. Pan, T. Hinderer, M. Boyle *et al.*, Phys. Rev. D **89**, 061502 (2014).
[51] M. Pürrer, Classical Quantum Gravity **31**, 195010 (2014).
[52] S. M. N. Alam and Z. J. Haas, in Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (ACM, New York, 2006), pp. 346–357.