Spline based search method for unmodeled transient gravitational wave chirps

Soumya D. Mohanty[*]

Department of Physics and Astronomy, The University of Texas Rio Grande Valley,
One West University Blvd., Brownsville, Texas 78520, USA
(Received 28 September 2017; published 30 November 2017)

A method is described for the detection and estimation of transient chirp signals that are characterized by smoothly evolving, but otherwise unmodeled, amplitude envelopes and instantaneous frequencies. Such signals are particularly relevant for gravitational wave searches, where they may arise in a wide range of astrophysical scenarios. The method uses splines with continuously adjustable breakpoints to represent the amplitude envelope and instantaneous frequency of a signal, and estimates them from noisy data using penalized least squares and model selection. Simulations based on waveforms spanning a wide morphological range show that the method performs well in a signal-to-noise ratio regime where the time-frequency signature of a signal is highly degraded, thereby extending the coverage of current unmodeled gravitational wave searches to a wider class of signals.

I. INTRODUCTION

The tally of confirmed direct gravitational wave (GW) detections now stands at five events. Across two observing runs, the twin Advanced Laser Interferometer Gravitational-Wave Observatory (LIGO) [1] detectors found GW150914 [2], GW151226 [3], and GW170104 [4] in two-way coincidence. All three signals are consistent with binary black hole (BBH) mergers. (A fourth BBH merger, LVT151012, was discovered in the first observing run but with marginal significance.) These were followed by an additional BBH merger signal, GW170814 [5], that was also the first event to be discovered in a three-way coincidence between the LIGO and Advanced Virgo [6] detectors. The second observing run concluded with the remarkable discovery by LIGO and Virgo of GW170817 [7], a binary neutron star inspiral.

Along with further improvements in the sensitivity of the LIGO and Virgo detectors, additional second -generation detectors—KAGRA [8] and LIGO-India [9]—are scheduled to come online over the next several years. Besides significantly enhancing overall search sensitivity, combining the data from the upcoming network of detectors will better localize sources on the sky, thereby increasing the chances of finding their electromagnetic counterparts.

The type of signals detected by LIGO so far have waveforms that can be calculated theoretically as a function of system parameters, allowing parametric data analysis methods, such as matched filtering [10], to be used for their detection and estimation. This approach is inapplicable, however, to astrophysical sources that are unanticipated or that emit inherently unpredictable signals. Search methods for such unmodeled signals—known as GW bursts when they are transient—must use minimal prior assumptions about their waveforms.

Due to their broad scope, burst search methods can also detect sufficiently strong parametric signals. In fact, GW150914 was first detected [11] by a burst search method [12] that implements a regularized maximum likelihood analysis [13–15] of data from a network of detectors. Refined estimates of the source parameters, such as the component masses, were obtained subsequently using parametric methods.

Among burst signals, the most challenging to search for are those that do not have compact time-frequency signatures. We know of several possible astrophysical scenarios where such signals may arise. Among these are (i) the post core-bounce phase of a core-collapse supernova (CCSN)[16,17], (ii) dynamical instabilities in rotating newborn neutron stars [18,19], and (iii) clump formation or dynamical instabilities in the accretion disc surrounding a newly formed black hole in a collapsar [20,21]. Such signals may be generic to GW emissions powered by the rotational energy of a compact engine.

In many of these scenarios, the burst signals spread their total energy over well-defined "tracks" in the time-frequency plane. Such signals are generally called *chirps* in the signal processing literature. (BBH merger signals are examples of parametric chirps.) Taking the analytic representation, $a(t) \exp(i\phi(t))$, of a signal $s(t)$, one expects a track-like feature in the time-frequency representation of $s(t)$ when the amplitude envelope $a(t)$ and instantaneous frequency $f(t) = \dot{\phi}(t)$ evolve adiabatically—$f(t) \gg \dot{a}/a$ and $f^2(t) \gg \dot{f}(t)$—relative to the instantaneous period $1/f(t)$. The sharpness of the track is determined by $\dot{a}(t)$, with a smaller $\dot{a}$ leading to a sharper track.

Several search methods have been developed in the GW data analysis literature for short duration (~1 sec) unmodeled chirps for which $a(t)$, or $f(t)$, or both are unknown.

---
[*]soumya.mohanty@utrgv.edu

The track-search method [22] uses an image processing approach to search for track-like features in the Wigner-Ville (WV) time-frequency distribution [23]. This method can detect signals with arbitrary $a(t)$ and $f(t)$ provided its track is distinguishable from the spurious features that appear in the WV transform due to its nonlinear nature. A significant advance has recently been made in mitigating these spurious features by applying sparsity regularization to the WV transform [24] but it remains to be integrated with methods such as track-search.

A natural approach to the detection of unmodeled chirps is to approximate $f(t)$ by a piecewise linear curve. Each piece represents a transient linear chirp signal, called a chirplet, and the signal is assumed to be a sequence of connected chirplets called a chirplet chain. [Note that a chirplet chain only approximates $f(t)$ and that additional degrees of freedom are needed to model $a(t)$.] The main challenge in this approach is the extremely high computational cost of searching the space of all chirplet chains to find the one that best fits the data. Different methods have been proposed to address this issue.

The best chirplet chain method [25] approximates the chirplet chain approach, for the special case where the duration (or scale) of chirplets and the length of a chain (number of chirplets) are fixed, with path integrals of the WV transform. The chirplet path pursuit (CPP) method [26], uses a different approach in which the chains are constructed out of a discrete set of multiscale chirplets. The use of multiscale chirplets allows greater flexibility in the signal model since slowly evolving parts of $f(t)$ can be approximated by longer chirplets. This leads to chains with variable lengths and a selection of the best-fit chirplet chain requires balancing the fitness of a chain against its complexity using a penalty on the chain length.

Track detection [27] or path integrals [28] have also been studied for detecting long duration ($\gtrsim 10$ sec) GW bursts in data from multiple detectors. Unlike the single detector methods mentioned above, these methods use the short-time cross-spectra of data from pairs of GW detectors. In the following, we consider only the single detector case, leaving multiple detectors to future work.

While the chirplet chain approach is designed to approximate $f(t)$, it is important to consider $a(t)$ also. In particular, there is no reason for $a(t)$ to remain constant over the duration of a single chirplet. This issue was addressed in CPP by allowing the amplitude of each chirplet to evolve as a polynomial in time. However, the degree of the polynomial is fixed for all the chirplets at a given scale and has to be prescribed in advance. This is difficult to do when $a(t)$ has an unknown and complex evolution.

In this paper, we present a search method for chirp signals that explicitly takes amplitude modulation into account. The structure of the method follows logically from modeling $a(t)$ and $f(t)$ as independent splines and

seeking a computationally feasible solution to the resulting high-dimensional nonlinear regression problem. Early and intermediate steps in the development of the method were reported in Refs. [29–31].

The detection and estimation performance of the method is quantified using simulated data, incorporating a wide range of signal waveform morphologies, in a signal-to-noise ratio (SNR) regime where the signal track in the time-frequency plane is easily disrupted and masked by noise. To keep computational costs under control, we focus only on signals with durations of $O(1)$ sec in this paper although the general idea can, in principle, be applied to much longer signals.

The rest of the paper is organized as follows. Section II sets up the notation and the models used in this paper for noise and signal. Section III presents a description of the method. The simulation setup used in assessing its detection and estimation performance is described in Sec. IV. The results obtained from the simulations are presented in Sec. V. Section VI compares the performance of the method with that of time-frequency clustering, a key component of the burst search methods currently used in LIGO. This is followed, in Sec. VII, by comparisons with track-search and CPP. Section VIII presents the conclusions from our study.

II. STATISTICAL MODEL

In the following, a symbol such as $\bar{s} \in \mathbb{R}^N$ denotes a row vector with N elements, and s_j, $j = 0, 1, \ldots, N - 1$, or $[\bar{s}]_j$, denotes its jth element. When $\bar{s}$ is a finite length discrete-time sequence of sample values of an underlying continuous-time function $s(t)$, the sampling times are denoted by t_i, $i = 0, 1, \ldots, N - 1$, and $s_i = s(t_i)$. A symbol such as $\hat{s}$ denotes either a solution to an optimization problem or a quantity estimated from data.

Boldface symbols, such as $\mathbf{A}$, denote matrices with the element in its ith row and jth column denoted by A_{ij} or $[\mathbf{A}]_{ij}$. The identity matrix is denoted by $\mathbf{I}$.

We use $\tilde{s}$ to denote the discrete Fourier transform (DFT) of $\bar{s}$,

$$\tilde{s}^T := \mathbf{F}\bar{s}^T, \qquad (1)$$

$$F_{\mathrm{km}} = e^{-2\pi i km/N}, \qquad (2)$$

with $\tilde{s}_j$ being its jth element. The inverse DFT is given by,

$$\mathbf{F}^{-1} = \frac{1}{N}\mathbf{F}^{\dagger}. \qquad (3)$$

The symbol "./" denotes element-by-element division, and the supremum of integers less than or equal to $x \in \mathbb{R}_{>0}$ is denoted by $\lfloor x \rfloor$.

A. Noise model

We will denote a segment of GW detector output, sampled uniformly with a sampling frequency f_s, by $\bar{y} \in \mathbb{R}^N$. Under the null and alternative hypotheses, denoted by H_0 and H_1 respectively,

$$\bar{y} = \begin{cases} \bar{n} & ; H_0, \\ \bar{s} + \bar{n} & ; H_1, \end{cases} \tag{4}$$

where $\bar{s}$ is a GW signal and $\bar{n}$ is a realization of noise. Our noise model assumes that $\bar{n}$ is drawn from a zero mean, Gaussian, stationary stochastic process. Let $\mathbf{C}$, $C_{ij} = E[n_i n_j]$, where $E[A]$ denotes the ensemble average of a random variable A, be the covariance matrix of the noise segment.

Since $\mathbf{C}$ is symmetric and positive definite, an inner product can be defined on $\mathbb{R}^N$,

$$\langle \bar{x}, \bar{y} \rangle = \bar{x} \mathbf{C}^{-1} \bar{y}^T. \tag{5}$$

The norm induced by this inner product will be denoted by $\|\bar{x}\|^2 = \langle \bar{x}, \bar{x} \rangle$. It can be shown that

$$\|\bar{x}\|^2 = \frac{1}{N} \tilde{x}^* (\mathbf{FCF}^{-1})^{-1} \tilde{x}^T$$

$$\approx \frac{1}{N f_s} \tilde{x}(\tilde{x}^\dagger ./ \bar{S}^T), \tag{6}$$

where $\bar{S}$ is the two-sided power spectral density of the noise defined by

$$S_i = \frac{1}{N f_s} E[|\tilde{n}_i|^2] = \frac{1}{f_s} (\mathbf{FCF}^{-1})_{ii}. \tag{7}$$

It follows that,

$$\delta_f \sum_{m=0}^{N-1} S_m = \sigma^2, \tag{8}$$

where σ^2 is the variance of the noise and $\delta_f = f_s/N$ is the spacing between consecutive frequencies in the DFT.

The approximation in Eq. (6) arises from neglecting off-diagonal terms in $\mathbf{FCF}^{-1}$. However, the approximation approaches equality very rapidly with an increase in N.

B. Signal model

As discussed earlier, the amplitude envelope $a(t)$ and instantaneous frequency $f(t) = \dot{\phi}(t)$ of a chirp signal evolve smoothly on the time scale of the instantaneous period $1/f(t)$. We model this smoothness behavior by prescribing $a(t)$ and $f(t)$ to be splines. The motivation behind using splines in particular is discussed further in Sec. III. Appendix A provides a brief review of splines and B-spline functions.

Let $a(t; \bar{\alpha}, \bar{\tau}_a)$ denote the spline for $a(t)$, where $\bar{\tau}_a$ are the breakpoints,

$$a(t; \bar{\alpha}, \bar{\tau}_a) = \sum_{j=0}^{M-1} \alpha_j \mathcal{B}_{j,k}(t; \bar{\tau}_a), \tag{9}$$

and $\mathcal{B}_{j,k}(t; \bar{\tau}_a)$ is a B-spline function [32] of order k. Since B-splines have compact support, $a(t; \bar{\alpha}, \bar{\tau}_a) = 0$ for $t \notin [\tau_{a,0}, \tau_{a,M-1}]$, where $\tau_{a,i} = [\bar{\tau}_a]_i$. As we will see later, the linear dependence of $a(t; \bar{\alpha}, \bar{\tau}_a)$ on $\bar{\alpha}$ allows considerable simplification in the analysis.

Let $f(t; \bar{\nu}, \bar{\tau}_f)$ be the spline corresponding to $f(t)$, where $\bar{\tau}_f \in \mathbb{R}^K$ and $\bar{\nu} \in \mathbb{R}^K$ denote the breakpoints and corresponding instantaneous frequencies that the spline must interpolate. Unlike $a(t; \bar{\alpha}, \bar{\tau}_a)$, there is no particular advantage gained by expressing $f(t; \bar{\nu}, \bar{\tau}_f)$ in terms of B-spline functions. We use Steffen's method [33] for spline interpolation, which guarantees the monotonicity of the interpolating function between given data points, in order to prevent spurious oscillations in $f(t; \bar{\nu}, \bar{\tau}_f)$.

With $\bar{\nu}$, $\bar{\tau}_a$, and $\bar{\tau}_f$ denoted collectively by $\bar{\theta}$, the signal model is given by,

$$s(t_i; \bar{\alpha}, \bar{\theta}, \phi_0) = a(t_i; \bar{\alpha}, \bar{\tau}_a) \sin(\phi(t_i; \bar{\nu}, \bar{\tau}_f) + \phi_0), \tag{10}$$

$$\phi(t; \bar{\nu}, \bar{\tau}_f) = \begin{cases} 0, & t < \tau_{a,0}, \\ \int_{\tau_{a,0}}^{t} dt' f(t'; \bar{\nu}, \bar{\tau}_f), & t \leq \tau_{a,M-1}. \end{cases} \tag{11}$$

Let $\mathbf{X}_0$ and $\mathbf{X}_1$ denote matrices given by

$$[\mathbf{X}_0(\bar{\theta})]_{jm} = \mathcal{B}_{j,k}(t_m; (\bar{\tau}_a) \sin(\phi(t_m; \bar{\nu}, \bar{\tau}_f)), \tag{12}$$

and

$$[\mathbf{X}_1(\bar{\theta})]_{jm} = \mathcal{B}_{j,k}(t_m; \bar{\tau}_a) \cos(\phi(t_m; \bar{\nu}, \bar{\tau}_f)). \tag{13}$$

In terms of these matrices, the signal sequence is,

$$\bar{s}(\bar{\alpha}, \bar{\theta}, \phi_0) = \bar{\beta} \mathbf{X}(\bar{\theta}), \tag{14}$$

$$\bar{\beta} = \bar{\alpha} \Phi_0,$$

$$\Phi_0 = (\cos\phi_0 \mathbf{I} \ \sin\phi_0 \mathbf{I}), \tag{15}$$

$$\mathbf{X}(\bar{\theta}) = \begin{pmatrix} \mathbf{X}_0(\bar{\theta}) \\ \mathbf{X}_1(\bar{\theta}) \end{pmatrix}. \tag{16}$$

While the signal model in Eq. (10) captures the basic idea of smoothness in the evolution of $a(t)$ and $f(t)$, it does not enforce the adiabaticity requirement. This is mainly because it is technically difficult to incorporate this constraint at present. As a result, the scope of the model actually encompasses a broader set of signals than just well-defined chirps.

III. DESCRIPTION OF THE METHOD

Based on the fundamental use of splines in the signal model given by Eq. (10) and the fact that the model represents signals that are effectively, but not only, chirps, we call the method presented here "Spline Enabled Effectively-Chirp Regression" (SEECR).

Some of the principal design choices behind SEECR are motivated by issues encountered in the simpler problem of fitting a smooth curve to noisy data. We briefly review these issues first before presenting a description of SEECR.

A formal approach to the problem of fitting a smooth curve to noisy data is to use regularized least-squares with a roughness penalty [34],

$$\hat{s}(t) = \underset{s(t)}{\arg\min} \sum_{i=0}^{N-1} (y_i - s_i)^2 + \lambda \int_{t_0}^{t_{N-1}} dt \ddot{s}^2(t). \quad (17)$$

This method is known as *smoothing spline* [35,36] since the solution turns out to be a cubic spline with the sampling times t_i, $i = 0, 1, \ldots, N$, as the breakpoints. The influence of the roughness penalty on the solution $\hat{s}(t)$ is controlled by the regulator gain λ. For $\lambda = 0$, the best-fit solution simply matches the data itself, while for $\lambda \to \infty$, it approaches a straight line. Between these two extremes lies a solution that is useful for drawing meaningful inferences from the data.

The natural emergence of splines under a smoothness requirement is the main motivation behind our modeling the amplitude envelope and instantaneous frequency of a chirp as splines. However, estimating these components by directly applying the roughness penalty on them appears to be technically difficult. Instead, we take recourse to another smoothness regularization approach that forms the bases of the *regression spline* [37] method. In this method, regularization is achieved by choosing $s(t)$ to be a spline *ab initio* but limiting the number of breakpoints to be $\ll N$.

A disadvantage of the regression spline method is that the regularization parameter, namely the number of breakpoints, is now discrete and, hence, does not allow fine-grained control over smoothness. Moreover, the placement of the breakpoints now plays an important role in determining the quality of the fit.

For a predetermined placement of a limited number of breakpoints, the *penalized spline* method [38] allows continuous control of smoothness. In the context of the simple curve fitting problem, the penalized spline method models the curve as a linear combination of B-splines and solves

$$\hat{\alpha} = \underset{\bar{\alpha}}{\arg\min} \sum_{i=0}^{N-1} (y_i - \bar{\alpha} \mathbf{A}(\bar{\tau}_a))^2 + \lambda \bar{\alpha} \bar{\alpha}^T, \quad (18)$$

where $A_{jm} = \mathcal{B}_{j,k}(t_m; \bar{\tau}_a)$.

Finding the optimum placement of breakpoints is a challenging nonlinear and nonconvex problem. Methods proposed in the literature to address this problem generally follow the approach of knot insertion and deletion. Only recently have optimization methods been developed that are capable of treating breakpoints as completely free parameters. In particular, particle swarm optimization (PSO) [39,40] has been applied to this problem [29,41] and found to have a good performance.

Along with the placement of breakpoints, the number of breakpoints and the regulator gain have a significant effect on the quality of estimation. For determining the regulator gain, generalized cross-validation (GCV) [42] provides a fast method. The number of breakpoints can be selected using the Akaike information criterion (AIC) [43].

SEECR combines the different elements outlined above, namely, penalized spline, GCV, breakpoint optimization using PSO, and AIC. The description of the algorithm now follows.

A. Regression using penalized spline

The signal model in Eq. (14) is estimated in SEECR by minimizing the penalized least-squares function,

$$\Lambda(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y}, \lambda) = R(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y}) + \lambda \bar{\alpha} \bar{\alpha}^T, \quad (19)$$

where

$$R(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y}) = \|\bar{y} - \bar{s}(\bar{\alpha}, \bar{\theta}, \phi_0)\|^2, \quad (20)$$

is the residual norm squared, over all the signal parameters. Henceforth, we drop the explicit listing of parameters wherever it aids clarity.

The positivity of the amplitude envelope, $a(t) \geq 0$, and B-splines, $\mathcal{B}_{j,k}(t; \bar{\tau}_a) \geq 0$, $\forall$ t, (see Appendix A) requires that the minimization of Λ be performed under a positivity constraint on $\bar{\alpha}$.

The estimate of the signal model is obtained using the following program of nested minimizations:

$$\min_{\bar{\alpha}, \bar{\theta}, \phi_0} \Lambda = \min_{\bar{\theta}} (\min_{\phi_0} (\min_{\bar{\alpha}} \Lambda)), \quad (21)$$

$$\alpha_i \geq 0, \quad \forall \ i. \quad (22)$$

The order of minimization above, from inner to outer, corresponds to parameters that can be treated semianalytically to those that need a fully numerical approach. The steps in solving the program are described below, starting from the innermost minimization. As mentioned earlier, the regulator gain, λ, is determined using GCV, which is merged into the minimization program at the second step.

B. Innermost minimization

First, we address the unconstrained minimization over $\bar{\alpha}$. To do so, we use Eqs. (14)–(16) to rewrite Λ in a more convenient form:

$$\Lambda = \|\bar{y}\|^2 + \bar{\alpha} \mathbf{K} \bar{\alpha}^T - 2\bar{q} \bar{\alpha}^T, \qquad (23)$$

$$\mathbf{K} = \mathbf{\Phi}_0 \mathbf{G} \mathbf{\Phi}_0^T,$$

$$\mathbf{G} = \mathbf{X} \mathbf{C}^{-1} \mathbf{X}^T + \lambda \mathbf{I}, \qquad (24)$$

$$\bar{q} = \bar{\eta} \mathbf{\Phi}_0^T, \qquad (25)$$

$$\bar{\eta} = \bar{y} \mathbf{C}^{-1} \mathbf{X}^T. \qquad (26)$$

$\mathbf{K}$ is symmetric and positive definite since $\bar{x} \mathbf{K} \bar{x}^T = \bar{x}' \mathbf{G} \bar{x}'^T > 0$ for any $\bar{x}$, where $\bar{x}'^T = \mathbf{\Phi}_0 \bar{x}^T$. It then follows that

$$\bar{r} = \arg \min_{\bar{\alpha}} \Lambda = \bar{q} \mathbf{K}^{-1}, \qquad (27)$$

is the solution to the unconstrained inner minimization.

The solution to the constrained minimization problem can be obtained from the Karush-Kuhn-Tucker conditions [44]. These conditions essentially state that the solution is either already in the convex cone of $\mathbb{R}^M$ defined by $\alpha_i > 0$, $\forall i$, or on one of its faces.

Thus, given the unconstrained minimizer $\bar{r}$ in Eq. (27), if $r_i \geq 0, \forall i$, then $\bar{r}$ itself is the constrained minimizer. If not, one has to find the projection of $\bar{r}$ on the faces. (The inner product to use for the projection is $\langle \bar{x}, \bar{w} \rangle = \bar{x} \mathbf{K} \bar{w}^T$.) For this task, we use the mixed primal-dual bases algorithm developed by Fraser and Massam [45], which returns the edge vectors of the face of the cone that contains the projection of $\bar{r}$.

Let the projection operator for the subspace $\mathcal{L}$ spanned by these edge vectors be $\mathbf{P}_{\mathcal{L}}$. Then the solution to the constrained minimization problem is

$$\hat{\alpha}_{\lambda, \phi_0}^T = \mathbf{P}_{\mathcal{L}} \bar{r}^T, \qquad (28)$$

and the estimated signal at this step in the minimization program is,

$$\hat{s}_{\lambda, \phi_0} = \bar{y} \mathbf{H}_{\lambda, \phi_0}^T, \qquad (29)$$

$$\mathbf{H}_{\lambda, \phi_0}^T = \mathbf{C}^{-1} \mathbf{X}^T \mathbf{\Phi}_0^T \mathbf{K}^{-1} \mathbf{P}_{\mathcal{L}}^T \mathbf{\Phi}_0 \mathbf{X}. \qquad (30)$$

The subscripts in $\hat{\alpha}_{\lambda, \phi_0}^T$, $\hat{s}_{\lambda, \phi_0}$ and $\mathbf{H}_{\lambda, \phi_0}$ make the dependence of these quantities on λ and ϕ_0 explicit.

C. Minimization over ϕ_0 and GCV

Consider the simpler case where GCV is used to determine λ before the minimization over ϕ_0. Let $\lambda_{\mathrm{GCV}}(\phi_0)$ be the resulting value. Then,

$$\lambda_{\mathrm{GCV}}(\phi_0) = \arg \min_{\lambda} \mathrm{GCV}(\lambda; \phi_0), \qquad (31)$$

$$\mathrm{GCV}(\lambda; \phi_0) = \frac{R(\hat{\alpha}_{\lambda, \phi_0}^T, \bar{\theta}, \phi_0 | \bar{y})}{(1 - \mathrm{Tr}(\mathbf{H}_{\lambda, \phi_0})/N)^2}, \qquad (32)$$

where $\mathrm{Tr}(\mathbf{H}_{\lambda, \phi_0})$ is the trace of $\mathbf{H}_{\lambda, \phi_0}$.

Numerical experiments show that computing $\lambda_{\mathrm{GCV}}(\phi_0)$ before minimizing ϕ_0 gives very unstable results. This is because the positivity constraint can introduce abrupt changes in the projection $\hat{\alpha}_{\lambda, \phi_0}^T$, by making it switch from one face of the convex cone to another, as ϕ_0 is varied. Independently of this empirical reason, putting GCV outside the minimization over ϕ_0 also makes sense because it is an approximation to cross-validation, and the right place for the latter is always after minimization over all relevant signal parameters.

Thus, the regulator gain in SEECR is determined as follows:

$$\lambda_{\mathrm{GCV}} = \arg \min_{\lambda} \mathrm{GCV}(\lambda; \phi_0(\lambda)), \qquad (33)$$

$$\phi_0(\lambda) = \arg \min_{\phi_0} \Lambda(\hat{\alpha}_{\lambda, \phi_0}, \bar{\theta}, \phi_0 | \bar{y}). \qquad (34)$$

Both of the minimizations above are performed numerically.

D. Outer minimization

Let

$$\hat{\alpha} = \hat{\alpha}_{\lambda_{\mathrm{GCV}}, \phi_0(\lambda_{\mathrm{GCV}})}, \qquad (35)$$

and let the corresponding value of Λ be denoted by

$$F(\bar{\theta} | \bar{y}) = \Lambda(\hat{\alpha}, \bar{\theta}, \phi_0(\lambda_{\mathrm{GCV}}) | \bar{y}), \qquad (36)$$

which we call the *fitness function* in the following. The next step in the program given by Eq. (21) is the minimization of the fitness function over the parameters $\bar{\tau}_a$, $\bar{\nu}$, and $\bar{\tau}_f$.

There are two principal challenges in this task. One is the high dimensionality, given by $M + 2K$, of the search space, and the other is the degeneracy caused by different permutations of the breakpoint sequences giving rise to the same splines. Degeneracies create strong local minima which increase the difficulty of locating the global minimum.

To address the issue of high dimensionality, we lower the number of parameters as follows. First, we set $[\bar{\tau}_f]_0 = \tau_{a,0}$ and $[\bar{\tau}_f]_{K-1} = \tau_{a,M-1}$ because the amplitude envelope spline, and hence the signal itself, is zero outside the interval $[\tau_{a,0}, \tau_{a,M-1}]$.

Second, based on the Cramer-Rao lower bound on the estimation error in the amplitude of a monochromatic signal being higher than its frequency, we can expect that the error in the estimation of the amplitude envelope $a(t)$ of a chirp is higher than its instantaneous frequency $f(t)$. (This is illustrated later in Sec. V C.) A corollary is that one need not invest as much effort in modeling the $a(t)$ spline as the $f(t)$ one. Therefore, we can simplify the placement of breakpoints for $a(t)$ considerably, and we do so by spacing them uniformly. This reduces the number of free $a(t)$ breakpoints from M to just two, namely, $\tau_{a,0}$ and $\tau_{a,M-1}$.

The total dimensionality of the search space for the outer minimization now reduces to $2K$: the two end breakpoints for the $a(t)$ spline, the $K - 2$ interior breakpoints for the $f(t)$ spline, and the K instantaneous frequency values in $\bar{\nu}$.

One approach to addressing the issue of degeneracy arising from the permutation symmetry of breakpoints is to constrain the two breakpoint sequences to be monotonic. That is, we enforce $\tau_{a,M-1} > \tau_{a,0}$ and $[\bar{\tau}_f]_i > [\bar{\tau}_f]_j$ for $i > j$ when searching for the minimum of the fitness function. However, this means that the search volume no longer has the simple shape of a box, a factor that is known to be detrimental to the performance of PSO.

An alternative is to reparametrize breakpoints such that every point in the new search space is guaranteed to be a monotonic sequence. For any breakpoint sequence $\bar{\tau} = (\tau_0, \tau_1, \ldots, \tau_{P-1})$, a simple reparametrization that leads to monotonicity is,

$$x_0 = \tau_0, \tag{37}$$

$$x_{0 < i \leq P-1} = \frac{\tau_i - \tau_{i-1}}{t_1 - \tau_{i-1}}. \tag{38}$$

The new parameters $\bar{x} = (x_0, x_1, \ldots, x_{P-1})$ for $i > 0$ are simply distance ratios, with $x_{i>0} \in [0, 1]$ (and $x_0 \in [t_0, t_{N-1})$). The search space in $\bar{x}$ is a box and no additional constraints are needed to ensure the monotonicity of a breakpoint sequence.

There is, however, a disadvantage to the reparametrization scheme presented above, which is that a uniformly spaced breakpoint sequence is pushed towards the boundary of the box. This is not of much concern for the amplitude envelope spline since we have reduced the number of free breakpoints to just two. However, the variant of PSO used in this paper is generally known to perform better if a global minimum is located towards the central region of a search space. Hence, its performance would suffer with the above reparametrization if an instantaneous frequency spline were best represented by uniformly spaced breakpoints.

A clever scheme that circumvents this problem, while still preserving monotonicity, was proposed in Ref. [46]:

$$x_0 = \tau_0, \tag{39}$$

$$x_{1 \leq i \leq P-2} = \frac{\tau_i - \tau_{i-1}}{\tau_{i+1} - \tau_{i-1}}. \tag{40}$$

$$x_{P-1} = \tau_{P-1}. \tag{41}$$

Here, the distance ratios in Eq. (40) are relative to the gap between the enclosing knots rather than, as in Eq. (38), a knot and the end point of the data.

With the reparametrization in Eqs. (39)–(41), no obvious degeneracy is left in the fitness function. However, that does not mean that there are no local minima in the fitness

function. In fact, as with the estimation of any oscillatory signal, multiple local minima may be expected that may be scattered widely in the search space. Therefore, the search for the global minimum cannot be performed with deterministic local minimizers and a method such as PSO must be used. (Despite the reduction in the number of parameters, the dimensionality of the search space is high enough that grid-based search strategies would simply be computationally infeasible.)

E. Model selection

All of the preceding description relates to fixed numbers, M and K respectively, of breakpoints for the amplitude envelope and instantaneous frequency splines. The final step in SEECR is an automated determination of their best values using AIC. The general expression for AIC is

$$\text{AIC} = 2N_{\text{params}} - 2 \ln \hat{L}, \tag{42}$$

where N_{params} is the total number of free parameters involved in a given model and $\hat{L}$ is the maximum value, over the space of these parameters, of the likelihood function. The best among a set of models is the one that has the minimum AIC value.

In our case, $N_{\text{params}} = M + 2K + 1$, where M is the number of B-spline coefficients $\bar{\alpha}$, $2K$ is the total number of breakpoints and corresponding instantaneous frequency values (Sec. III D), and 1 is for the ϕ_0 parameter.

For Gaussian stationary noise, the log-likelihood can be expressed as $-2R(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y})$ [see Eq. (20)]. Hence, maximizing the former is equivalent to minimizing the latter. In the case of SEECR, $R(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y})$ is replaced by $\Lambda(\bar{\alpha}, \bar{\theta}, \phi_0 | \bar{y})$ [see Eq. (19)]. Its minimization over the parameters $\bar{\alpha}$ and ϕ_0 yields the fitness function, $F(\bar{\theta} | \bar{y})$, defined in Eq. (36). Thus, $-2 \ln \hat{L}$ in Eq. (42) is replaced by the minimum value, $\hat{F}_{M,K}$, of the fitness function,

$$\hat{F}_{M,K} = \min_{\bar{\theta}} F(\bar{\theta} | \bar{y}). \tag{43}$$

Hence, the value of AIC in our case is given by

$$\text{AIC} = 2(M + 2K) + \hat{F}_{M,K}, \tag{44}$$

where we have dropped constants that do not affect the minimization of AIC. The number of breakpoints in the model that minimizes AIC will be denoted by $\hat{M}$ and $\hat{K}$ in the following.

F. Amplitude envelope and instantaneous frequency estimates

Let the final estimated signal sequence, obtained from the best model selected by AIC, be denoted by $\hat{s}$. To obtain the best-fit sequences for the amplitude envelope, $\hat{a}$, and

instantaneous frequency, $\hat{f}$, we construct the analytic sequence $\hat{s}^{(\text{anlt})}$,

$$\hat{s}^{(\text{anlt})} = \hat{s} + i\mathcal{H}[\hat{s}], \tag{45}$$

where $\mathcal{H}$ is the discrete Hilbert transform [47] operator. Then

$$\hat{a}_j = |\hat{s}_j^{(\text{anlt})}|, \tag{46}$$

$$\hat{f}_i = \frac{\hat{\phi}_{i+1} - \hat{\phi}_i}{t_{i+1} - t_i}, \tag{47}$$

$$\hat{\phi}_j = \arg(\hat{s}_j^{(\text{anlt})}), \tag{48}$$

where $j = 0, 1, …, N - 1$, $i = 0, 1, …, N - 2$, and continuity is enforced across jumps of $\pm\pi$ in $\hat{\phi}$.

We do not obtain $\hat{a}$ and $\hat{f}$ directly from their respective estimated splines because the two interact nonlinearly in $\hat{s}$ to give a better estimate of the signal than what is possible with the splines alone. However, a minor downside of using the Hilbert transform is that it creates artifacts in $\hat{f}$. Usually these are samples that are negative or very close to the Nyquist rate, and easily eliminated by setting them to zero. The $\hat{a}$ sequence generally does not present such artifacts.

In the following, exactly the same process as above is used to get the amplitude envelope and instantaneous frequency of the true signal.

G. Evolution of SEECR

As mentioned earlier, SEECR is the culmination of a sequence of intermediated methods [29–31]. Here, we briefly summarize the similarities and differences between SEECR and the preceding methods.

In Ref. [29], the simple problem of fitting data with a spline was considered. Thus, the signal model used was, $s(t; \bar{\alpha}, \bar{\tau}_a) = a(t; \bar{\alpha}, \bar{\tau}_a)$, with $a(t; \bar{\alpha}, \bar{\tau}_a)$ given by Eq. (9). PSO was proposed for optimizing the residual norm squared [Eq. (20)] over $\bar{\tau}_a$ without a monotonicity ($[\bar{\tau}_a]_{i>j} > [\bar{\tau}_a]_j$) constraint. (In addition, the variant of PSO used in Ref. [29] is different from the one used in SEECR.)

The signal model used here [Eq. (10)] was introduced in a more restricted form in Ref. [30]: it was assumed that $\phi(t) \to 2\pi f_0 t + \phi(t)$, with $\phi(t)$ changing over a much longer time scale than the period, $1/f_0$, of the carrier. This restriction allows the signal to be heterodyned, yielding the two quadratures $a(t)\cos\phi(t)$ and $a(t)\sin\phi(t)$. The method in Ref. [29] was then used to estimate the quadratures independently. While the importance of modeling both the amplitude and phase evolution of a signal through splines was emphasized in Ref. [30], the heterodyning approach is completely different from what is done in SEECR.

In Ref. [31], the signal model was generalized to essentially match Eq. (10). However, the initial phase parameter, ϕ_0, was not included in the model, which simplifies the steps involved in Sec. III B considerably. In addition $f(t)$ was modeled with a linear, not cubic, spline. The number of breakpoints, M and K, were not varied and model selection (see Sec. III E) was not used. The use of GCV was introduced but did not face the complication, described in Sec. III C, involved in meshing it with the minimization over ϕ_0.

IV. DESCRIPTION OF THE SIMULATIONS

We quantify the performance of SEECR using statistically independent simulated data realizations corresponding to the data model in Eq. (4). H_0 data realizations are drawn from a zero mean Gaussian white noise process with unit variance [i.e., an i.i.d $N(0, 1)$ sequence]. There is no loss of generality because the inner product in Eq. (5) is equivalent in the Fourier domain to the Euclidean inner product of a white noise sequence with a whitened signal. Since the choice of waveforms for unmodeled signals is arbitrary to begin with, they can be assumed to be those of the whitened signals.

A. Simulated signal waveforms

We use the following simulated signal waveforms, covering a wide range in the behavior of the amplitude envelope and instantaneous frequency. Each signal is assigned a label followed by pertinent information about it. For the signals where expressions for $a(t)$ and $\phi(t)$ are given, $s(t) = a(t)\sin(\phi(t))$. We have taken care to set some of the signal parameters, such as the start time or the carrier frequency, at values that are not related in a special way to the sampling grid in either the temporal or the Fourier domain. All data realizations containing the signals listed below have a duration of 2.0 sec with a sampling frequency of 4096 Hz.

TS: Transient sinusoid with $a(t) = 1$ for $t \in [0.4, 1.4]$ sec and zero otherwise. $\phi(t) = 2\pi f_0(t - t_0)$, with $f(t) = f_0 = 473.0$ Hz, and $t_0 = 0.4$ sec.

SG: Sine-Gaussian signal with constant $f(t) = f_0 = 204.8$ Hz and $a(t)$ having a Gaussian shape that is symmetric with respect to the midpoint of the signal. $a(t) = \exp(-(t - t_0)^2/(2 \times (\text{FWHM}/2.355)^2))$, for $t \in [0.4, 0.9]$ sec and zero otherwise. The peak of $a(t)$ is at $t_0 = 0.65$ sec and FWHM = 0.29 sec is its full width at half maximum. $\phi(t) = 2\pi f_0(t - t_0) + \pi/2$.

3PS: Monochromatic signal with three Gaussian peaks in the amplitude envelope. This signal is obtained by concatenating three SG signals. (The middle signal is the negative of the SG in order to reduce the effect of phase discontinuities at its boundaries.) $a(t) \neq 0$ for $t \in [0.3, 1.8]$ sec and zero otherwise.
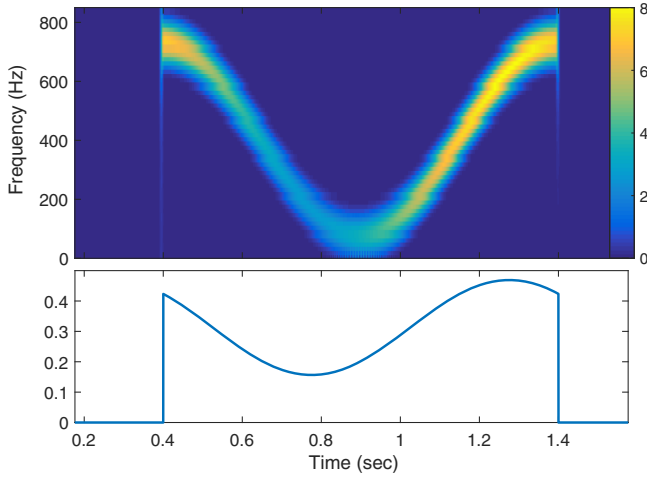
FIG. 1. (Top) Spectrogram of the CC signal showing the behavior of its instantaneous frequency $f(t)$. (Bottom) The amplitude envelope, $a(t)$, of the CC signal for SNR = 15. The locations of the minima in $f(t)$ and $a(t)$ do not coincide, leading to the lopsided distribution of signal power, with the part after the minimum in $f(t)$ being stronger.

LC: Linear chirp (quadratic phase) with constant amplitude. $\phi(t) = 2\pi(f_0 t + f_1 t^2)$, with $f_0 = 200$ Hz and $f_1 = 300$ Hz^2, and $a(t) = 1$ for $t \in [0.4, 1.4]$ sec and zero otherwise.

QC: Quadratic chirp (cubic phase) with constant amplitude as defined in Ref. [26]. $\phi(t) = 2\pi(f_0 t + f_1 t^3)$, where $f_0 = (2\pi)^{-1} 256$ Hz and $f_1 = (2\pi)^{-1}(512/3)$ Hz^3. $a(t) = 1$ for $t \in [0.4, 1.4]$ and zero otherwise. The start and end frequencies are 40.7 and 122.2 Hz respectively.

CC: Cosine phase chirp with cosine modulated amplitude as defined in Ref. [26]. $a(t) = 2 + \cos(2\pi f_0 (t - t_0) + \pi/4)$ for $t \in [0.4, 1.4]$ sec and zero otherwise. Here, $f_0 = 1$ Hz and $t_0 = 0.4$ sec. $\phi(t) = \phi_m \sin(2\pi f_0 (t - t_0)) + 2\pi f_1 (t - t_0)$, with $\phi_m = 1024/\pi$ rad and $f_1 = 400$ Hz. The behavior of $f(t)$ can be seen from its spectrogram in Fig. 1. (See Appendix B for the precise definition of a spectrogram as used in this paper.)

s11WW: A CCSN waveform obtained from Ref. [48] corresponding to the acoustic supernova model [16]. The waveform time series was antialiased and downsampled to $f_s = 4096$ Hz, leaving no discernible changes as most of the power in the signal lies below ~1.5 kHz. Both $a(t)$ and $f(t)$ have a complex evolution for this waveform due to the simultaneous presence of multiple chirping components as can be seen from the spectrogram of this signal in Fig. 2. However, there is a single component that dominates in power, making the single chirp model assumed in SEECR a good fit. In each data realization, the signal starts at $t = 0.4$ sec and terminates at 1.173 sec.
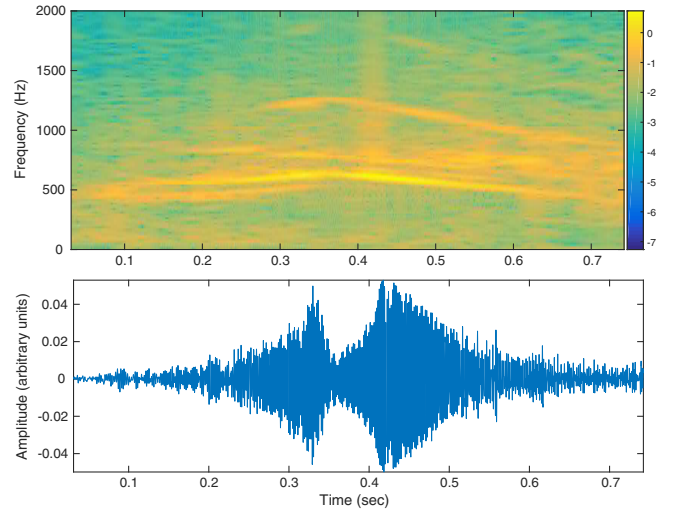

FIG. 2. (Top) Spectrogram of the s11WW signal with the magnitude shown on a log scale in order to elucidate the multiple chirping components more clearly. (Bottom) The signal time series where the amplitude has been scaled such that SNR = 1.

When constructing an H_1 data realization, the signal amplitude is normalized such that it has a certain matched filtering SNR. The SNR of a signal characterizes the performance of the optimal statistic, namely the log-likelihood ratio (LLR), for the binary hypotheses test where there is only one signal waveform and it is completely known *a priori*. For the Gaussian white noise process used in the simulations,

$$\text{SNR} = \frac{E[\text{LLR}|H_1] - E[\text{LLR}|H_0]}{[E[(\text{LLR} - E[\text{LLR}|H_0])^2|H_0]]^{1/2}},$$

$$= \left[\sum_{i=0}^{N-1} s_i^2\right]^{1/2}, \quad (49)$$

where $E[\text{LLR}|H_i]$, $i = 0$ or 1, denotes expectation under hypothesis H_i. For generating data realizations under H_1, we use three SNR values, SNR = 10, 12, 15, for each of the simulated signals.

B. GW150914 analysis

The simulated waveforms listed so far have durations of ≥ 1 sec, with the exception of SG that has a duration of 0.5 sec. Although the main target for SEECR are signals in this duration range, it is interesting to quantify its performance for a significantly shorter chirp.

For this purpose, we simply use the real event, GW150914, which furnishes a chirp of duration < 0.2 sec. However, GW150914 had an exceptionally high observed network SNR of 24, and a single detector SNR of ≈ 20 [49] in the Hanford detector, making it an easy case for burst search algorithms. To test if SEECR could have detected this signal at weaker strengths, we use the real GW150914 data as a seed to generate new realizations in which the observed SNR is reduced to ≈ 10.
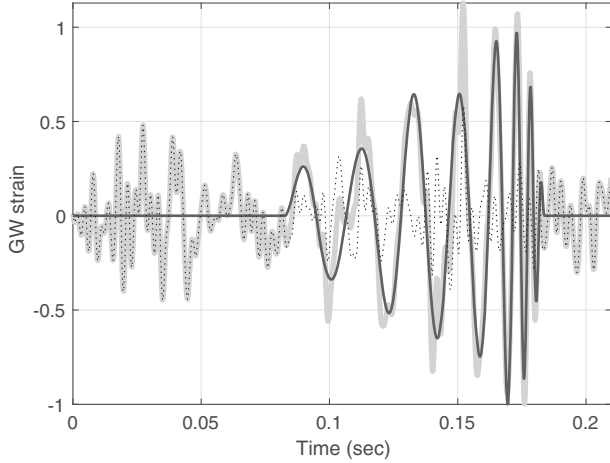
FIG. 3.   The thick gray curve is the GW150914 data from the Hanford detector. The solid black curve is the signal estimated by SEECR. The dashed curve shows the residual after subtracting the estimated signal from the data.
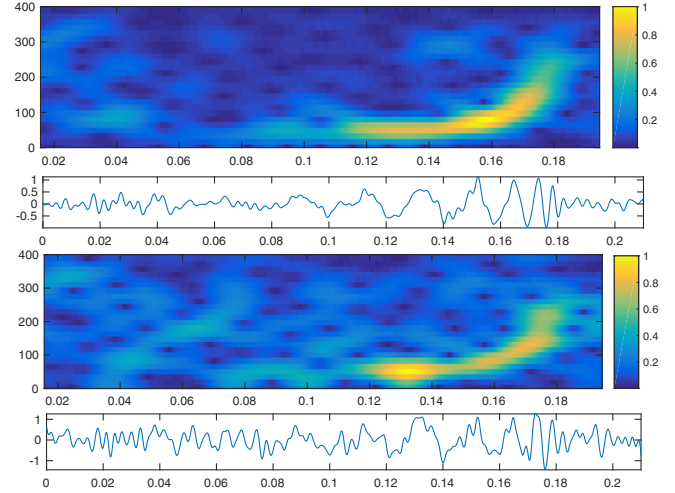

FIG. 4.   GW150914 data from the Hanford detector compared with a realization obtained by adding excess pseudorandom white noise. The first and second panels from the top show the spectrogram, obtained with a window length of 128 and overlap between consecutive windows of 127 samples, and the data time series respectively. The bottom two panels are the corresponding plots for a data realization where the observed SNR has been reduced by a factor of 2. In all panels, the horizontal axis shows time in seconds. The vertical axes in the case of spectrograms shows frequency (Hz). In the time series plots, the vertical axis shows the (whitened) GW strain ($\times 10^{-21}$).

First, we take the time series from the Hanford detector, shown in Fig. 1 of Ref. [2] and estimate the standard deviation of the noise in the data. This is done by estimating the signal using SEECR and subtracting it from the data to obtain the residual. Figure 3 shows the data, the estimated signal, and the residual. The residual has a sample standard deviation of 0.16.

Next, a realization of pseudorandom noise is generated and added to the original data. The noise realization is first generated as white noise with unit variance and then low-pass filtered, using an order 40 finite impulse response filter, to the band [0, 450] Hz. The resulting time series, having a standard deviation of σ_{filt}, is then scaled by $[\sqrt{3} \times 0.16/\sigma_{\mathrm{filt}}]$. Modulo the sampling error in the standard deviation estimate, the observed SNR of the signal in the new realization is reduced by a factor of 2. To generate H_0 data, we follow the same procedure but use a scaling factor of $2 \times 0.16/\sigma_{\mathrm{filt}}$.

As with the simulated signals, independent realizations of H_1 and H_0 data are generated for GW150914. Each data realization has a duration of 0.21 sec with a sampling frequency of 4096 Hz. Figure 4 compares the original data with one such realization.

C. SEECR parameter settings

The principal user-determined parameters governing SEECR are the number of breakpoints, M and K, for the amplitude envelope and instantaneous frequency splines respectively. The user provides a set of values for M and K and, as described in Sec. III E, AIC is used to pick the best combination.

In principle, one need only specify the maximum values of M and K and let AIC examine all the integers below them. However, this is wasteful since the signal estimates, and hence the AIC values, may not differ much between

nearby models. This is particularly true at higher values for the number of breakpoints where nearby models start differing less and less in their fit quality. Hence, computational costs can be reduced substantially by spacing models out judiciously.

Based on the above and keeping computational costs in mind, we arrived at the sets $\{5, 6, 7, 9, 11\}$ and $\{3, 4, 5, 7\}$ for M and K respectively, resulting in 20 different models, that are kept fixed throughout this paper.

Besides the above parameters, there are the parameters associated with PSO and the range, $[\nu_{\min}, \nu_{\max}]$, for the K instantaneous frequency values ν_i, that it needs to search. (The range for the amplitude envelope end breakpoint parameters is set so that the entire data segment is covered.) A virtue of the PSO algorithm is the robustness of its parameter settings. This allows us to simply keep the same settings [50] as used in Ref. [51], to which we refer the reader for further details. For the above set of K values, the dimensionality of the search space for PSO ranges between 6 and 14.

Like all stochastic global optimizers, PSO is not guaranteed to converge to the global minimum. However, the probability of success can be increased exponentially by doing multiple runs of PSO, with statistically independent initial states, on the same data realization and picking the run that returns the best fitness value. The number of independent PSO runs is set to eight in this paper.

We keep ν_{max} slightly below the Nyquist frequency of the data to prevent too many PSO particles from escaping the search region to explore physically invalid frequencies. Except Sec. VII, where $\nu_{\text{max}} = 510$ Hz, we set $\nu_{\text{min}} = 0$ and $\nu_{\text{max}} = 2000$ Hz.

Finally, we use splines of order 4 (cubic splines) for both the amplitude envelope and instantaneous frequency.

V. RESULTS

The presentation of the simulation results is organized as follows. In Sec. VA, we focus on the detection performance of SEECR. Section VB describes its performance in estimating the amplitude envelope and instantaneous frequency of a signal. The results in Sec. VA and Sec. VB use the set of signals described in Sec. IVA with 500 realizations of H_0 and a minimum of 50 realizations of H_1 data for each signal and each SNR. Section VC presents results from the GW150914 analysis described in Sec. IVB. For these results we use 100 H_0 and 50 H_1 data realizations.

A. Detection performance

For SEECR to function as a detector, we must choose a detection statistic, and a natural choice for it is the LLR evaluated at the best-fit model. Following the discussion in Sec. III E regarding the relation between the log-likelihood and $\hat{F}_{\hat{M},\hat{K}}$ [defined in Eq. (43)],

$$\text{LLR} = \|\bar{y}\|^2 - \hat{F}_{\hat{M},\hat{K}}. \tag{50}$$

To obtain the threshold corresponding to a given false alarm probability, we estimate the probability density function (PDF) of LLR from the H_0 data realizations. Figure 5 shows the estimated PDF along with the best-fit lognormal PDF. We pick the lognormal PDF,

$$p(x) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right), \tag{51}$$

because it provides a good match to the asymmetry of the estimated distribution around its mode, as well as its heavy tail, with only two free parameters.

We quote detection probabilities at two values of the false alarm probability: $1/500 = 2 \times 10^{-3}$ and 2×10^{-4}. Since each data realization is 2 sec long, the values of the false alarm rate (FAR) are 10^{-3} and 10^{-4} events/ sec respectively. (The resulting FAR for coincidence-based detection between a pair of GW detectors is discussed in Sec. VIII.) The corresponding thresholds on LLR obtained from the best-fit lognormal are 80.3 and 86.9 respectively.

Table I reports the detection probabilities for the simulated signals in Sec. IVA at the different SNR values used in this study. The error interval associated with each detection probability corresponds to $\pm 1\sigma$, where
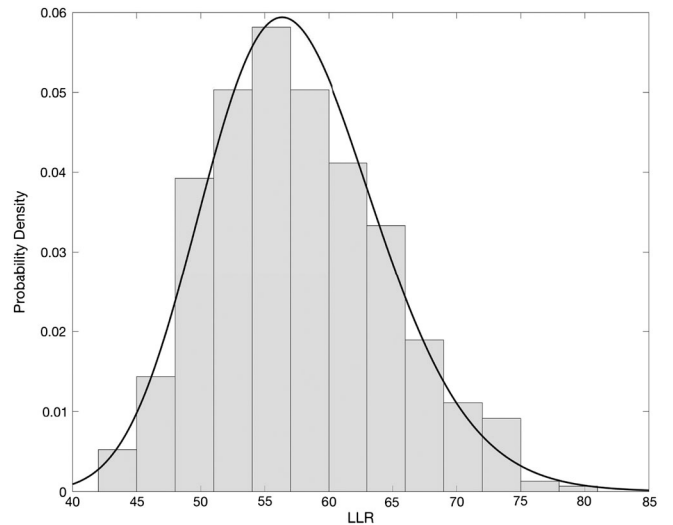

FIG. 5. Estimated distribution of the SEECR detection statistic, LLR, under the null hypothesis. The distribution is estimated from 500 realizations of an i.i.d $N(0, 1)$ sequence with 8192 samples. The bars show the histogram, with the count in each bin normalized to represent the PDF. The solid curve shows the best-fit lognormal PDF, obtained for $\mu = 4.04563$, and $\sigma = 0.11836$.

$$\sigma = [p(1 - p)/N_{\text{trials}}]^{1/2}, \tag{52}$$

with p being the estimated detection probability and N_{trials} being the number of H_1 data realizations used. Note that the estimated detection probability does not have a normal distribution, and the error interval above is not strictly appropriate, for p close to unity or zero. For extreme values of p, one may use the Clopper-Pearson confidence interval [52] to assess the error in p. In our case, the extreme value of concern is $p = 1$, for which the interval is given by $((\alpha/2)^{(1/N_{\text{trials}})}, 1)$, where α is the confidence level. For $\alpha = 0.95$ and $N_{\text{trials}} \geq 50$, the interval is $(\geq 0.9852, 1)$. Here, and in the rest of the paper, a quoted estimated detection probability of unity is understood to refer to the above confidence interval.

TABLE I. Estimated detection probabilities, and their 1σ error intervals at two different false alarm rates. The detection probability for each SNR value is estimated using 50 H_1 data realizations except for QC and CC, where the number of realizations is 120. The detection probability at SNR = 15 is unity at both FARs for every signal and, hence, not listed explicitly.

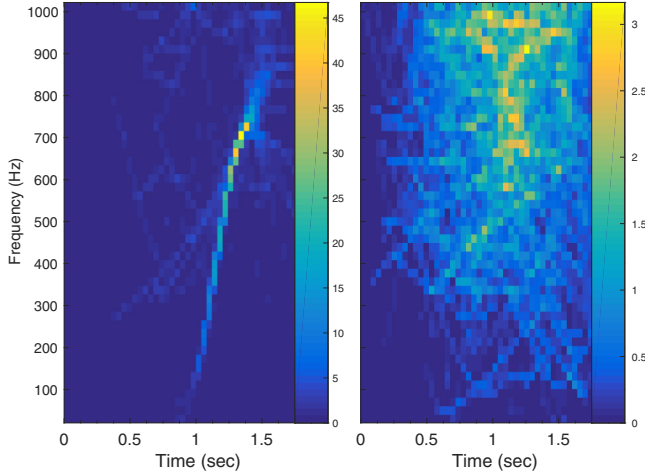|	FAR = 10^{-3} events/ sec		FAR = 10^{-4} events/ sec	
Signal	SNR = 10	SNR = 12	SNR = 10	SNR = 12
TS	0.98 ± 0.02	1.00 ± 0.00	0.96 ± 0.03	1.00 ± 0.00
SG	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.02	1.00 ± 0.00
3PS	0.84 ± 0.05	0.94 ± 0.03	0.82 ± 0.05	0.92 ± 0.04
LC	0.52 ± 0.07	0.90 ± 0.04	0.40 ± 0.07	0.84 ± 0.05
QC	0.61 ± 0.04	0.97 ± 0.01	0.48 ± 0.05	0.95 ± 0.02
CC	0.22 ± 0.04	0.68 ± 0.04	0.092 ± 0.03	0.53 ± 0.05
s11WW	0.86 ± 0.05	0.98 ± 0.02	0.72 ± 0.06	0.98 ± 0.02

FIG. 6. Two-dimensional (2D) histogram of estimated instantaneous frequencies for (left panel) H_1 data realizations containing the CC signal with SNR = 10, and (right panel) H_0 data. Each histogram is constructed by plotting all the estimated frequencies and counting the number of plotted points in a regular grid of 2D bins. There are 50 bins along each dimension. For the left panel, only those data realizations are included that had LLR values less than the detection threshold. All realizations of H_0 data are used for the right panel. The counts in each panel are normalized by the respective number of trials used.

We see that, at SNR = 10 and a FAR of 10^{-3} events/ sec, SEECR attains a detection probability of $\gtrsim 0.5$ for all the signals except CC. In itself, the reduced power for this signal is not surprising given that it has the most extreme amplitude and instantaneous frequency variation. However, an additional reason appears to be the lopsided distribution of signal power as seen in Fig. 1. Its effect on the estimated signal is shown in Fig. 6. We see that the estimated instantaneous frequency tends to match only the part of the signal that is louder, and the initial half of the signal is missed completely.

Figure 6 also shows the estimated instantaneous frequencies from H_0 data. It is interesting that the imprint of the signal on the distribution of estimated instantaneous frequencies is quite clear even at SNR = 10.

The detection probability of the CC signal is reduced substantially for the FAR of 10^{-4} events/ sec but then climbs to 0.53 ± 0.05 at SNR = 12. SEECR achieves a detection probability $\gtrsim 0.8$ for all the other signals at SNR = 12 for this FAR.

B. Estimation performance

Gauging the performance of any method on the estimation of chirp signals requires metrics that go beyond the simple mean squared error (MSE) $\|\bar{s} - \hat{s}\|^2$ between the true signal, $\bar{s}$, and its estimate $\hat{s}$. This is because, as discussed in Sec. III D, the error in estimating the amplitude envelope $a(t)$ of a chirp can be significantly higher than that for its instantaneous frequency $f(t)$ but they are conflated in the MSE without any kind of weighting.

Moreover, going by the case of binary inspiral signals, most of the physically important information carried by a GW chirp is likely to reside in $f(t)$ and one would like to study the error in estimating it independently of $a(t)$. This motivates the introduction of a set of metrics to separately quantify the estimation performance for $a(t)$ and $f(t)$.

1. Estimation metrics

The metrics proposed here are based on the physically relevant information one would like to extract from any estimated signal. At the most basic level, this consists of the time of arrival, the duration, and as much of $f(t)$ as possible.

In a parametric search method, the time of arrival and duration are explicit parameters of the signal model and are measured as such. In the case of unmodeled chirps, however, the measured quantities are $a(t)$ and $f(t)$, and the time of arrival and duration must be derived from them. Among the two, it is natural to use $a(t)$ for this inference but due account must be taken of estimation error, which can be expected to be higher where the true $a(t)$ is smaller. Therefore, for example, simply using the start time of the estimated $a(t)$ as the time of arrival is not a good idea because the start of a signal is precisely where the true $a(t)$ decays to zero and the estimation error is likely to be highest.

Consider a finite duration amplitude envelope $a(t)$, with $a(t) = 0$ for $t \notin [t_1, t_2]$. Given that $a(t) \geq 0$ everywhere and integrable, one can normalize it to construct a PDF over t,

$$p_a(t) = \frac{a(t)}{\int_{t_1}^{t_2} dt a(t)}. \tag{53}$$

We define the time of arrival, denoted by t_a, as the median of this PDF,

$$\int_{t_1}^{t_a} p_a(t) dt = \frac{1}{2}. \tag{54}$$

The duration t_D is defined as the interquartile range (IQR)—the difference between the first and the third quartiles—of the PDF,

$$t_D = q(0.75) - q(0.25), \tag{55}$$

$$\int_{t_1}^{q(\alpha)} dt p_a(t) = \alpha. \tag{56}$$

The median is preferable to the mean of $p_a(t)$ as an estimator of t_a because it is more robust against the increased error in the tails of $p_a(t)$ near the start and end of a signal. Generally, these errors need not be equal at the two ends, giving rise to a larger bias in the mean than in the median. For the same reason, the IQR is a more robust measure of the duration than the standard deviation. For

reference, the IQR of a normal distribution with standard deviation σ is 1.34σ.

We denote the metrics associated with the time of arrival and duration by δt_a and δD respectively. The metric δt_a is simply the offset

$$\delta t_a = \hat{t}_a - t_a, \tag{57}$$

where $\hat{t}_a$ and t_a are the times of arrival associated with the estimated and true amplitude envelopes respectively. Similarly, the metric δD is

$$\delta D = \hat{t}_D - t_D, \tag{58}$$

where $\hat{t}_D$ and t_D are the interquartile ranges associated with the estimated and true amplitude envelopes respectively.

For $f(t)$, we adopt the following metric. Let $\mathcal{F}$ be the set of time samples within the start and stop times of the true signal. Let $\hat{f}$ and $\bar{f}$ be the estimated and true instantaneous frequency sequences respectively. Note that the set of time instants over which each is supported will not be identical in general. Define

$$\mathcal{G}(\epsilon_f) = \{i | t_i \in \mathcal{F}, |[\bar{f}]_i - [\hat{f}]_i| \le \epsilon_f\}. \tag{59}$$

In words, $\mathcal{G}(\epsilon_f)$ is that part of the true signal where the estimated and true instantaneous frequencies differ by less than $\pm\epsilon_f$. The metric is then defined as,

$$\rho^2(\epsilon_f) = \frac{\sum_{i \in \mathcal{G}(\epsilon_f)} a_i^2}{\sum_{i \in \mathcal{F}} a_i^2}. \tag{60}$$

The numerator is the squared norm of the true amplitude envelope, $\bar{a}$, restricted to the samples in the set $\mathcal{G}(\epsilon_f)$. The denominator is the squared norm of the full $\bar{a}$.

The metric $\rho(\epsilon_f)$ takes account of the fact that the error in frequency estimation can be expected to be larger where the true signal amplitude is weaker. Thus, we must somehow weight the error by the instantaneous amplitude of the signal before combining them. However, a straightforward weighted average of $\hat{f} - \bar{f}$, with the weight given by $p_a(t)$, is not found to perform well. This is because SEECR does not put any constraint on how fast $f(t)$ can vary and this allows the estimated frequency to change rapidly near the beginning and end of a signal where its true amplitude is small (or zero). (This effect is visible as a flaring of the estimates in Fig. 6 around the end of the signal.) The resulting errors turn out to be too large to be compensated by the decaying amplitude envelope near these locations. By confining our attention to the interval $\mathcal{G}(\epsilon_f)$, where the estimated and true instantaneous frequencies agree well, and constructing the metric out of the amplitude envelope, we cut out these spurious end effects and fold in the required weighting at the same time.
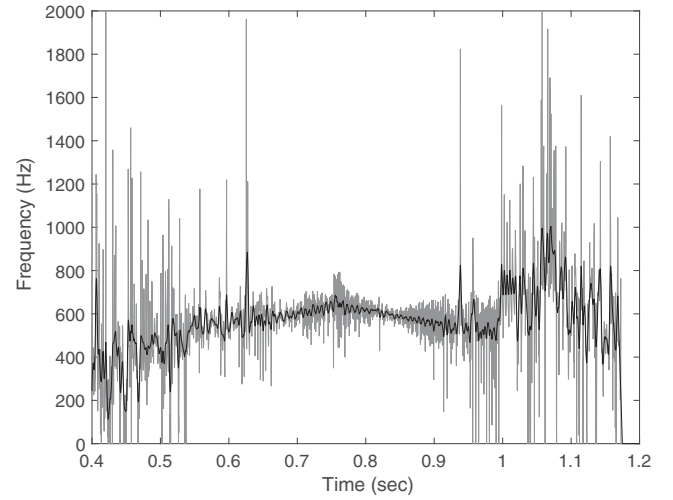

FIG. 7. The instantaneous frequency, $\bar{f}$, of the s11WW signal (in gray) and its running average (in black). $\bar{f}$ is obtained from the analytic form of the signal as described in Sec. III F. The large spikes in $\bar{f}$, that cross a band of $\approx \pm 300$ Hz around the running average, are artifacts of this process and should be ignored. The running average is computed over a block of ten samples. The slope of the trend changes from positive to negative somewhere in the interval [0.75, 0.77] sec. Over this interval, the sample standard deviation of $\bar{f}$ is 98.3 Hz.

While $\rho(\epsilon_f)$ as defined above is appropriate for a smoothly evolving instantaneous frequency, it needs to be modified for signals where this is not true. As can be seen from Fig. 7, the s11WW signal presents such a situation, where, In addition to an underlying trend, a fair amount of scatter (excluding the spurious spikes) is evident in the true instantaneous frequency. The trend can be elucidated by taking a running average, which is also shown in the figure. The scatter must be accounted for when comparing estimated and true instantaneous frequencies because no semiparametric method, such as SEECR, can hope to match the scatter in detail without having a degree of freedom that is so large as to make it practically useless.

For the s11WW signal, therefore, $\rho(\epsilon_f)$ is calculated with $\bar{f}$ replaced in Eq. (59) by its running average. It should be noted that setting ϵ_f to be less than the standard deviation of the running average itself will again show up as an apparent loss in performance. The running average used here is computed over a block of ten samples, and given that the standard deviation of f_i around the running average is $\approx$100 Hz, the standard deviation of the running average itself is $\approx$30 Hz.

2. Metric distributions

Figures 8, 9, and 10 summarize the sampling distributions of δt_a, δD, and $\rho(\epsilon_f)$ respectively in the form of box-and-whisker plots. For each box, the "⊙" mark indicates the median of the distribution, while the bottom and top
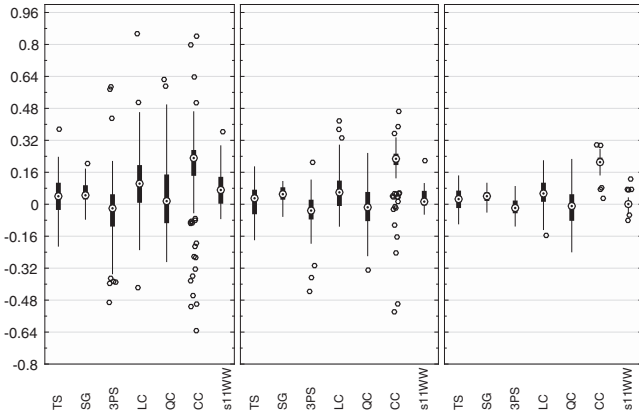
FIG. 8. Box-and-whisker plots of the metric δt_a. Each box-and-whisker summarizes the sampling distribution, as described in Sec. V B 2, of δt_a for one signal and one SNR. The name of the signal is shown on the x axis. The box-and-whisker plots corresponding to the same SNR are grouped in one panel. From left to right, the panels correspond to SNR values of $[10, 12, 15]$ respectively.
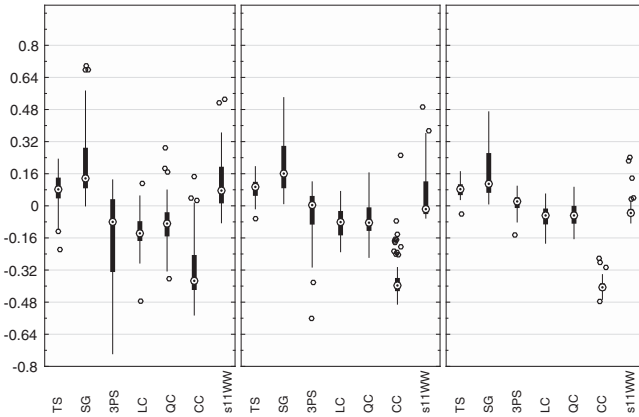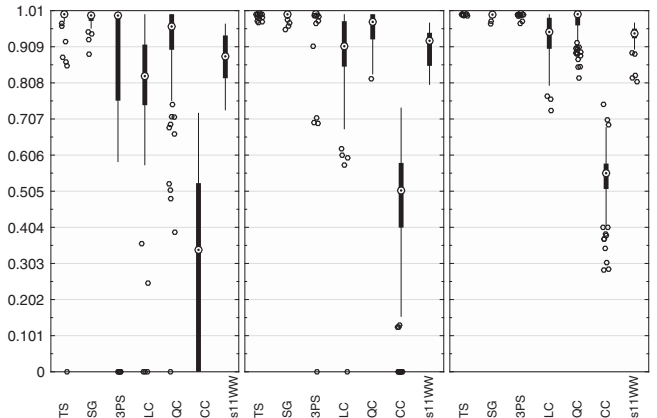

FIG. 10. Box-and-whisker plots of the metric $\rho(\epsilon_f)$ for $\epsilon_f = 64$ Hz. Each box-and-whisker summarizes the sampling distribution, as described in Sec. V B 2, of $\rho(\epsilon_f)$ for one signal and one SNR. The name of the signal is shown on the x axis. The box-and-whisker plots corresponding to the same SNR are grouped in one panel. From left to right, the panels correspond to SNR values of $[10, 12, 15]$ respectively.

FIG. 9. Box-and-whisker plots of the metric δD. Each box-and-whisker summarizes the sampling distribution, as described in Sec. V B 2, of δD for one signal and one SNR. The name of the signal is shown on the x axis. The box-and-whisker plots corresponding to the same SNR are grouped in one panel. From left to right, the panels correspond to SNR values of $[10, 12, 15]$ respectively.

edges correspond to its 25th and 75th percentiles respectively. Thus, the length of a box corresponds to the IQR and contains 50% of the probability. The whiskers (thin lines) extend to the extreme data points that are not outliers. A sample value is deemed to be an outlier if it is separated from the median by more than twice the IQR. (Outliers are shown as open circles that are dithered horizontally by small amounts to aid visual clarity.)

For reference, the length of each whisker is 2.68σ for a normal distribution having a standard deviation σ. As such, more than 99% of the probability under a normal distribution is contained between the ends of the two whiskers.

While this is also true for the observed distributions in general, there are some exceptions. The correct probability coverage in such cases can be obtained by simply counting the number of outliers in the plot and subtracting it from the number of trials (see Table I for the exact number of trials).

The distributions of δt_a and δD show that reducing the errors in time of arrival and duration, if they are obtained from the estimated amplitude envelope, to levels where they are significantly smaller than the duration of the signals requires SNR$\gtrsim 15$. The lowest error at this SNR is in the range ± 0.08 sec, with a probability of $\gtrsim 0.99$, for the s11WW signal (1 sec duration). With the same probability, QC shows the broadest range for the error at about ± 0.24 sec.

While the time of arrival is generally estimated with negligible bias, it is significant for the CC signal due to its partial reconstruction (see Fig. 6). However, the bias is fairly independent of SNR and, hence, will not affect the offsets between the estimated times of arrivals for CC signals in a network of detectors. Excluding CC, the largest range (≈ 0.99 probability) in time of arrival error at SNR $= 10$, which occurs for the QC signal, is ± 0.5 sec.

The bias in duration estimation, on the other hand, is non-negligible for several signals even at SNR $= 15$. The anomaly in the duration estimation is the SG signal, for which the error has a distinctly asymmetrical distribution around the median. This is because the estimated amplitude envelope for this signal has a peak that is well localized around that of the true signal, as evident from its δt_a distribution, but it is biased away from having a symmetrical shape around the peak. This illustrates the problem, mentioned earlier, with using the start and stop times of the estimated amplitude envelope directly for deriving duration and time of arrival.
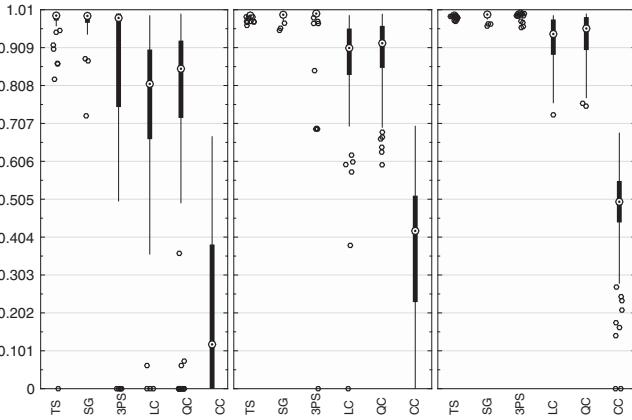
FIG. 11. Box-and-whisker plots of the metric $\rho(\epsilon_f)$ for $\epsilon_f = 4$ Hz. Each box-and-whisker summarizes the sampling distribution, as described in Sec. V B 2, of $\rho(\epsilon_f)$ for one signal and one SNR. The name of the signal is shown on the x axis. The box-and-whisker plots corresponding to the same SNR are grouped in one panel. From left to right, the panels correspond to SNR values of $[10, 12, 15]$ respectively.

From Fig. 10, we see that with a tolerance of $\epsilon_f = 64$ Hz in frequency estimation error, SEECR is able to recover $\gtrsim 60\%$ of the frequency evolution at the lowest SNR with $\gtrsim 0.99$ probability. The only exception is the CC signal and the reason is again its partial reconstruction. Excluding this signal, the whiskers of all the distributions lie above $\approx 70\%$ at SNR = 12 and $\approx 80\%$ at SNR = 15.

Figure 11 shows the distribution of $\rho(\epsilon_f)$ for a much tighter tolerance of $\epsilon_f = 4$ Hz. The distribution for the s11WW signal is not included in this figure because ϵ_f is smaller than the standard deviation of the running average of its instantaneous frequency (see Sec. V B 1). The changes in the distributions of $\rho(\epsilon_f)$ are relatively small for all the other signals and $\gtrsim 75\%$ of frequency evolution is still recovered at SNR = 15 with $\gtrsim 0.99$ probability.

C. GW150914 analysis

As described in Sec. IV B, pseudorandom noise was added to GW150914 data to reduce the observed SNR of the signal by a factor of 2. Figure 12 shows the cumulative distribution function of the LLR [Eq. (50)] under H_0, along with a lognormal fit, and H_1. The two distributions do not overlap. Based on the lognormal fit, SEECR can detect a signal like GW150914 at an SNR = 10 with a probability of unity even at a false alarm probability of 2×10^{-16} (corresponding to a threshold of LLR = 50).

Figures 13 and 14 show the 2D histograms, following the construction described in Fig. 6, of all the estimated amplitude envelopes and instantaneous frequencies respectively, along with box-and-whisker plots of the metrics δt_a and $\rho(\epsilon_f)$. Comparison of the 2D histograms clearly illustrates the discussion in Sec. III D that the estimation
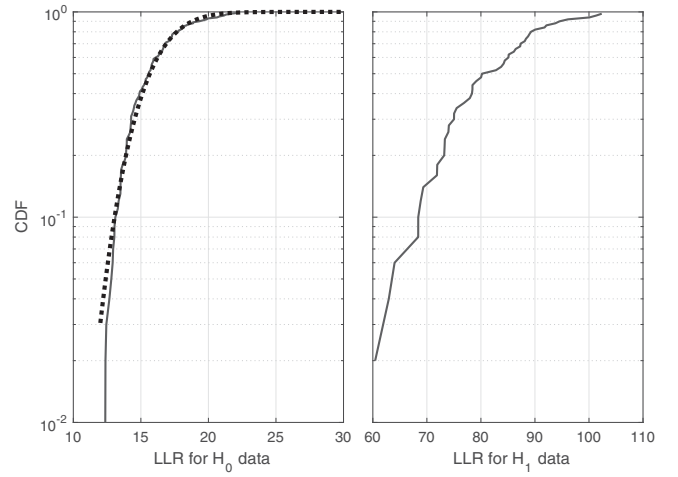

FIG. 12. Estimated cumulative distribution functions (CDFs) of the LLR obtained from the SEECR analysis of GW150914 data. The CDF on the left (solid curve) is estimated from noise-only data (generated independently of GW150914 data) while the one on the right corresponds to the real GW150914 data with added pseudorandom noise. Also shown on the left (dotted curve) is the CDF of the best lognormal fit.

error for the amplitude envelope of a chirp is significantly higher than that for its instantaneous frequency.

From the δt_a distribution, we conclude that, with a probability of about ≈ 0.5 and ≈ 0.99 respectively, SEECR
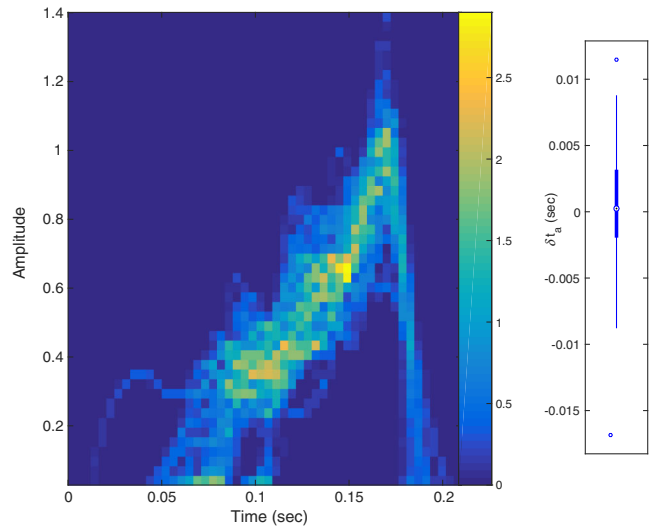

FIG. 13. The 2D histogram of estimated amplitude envelopes (left panel) using 50 data realizations of GW150914 data with added pseudorandom noise. The histogram is constructed by plotting all the estimated amplitude envelopes and counting the number of plotted points in a regular grid of 2D bins. There are 50 bins along each dimension. The counts have been normalized by the number of realizations used. The distribution of the metric, δt_a, is shown as a box-and-whisker plot (right panel). The true time of arrival was taken to be the one associated with the signal estimated by SEECR from the original GW150914 data.
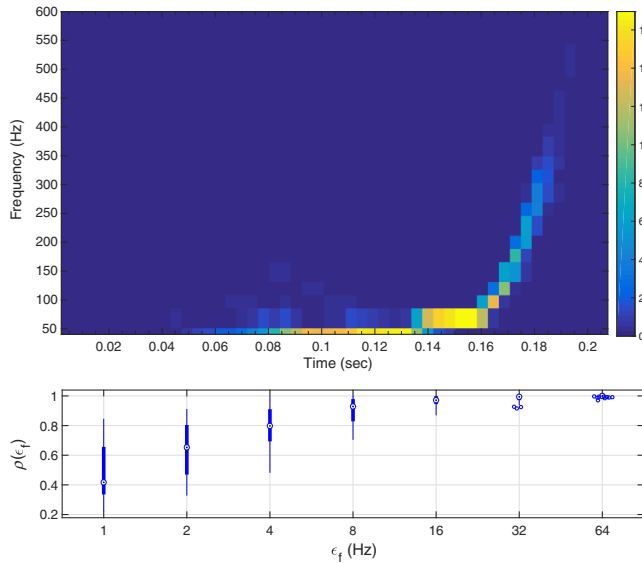
FIG. 14. The 2D histogram of estimated instantaneous frequencies (top panel) using 50 data realizations of GW150914 data with added pseudorandom noise. The histogram is constructed by plotting all the estimated instantaneous frequencies and counting the number of plotted points in a regular grid of 2D bins. There are 50 bins along each dimension. The counts have been normalized by the number of realizations used. The distribution of the metric, $\rho(\epsilon_f)$, is shown as box-and-whisker plots (bottom panel) for $\epsilon_f \in \{1, 2, 4, 8, 16, 32, 64\}$ Hz. For the calculation of $\rho(\epsilon_f)$, the true amplitude envelope was taken to be that of the signal estimated by SEECR from the original GW150914 data.

was able to pin down the time of arrival of the signal to within about ± 2.5 and ± 10 msec. This is a vast improvement over the situation seen in Sec. V B 2 for the case of long-duration signals. The $\rho(\epsilon_f)$ distribution, on the other hand, shows a worsening relative to the longer duration signals. For example, compared to the SG signal at SNR $= 10$, which shows $\gtrsim 90\%$ recovery with 0.99 probability for $\epsilon_f = 4$ Hz, the same performance for GW150914 requires an error tolerance of $\epsilon_f \gtrsim 16$ Hz.

VI. COMPARISON WITH TIME-FREQUENCY CLUSTERING

All of the principal search algorithms used in LIGO for GW burst search [12,53,54] use some form of time-frequency (or time-scale) clustering. It is assumed that the presence of a signal in noisy data produces areas of locally high power, or *clusters*, in the time-frequency plane. Depending on the properties used for distinguishing between signal and noise-induced clusters, there is a wide variation in how clustering is implemented, ranging from a nearest-neighbor-based approach [55] to a proximity prior [53].

For a given SNR, the sensitivity of any clustering-based method is naturally lower for signals that do not produce strong clusters. This is a particularly relevant issue for
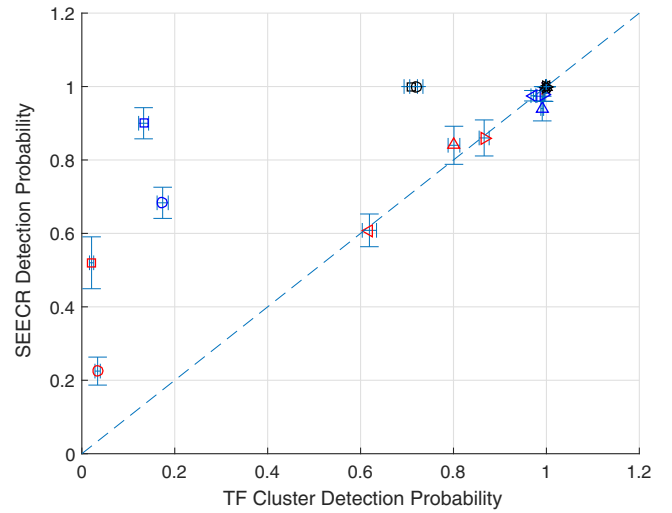

FIG. 15. Scatterplot of detection probability attained by time-frequency clustering (x axis) and SEECR (y axis) for the simulated signals (see Sec. IV) and SNR $= 10$, 12, 15. (The axes have been extended beyond a probability of unity for clarity.) The marker shapes correspond to the signal waveforms as follows: TS ($\bullet$), SG ($*$), 3PS ($\triangle$), LC ($\square$), QC ($\triangleleft$), CC ($\circ$), s11WW ($\triangleright$). The color of a marker indicates the SNR with the correspondence 10 (red), 12 (blue), and 15 (black). The error bars in each direction correspond to the respective 1σ intervals [cf., Eq. (52)].

chirps since they spread their total energy over an extended track. Therefore, it is interesting to compare the performance of SEECR with time-frequency clustering.

Since a full-fledged comparison with the search methods used in LIGO is outside the scope of this paper, we construct an *ad hoc* clustering-based search method that is simpler but, at the same time, captures the principal features of clustering used in the more sophisticated methods. We refer the reader to Appendix D for a description of the clustering-based search method. Here, we focus entirely on the results obtained with this method and its comparison with SEECR.

To quantify the performance of the clustering-based search method, we generate data realizations in exactly the same way as described in Sec. IV. However, due to the use of multiresolution analysis (see Appendix D), the overall FAR is split across the different resolution levels and, consequently, a much larger number of data realizations is required to reduce sampling errors. Consequently, we generate 10^4 and 10^3 H_0 and H_1 data realizations respectively. For the same reason, we only compare the clustering-based method and SEECR at the larger FAR of 10^{-3} events/sec.

Figure 15 shows a scatterplot of the detection probability attained by SEECR (from Table I) and the clustering-based method across all signals and SNR values. We see from the points that are far away from the line of equal detection probabilities that the performance of SEECR is significantly
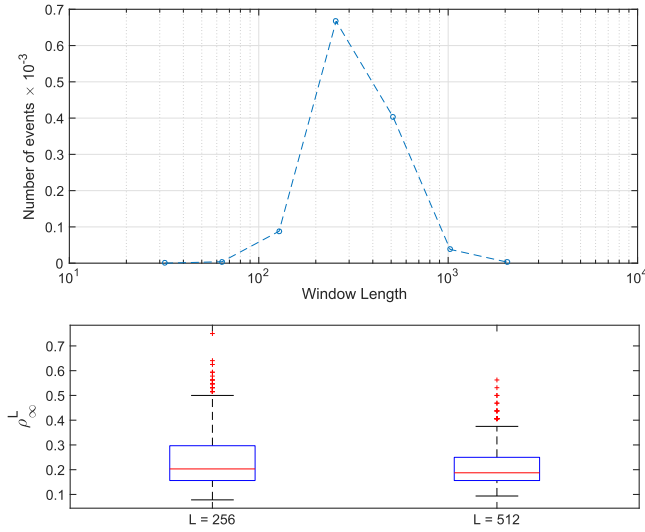
FIG. 16. The top panel shows the number of time-frequency events found, as a fraction of the number of data realizations (= 1000), for each of the window lengths, $L = 2^n$, $n = 5, 6, …, 11$, used in the clustering-based search method. Each data realization contains the LC signal with an SNR = 15. (The dashed line is included as a visual aid only, with the actual data points shown as open circles.) The bottom panel shows the distribution of ρ_∞^L for $L = 256$ and $L = 512$, the window lengths that produce the bulk of detected events, in the form of box-and-whisker plots. In these plots, the median is indicated by the red line in each box. The edges of each box and the size of the whisker carry the same meaning as described in Sec. V B 2. Outliers are denoted by the "+" marker.

better than clustering for the CC and LC signals. For the remaining signals, the two have essentially the same performance.

At SNR = 15, the detection probabilities attained by clustering for the LC and CC signals are 0.71 ± 0.014 and 0.72 ± 0.014 respectively while they are unity for SEECR in both cases. The performance of clustering worsens rapidly for these signals as SNR is reduced, with the detection probabilities at SNR = 12 being 0.133 ± 0.011 and 0.174 ± 0.011 for LC and CC respectively. (The corresponding probabilities are 0.90 ± 0.04 and 0.68 ± 0.04 for SEECR.) While a reduction in performance of clustering is expected, due to the spreading of signal power across a track, the extent to which it degrades for a simple signal such as LC is quite surprising.

While clustering is a detection, not an estimation, method, estimation is possible as a follow-up step to clustering-based detections. However, if the estimation algorithm focuses on only the time-frequency regions identified as significant by the clustering step, the errors in the estimation can become quite large. This is evident from Fig. 16 where we have taken the case of data realizations containing the LC signal at SNR = 15 and analyzed the associated time-frequency events as described

below. (See Appendix C for the definition of a time-frequency event.)

Let C_0^L be the set of spectrogram columns constituting the support of the true signal for window length L, and let C^L be the set of columns constituting an event. The ratio $n(C^L \cap C_0^L)/n(C_0^L)$, where $n(A)$ is the cardinality of a set A, is a simple measure of how well clustering can indicate the time-frequency region for follow-up analysis by estimation algorithms. For a signal such as LC that has a constant amplitude envelope over its entire duration, this ratio is equivalent to the metric $\rho(\epsilon_f)$ defined in Eq. (60) but with ϵ_f set to be the entire frequency range of the spectrogram. To indicate this connection, we denote the ratio above as ρ_∞^L.

As can be seen from the box-and-whisker plots in Fig. 16, for the window lengths $L = 256$ and $L = 512$ that produce the bulk of the detected events, ρ_∞^L is $\lesssim 0.3$ with a probability of 0.75 and, consequently, clustering flags $\lesssim 30\%$ of the region of the time-frequency plane containing the true signal. (This fraction would be reduced further if the error in frequency estimation is also taken into account.) In contrast, we see from Fig. 11 that SEECR recovers $\approx 90\%$ of the LC signal at the same SNR and probability with a frequency estimation error of ± 4 Hz.

VII. COMPARISON WITH TRACK-SEARCH AND CPP

The GW150914 system, with each of its components having a measured mass of $\approx 30 \, M_\odot$, falls within the range of simulated BBH signals used in Ref. [22] for analyzing the performance of track-search. Hence, the results in Sec. V C pertaining to the analysis of GW150914 data at an observed SNR = 10 can be used to compare the performance of SEECR with that of track-search.

The analysis in Ref. [22] used a false alarm probability of 3.4×10^{-5} for segments that are 0.415 sec long, sampled at a frequency of 9868.42 Hz. This corresponds to a FAR of 8.2×10^{-5} events/ sec. Assuming that the FAR scales linearly with the frequency search range, and that the range used in track-search extended to the Nyquist frequency of 4934.21 Hz, the equivalent FAR for SEECR is $(450/4934.21) \times 8.2 \times 10^{-5} = 7.5 \times 10^{-6}$ events/ sec. Here, we have used the actual bandwidth of [0, 450] Hz of the GW150914 data even though SEECR was run with a frequency search range that extends to 2000 Hz. Finally, translating this FAR back to false alarm probability for the GW150914 data segment length of 0.21 sec, we get 1.6×10^{-6}.

Based on the log-normal fit in Fig. 12, the false alarm probability derived above corresponds to a threshold of 30.4 on the LLR statistic. At this threshold, the detection probability attained by SEECR is unity. The detection probability for track-search can be read off from Fig. 5 of Ref. [22] to be ≈ 0.8 for a system with a total mass of

60 $M_\odot$ at SNR = 10. With the caveat that a proper comparison requires analysis of the same data realizations with compatible search parameter settings and a wide range of waveform morphologies, we find that SEECR has a performance that is comparable to or better than that of track-search.

For comparing SEECR with CPP, we use a different simulation setup than the one in Sec. IV A. This was necessitated by the high RAM requirement of the public domain CPP code (CHIRPLAB), which results in very large execution times when applied to the data realizations in Sec. IV. Each data realization is now shorter, with a duration of 0.5 sec at a sampling frequency of 2048 Hz, leading to 1024 samples per realization.

We generate 500 realizations of H_0 data following the same noise model as in Sec. IV. For H_1 data, we use a Newtonian inspiral signal that starts at 0.1 sec and lasts 0.24 sec. This signal corresponds to an equal-mass binary with a total mass of 45.0 $M_\odot$ and a lower frequency cutoff of 40 Hz. To stay within the range of signal strengths used in Ref. [26], the signal is normalized to have a value of 0.25 for the SNR as defined in Ref. [26], which corresponds to SNR = 5.5 as per the definition in this paper. (The latter is a factor of $\sqrt{N}$ higher than the former, where $N = 492$ is the number of samples in the signal waveform.)

We use the best path statistic [26] for path lengths $[1, 2, 4, 8, 16]$ and the routines provided in CHIRPLAB for estimating detection probability. At a false alarm probability of 0.05, which is the fiducial value used in Ref. [26], CPP is found to attain a detection probability of 0.242. Within sampling error, and for the same false alarm probability, SEECR gives a nearly identical detection probability of 0.25. Thus, CPP and SEECR seem to be comparable in performance.

The CPP algorithm assumes that the signal waveform occupies the whole of the data segment being analyzed. Although this condition is violated by the H_1 data described above, it provides a more realistic test since the true duration of an unmodeled signal is unknown by definition. That said, a version of CPP that is not limited by the above assumption should be used in future studies for a fair comparison.

VIII. CONCLUSIONS

We have presented a novel algorithm, called SEECR, for the detection and estimation of unmodeled transient chirp signals. The algorithm makes no assumptions about the time evolution of the amplitude envelope, $a(t)$, or the instantaneous frequency, $f(t)$, of a chirp signal except that they are smooth. The smoothness requirement is incorporated by modeling $a(t)$ and $f(t)$ with mutually independent splines. An important feature of SEECR is its small number of free parameters. Given enough computing power, these can be reduced to just two, namely, the maximum number of breakpoints to use for the two splines.

SEECR was tested on seven chirp signals spanning a wide range of amplitude and frequency evolution morphology, and found to achieve a detection probability $\gtrsim 0.5$ in the low SNR range (10 ≤ SNR ≤ 12) at a FAR between 10^{-3} and 10^{-4} events/sec. Hence, it is capable of achieving good sensitivity at astrophysically realistic signal strengths.

In terms of estimation, the instantaneous frequency of a signal is estimated much better, as expected, than the amplitude envelope. For example, excluding the CC signal due to its partial reconstruction, more than 60% of the instantaneous frequency evolution of a signal can be recovered with an error of ±64 Hz at SNR = 10.

Errors in the time of arrival were found to depend strongly on the true signal duration. For example, the smallest error range found is about ±80 msec for the s11WW signal (1 sec duration) at SNR = 15 but it reaches the ±10 msec level, comparable to the maximum light travel time between the two LIGO detectors, for GW150914 (0.2 sec duration) at a lower SNR of 10.

If SEECR is used in a temporal coincidence scheme across two GW detectors with an acceptance window of ±0.5 sec, which is the largest error range at the lowest SNR, the coincidence FAR becomes $(10^{-4})^2$ events/sec, or 1 event in 3.17 years, for a single detector FAR of 10^{-4} events/sec. At SNR = 12, the lowest corresponding two-detector coincidence detection probability, excluding the CC signal, is $0.84^2 \approx 0.7$ for the LC signal.

The coincidence FAR can be reduced substantially if instead of a temporal scheme, coincidence is imposed on the estimated frequency evolution. However, we did not explore coincidence schemes further in this paper because it is not the optimal way to utilize multiple GW detectors. The proper generalization of SEECR, which is a major future direction for its evolution, is coherent network analysis where each of the two GW polarizations is an independent instance of the single-detector signal model used in this paper. An early step in this direction was reported in Ref. [46] for the much simplified case where each GW polarization waveform itself is assumed to be a spline.

Based on an *ad hoc* time-frequency clustering method, we found that SEECR significantly outperforms a clustering-based search for some of the signal waveforms (CC and LC). At a FAR of 10^{-3} events/sec and SNR = 12, the clustering-based method could only achieve detection probabilities in the $[0.133 \pm 0.011, 0.174 \pm 0.011]$ range while SEECR achieved $[0.90 \pm 0.04, 0.68 \pm 0.04]$. Our clustering method fully incorporates multiresolution analysis, which is the main driver of performance for such methods. Hence, we do not expect a significantly different outcome for more sophisticated approaches to the production of time-frequency clusters.

Since clustering is a key component of the burst search methods used by LIGO, SEECR can complement current

searches by extending their coverage of GW waveform morphologies. We also compared SEECR to track-search and CPP and found that it is comparable in performance to these methods.

The metrics proposed here to quantify the estimation performance of SEECR can prove useful for a comparative study of algorithms that target long-duration ($\gtrsim 1$ sec) chirp signals. Similarly, the set of waveforms used here can serve as a benchmarking test bed.

APPENDIX A: B-SPLINE FUNCTIONS

A spline is a piecewise polynomial function defined over a set of adjacent intervals, where the end points of the intervals are called breakpoints. The coefficients of the polynomials are determined by specifying conditions, such as continuity and differentiability, at the breakpoints. Additional conditions at a breakpoint can be specified by expanding the sequence of breakpoints into a sequence of *knots*, where multiple consecutive knots can have the same breakpoint value.

For a fixed set of L knots $\bar{\tau} = (\tau_0, \tau_1, ..., \tau_{L-1})$, the set of all splines defined by $\bar{\tau}$ and having polynomial order k ($= 4$ for a cubic polynomial) is a linear vector space of dimensionality $L - k$. The set of B-spline functions, denoted by $\mathcal{B}_{i,k}(t; \bar{\tau})$, $i = 0, 1, ..., L - k - 1$, constitutes a basis for this space. They can be obtained using the recursion relations [56],

$$\mathcal{B}_{i,1}(t; \bar{\tau}) = \begin{cases} 1, & \tau_i <= t < \tau_{i+1}, \\ 0, & \text{else}, \end{cases} \tag{A1}$$

$$\begin{aligned} \mathcal{B}_{i,k}(t; \bar{\tau}) &= \frac{t - \tau_i}{\tau_{i+k-1} - \tau_i} \mathcal{B}_{i,k-1}(t; \bar{\tau}) \\ &+ \frac{\tau_{i+k} - t}{\tau_{i+k} - \tau_{i+1}} \mathcal{B}_{i+1,k-1}(t; \bar{\tau}). \end{aligned} \tag{A2}$$

From Eq. (A1), $B_{i,1}(t) = 0$ when $\tau_i = t = \tau_{i+1}$, and any term in Eq. (A2) that has a zero in the denominator (due to knot multiplicity) will be set to zero by this condition. It can

be shown that $\mathcal{B}_{i,k}(t; \bar{\tau}) = 0$ for $t \notin [\tau_i, \tau_{i+k}]$ and is positive in the interior of this interval.

For generating B-splines numerically, we use routines in the GNU Scientific Library (GSL) [57]. In these routines, the end knots have a multiplicity of k for a spline of order k. Thus, the number of B-splines generated is two more than the number of breakpoints. However, since the B-splines in this scheme at the end breakpoints are discontinuous, we always set their corresponding coefficients to zero. Therefore, the amplitude envelope spline is a linear combination of M B-splines as shown in Eq. (9).

APPENDIX B: SPECTROGRAM

For a given data sequence $\bar{x}$ of length N, define the windowed sequence $\bar{x}_a^L$ of length $L < N$ and offset a, $\bar{x}_a^L = (x_a, x_{a+1}, ..., x_{a+L-1})$, $0 \leq a \leq N - L$. Then a spectrogram, $\mathbf{S}^L$, is given by,

$$[\mathbf{S}^L]_{mn} = |[\mathbf{F}(\bar{w}^L . * \bar{x}_n^L)]_m|^2 \tag{B1}$$

where $\bar{w}^L$ is a window sequence of length L, $m = 0, 1, ..., \lfloor L/2 \rfloor$, and $n = 0, K, 2K, ..., \lfloor (N-L)/K \rfloor - 1$ with $1 \leq K \leq L$. Here, $L - K$ specifies the overlap between consecutive windowed sequences. In this paper, $\bar{w}^L$ is always a Hamming window and $K = \lfloor 0.25L \rfloor$. An element at row i and column j of $\mathbf{S}^L$ is called a pixel, (i, j), and $[\mathbf{S}^L]_{ij}$ is its amplitude.

APPENDIX C: TIME-FREQUENCY CLUSTERING

We present the definition of a time-frequency cluster as used in this paper as well as the algorithm used for producing clusters. See Appendix B for the notation used here.

Given a spectrogram $\mathbf{S}^L$ and a threshold η^L, define the binary matrix $\mathbf{B}^L$,

$$[\mathbf{B}^L]_{mn} = \begin{cases} 0 & [\mathbf{S}^L]_{mn} < \eta^L, \\ 1 & [\mathbf{S}^L]_{mn} \geq \eta^L. \end{cases} \tag{C1}$$

One can represent $\mathbf{B}^L$ as an image with pixels colored black when they have amplitude 1 and white otherwise. This has led to the common terminology, following Ref. [55], where a pixel with amplitude 1 is called a black pixel (BP), $\mathbf{B}^L$ is called the BP map, and η^L is called the BP threshold.

Define pixels (i, j) and (p, q) to be nearest neighbors if $(p - i, q - j) \in \{-1, 0, 1\} \times \{-1, 0, 1\}$. We call a nonempty sequence of pixels a *path* if it is a sequence of only nearest neighbors, and two pixels are *connected* if they are members of a path. A nonempty set of black pixels is defined to be a cluster if each element of the set is connected to every element of the set by a path that consists of only the elements of the set.

To distinguish noise and signal-induced clusters, we put a threshold on the cluster integrated power P_C^L, which is defined as

$$P_C^L = \sum_{(i,j) \in C} [\mathbf{S}^L]_{ij}. \qquad (C2)$$

For a given data realization and window length L, the union of pixels from all the clusters for which P_C^L exceeds some threshold is called a time-frequency event, or just an event when there is no scope for confusion.

APPENDIX D: CLUSTERING-BASED SEARCH METHOD

The steps below describe the clustering-based search method used in this paper and how it is initialized in our simulations. See Appendices B and C for the notation used here.

(1) Choose a set of values of L to allow multiresolution analysis. The frequency spacing between pixels in a column of $\mathbf{S}^L$ is given by f_s/L Hz, where $f_s = 4096$ Hz is the sampling frequency (see Sec. IV). Following the frequency resolutions used in the analysis of GW150914 by the coherent WaveBurst algorithm [2], we pick $L = 2^n$, $n = 5, 6, \ldots, 11$, leading to frequency spacings of 128, 64, 32, 16, 8, 4, and 2 Hz respectively.

(2) Obtain the BP threshold, η^L, for a target BP rate, r_{BP}^L, in H_0 data. For the noise model used here, $[\mathbf{S}^L]_{mn}$ has an exponential distribution, and assuming that pixels are statistically independent, the BP threshold is given by

$$\eta^L = \|\bar{w}^L\|^2 \ln \left(\frac{(\lfloor L/2 \rfloor + 1)\lfloor (N-L)/K \rfloor}{r_{\text{BP}}^L \times N/f_s} \right). \quad (D1)$$

We set $r_{\text{BP}}^L = 1000/7$ BP/ sec, leading to an overall rate of 1000 BP/ sec across all seven window lengths.

(3) Estimate the threshold on cluster integrated power, P_C^L. We use a target rate of $10^{-3}/7$ clusters/ sec for each L in H_0 data. This results in ≈ 3 as the expected number of noise-induced clusters over the entire 2×10^4 sec of H_0 data for each L. The corresponding threshold on P_C^L is, therefore, taken to be the third largest integrated power over all the clusters found for that L.

With the thresholds determined as described above, we run the method on realizations of H_1 data for each signal and each SNR.

For estimating detection probability, we count events found across all the values of L for a single data realization as one instance of detection. This grouping is an essential part of any multiresolution analysis since the same signal can produce clusters across multiple levels of resolution.

Strictly speaking, the same grouping should also be used for clusters obtained from H_0 data realizations but this is unnecessary in practice because the probability of clusters appearing across multiple values of L for a single H_0 realization, at the low rate of $10^{-3}/7$ clusters/ sec per L, is extremely small. Hence, at low rates of cluster production in H_0 data, individual clusters can be identified with instances of detection. Thus, the overall rate of 10^{-3} clusters/ sec that was set above matches the FAR of 10^{-3} events/ sec used for SEECR in Sec. V A.

[1] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **116**, 131103 (2016).

[2] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **116**, 061102 (2016).

[3] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **116**, 241103 (2016).

[4] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **118**, 221101 (2017).

[5] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **119**, 141101 (2017).

[6] F. Acernese *et al.*, Classical Quantum Gravity **32**, 024001 (2015).

[7] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. Lett. **119**, 161101 (2017).

[8] K. Somiya, Classical Quantum Gravity **29**, 124007 (2012).

[9] C. S. Unnikrishnan, Int. J. Mod. Phys. D **22**, 1341010 (2013).

[10] C. W. Helstrom, *Statistical Theory of Signal Detection* (Pergamon, London, 1968).

[11] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. D **93**, 122004 (2016).

[12] S. Klimenko *et al.*, Phys. Rev. D **93**, 042004 (2016).

[13] S. Klimenko, S. Mohanty, M. Rakhmanov, and G. Mitselmakher, Phys. Rev. D **72**, 122002 (2005).

[14] S. D. Mohanty, M. Rakhmanov, S. Klimenko, and G. Mitselmakher, Classical Quantum Gravity **23**, 4799(2006).

[15] M. Rakhmanov, Classical Quantum Gravity **23**, S673 (2006).

[16] C. D. Ott, A. Burrows, L. Dessart, and E. Livne, Phys. Rev. Lett. **96**, 201102 (2006).

[17] C. D. Ott, Classical Quantum Gravity **26**, 063001 (2009).

[18] Y. T. Liu, Phys. Rev. D **65**, 124003 (2002).

[19] A. L. Piro and E. Thrane, Astrophys. J. **761**, 63 (2012).

[20] M. H. P. M. van Putten, Phys. Rev. Lett. **87**, 091101 (2001).

[21] K. Kiuchi, M. Shibata, P. J. Montero, and J. A. Font, Phys. Rev. Lett. **106**, 251102 (2011).

[22] W. G. Anderson and R. Balasubramanian, Phys. Rev. D **60**, 102001 (1999).

[23] L. Cohen, Proc. IEEE **77**, 941 (1989).

[24] P. Addesso, M. Longo, S. Marano, V. Matta, I. Pinto, and M. Principe, in *2015 3rd International Workshop on Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa)* (IEEE, New York, 2015), p. 154.

[25] E. Chassande-Mottin and A. Pai, Phys. Rev. D **73**, 042003 (2006).

[26] E. J. Candès, P. R. Charlton, and H. Helgason, Classical Quantum Gravity **25**, 184020 (2008).

[27] E. Thrane *et al.*, Phys. Rev. D **83**, 083004 (2011).

[28] E. Thrane and M. Coughlin, Phys. Rev. D **89**, 063012 (2014).

[29] S. D. Mohanty, Astron. Rev. **7**, 29 (2012).

[30] S. D. Mohanty, Astron. Rev. **7**, 4 (2012).

[31] S. D. Mohanty, in *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation* (ACM, New York, 2014), p. 173.

[32] C. de Boor, *A Practical Guide to Splines* (Springer, New York, 2001).

[33] M. Steffen, Astron. Astrophys. **239**, 443 (1990).

[34] P. J. Green and B. W. Silverman, *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach* (CRC Press, Boca Raton, FL, 1993).

[35] G. Wahba, *Spline Models for Observational Data* (SIAM, Philadelphia, 1990).

[36] W. Hardle, *Applied Nonparametric Regression* (Cambridge University Press, Cambridge, England, 1990).

[37] J. S. Racine, A primer on regression splines, https://cran.r -project.org/web/packages/crs/vignettes/spline_primer.pdf.

[38] D. Ruppert, M. P. Wand, and R. J. Carroll, *Semiparametric Regression* (Cambridge University Press, Cambridge, England, 2003).

[39] R. Eberhart and J. Kennedy, in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science* (IEEE, New York, 1995), p. 39.

[40] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence* (Wiley, New York, 2005).

[41] A. Gálvez and A. Iglesias, Comput.-Aided Des. Appl. **43**, 1683 (2011).

[42] G. H. Golub, M. Heath, and G. Wahba, Technometrics **21**, 215 (1979).

[43] H. Akaike, Information theory and an extension of the maximum likelihood principle, in *Selected Papers of Hirotugu Akaike*, edited by E. Parzen, K. Tanabe, and G. Kitagawa (Springer, New York, 1998), p. 199.

[44] H. W. Kuhn and A. W. Tucker, in *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability* (University of California Press, Berkeley, 1951), p. 481.

[45] D. Fraser and H. Massam, Scand. J. Stat. Theory Appl. **16**, 65 (1989).

[46] C. Leung, SIAM Undergraduate Research Online (SIURO) **8** (2015).

[47] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. (Prentice-Hall, Englewood Cliffs, NJ, 1998), Chap. 11.

[48] C. D. Ott, stellarcollapse.org.

[49] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Phys. Rev. D **93**, 122003 (2016).

[50] D. Bratton and J. Kennedy, in *2007 IEEE Swarm Intelligence Symposium (SIS)* (IEEE, New York, 2007), p. 120.

[51] Y. Wang, S. D. Mohanty, and F. A. Jenet, Astrophys. J. **815**, 125 (2015).

[52] M. Thulin, Electron. J. Stat. **8**, 817 (2014).

[53] N. J. Cornish and T. B. Littenberg, Classical Quantum Gravity **32**, 135012 (2015).

[54] R. Lynch, S. Vitale, R. Essick, E. Katsavounidis, and F. Robinet, Phys. Rev. D **95**, 104046 (2017).

[55] J. Sylvestre, Phys. Rev. D **66**, 102004 (2002).

[56] C. de Boor, J. Approx. Theory **6**, 50 (1972).

[57] B. Gough, *GNU Scientific Library Reference Manual* (Network Theory Ltd., Surrey, UK, 2009).