

# Use of an excess power method and a convolutional neural network in an all-sky search for continuous gravitational waves

Takahiro S. Yamamoto<sup>1</sup> and Takahiro Tanaka<sup>1,2</sup>

<sup>1</sup>*Department of Physics, Kyoto University, Kyoto 606-8502, Japan*

<sup>2</sup>*Center for Gravitational Physics, Yukawa Institute for Theoretical Physics, Kyoto University, Kyoto 606-8502, Japan*

 (Received 25 November 2020; accepted 18 March 2021; published 26 April 2021)

The signal of continuous gravitational waves has a longer duration than the observation period. Even if the waveform in the source frame is monochromatic, we will observe the waveform with modulated frequencies due to the motion of the detector. If the source location is unknown, a lot of templates having different sky positions are required to demodulate the frequency, and the required huge computational cost restricts the applicable parameter region of coherent search. In this work, we propose and examine a new method to select candidates, which reduces the cost of coherent search by following up only the selected candidates. As a first step, we consider an idealized situation in which only a single-detector having 100% duty cycle is available and its detector noise is approximated by the stationary Gaussian noise. Also, we assume that the signal has no spin-down, that the polarization angle, the inclination angle, and the initial phase are fixed to be  $\psi = 0$ ,  $\cos i = 1$ , and  $\phi_0 = 0$ , and that they are treated as known parameters. We combine several methods: (1) the short-time Fourier transform with the resampled data such that the Earth motion for the source is canceled in some reference direction, (2) the excess power search in the Fourier transform of the time series obtained by picking up the amplitude in a particular frequency bin from the short-time Fourier transform data, and (3) the deep learning method to further constrain the source sky position. The computational cost and the detection probability are estimated. The injection test is carried out to check the validity of the detection probability. We find that our method is worthy of further study for analyzing  $O(10^7)$  sec strain data.

DOI: [10.1103/PhysRevD.103.084049](https://doi.org/10.1103/PhysRevD.103.084049)

## I. INTRODUCTION

Advanced LIGO and Advanced Virgo detected the first event of gravitational waves from a binary black hole merger in 2015 [1]. After the three observation runs, a lot of binary coalescence events are found [2,3]. In addition to Advanced LIGO and Advanced Virgo, KAGRA [4], and LIGO India [5] are planning to join the gravitational wave detector network [6]. The gravitational wave astronomy is expected to get fruitful results for improving our understanding of the astronomical properties of compact objects [7–9], the true nature of gravity [10–12], the origin of the Universe [13], and so on (see [14] for a review).

All gravitational wave signals which are detected so far have duration  $O(10^{0-2})$  sec, which is much shorter than the observation period. By contrast, we also expect gravitational waves which last longer than the observation period. Such long-lived gravitational waves are called “continuous gravitational waves” (see [15,16] as textbooks). Continuous gravitational waves are defined by the following three properties: (1) small change rate of the amplitude, (2) almost constant fundamental frequency, and (3) duration longer than the observation period. Rotating anisotropic neutron stars are typical candidate sources of

continuous gravitational waves. In addition, there are several exotic objects proposed as possible candidates of the sources of continuous gravitational waves [17–19].

Continuous gravitational waves are modeled by simpler waveforms than those of coalescing binaries. The parameters characterizing a typical waveform are the amplitude, the initial frequency, and the frequency derivatives with time. Although the waveform generated by the source is analytically simple, the effect of the detector’s motion makes the data analysis for continuous gravitational waves challenging. The detector’s motion causes the modulation in the frequency, and resulting in the dispersion of the power in the frequency domain. If the source location is *a priori* known by electromagnetic observations, the modulation can be removed precisely enough. By contrast, for the unknown target search, we need to correlate the data with a tremendous amount of templates to cover the unknown source location on the sky. This severely restricts the applicability of the all-sky coherent search to strain data of long durations. Therefore, semicoherent methods, in which the strain data is divided into a set of segments and statistics calculated for respective segments are summed up appropriately, are often used. Various semicoherent

methods (e.g., time-domain  $\mathcal{F}$ -statistic [20], SkyHough [21], FrequencyHough [22]) were proposed so far, and they are actually used to analyze LIGO and Virgo's data. There is a tremendous effort to find continuous gravitational waves [23–27].

As another trend of the research, the deep learning method is introduced to the field of gravitational wave data analysis. After the pioneering work done by George and Huerta [28], there are many proposals to use deep learning for wide purposes, e.g., parameter estimation for binary coalescence [29–32], noise classification [33], and waveform modeling [34]. As for applications to the search for continuous gravitational waves, several groups already proposed deep learning methods. Dreissigacker *et al.* [35,36] applied neural networks to all-sky searches of signals with the duration  $10^5$  sec and  $10^6$  sec. They used Fourier transformed strain as inputs. Their methods can be applied to the signal located in broad frequency bands and to the case of multiple detectors and realistic noise. Also, it is shown that the synergies between the deep learning and standard methods or other machine learning techniques are also powerful [37,38].

In this paper, we propose a new method designed for detecting monochromatic waves, combining several transformations and the deep learning method. In Sec. II, the waveform model and some assumptions are introduced. The coherent matched filtering and the time resampling technique are briefly reviewed in Sec. III. In Sec. IV, we explain our strategy that combines several traditional methods such as the resampling, the short-time Fourier transform, and the excess power search with the deep learning method. We show the results of the assessment of the performance of our new method in Sec. V. Section VI is devoted to the conclusion.

## II. WAVEFORM MODEL

We consider a monochromatic gravitational wave. We denote by  $f_{\text{gw}}$  its frequency constant in time. With the assumption that the source is at rest with respect to the solar system barycenter (SSB), a complex-valued gravitational waveform in the source frame  $h^{\text{source}}(\tau)$  will be simply written as

$$h^{\text{source}}(\tau) = h_0 e^{2\pi i f_{\text{gw}} \tau + i \phi_0}, \quad (2.1)$$

where  $\tau$  is called ‘‘SSB time,’’ and  $h_0$  and  $\phi_0$  are the amplitude and the initial phase, respectively. In this work, for simplicity, we assume that

$$\phi_0 = 0, \quad (2.2)$$

and regard it as a known parameter. The phase of a gravitational wave is modulated due to the detector motion and the modulation depends on the source location. The normal vector pointing from the Earth's center to the sky

position specified by a right ascension  $\alpha$  and a declination angle  $\delta$  is defined by

$$\mathbf{n}(\alpha, \delta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \epsilon & \sin \epsilon \\ 0 & -\sin \epsilon & \cos \epsilon \end{pmatrix} \begin{pmatrix} \cos \alpha \cos \delta \\ \sin \alpha \cos \delta \\ \sin \delta \end{pmatrix}, \quad (2.3)$$

with the tilt angle between the Earth's rotation axis and the orbital angular momentum  $\epsilon$ . Here, we work in the SSB frame, in which the  $z$  axis is along the Earth's orbital angular momentum and the  $x$  axis points towards the vernal equinox. Defining the detector time  $t$  so as to satisfy

$$\tau = t + \frac{\mathbf{r}(t) \cdot \mathbf{n}(\alpha_s, \delta_s)}{c}, \quad (2.4)$$

we obtain the waveform in the detector frame

$$h(t) := h_0 e^{i\Phi(t)}, \quad (2.5)$$

with

$$\Phi(t) = 2\pi f_{\text{gw}} t + 2\pi f_{\text{gw}} \frac{\mathbf{r}(t) \cdot \mathbf{n}(\alpha_s, \delta_s)}{c}. \quad (2.6)$$

A subscript ‘‘s’’ indicates the quantity related to the gravitational wave source. Namely,  $(\alpha_s, \delta_s)$  means the sky position of the source. In the following, we use the notation  $\mathbf{n}_s := \mathbf{n}(\alpha_s, \delta_s)$ .

For the modeling of the detector motion, we adopt a little simplification, which we believe will not affect our main result. We assume that the position vector of the detector can be written by a sum of the Earth's rotation part  $\mathbf{r}_{\oplus}(t)$ , and the Earth's orbital motion part  $\mathbf{r}_{\odot}(t)$ . The Earth is assumed to take a circular orbit on the  $xy$  plane. Then, we can write  $\mathbf{r}_{\odot}(t)$  as

$$\mathbf{r}_{\odot}(t) = R_{\text{ES}} \begin{pmatrix} \cos(\varphi_{\odot} + \Omega_{\odot} t) \\ \sin(\varphi_{\odot} + \Omega_{\odot} t) \\ 0 \end{pmatrix}, \quad (2.7)$$

where  $R_{\text{ES}}$ ,  $\Omega_{\odot}$ , and  $\varphi_{\odot}$  are the distance between the Earth and the Sun, the angular velocity of the orbital motion and the initial phase, respectively. The detector motion due to the Earth's rotation can be described as

$$\mathbf{r}_{\oplus}(t) = R_{\text{E}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \epsilon & \sin \epsilon \\ 0 & -\sin \epsilon & \cos \epsilon \end{pmatrix} \begin{pmatrix} \cos \lambda \cos(\varphi_{\oplus} + \Omega_{\oplus} t) \\ \cos \lambda \sin(\varphi_{\oplus} + \Omega_{\oplus} t) \\ \sin \lambda \end{pmatrix}, \quad (2.8)$$

where  $R_{\text{E}}$ ,  $\lambda$ ,  $\Omega_{\oplus}$ , and  $\varphi_{\oplus}$  are the radius of the Earth, the latitude of the detector, the angular velocity of the Earth's rotation and the initial phase, respectively. The modulated phase  $\Phi(t)$  can be decomposed into

$$\Phi(t) = 2\pi f_{\text{gw}} t + \Phi_{\oplus}(t) + \Phi_{\odot}(t), \quad (2.9)$$

where

$$\Phi_{\oplus}(t) = 2\pi f_{\text{gw}} \frac{\mathbf{r}_{\oplus}(t) \cdot \mathbf{n}_s}{c}, \quad (2.10)$$

$$\Phi_{\odot}(t) = 2\pi f_{\text{gw}} \frac{\mathbf{r}_{\odot}(t) \cdot \mathbf{n}_s}{c}. \quad (2.11)$$

Finally, we take into account the amplitude modulation due to the detector's motion, which can be described by the antenna pattern function. In this work, the polarization angle and the inclination angle are, respectively, assumed to be

$$\psi = 0, \quad \cos i = 1, \quad (2.12)$$

and, similarly to  $\phi_0$ , they are treated as known parameters. Then, the gravitational wave to be observed by a detector can be written as

$$h_{\text{obs}}(t) = G(t)h(t) + G^*(t)h^*, \quad (2.13)$$

with

$$G(t) := \frac{F_+(t) + iF_{\times}(t)}{2}. \quad (2.14)$$

The definitions of  $F_+(t)$  and  $F_{\times}(t)$  are the same as those used in Jaranowski *et al.*, [20]. In this work, the antenna pattern function of LIGO Hanford is employed. The strain data is written as

$$s(t) = h_{\text{obs}}(t) + n(t), \quad (2.15)$$

where  $n(t)$  is the detector noise. We assume that the strain data from the detector has no gaps in time and the detector noise is stationary and Gaussian.

### III. COHERENT SEARCH METHOD

Before explaining our method, we briefly review the coherent search method and the time resampling technique [20].

If the expected waveforms can be modeled precisely and the noise is Gaussian, the matched filtering is the optimal method for the detection and the parameter estimation, besides the computational cost. The noise weighted inner product is defined by

$$(a|b) := 4\text{Re} \left[ \int_0^{\infty} df \frac{\tilde{a}(f)\tilde{b}^*(f)}{S_n(f)} \right], \quad (3.1)$$

where  $S_n(f)$  is the power spectral density of the detector noise. A signal-to-noise ratio (SNR) can be calculated with

the inner product between a strain  $s(t)$  and a template  $h_{\text{temp}}(t)$  as

$$\rho_{\text{MF}} := \frac{(s|h_{\text{temp}})}{\sqrt{(h_{\text{temp}}|h_{\text{temp}})}}. \quad (3.2)$$

Theoretically predicted waveforms  $h_{\text{temp}}(t)$  have various parameters characterizing the source properties and the geometrical information. A set of waveforms having different parameters is called a template bank. For each template in a template bank, we can assign the value of SNR calculated by Eq. (3.2). If the maximum value of SNR in the template set exceeds a threshold value, it is a sign that an actual signal may exist and the parameter inference is also obtained from the distribution of SNR.

Due to a long duration and a narrow frequency band of continuous gravitational waves, the inner product (3.1) can be recast into the time-domain expression as

$$(a|b) \simeq \frac{2}{S_n(f_{\text{gw}})} \text{Re} \left[ \int_0^{T_{\text{obs}}} dt a(t)b^*(t) \right], \quad (3.3)$$

where  $T_{\text{obs}}$  is the observation time. For a monochromatic source, the waveform can be modeled by Eq. (2.1). The modulation due to the detector motion is only in the phase of the waveform. The time resampling technique nullifies the phase modulation by redefining the time coordinate. If the position of the source is *a priori* known, the exact relation (2.4) can be obtained. Therefore, the phase modulation can be completely removed and the monochromatic waveform is applicable to the time-domain matched filtering (3.3). Calculation of the inner product (3.3) between the resampled signal and a monochromatic waveform is equivalent to the Fourier transform. Thus, the fast algorithm (i.e., fast Fourier transform) can be employed to rapidly search the gravitational wave frequency,  $f_{\text{gw}}$ .

When the source location is unknown, we need to search all-sky by placing a set of grid points  $\{\mathbf{n}_g^{(i)}\}_{i=1}^{N_{\text{grid}}}$  to keep the maximum loss of SNR within the acceptable range. For each grid point, we carry out the Fourier transform after the transformation

$$\zeta := t + \frac{\mathbf{r}(t) \cdot \mathbf{n}_g}{c}. \quad (3.4)$$

Here, we omit the superscript  $(i)$ , for brevity. The necessary number of grid points  $N_{\text{grid}}$  can be estimated by the angular resolution of gravitational wave sources. The angular resolution of gravitational wave sources can be roughly given by the ratio between the wavelength of the gravitational wave and the diameter of the Earth's orbit, i.e.,

$$(\delta\theta)_{\text{coh}} \sim \frac{\lambda_{\text{gw}}}{2R_{\text{ES}}} \sim 10^{-5} \text{ [rad]}. \quad (3.5)$$

Here, we adopt 100 Hz as the fiducial value for  $c\lambda_{\text{gw}}^{-1}$ . Thus, the required number of grid points is, at least,

$$N_{\text{grids}} \sim \frac{4\pi}{(\delta\theta)_{\text{coh}}^2} \sim 1.3 \times 10^{11}. \quad (3.6)$$

The time resampling and the Fourier transform are applied to each grid point. The number of floating point operations required for carrying out FFT is  $\sim 1.7 \times 10^{12}$  per grid point, with the signal of a duration  $10^7$  sec and a sampling frequency 1024 Hz. Even if we have a 1PFlops machine, the computational time becomes  $2.2 \times 10^8$  sec, which is longer than the signal duration. For this reason, a naïve coherent method with the time resampling is not suitable even for monochromatic sources (see [39] proposing a more efficient coherent method).

## IV. OUR METHOD

### A. Subtracting the effect due to the Earth's rotation

As stated in Sec. III, the time resampling technique can demodulate the phase, but complete demodulation is not available because of the limitation of computational resources. In our work, the time resampling technique is employed to eliminate only the effect caused by the Earth's diurnal rotation,  $\Phi_{\oplus}(t)$ . Assuming a representative grid point  $\mathbf{n}_g$ , we can rewrite the phase (2.6) as

$$\begin{aligned} \Phi(t) &= 2\pi f_{\text{gw}} t + \Phi_{\oplus}(t) + \Phi_{\odot}(t), \\ &= 2\pi f_{\text{gw}} \zeta + \delta\Phi_{\oplus}(t) + \delta\Phi_{\odot}(t), \end{aligned} \quad (4.1)$$

where

$$\delta\Phi_{\oplus}(t) := 2\pi f_{\text{gw}} \frac{\mathbf{r}_{\oplus}(t) \cdot \Delta \mathbf{n}}{c}, \quad (4.2)$$

$$\delta\Phi_{\odot}(t) := 2\pi f_{\text{gw}} \frac{\mathbf{r}_{\odot}(t) \cdot \Delta \mathbf{n}}{c}, \quad (4.3)$$

and  $\Delta \mathbf{n} := \mathbf{n}_s - \mathbf{n}_g$ . Since the residual phase varies with time, we will place grid points so that the amplitude of the residual phase in the worst case, i.e.,

$$\min_{\mathbf{n}_g} \max_t |\delta\Phi_{\oplus}(t)|$$

becomes smaller than a threshold  $\delta\Phi_{\epsilon}$  for any source direction  $\mathbf{n}_s$  within the area covered by the grid point  $\mathbf{n}_g$ . To optimize the grid placement, we employ the method proposed in Ref. [40]. The residual phase  $\delta\Phi_{\oplus}$  is expanded up to the first order of  $\Delta\alpha := \alpha_s - \alpha_g$  and  $\Delta\delta := \delta_s - \delta_g$ . Then, we get

$$\begin{aligned} \delta\Phi_{\oplus} &\simeq \frac{2\pi f_{\text{gw}}}{c} R_{\text{E}} \cos \lambda \{ -\Delta\delta \sin \delta_g \cos(\alpha_g - \varphi_{\oplus} - \Omega_{\oplus} t) \\ &\quad - \Delta\alpha \cos \delta_g \sin(\alpha_g - \varphi_{\oplus} - \Omega_{\oplus} t) \}. \end{aligned} \quad (4.4)$$

Here, the constant term is neglected because it degenerates with the initial phase  $\phi_0$ . The maximum value of the residual phase is

$$\begin{aligned} \max_t |\delta\Phi_{\oplus}| &= \frac{2\pi f_{\text{gw}}}{c} R_{\text{E}} |\cos \lambda| \\ &\quad \times \sqrt{(\Delta\delta)^2 \sin^2 \delta_g + (\Delta\alpha)^2 \cos^2 \delta_g}. \end{aligned} \quad (4.5)$$

The grid points are to be determined to satisfy  $\max_t |\delta\Phi_{\oplus}| < \delta\Phi_{\epsilon}$  for any source direction.

Because the residual phase (4.5) is symmetric under the transformation  $\delta_g \rightarrow -\delta_g$ , the placement of grids on the negative  $\delta$  side can be generated by inverting the sign of the grids on the positive  $\delta$  side. Therefore, we focus on the case with  $0 \leq \delta \leq \pi/2$ .

Since the residual phase depends only on  $\delta$  at  $\delta_g = \pi/2$ , a single template can cover the neighbor of  $\delta = \pi/2$ . In fact, at  $\delta = \pi/2$ , Eq. (4.5) becomes

$$\max_t |\delta\Phi_{\oplus}| = \frac{2\pi f_{\text{gw}}}{c} R_{\text{E}} |\Delta\delta| \cos \lambda. \quad (4.6)$$

Therefore, the condition  $\max_t |\delta\Phi_{\oplus}| \leq \delta\Phi_{\epsilon}$  gives the lower bound of  $\delta_1$  such that the region  $\delta_1 \leq \delta \leq \pi/2$  can be covered by a single patch represented by  $\{(\alpha_g, \delta_g) = (0, \pi/2)\}$ , to find

$$\delta_1 := \frac{\pi}{2} - \delta\Phi_{\epsilon} \times \frac{c}{2\pi f_{\text{gw}} R_{\text{E}} \cos \lambda}. \quad (4.7)$$

Plural patches are necessary to cover the strip of a constant  $\delta$  in the other range. We introduce a two-dimensional metric corresponding to the residual phase (4.5),

$$d\sigma^2 = \cos^2 \delta d\alpha^2 + \sin^2 \delta d\delta^2. \quad (4.8)$$

In general, a metric in a 2-dimensional manifold can be transformed into a conformally flat metric by an appropriate coordinate transformation. When the space is conformally flat, the curve of a small constant distance measured from an arbitrary chosen point can be approximated by a circle. Therefore, a template spacing in the two-dimensional parameter space becomes relatively easy. By defining new variables  $X := \alpha$  and  $Y := -\log |\cos \delta|$ , the metric can be transformed into

$$d\sigma^2 = e^{-2Y} (dX^2 + dY^2). \quad (4.9)$$

Along with [40], we can construct the sky patches covering the half-sky region with  $0 \leq \delta \leq \delta_1$ . Figure 1 shows a part of grid points constructed under the condition

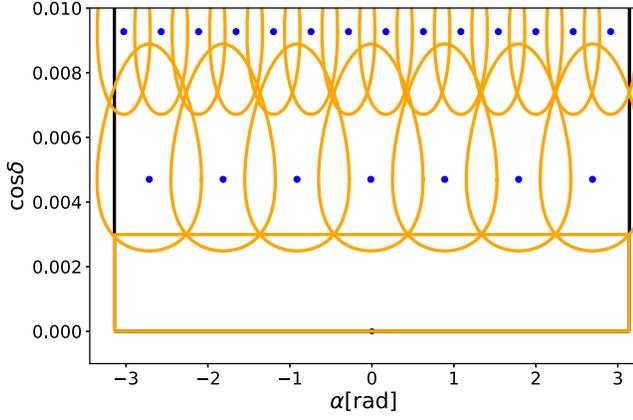


FIG. 1. Grid point placement on a fraction of  $(\alpha, \cos \delta)$  plane. Blue dots are grid points and orange contours show the  $\max_t |\delta \Phi_{\oplus}(t)| = \delta \Phi_e$  contours for each grid point. The region  $\{(\alpha, \delta) | \delta < \delta_1\}$  is covered by a single template  $(\alpha_g, \delta_g) = (0, \pi/2)$  and the shape of the patch is square on this plane.

$$\delta \Phi_e = 0.058, \quad (4.10)$$

which we adopt throughout this paper. The total number of grid points to cover the whole sky is

$$N_{\text{grid}} = 352,436, \quad (4.11)$$

for  $f_{\text{gw}} = 100$  Hz.

## B. Modeling the effect due to the Earth's orbital motion

As we choose  $\delta \Phi_e$  to be sufficiently small, we neglect  $\delta \Phi_{\oplus}$  in the following discussion. Then, after subtracting the phase modulation due to the Earth's rotation, the phase of the gravitational wave (4.1) becomes

$$\Phi(t) = 2\pi f_{\text{gw}} \zeta + \delta \Phi_{\odot}(t). \quad (4.12)$$

We apply the short-time Fourier transform (STFT) to the time-resampled strain,

$$s(\zeta) = h_{\text{obs}}(\zeta) + n(\zeta). \quad (4.13)$$

In the rest of the paper, we treat only the time-resampled data. Therefore, without confusion, the time-resampled data in Eq. (4.13) can be denoted by the same character as the original one. The strain is divided into  $N_{\text{seg}}$  segments having the duration  $T_{\text{seg}}$  and their start times are denoted by  $\zeta_j := jT_{\text{slide}}$ , ( $j = 0, 1, \dots, N_{\text{seg}} - 1$ ).  $T_{\text{slide}}$  is not necessary to be equal to  $T_{\text{seg}}$ . The output of STFT with the window function  $w(\zeta)$  is defined by

$$s_{j,k}^{\text{STFT}} = h_{j,k}^{\text{STFT}} + n_{j,k}^{\text{STFT}}, \quad (4.14)$$

where

$$h_{j,k}^{\text{STFT}} = \frac{1}{T_{\text{seg}}} \int_{\zeta_j}^{\zeta_j + T_{\text{seg}}} d\zeta' w(\zeta' - \zeta_j) h_{\text{obs}}(\zeta') e^{-2\pi i f_k \zeta'}, \quad (4.15)$$

$$n_{j,k}^{\text{STFT}} = \frac{1}{T_{\text{seg}}} \int_{\zeta_j}^{\zeta_j + T_{\text{seg}}} d\zeta' w(\zeta' - \zeta_j) n(\zeta') e^{-2\pi i f_k \zeta'}, \quad (4.16)$$

and  $f_k := k\Delta f = k/T_{\text{seg}}$  is the frequency of the  $k$ th element of STFT. Let us focus on the positive frequency modes, i.e.,  $f_k > 0$ . Then, the second term of Eq. (2.13) can be neglected and Eq. (4.15) can be approximated by

$$h_{j,k}^{\text{STFT}} \simeq \frac{1}{T_{\text{seg}}} \int_{\zeta_j}^{\zeta_j + T_{\text{seg}}} d\zeta' \{w(\zeta' - \zeta_j) \times G(t(\zeta')) e^{2\pi i \delta f_k \zeta'} e^{i\delta \Phi_{\odot}(\zeta')}\}, \quad (4.17)$$

with  $\delta f_k := f_{\text{gw}} - f_k$ . In the expression of  $G(t(\zeta))$ , the SSB time  $\zeta$  appears only through the combination  $\Omega_{\oplus} t(\zeta)$ . The difference between  $\Omega_{\oplus} t(\zeta)$  and  $\Omega_{\oplus} \zeta$  is negligibly small. Therefore, in Eq. (4.17),  $G(t(\zeta))$  can be replaced by  $G(\zeta)$ . The duration  $T_{\text{seg}}$  is chosen so that  $G(t(\zeta))$  can be approximated by a constant in each segment. With this choice of  $T_{\text{seg}}$ , the factor  $e^{i\delta \Phi_{\odot}(t)}$  also can be seen as a constant in each segment because it varies slower than the antenna pattern function. Therefore, Eq. (4.17) can be approximated by

$$h_{j,k}^{\text{STFT}} \simeq h_0 e^{i\delta \Phi_{\odot}(\zeta_j)} G(\zeta_j) W_k(\zeta_j), \quad (4.18)$$

where

$$W_k(\zeta_j) := \frac{1}{T_{\text{seg}}} \int_{\zeta}^{\zeta_j + T_{\text{seg}}} d\zeta' w(\zeta' - \zeta_j) e^{2\pi i \delta f_k \zeta'}. \quad (4.19)$$

In this work, we use the Tukey window,

$$w(\zeta) = \begin{cases} \frac{1}{2} - \frac{1}{2} \cos\left(\frac{2\pi\zeta}{\alpha T_{\text{seg}}}\right), & (0 \leq \frac{\zeta}{T_{\text{seg}}} < \frac{\alpha}{2}), \\ 1, & (\frac{\alpha}{2} \leq \frac{\zeta}{T_{\text{seg}}} \leq 1 - \frac{\alpha}{2}), \\ \frac{1}{2} - \frac{1}{2} \cos\left(\frac{2\pi(T_{\text{seg}} - \zeta)}{\alpha T_{\text{seg}}}\right), & (1 - \frac{\alpha}{2} < \frac{\zeta}{T_{\text{seg}}} \leq 1). \end{cases} \quad (4.20)$$

We set the parameter  $\alpha$  to 0.125. With  $\beta_k := T_{\text{seg}} \delta f_k$ , Eq. (4.19) can be calculated as

$$W_k(\zeta_j) = e^{2\pi i \delta f_k \zeta_j} \frac{(1 + e^{i\alpha\beta_k})(1 - e^{2\pi i\beta_k(1-\alpha/2)})}{4\pi i \beta_k (\alpha^2 \beta_k^2 - 1)}. \quad (4.21)$$

Using the Jacobi-Anger expansion, we can expand the factor  $e^{i\delta \Phi_{\odot}(\zeta_j)}$  that appears in Eq. (4.18) as

$$e^{i\delta\Phi_{\circ}(\zeta_j)} = \sum_{\ell=-\infty}^{\infty} i^{\ell} J_{\ell}(X) e^{i\ell\Omega_{\circ}\zeta_j} e^{i\ell(\varphi_{\circ}-\phi_X)}, \quad (4.22)$$

where  $J_{\ell}(z)$  is the Bessel function of the first kind and

$$X := \frac{2\pi f_{\text{gw}} R_{\text{ES}}}{c} \sqrt{(\Delta n_x)^2 + (\Delta n_y)^2}, \quad (4.23)$$

$$e^{i\phi_X} := \Delta n_x + i\Delta n_y. \quad (4.24)$$

Therefore, Eq. (4.18) can be expressed as

$$\begin{aligned} h_{j,k}^{\text{STFT}} &\simeq h_0 G(\zeta_j) W_k(0) e^{2\pi i \delta f_k \zeta_j} \\ &\times \sum_{\ell=-\infty}^{\infty} i^{\ell} J_{\ell}(X) e^{i\ell(\varphi_{\circ}-\phi_X)} e^{i\ell\Omega_{\circ}\zeta_j}. \end{aligned} \quad (4.25)$$

The Fourier transform of  $h_{j,k}^{\text{STFT}}$  with a fixed integer  $k$  is defined by

$$\mathbf{H}_{\ell,k} := \frac{1}{N_{\text{seg}}} \sum_{j=0}^{N_{\text{seg}}-1} h_{j,k}^{\text{STFT}} e^{-2\pi i j \ell / N_{\text{seg}}}. \quad (4.26)$$

We refer to  $\mathbf{H}_{\ell,k}$  as the  $\ell$ -domain signal. To understand the pattern hidden in  $\mathbf{H}_{j,k}$ , we set aside the factor  $G(\zeta_j)$  for a while. Then Eq. (4.26) can be estimated as

$$\mathbf{H}_{\ell,k} \sim h_0 W_k(0) i^{\ell'} J_{\ell'}(X) e^{i\ell'(\varphi_{\circ}-\phi_X)}, \quad (4.27)$$

with  $\ell' \sim \ell + 2\pi\Omega_{\circ}^{-1}\delta f_k$ . Because of the fact that  $J_{\ell}(z) \simeq 0$  for  $|\ell| \gtrsim |z|$  and  $X \lesssim O(10^3)$  for  $f_{\text{gw}} = 100$  Hz, the signal is localized within the region where only few thousand bins in the  $\ell$  domain. Putting back the antenna pattern  $G(\zeta_j)$ , we expect that  $\ell$ -domain signals lose their amplitude and their

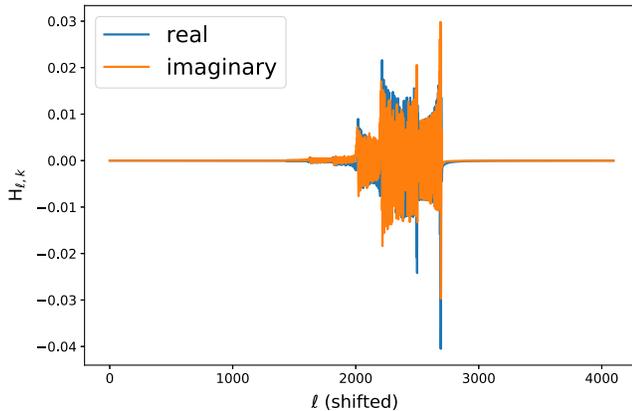


FIG. 2. An example of the  $\ell$ -domain waveform. The length of the  $\ell$ -domain waveform is  $2^{19}$ . This figure is an enlargement around the region at where the signal is localized. The amplitude is  $h_0 = 1.0$ .

localizations become worse than those for the idealized cases. Figure 2 shows an example of the  $\ell$ -domain signal.

### C. Excess power method for finding candidates

By the method shown in the previous subsection, for every grid point  $\mathbf{n}_g$  and every frequency bin  $f_k$ , we obtain an  $\ell$ -domain strain defined by

$$\mathbf{S}_{\ell,k} := \mathbf{H}_{\ell,k} + \mathbf{N}_{\ell,k}, \quad (4.28)$$

with

$$\mathbf{N}_{\ell,k} := \frac{1}{N_{\text{seg}}} \sum_{j=0}^{N_{\text{seg}}-1} n_{j,k}^{\text{STFT}} e^{-2\pi i j \ell / N_{\text{seg}}}. \quad (4.29)$$

There are  $T_{\text{obs}}/T_{\text{seg}} \sim O(10^6)$  data points in a single  $\ell$ -domain strain and we know that the signal in  $\ell$ -domain will be localized within a small region  $\sim O(10^3)$ . Thus, the excess power method [41] is useful for selecting the candidates with a minimal computational cost. We here divide an  $\ell$ -domain signal into short chunks so that each chunk has the length  $\delta\ell$  and neighbored segments have an overlap by  $\delta\ell/2$ , which is one of the simplest choices but not the optimal one. Then, we obtain  $N_{\text{chunk}/\text{signal}} = 2(N_{\text{seg}} - \delta\ell)/\delta\ell$  chunks from one  $\ell$ -domain signal. The excess power statistic for the grid point  $\mathbf{n}_g$ , the frequency bin  $f_k$ , and the  $c$ th chunk ( $c = 0, 1, \dots, N_{\text{chunk}/\text{signal}} - 1$ ) is defined by

$$\mathcal{E}(\mathbf{n}_g, f_k, c) := 4 \sum_{\ell=c\delta\ell/2}^{(c+2)\delta\ell/2-1} \frac{|\mathbf{S}_{\ell,k}|^2}{\tilde{\sigma}_k^2}, \quad (4.30)$$

where

$$\langle \mathbf{N}_{\ell,k} \mathbf{N}_{\ell',k}^* \rangle = \frac{1}{2} \tilde{\sigma}_k^2 \delta_{\ell\ell'}. \quad (4.31)$$

The variance of noise in the  $\ell$  domain,  $\tilde{\sigma}_k$ , is estimated as

$$\tilde{\sigma}_k^2 = \frac{S_n(f_k)}{N_{\text{seg}} T_{\text{seg}}} \times \mathcal{W}, \quad (4.32)$$

where  $\mathcal{W}$  is the factor coming from the window function and defined by

$$\mathcal{W} := \int_{-0.5}^{0.5} dx [w(x)]^2. \quad (4.33)$$

The derivation of Eq. (4.32) is summarized in Appendix.

We define the SNR of the excess power by

$$\rho_{\text{EP}}(\mathbf{n}_g, f_k, c) := \frac{\mathcal{E}(\mathbf{n}_g, f_k, c) - \langle \mathcal{E} \rangle_n}{\sigma_n(\mathcal{E})}, \quad (4.34)$$

where

$$\langle \mathcal{E} \rangle_n = 2\delta\ell, \quad (4.35)$$

and

$$\sigma_n(\mathcal{E}) := \sqrt{\langle (\mathcal{E} - \langle \mathcal{E} \rangle_n)^2 \rangle_n} = 2\sqrt{\delta\ell}, \quad (4.36)$$

are, respectively, the expectation value and the standard deviation of  $\mathcal{E}$  when only noise exists. We select the candidate set of parameter values  $\{\mathbf{n}_g, f_k, c\}$ , when

$$\rho_{\text{EP}}(\mathbf{n}_g, f_k, c) > \hat{\rho}_{\text{EP}}$$

is satisfied with a threshold value  $\hat{\rho}_{\text{EP}}$ . Strictly speaking, since the excess power statistic  $\mathcal{E}$  is the sum of  $2\delta\ell$  squared Gaussian random variables with the variance  $1/2\sqrt{\delta\ell}$ ,  $\mathcal{E}$  follows a chi square distribution with the degree of freedom  $2\delta\ell$ . However, since here we choose  $\delta\ell$  to be large, the distribution of  $\mathcal{E}$  can be approximated by a Gaussian distribution with the average  $2\delta\ell$  and the standard deviation  $2\sqrt{\delta\ell}$ . Therefore, in the absence of gravitational wave signal, the probability distribution of  $\rho_{\text{EP}}$  is a Gaussian distribution with zero mean and unit variance.

Also in the presence of some signal, the excess power statistics  $\rho_{\text{EP}}$  is given by a sum of many statistical variables. Thus, the statistical distribution of  $\rho_{\text{EP}}$  can be approximated by the Gaussian distribution whose mean and standard deviation are calculated as

$$\mu_{\text{EP}}(\boldsymbol{\xi}) = \frac{2P_k(\boldsymbol{\xi})}{\tilde{\sigma}_k^2 \sqrt{\delta\ell}}, \quad (4.37)$$

and

$$\sigma_{\text{EP}}(\boldsymbol{\xi}) = \sqrt{1 + \frac{4P_k(\boldsymbol{\xi})}{\tilde{\sigma}_k^2 \delta\ell}}, \quad (4.38)$$

Here, we define

$$P_k(\boldsymbol{\xi}) := \sum_{\ell} |H_{\ell,k}(\boldsymbol{\xi})|^2, \quad (4.39)$$

and we define  $\boldsymbol{\xi}$  as a set of parameters  $\boldsymbol{\xi} := (h_0, \vec{\xi}) = (h_0, f_{\text{gw}}, \alpha_s, \delta_s)$ . The false alarm rate and the detection efficiency will be assessed with this Gaussian approximation.

## D. Neural network for localizing

### 1. Fundamentals

Deep learning is one of the approaches for finding features being hidden in the data (see [42] as a textbook).

Artificial neural networks (ANNs) are the architectures playing the central roll in deep learning. An ANN consists of consecutive layers and each layer is formed by a lot of units (neurons). Each layer takes inputs from the previous layer and processed data is passed to the next layer. As a simple example, the process occurring in each layer can be written as the combination of affine transformation and a nonlinear transformation, i.e.,

$$x_i^{(\ell+1)} = g\left(\sum_{j=1}^{N^{(\ell)}} w_{ij}^{(\ell)} x_j^{(\ell)} + b^{(\ell)}\right) \quad (i = 1, 2, \dots, N^{(\ell+1)}), \quad (4.40)$$

where  $x^{(\ell)}$  is a set of input data on the  $\ell$ th layer and  $g$  is a nonlinear function, which is called an activation function. We use a rectified linear unit (ReLU) [43], defined by

$$g(z) = \max[z, 0]. \quad (4.41)$$

The parameters  $w$  and  $b$  are, respectively, called weights and biases. They are tunable parameters and optimized to capture the features of data. The process to optimize weights and biases is called training. Frequently, the affine transformation and the nonlinear transformation are divided into two layers, called a linear layer and a nonlinear transformation layer, respectively.

In addition to the layers as given by Eq. (4.40), many variants are proposed so far. In this work, we use also one-dimensional convolutional layers [44] and max-pooling layers [45]. The input of a convolutional layer, denoted by  $x_i^c$ , is a set of vectors. For example, in the case of color images, each pixel has three channels corresponding to three primary colors of light. Therefore, the input data is a set of three two-dimensional arrays. The discrete convolution, which is represented as

$$o_i^{c'} = \sum_{c=0}^{C-1} \sum_{k=0}^{K-1} f_k^{c,c'} x_{i+k}^c + b^{c'}, \quad (4.42)$$

is calculated in a convolutional layer. Here,  $x$  is the input and  $o$  is the output data of the layer.  $C$  and  $K$  are, respectively, the number of channels and the width of the kernel. Each pixel of the data is specified by an index  $i$ . The parameters  $f$  and  $b$  are optimized during the training. A max pooling layer, whose operation can be written as

$$o_i^c = \max_{k=0,1,\dots,K-1} [x_{si+k}^c], \quad (4.43)$$

with the kernel size  $K$  and the stride  $s$ , reduces the length of the data and hence the computational cost.

In supervised learning, a given dataset consists of many pairs of input data and target values. An ANN learns the relation between input data and target values from the

dataset and predicts values corresponding to newly given input data. In order to train an ANN, the deviation between the predicted values and the target value is quantified by a loss function. For a regression problem, the mean square loss,

$$L[\mathbf{y}(w), \mathbf{t}] = \frac{1}{2} \sum_{i=1}^d |y_i(w) - t_i|^2, \quad (4.44)$$

is often employed. Here,  $\mathbf{y}$  and  $\mathbf{t}$  are a set of predicted values and that of target values, respectively, and they are expressed as  $d$ -dimensional vectors. The prediction depends on the weights of the neural network, which are denoted by a single symbol  $w$ . An ANN is optimized so as to minimize the loss function for a given dataset, which is the sum of the loss functions for all data contained in the training dataset. Because the complete minimization using all dataset cannot be done, the iterative method is used. The weight  $w$  is updated by the replacement algorithm given by

$$w \rightarrow w - \eta \nabla_w \sum_{n=1}^{N_{\text{train}}} L[\mathbf{y}_n(w) - \mathbf{t}_n], \quad (4.45)$$

where  $N_{\text{train}}$  is the number of data contained in the dataset and  $\eta$  is called learning rate and characterizes the strength of each update. The algorithm shown in Eq. (4.45) is called gradient descent, which is the simplest procedure to update the weights, and many variants (e.g., momentum [46], rms prop [47], Adam [48]) are proposed so far. Regardless of the choice of the update algorithm, the gradients of a loss function is required and they can be quickly calculated by the backpropagation scheme [49]. In Eq. (4.45), all data in the dataset are used for each iteration. In practice, the loss function for a subset of the dataset is calculated. The subset is called a batch and chosen randomly in every iteration. This procedure is called a minibatch training.

In the training process, we optimize a neural network so that the loss function is minimized for a dataset. However, this strategy cannot be straightforwardly applied to practical situations. First, the trained neural network may fall in overfitting. Then, the neural network does not have an expected ability to correctly predict the label for a newly given input data which is not used for training. Second, we have to optimize the neural network model and the update procedure, too. For this purpose, we have to appropriately select the hyperparameters, such as the number of neurons of the  $\ell$ th layer ( $N^{(\ell)}$ ) and the learning rate ( $\eta$ ). They are not automatically tuned during the training process.

To solve these problems, we prepare a validation dataset that is independent from the training dataset. The weights of the neural network are optimized so that the loss function for the training dataset is minimized. The validation data is used for monitoring the training process and assessing which model is better for the problem that the user wants to

solve. To prevent the overfitting, the training should be stopped when the loss for the validation dataset tend to deviate from that for the training dataset (early stopping). To optimize the hyperparameters, many neural network models having various structures are trained with different training schemes. Among them, we choose the one performing with the smallest loss for the validation dataset.

## 2. Setup in our analysis

The whole architecture of the neural network we used is shown in Table I. The input data of the neural network is the complex valued numbers taken from a short chunk of the  $\ell$ -domain signal, and the output is the predicted sky position. The  $\ell$ -domain waveform  $\mathbf{H}_{\ell,k}$  is determined mainly by the residual phase  $\delta\Phi_{\odot}$ , which depends on the sky position  $(\alpha_s, \delta_s)$  through the vector  $\Delta\mathbf{n}$ . Because  $z_{\odot} = 0$ , only  $x$  and  $y$  components of  $\Delta\mathbf{n}$  affect  $\delta\Phi_{\odot}$ . Therefore, we label each waveform with the values of  $\Delta n_x$  and  $\Delta n_y$ , which are the targets of the prediction of the neural network. The outputs of the neural network are inverted to the predicted values which are denoted by  $(\alpha_p, \delta_p)$ . We apply the neural network to each candidate, selected by the excess power method, in order to narrow down the possible area in which the source is likely to be located. For simplicity, the  $(\alpha, \delta)$  plane is regarded as a two-dimensional Euclidean space, and the shape of the predicted region is assumed to be a disk on the  $(\alpha, \delta)$  plane. For each candidate, the origin of the disk is

TABLE I. The architecture of the neural network used in this work. For convolution and max pooling layers, the input and the output are characterized by  $(C, N)$  where  $C$  is the number of channels and  $N$  is the length of the data. For convolutional layers, the lengths of kernels are 16, 16, 8, 8, 4, and 4 from the earlier to the later layer. The kernel size of the max pooling layers is 4.

Layer	Input	Output
1D convolution	(2, 2048)	(64, 2033)
ReLU	(64, 2033)	(64, 2033)
1D convolution	(64, 2033)	(64, 2018)
ReLU	(64, 2018)	(64, 2018)
Max pooling	(64, 2018)	(64, 504)
1D convolution	(64, 504)	(128, 497)
ReLU	(128, 497)	(128, 497)
1D convolution	(128, 497)	(128, 490)
ReLU	(128, 490)	(128, 490)
Max pooling	(128, 490)	(128, 122)
1D convolution	(128, 122)	(256, 119)
ReLU	(256, 119)	(256, 119)
1D convolution	(256, 119)	(256, 116)
ReLU	(256, 116)	(256, 116)
Max pooling	(256, 116)	(256, 29)
Dense	$256 \times 29$	64
ReLU	64	64
Dense	64	64
ReLU	64	64
Dense	64	2

set to the predicted point. The radius of the disk, denoted by  $r_{\text{NN}}$ , is fixed to a constant value. In the follow-up stage, the finer grids are placed to cover whole region of the disk.

In order to train the neural network, we need to prepare the training dataset and the validation dataset. We use Eq. (4.26) as the model waveform and pick up only a short chunk containing the signal. The length of chunk is  $\delta\ell = 2048$ . We prepare 200 000 waveforms for the training and 10 000 waveforms for validation. At that time, we set  $h_0 = 1$ . We assume that we use only a single detector and use the geometry information (e.g., the latitude of the detector) of LIGO Hanford in calculating the antenna pattern function as an example. In this work, we focus on one sky patch covered by a single grid point and a frequency bin fixed at  $f_k = 100$  Hz since the scaling to the search over the whole sky and the wider frequency band is straightforward. The sources are randomly distributed within the sky patch. The parameters  $\beta_k$  are randomly sampled from a uniform distribution on  $[-0.5, 0.5]$ . The original strain has the duration  $2^{24}$  sec and the sampling frequency 1024 Hz. We introduce the normalized gravitational wave amplitude by

$$\hat{h}_0 := h_0 \left( \frac{S_n(f_{\text{ref}})}{1 \text{ Hz}^{-1}} \right)^{-1/2}. \quad (4.46)$$

Here, we set  $f_{\text{ref}} = f_k$ . At each training step, the amplitude whose logarithm is randomly chosen from the uniform distribution on  $-2.1 \leq \log_{10} \hat{h}_0 \leq -1.0$  is multiplied to the waveforms, and they are injected into the simulated noise. The different realizations of noise are sampled for every iterations. The real part and the imaginary part of the noise data mimicking  $N_{\ell,k}$  are generated from a Gaussian distribution with a zero mean and a variance

$$\frac{\mathcal{W}}{4N_{\text{seg}}T_{\text{seg}}}, \quad (4.47)$$

[see Eqs. (4.31) and (4.32)].

TABLE II. The values of the parameters we used in this work.

Symbol	Parameters	Value
$T_{\text{obs}}$	Observation period	$2^{24}$ sec
$f_s$	Sampling frequency	1024 Hz
$N_{\text{grid}}$	No. of grids	352436
$N_{\text{bin}}$	No. of frequency bins of STFT	3200
$T_{\text{seg}}$	Duration of a STFT segment	32 sec
$T_{\text{slide}}$	Dilation of STFT segment	32 sec
$\delta\ell$	Length of chunk	2048
$(\delta\ell)_{\text{slide}}$	Dilation of chunk	128
$f_k$	Fixed frequency bin	100 Hz
$\alpha_g$	Right ascension of grid	$-0.158649$ rad
$\delta_g$	Declination of grid	1.02631 rad

We employ the minibatch training. We set the batch size to 256. The Adam [48] is used for the update algorithm. We implement with the PYTHON library PYTORCH [50] and use a GPU GeForce 1080Ti. The parameter values we used are listed in Table II.

### E. Follow-up analysis by coherent matched filtering

After selecting candidates and narrowing down the possible area at which the source is likely to be located, we apply the coherent matched filtering for the follow-up analysis. The grid points with the resolution shown in Eq. (3.5) are placed to cover the selected area. Assuming a grid point, we can carry out the demodulation of the phase by using the time resampling technique. If the deviation between the directions of the grid point and the source is smaller than the resolution, the residual phase remaining after the time resampling is sufficiently small to avoid the loss of SNR.

In this operation, heterodyning and downsampling can significantly reduce the data length and hence the computational cost [51]. Let us assume that we have a candidate labeled with  $\{n_g, f_k, c\}$ . If the candidate is the true event, the gravitational wave frequency  $f_{\text{gw}}$  should take the value in the narrow frequency band indicated by

$$f_k - \frac{1}{2T_{\text{seg}}} \leq f_{\text{gw}} \leq f_k + \frac{1}{2T_{\text{seg}}}. \quad (4.48)$$

By multiplying the factor  $e^{-2\pi i f_k \zeta}$  to the resampled strain, we can convert the gravitational wave signal frequency to near dc components (heterodyning). After that, the gravitational wave signal has a lower frequency than  $1/2T_{\text{seg}}$  Hz. Therefore, downsampling by appropriately averaging the resampled strain data with a sampling frequency  $\sim 1/T_{\text{seg}}$  reduces the number of data points without loss of the significance of the gravitational wave signal.

The coherent matched filtering follows the heterodyning and the downsampling processes. As stated in Eqs. (2.2) and (2.12), we fix  $\psi = 0$ ,  $\cos \iota = 1$ , and  $\phi_0 = 0$  and treat them as known parameters. Also, we assume that the signal waveform and the template completely match. The definition of a match is already given in Eq. (3.2). The gravitational waveform is written as

$$h(\xi) = h_0 \cdot h_{\text{temp}}(\vec{\xi}). \quad (4.49)$$

Among these parameters, the amplitude  $h_0$  can be analytically marginalized to maximize the likelihood. Then we obtain the signal-to-noise ratio in Eq. (3.2) and use it as the detection statistic. When only the detector noise dominates the strain data,  $\rho_{\text{MF}}$  follows the standard normal distribution. On the other hand, if the signal exists, then the SNR follows a Gaussian distribution with a mean

$$\mu_{\text{MF}}(\xi) = h_0 \cdot \sqrt{(h_{\text{temp}}(\vec{\xi})|h_{\text{temp}}(\vec{\xi}))}, \quad (4.50)$$

and a unit variance.

## V. RESULTS

### A. Computational cost

Our procedure is characterized by three parameters  $\rho := (\text{FAP}_{\text{EP}}, r_{\text{NN}}, \hat{\rho}_{\text{MF}})$ , i.e.,

- (i)  $\text{FAP}_{\text{EP}}$ , false alarm probability for each chunk,
- (ii)  $r_{\text{NN}}$ , the radius of the predicted region to which the follow-up analysis is applied,
- (iii)  $\hat{\rho}_{\text{MF}}$ , the threshold of the SNR of the coherent matched filtering.

$\mathcal{N}_{\text{EP}}$  denotes the computational cost of the excess power method.  $2N_{\text{seg}}$  multiplications and  $2N_{\text{seg}}$  additions of real numbers are required to calculate the excess powers for all chunks in one  $\ell$ -domain signal. The computational cost for calculating the excess powers for all chunks can be estimated as

$$\mathcal{N}_{\text{EP}} = 4N_{\text{seg}} \times N_{\text{grid}} \times N_{\text{bin}} \sim 4.7 \times 10^{15}, \quad (5.1)$$

in the unit of the number of floating point operations. As we see in the following, this cost can be neglected. Next, we check the computational time of the neural network analysis. We estimate the computational time of the neural network by measuring the elapsed time for analyzing ten thousand data. Because the elapsed time is 1.4 sec, the total computational time of the neural network is estimated as

$$\mathcal{T}_{\text{NN}} \simeq \frac{1.4 \text{ sec}}{10^4 \text{ data}} \times N_{\text{candidate}}, \quad (5.2)$$

where  $N_{\text{candidate}}$  is the number of candidates which are selected by the excess power method and is estimated as

$$\begin{aligned} N_{\text{candidate}} &= N_{\text{chunk}} \times \text{FAP}_{\text{EP}}, \\ &= N_{\text{grid}} \cdot N_{\text{bin}} \cdot \frac{N_{\text{seg}}}{(\delta\ell)_{\text{slide}}} \times \text{FAP}_{\text{EP}}, \\ &\simeq 4.6 \times 10^{10} \left( \frac{\text{FAP}_{\text{EP}}}{10^{-2}} \right). \end{aligned} \quad (5.3)$$

Substituting it, we obtain

$$\mathcal{T}_{\text{NN}} \simeq 6.4 \times 10^6 \text{ sec} \left( \frac{\text{FAP}_{\text{EP}}}{10^{-2}} \right). \quad (5.4)$$

Therefore, we focus on the case  $\text{FAP}_{\text{EP}} \leq 10^{-2}$ . The computational cost of our analysis is dominated mainly by the preprocessing of the observed strain data and the follow-up analysis. The computational cost of the entire analysis is denoted by  $\mathcal{N}_{\text{comp}}$  and is approximately calculated by

$$\mathcal{N}_{\text{comp}} = \mathcal{N}_{\text{preprocess}} + \mathcal{N}_{\text{follow-up}}, \quad (5.5)$$

where  $\mathcal{N}_{\text{preprocess}}$  and  $\mathcal{N}_{\text{follow-up}}$  are the computational cost of the preprocessing and the follow-up analysis, respectively. In this work, we fix the STFT segment duration and the length of the chunk. Thus, the computational cost of the preprocessing is a constant.

$$\mathcal{N}_{\text{preprocess}} = N_{\text{grid}} \times (\mathcal{N}_{\text{STFT}} + \mathcal{N}_{\text{FFT}} \times N_{\text{bins}}). \quad (5.6)$$

The computational cost of the STFT is

$$\mathcal{N}_{\text{STFT}} = N_{\text{seg}} \cdot 5T_{\text{seg}}f_s \log_2 T_{\text{seg}}f_s, \quad (5.7)$$

and the computational cost of FFT is

$$\mathcal{N}_{\text{FFT}} = 5N_{\text{seg}} \log_2 N_{\text{seg}}. \quad (5.8)$$

With  $T_{\text{obs}} = 2^{24}$  sec,  $T_{\text{seg}} = 2^5$  sec,  $f_s = 2^{10}$  Hz, and  $N_{\text{grid}} \sim 3.5 \times 10^5$ , the computational cost of the preprocess is estimated as

$$\mathcal{N}_{\text{preprocess}} \simeq 2.3 \times 10^{18}. \quad (5.9)$$

On the other hand, the computational cost of the follow-up analysis is determined by the combination of  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$ . The computational cost of the follow-up analysis is

$$\mathcal{N}_{\text{follow-up}} = N_{\text{candidate}} \times \frac{\pi(r_{\text{NN}})^2}{(\delta\theta)_{\text{coh}}^2} \times \mathcal{N}_{\text{FFT,coh}}. \quad (5.10)$$

Here,  $(\delta\theta)_{\text{coh}}^2$  is the typical area of region where each grid point of the coherent analysis covers [see Eq. (3.5)]. The computational cost of taking match is dominated by the Fourier transform and calculated as

$$\mathcal{N}_{\text{FFT,coh}} = 5(T_{\text{obs}}f_{s,\text{coh}}) \log_2(T_{\text{obs}}f_{s,\text{coh}}) \simeq 5.0 \times 10^7, \quad (5.11)$$

where  $f_{s,\text{coh}} = 1/T_{\text{seg}} = 2^{-5}$  Hz. Therefore, we estimate the computational cost of the follow-up analysis as

$$\mathcal{N}_{\text{follow-up}} = 9.0 \times 10^{22} \left( \frac{r_{\text{NN}}}{10^{-3} \text{ rad}} \right)^2 \left( \frac{\text{FAP}_{\text{EP}}}{10^{-2}} \right). \quad (5.12)$$

Substituting Eqs. (5.9) and (5.12) into Eq. (5.5), we can assess the computational cost of the entire analysis as a function of  $r_{\text{NN}}$  and  $\text{FAP}_{\text{EP}}$ . Figure 3 shows the computational cost for various combinations of  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$ . One can read a feasible combination of  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$  depending on one's available computational resources.

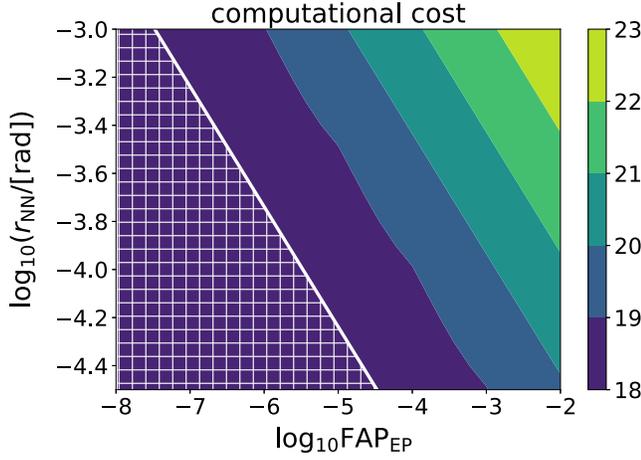


FIG. 3. The logarithm of the evaluated computational cost in the unit of the number of floating point operations. In the white hatched region, the computational cost is dominated by that of the preprocessing, i.e.,  $\mathcal{N}_{\text{follow-up}} \leq \mathcal{N}_{\text{preprocess}}$ . As the false alarm probability of excess power is set to be smaller, the computational cost is reduced because the number of candidates decreases. Also, the computational cost becomes smaller as the parameter  $r_{\text{NN}}$  is shrunk.

### B. False alarm probability

The false alarm probability of the entire process (see [52,53]) is

$$p_{\text{fa}}(\boldsymbol{\rho}) = \{1 - (\text{Prob}[\rho_{\text{EP}} < \hat{\rho}_{\text{EP}} | \rho_{\text{EP}} \sim \mathcal{N}(0, 1)])^{N_{\text{chunk}}}\} \times \{1 - (\text{Prob}[\rho_{\text{MF}} < \hat{\rho}_{\text{MF}} | \rho_{\text{MF}} \sim \mathcal{N}(0, 1)])^{N_t}\}, \quad (5.13)$$

where  $N_t$  is the number of required templates for the coherent search. It can be estimated by

$$N_t = N_{\text{candidate}} \times \frac{\pi(r_{\text{NN}})^2}{(\delta\theta^2)_{\text{coh}}} \times N_{\text{bin,coh}}, \quad (5.14)$$

where  $N_{\text{bin,coh}}$  is the number of the frequency bins of the coherent search. Using the value listed in Table II, we obtain

$$N_t \simeq 5.6 \times 10^{21} \left( \frac{r_{\text{NN}}}{10^{-3} \text{ rad}} \right)^2 \left( \frac{\text{FAP}_{\text{EP}}}{10^{-2}} \right). \quad (5.15)$$

Because the false alarm probability of the follow-up stage determines that of the entire process, we can approximate it as

$$p_{\text{fa}}(\boldsymbol{\rho}) \simeq \{1 - (\text{Prob}[\rho_{\text{MF}} < \hat{\rho}_{\text{MF}} | \rho_{\text{MF}} \sim \mathcal{N}(0, 1)])^{N_t}\}. \quad (5.16)$$

Furthermore, because  $N_t \gg 1$ , we can approximate

$$p_{\text{fa}}(\boldsymbol{\rho}) \simeq N_t \cdot \text{Prob}[\rho_{\text{MF}} > \hat{\rho}_{\text{MF}} | \rho_{\text{MF}} \sim \mathcal{N}(0, 1)]. \quad (5.17)$$

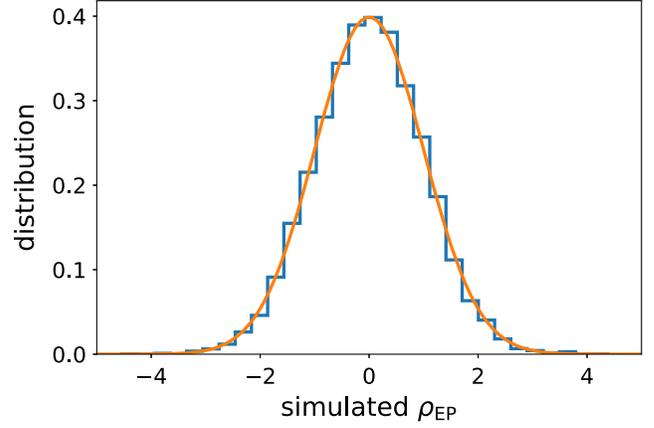


FIG. 4. The histogram of the simulated  $\rho_{\text{EP}}$ . The blue line is the histogram, and the orange line indicates the standard normal distribution. They match well. The  $p$  value of the Kolmogorov-Smirnov test is 0.753254.

In this work, the threshold  $\hat{\rho}_{\text{MF}}$  is chosen so that the false alarm probability of the entire process is 0.01. As shown in Eq. (5.15), the number of templates depends on  $(r_{\text{NN}}, \text{FAP}_{\text{EP}})$ , and the same is true for  $\hat{\rho}_{\text{MF}}$ .

The false alarm probability of the matched filtering has already been studied in the literature. Therefore, in this work, we check only the validity of the statistical properties of the excess power method. It is computationally difficult to treat a whole signal of a duration  $T_{\text{obs}} = 2^{24}$  sec. Therefore, we generate  $N_{\text{seg}}$  short noise data of a duration  $T_{\text{seg}}$  assuming  $S_n(f) = 1$ . After applying a window function, FFT is carried out to each short strain. We pick up a FFT of  $k$ th frequency bin from each FFT data and regard them as  $\{n_{j,k}^{\text{STFT}}\}_{j=1}^{N_{\text{seg}}}$ . We obtain  $\mathbf{N}_{\ell,k}$  by taking the Fourier transform of  $\{n_{j,k}^{\text{STFT}}\}_{j=1}^{N_{\text{seg}}}$  and divide it into  $N_{\text{seg}}/\delta\ell = 128$  chunks. After repeating above procedures for 80 times, 10, 240 chunks are generated. For each chunk, the excess power statistics  $\mathcal{E}$  and SNRs  $\rho_{\text{EP}}$  are calculated. The histogram of the simulated values of  $\rho_{\text{EP}}$  is shown in Fig. 4. It seems to match the standard normal distribution. Additionally, we carry out the Kolmogorov-Smirnov test and obtain a  $p$  value of 0.753254. It is numerically confirmed that the SNR of noise data follows the standard normal distribution.

### C. Detection probability

The detection probability of the signal with an amplitude  $h_0$  can be estimated by

$$\begin{aligned} p_{\text{det}}(h_0; \boldsymbol{\rho}) &= \langle p_{\text{det}}(\boldsymbol{\xi}; \boldsymbol{\rho}) \rangle_{\tilde{\xi}} \\ &= \langle \text{Prob}[\rho_{\text{EP}} > \hat{\rho}_{\text{EP}} | \rho_{\text{EP}} \sim \mathcal{N}(\mu_{\text{EP}}(\boldsymbol{\xi}), \sigma_{\text{EP}}(\boldsymbol{\xi}))] \\ &\quad \times \text{Prob}[\text{source is located in a predicted region} | \boldsymbol{\xi}, r_{\text{NN}}] \\ &\quad \times \text{Prob}[\rho_{\text{MF}} > \hat{\rho}_{\text{MF}} | \rho_{\text{MF}} \sim \mathcal{N}(\mu_{\text{MF}}(\boldsymbol{\xi}), 1)] \rangle_{\tilde{\xi}}, \end{aligned} \quad (5.18)$$

where

$$\langle \cdots \rangle_{\vec{\xi}} := \int d\vec{\xi} (\cdots) \pi(\vec{\xi}) \quad (5.19)$$

is the average over the source parameters  $\vec{\xi}$  with the probability density function  $\pi(\vec{\xi})$ . As explained in Sec. IV D, the neural network is trained with the waveforms sampled only from the vicinity of the reference grid point and the narrow frequency band. It is envisaged that the trained neural network does not work well for signals outside of the reference patch and the frequency band. Therefore, we only test for the limited parameter region. Correspondingly, the average operation is also taken over such narrow parameter space. To quantify the detection power, the amplitude parameter  $h_0^{95\%}$  is defined by

$$p_{\text{det}}(h_0^{95\%}; \boldsymbol{\rho}) = 0.95, \quad (5.20)$$

and correspondingly,

$$\hat{h}_0^{95\%} := h_0^{95\%} \left( \frac{S_n(f_{\text{ref}})}{1 \text{ Hz}^{-1}} \right)^{-1/2}. \quad (5.21)$$

The parameters,  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$ , are optimized so that  $h_0^{95\%}$  takes the smallest value under the condition of the computational power.

To explore the parameter space of  $(\text{FAP}_{\text{EP}}, r_{\text{NN}})$ , we place the regular grid on  $\log_{10} \text{FAP}_{\text{EP}}$  from  $-8$  to  $-2$  by a step of 1, and the regular grid on  $\log_{10} r_{\text{NN}}$  from  $-4.5$  to  $-3.0$  by a step of 0.05. For every pair of  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$ , we calculate  $\hat{h}_0^{95\%}$  by the following procedure. First, we place a regular grid on  $\log_{10} \hat{h}_0$  from  $-2.3$  to  $-1.0$  by a step of 0.05. For one sample of the amplitude, the parameters  $\vec{\xi}$  are randomly sampled. The sampled parameters are denoted by  $\{\vec{\xi}^{(i)}\}_{i=1}^M$ . The waveforms are generated with the sampled parameters. Each waveform is injected into different noise data in the same manner as the method explained in Sec. IV D. The fraction of the events detected is employed as the estimator of the detection probability of the signal with parameter  $(h_0, \vec{\xi}^{(i)})$ . Repeating these procedure for every sampled parameters  $\{\vec{\xi}^{(i)}\}_{i=1}^M$ , we obtain the set of the estimated detection probabilities. Then, the detection probability  $p_{\text{det}}(h_0; \boldsymbol{\rho})$  is estimated by

$$p_{\text{det}}(h_0; \boldsymbol{\rho}) \simeq \frac{1}{M} \sum_{i=1}^M p_{\text{det}}(h_0, \vec{\xi}^{(i)}; \boldsymbol{\rho}). \quad (5.22)$$

Changing the value of the amplitude  $\hat{h}_0$ , we get the estimated detection probability as a function of  $h_0$  for a certain values of  $\text{FAP}_{\text{EP}}$  and  $r_{\text{NN}}$ . If the estimated detection probability exceeds 95% for one or more samples of the amplitude, the obtained detection probabilities are fitted by a sigmoidlike function,

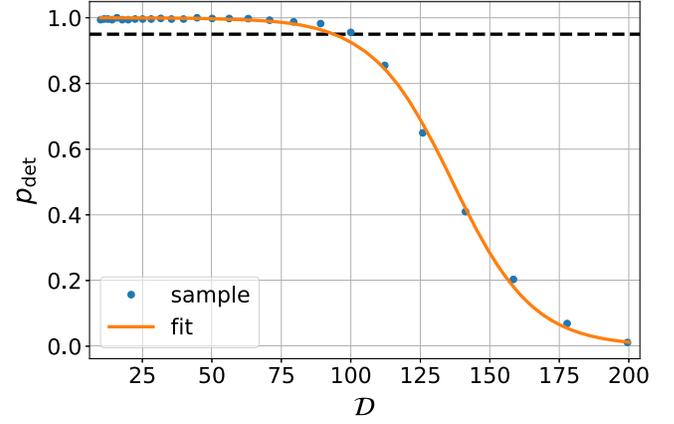


FIG. 5. Example of fitting of detection probability. We set  $\text{FAP}_{\text{EP}} = 10^{-3}$  and  $r_{\text{NN}} = 10^{-3.8}$  rad. Blue dots are estimated values of  $p_{\text{det}}$ , and orange solid line is the fitted sigmoid curve.

$$\zeta(\mathcal{D}; a, b) = \frac{1}{1 + e^{(\mathcal{D}-a)/b}}, \quad (5.23)$$

where

$$\mathcal{D} := (\hat{h}_0)^{-1}, \quad (5.24)$$

is called “the sensitivity depth,” and the parameters  $(a, b)$  of a sigmoid function is to be optimized. Using the optimized parameters  $(a^*, b^*)$ , the estimated value of  $\mathcal{D}^{95\%} := (\hat{h}_0^{95\%})^{-1}$  can be obtained as

$$\mathcal{D}^{95\%}(\text{FAP}_{\text{EP}}, r_{\text{NN}}) = a^* - b^* \ln \frac{1 - 0.95}{0.95}, \quad (5.25)$$

which is the inverse of  $\zeta(\mathcal{D})$ . In this work, we set  $M = 1024$  and the number of noise realization for each parameter set to be 512. Figure 5 shows an example of the fitting. The estimated values of  $\mathcal{D}^{95\%}$  is shown in Fig. 6.

To confirm that the signals with the amplitude  $h_0^{95\%}$  are detected with 95% detection probability, we perform the injection test. To save the computational cost, we skip the follow-up stage and assume that the detection probability is determined by the excess power selection and the neural network analysis. We only use a short chunk centered at the support of the signal as an injection waveform. Ten thousand chunks with various signal parameters  $\vec{\xi}$  are prepared and injected into Gaussian noise data. The waveform model and the noise property are the same as those of the training dataset of the neural network. The excess power is calculated for each chunk, and the neural network analysis is carried out if a chunk is selected as a candidate. Counting the number of detected events, we obtain the recovered value of the detection probability. The procedure shown above is repeated for each combination of  $(\text{FAP}_{\text{EP}}, r_{\text{NN}})$ . Figure 7 shows the result of the injection test. For all combinations of  $(\text{FAP}_{\text{EP}}, r_{\text{NN}})$ , the detection

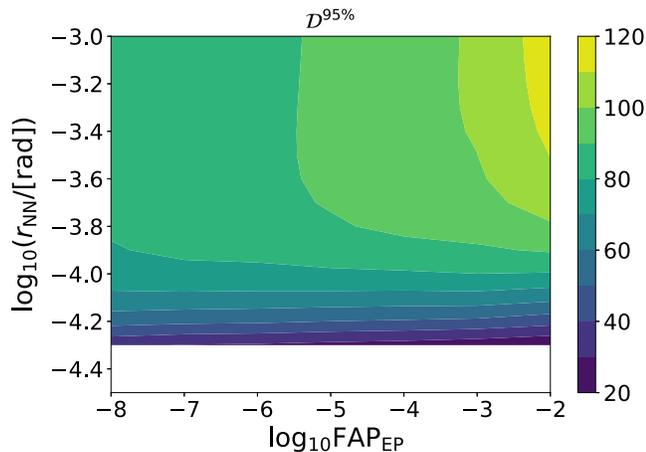


FIG. 6. Estimated sensitivity depths  $\mathcal{D}^{95\%}$ . For parameters within a blank region, the detection probabilities do not reach 95% for a surveyed range of amplitude  $\hat{h}_0$ . For  $r_{\text{NN}} \lesssim 10^{-4}$  rad, the neural network controls the detection probability. On the other hand, for  $r_{\text{NN}} \gtrsim 10^{-3.6}$  rad, the excess power method controls the detection probability because the predicted region is large enough to contain the true location of the source.

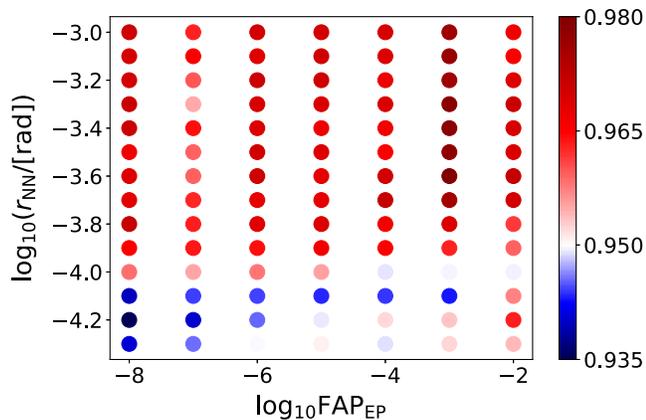


FIG. 7. Recovered values of the detection probability. For all parameters, the detection probabilities are recovered to 95% with the error of only a few percent.

probabilities are close to 95%. Therefore, our estimation of the detectable amplitude  $\hat{h}_0^{95\%}$  is convincing.

## VI. CONCLUSION

We proposed a new method of an all-sky search for continuous gravitational waves, combining the excess power and the deep learning methods. The time resampling and the STFT are used for localizing the signal into a relatively small number of elements in the whole data. Then, the excess power method selects the candidates of the grid point in the sky and the frequency bins where the signal likely exists. The deep neural network narrows down the region to be explored by the follow-up search by two

orders of magnitude than the original area of the sky patch. Before the follow-up coherent search, the heterodyning and the downsampling can reduce the computational cost. We calculated the computational cost of our method. Most of the computational costs are spent by preprocessing the strain data and the follow-up coherent matched filtering search. The computational costs of the excess power method are negligibly small, and the computational time of the neural network can be suppressed to an acceptable level by setting  $\text{FAP}_{\text{EP}} \leq 10^{-2}$ . We estimated the detection abilities of our method with the limited setup where the polarization angle, the inclination angle, and the initial phase are fixed and assumed as known parameters. The dataset for training the neural network and testing is generated from a very narrow parameter space of  $(f_{\text{gw}}, \alpha_s, \delta_s)$ . With a reasonable computational power, the sensitivity depth can be achieved  $\mathcal{D}^{95\%} \gtrsim 80$ .

Our training data, which is used for training the neural network, span the restricted parameter region. Namely, the gravitational wave frequencies of the training data are distributed within the small frequency band centered at 100 Hz of width  $\pm 1/(2T_{\text{seg}})$  and the source locations are sampled from very narrow regions around the fixed grid point. Nevertheless, we can expect our method can be applied to the all-sky search and the frequency band below 100 Hz. If the gravitational wave frequency becomes lower than 100 Hz, the strength of the phase modulation becomes weaker [see Eq. (2.6)]. Therefore, even if  $f_{\text{gw}} < 100$  Hz, the signal power in the  $\ell$  domain would still be concentrated in a narrow region, and it can be expected that the efficiency of the excess power method is maintained. We can employ a similar discussion also for the dependency on the source location. The power concentration in the  $\ell$  domain is still valid even if we take into account the dependency of the source location, while it causes the variation of the signal amplitude. From the above discussion, only slight modifications of the construction of the training data and our neural network structure are enough to apply our strategies to an all-sky search of monochromatic sources having a frequency lower than  $\sim 100$  Hz.

In addition to the above points, there are several avenues for improving our method. We fixed various parameters such as the width of the STFT  $T_{\text{seg}}$  and the length of each chunk  $\delta\ell$  in a little hand-waving manner. Surveying and optimizing these parameters may improve the detection efficiency of our method. Especially, the sampling frequency when downsampling might reduce the computational cost significantly. As can be seen from Eq. (4.27), the deviation  $\delta f_k$  causes the translation of the signal in the  $\ell$  domain. It is expected that we can further constrain the gravitational wave frequency than  $\sim T_{\text{seg}}^{-1}$ . Considering this effect, we can set the sampling frequencies of down-sampled strains to a lower value than our current choice. This optimization would result in the further reduction of the computational time of the follow-up coherent search.

In the present paper, we assumed that the stationary Gaussian detector noise and 100% duty cycle. We also simplified the waveform model, e.g., the frequency change  $df/dt$  is not incorporated. In spite of these simplifications, the obtained results can be regarded as a proof of principle and are enough to convince that our method has the potential for improving the all-sky search for continuous gravitational waves with the duration of  $O(10^7)$  sec. Relaxing these assumptions is beyond the scope of this paper and left as future work.

### ACKNOWLEDGMENTS

We thank Yousuke Itoh for fruitful discussions. This work was supported by JSPS KAKENHI Grant No. JP17H06358 (and also Grant No. JP17H06357), *A01: Testing gravity theories using gravitational waves*, as a part of the innovative research area, ‘‘Gravitational wave physics and astronomy: Genesis’’, and Grant No. JP20K03928. Some part of calculation has been performed by using a GeForce 1080Ti GPU at Nagaoka University of Technology.

### APPENDIX: NOISE STATISTICS IN THE $\ell$ DOMAIN

In general, the power spectral density of a stochastic process  $n(t)$  is defined by

$$\langle \tilde{n}(f)\tilde{n}^*(f') \rangle = \frac{1}{2} S_n(f)\delta(f-f'), \quad (\text{A1})$$

where the Fourier transform of  $n(t)$  is defined by

$$\tilde{n}(f) = \int_{-\infty}^{\infty} dt n(t) e^{-2\pi i f t}, \quad (\text{A2})$$

while we define the STFT by Eq. (4.16). Ignoring the effect of the window function, the variance of  $n_{j,k}^{\text{STFT}}$  can be approximated by

$$\langle (n_{j,k}^{\text{STFT}})(n_{j',k'}^{\text{STFT}})^* \rangle = \frac{1}{2T_{\text{seg}}} S_n(f_k) \delta_{kk'} \delta_{jj'}. \quad (\text{A3})$$

Here, we assume that different STFT bins are statistically independent. The variance of  $\mathbf{N}_{\ell,k}$  is

$$\begin{aligned} \langle \mathbf{N}_{\ell,k} \mathbf{N}_{\ell',k'}^* \rangle &= \frac{1}{N_{\text{seg}}^2} \sum_{j=1}^{N_{\text{seg}}} \sum_{j'=1}^{N_{\text{seg}}} \langle (n_{j,k}^{\text{STFT}})(n_{j',k'}^{\text{STFT}})^* \rangle e^{-2\pi i(j\ell-j'\ell')/N_{\text{seg}}}, \\ &= \frac{S_n(f_k)}{2T_{\text{seg}} N_{\text{seg}}} \delta_{\ell\ell'} \delta_{kk'}. \end{aligned} \quad (\text{A4})$$

Therefore, we get

$$\tilde{\sigma}_k^2 = 2 \langle \mathbf{N}_{\ell,k} \mathbf{N}_{\ell,k}^* \rangle = \frac{S_n(f_k)}{T_{\text{seg}} N_{\text{seg}}}. \quad (\text{A5})$$

- 
- [1] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. Lett.* **116**, 0621102 (2016).
  - [2] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. X* **9**, 031040 (2019).
  - [3] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), [arXiv:2010.14527](https://arxiv.org/abs/2010.14527).
  - [4] Y. Aso, Y. Michimura, K. Somiya, M. Ando, O. Miyakawa, T. Sekiguchi, D. Tatsumi, and H. Yamamoto (KAGRA Collaboration), *Phys. Rev. D* **88**, 043007 (2013).
  - [5] I. Bala, S. Tarun, U. CS, D. Sanjeev, R. Sendhil, and S. Anand, <https://dcc.ligo.org/LIGO-M1100296/public> (2011).
  - [6] B. P. Abbott *et al.* (KAGRA, LIGO Scientific, and Virgo Collaborations), *Living Rev. Relativity* **23**, 3 (2020).
  - [7] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. X* **9**, 011001 (2019).
  - [8] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Astrophys. J. Lett.* **900**, L13 (2020).
  - [9] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), [arXiv:2010.14533](https://arxiv.org/abs/2010.14533).
  - [10] C. M. Will, *Living Rev. Relativity* **17**, 4 (2014).
  - [11] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. D* **100**, 104036 (2019).
  - [12] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), [arXiv:2010.14529](https://arxiv.org/abs/2010.14529).
  - [13] M. Maggiore, *Phys. Rep.* **331**, 283 (2000).
  - [14] B. S. Sathyaprakash and B. F. Schutz, *Living Rev. Relativity* **12**, 2 (2009).
  - [15] M. Maggiore, *Gravitational Waves. Vol. 1: Theory and Experiments*, Oxford Master Series in Physics (Oxford University Press, New York, 2007), ISBN.
  - [16] J. D. E. Creighton and W. G. Anderson, *Gravitational-Wave Physics and Astronomy: An Introduction to Theory, Experiment and Data Analysis* (Wiley-VCH, Weinheim, Germany, 2011).
  - [17] S. J. Zhu, M. Baryakhtar, M. A. Papa, D. Tsuna, N. Kawanaka, and H.-B. Eggenstein, *Phys. Rev. D* **102**, 063020 (2020).
  - [18] A. Arvanitaki, M. Baryakhtar, and X. Huang, *Phys. Rev. D* **91**, 084011 (2015).
  - [19] R. Brito, S. Ghosh, E. Barausse, E. Berti, V. Cardoso, I. Dvorkin, A. Klein, and P. Pani, *Phys. Rev. D* **96**, 064050 (2017).

- [20] P. Jaranowski, A. Krolak, and B. F. Schutz, *Phys. Rev. D* **58**, 063001 (1998).
- [21] B. Krishnan, A. M. Sintes, M. A. Papa, B. F. Schutz, S. Frasca, and C. Palomba, *Phys. Rev. D* **70**, 082001 (2004).
- [22] P. Astone, A. Colla, S. D’Antonio, S. Frasca, and C. Palomba, *Phys. Rev. D* **90**, 042002 (2014).
- [23] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. D* **96**, 062002 (2017).
- [24] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. D* **97**, 102003 (2018).
- [25] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), *Phys. Rev. D* **100**, 024004 (2019).
- [26] V. Dergachev and M. A. Papa, *Phys. Rev. Lett.* **125**, 171101 (2020).
- [27] V. Dergachev and M. A. Papa, *Phys. Rev. D* **103**, 063019 (2021).
- [28] D. George and E. A. Huerta, *Phys. Rev. D* **97**, 044039 (2018).
- [29] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, [arXiv:1909.06296](https://arxiv.org/abs/1909.06296).
- [30] A. J. K. Chua and M. Vallisneri, *Phys. Rev. Lett.* **124**, 041102 (2020).
- [31] H. Nakano, T. Narikawa, K.-I. Oohara, K. Sakai, H.-A. Shinkai, H. Takahashi, T. Tanaka, N. Uchikata, S. Yamamoto, and T. S. Yamamoto, *Phys. Rev. D* **99**, 124032 (2019).
- [32] T. S. Yamamoto and T. Tanaka, [arXiv:2002.12095](https://arxiv.org/abs/2002.12095).
- [33] D. George, H. Shen, and E. A. Huerta, *Phys. Rev. D* **97**, 101501(R) (2018).
- [34] A. J. K. Chua, M. L. Katz, N. Warburton, and S. A. Hughes, *Phys. Rev. Lett.* **126**, 051102 (2021).
- [35] C. Dreissigacker, R. Sharma, C. Messenger, R. Zhao, and R. Prix, *Phys. Rev. D* **100**, 044009 (2019).
- [36] C. Dreissigacker and R. Prix, *Phys. Rev. D* **102**, 022005 (2020).
- [37] F. Morawski, M. Bejger, and P. Ciecielag, *Mach. Learn. Sci. Tech.* **1**, 025016 (2020).
- [38] B. Beheshtipour and M. A. Papa, *Phys. Rev. D* **101**, 064009 (2020).
- [39] V. Dergachev, *Phys. Rev. D* **85**, 062003 (2012).
- [40] H. Nakano, H. Takahashi, H. Tagoshi, and M. Sasaki, *Phys. Rev. D* **68**, 102003 (2003).
- [41] W. G. Anderson, P. R. Brady, J. D. E. Creighton, and E. E. Flanagan, *Phys. Rev. D* **63**, 042003 (2001).
- [42] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016), <http://www.deeplearningbook.org>.
- [43] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, in *2009 IEEE 12th International Conference on Computer Vision (ICCV)* (IEEE Computer Society, Los Alamitos, CA, USA, 2009), pp. 2146–2153, ISSN 1550-5499, <https://doi.ieeecomputersociety.org/10.1109/ICCV.2009.5459469>.
- [44] Y. LeCun and Y. Bengio, in *The Handbook of Brain Theory and Neural Networks* (MIT Press, Cambridge, MA, USA, 1998), pp. 255–258, ISBN .
- [45] Y. Zhou and R. Chellappa, in *IEEE 1988 International Conference on Neural Networks* (IEEE, San Diego, CA, USA, 1988), pp. 71–78.
- [46] B. T. Polyak, *USSR Computational Mathematics and Mathematical Physics* **4**, 1 (1964).
- [47] G. Hinton, *Neural Networks for Machine Learning* (2012), <http://www.cs.toronto.edu/~hinton/coursera/lecture6/lec6.pdf>.
- [48] D. P. Kingma and J. Ba, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [49] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Nature (London)* **323**, 533 (1986).
- [50] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, in *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., Red Hook, NY, USA, 2019), pp. 8024–8035.
- [51] R. J. Dupuis and G. Woan, *Phys. Rev. D* **72**, 102002 (2005).
- [52] K. Wette, *Phys. Rev. D* **85**, 042003 (2012).
- [53] C. Dreissigacker, R. Prix, and K. Wette, *Phys. Rev. D* **98**, 084058 (2018).