Deep-learning classification and parameter inference of rotational core-collapse supernovae

Solange Nunes[1,*] Gabriel Escrig[2,†] Osvaldo G. Freitas[1,3,‡] José A. Font[3,4,§] Tiago Fernandes[1,‖]
Antonio Onofre[1,¶] and Alejandro Torres-Forné[3,**]

[1]*Centro de Física das Universidades do Minho e do Porto (CF-UM-UP),
Universidade do Minho, 4710-057 Braga, Portugal*
[2]*Departamento de Física Teórica, Universidad Complutense de Madrid,
Plaza de las Ciencias 1, 28040 Madrid, Spain*
[3]*Departamento de Astronomía y Astrofísica, Universitat de València,
Dr. Moliner 50, 46100, Burjassot (València), Spain*
[4]*Observatori Astronòmic, Universitat de València, Catedrático José Beltrán 2,
46980, Paterna (València), Spain*

We test deep-learning (DL) techniques for the analysis of rotational core-collapse supernovae (CCSN) gravitational-wave (GW) signals by performing classification and parameter inference of the maximum (peak) frequency and the GW strain amplitude (Δh) multiplied by the luminosity distance (D) attained at core bounce, respectively, (f_{peak}) and ($D \cdot \Delta h$). Our datasets are built from a catalog of numerically generated CCSN waveforms assembled by Richers *et al.* 2017. Those waveforms are injected into noise from the Advanced Laser Interferometer Gravitational Wave Observatory and Advanced Virgo detectors corresponding to the O2 and O3a observing runs. For a network signal-to-noise ratio (SNR) above 5, our classification network using time series detects Galactic CCSN GW signals buried in detector noise with a false positive rate of 0.10% and a 98% accuracy, being able to detect all signals with SNR > 10. The inference of f_{peak} is more accurate than for $D \cdot \Delta h$, particularly for our datasets with the shortest time window (0.25 s) and for a minimum SNR = 15. From the calibration plots of predicted versus true values of the two parameters, the standard deviation (σ) and the slope deviation with respect to the ideal value are computed. We find $\sigma_{D \cdot \Delta h} = 52.6$ cm and $\sigma_{f_{\text{peak}}} = 18.3$ Hz, with respective slope deviations of 11.6% and 8.3%. Our best model is also tested on waveforms from a recent CCSN catalog built by Mitra *et al.* 2023, different from the one used for the training. For these new waveforms, the true values of the two parameters are mostly within the 1σ band around the network's predicted values. Our results show that DL techniques hold promise to infer physical parameters of Galactic rotational CCSN events.

I. INTRODUCTION

The gravitational collapse of the core of massive stars and the subsequent explosion as a supernova (a core-collapse supernova event, or CCSN event hereafter) is one of the most interesting sources of gravitational waves (GWs) to be detected in the coming years (see [1] for a recent review). CCSN events are thought to be the end result of stars with a zero age main sequence mass larger than about $8M_\odot$. The core of these stars, consisting of iron-group nuclei, can no longer sustain nuclear burning and collapses. As the core reaches densities above nuclear saturation density and the equation of state (EOS) stiffens, the infalling material bounces back launching an outward-moving shock wave. Numerical simulations have long helped understand the processes by which the shock wave can gain sufficient energy to power a supernova explosion. Two main mechanisms have been proposed, neutrino-driven explosions and magneto-rotational explosions (see [1,2] and references therein). Rotation strongly affects the dynamics of CCSN events and, in turn, its GW emission. Most CCSN events are nonrotating or slowly rotating [3]. In these models, the GW emission is largely stochastic as it is triggered by convection and by the standing accretion shock instability. In rapidly rotating stars, however, the early part of the GW signal is well determined. In these

*Contact author: solangesilnunes@gmail.com
†Contact author: gescrig@ucm.es
‡Contact author: ogf1996@gmail.com
§Contact author: j.antonio.font@uv.es
‖Contact author: tiagosfernandes@ua.pt
¶Contact author: antonio.onofre@cern.ch
**Contact author: alejandro.torres@uv.es

models the explosion is powered by the rotational kinetic energy (see [4] and references therein), leading to a bounce GW signal that is well understood. The GW strain of rapidly-rotating cores reaches its maximum right before bounce, and its value depends on the corresponding degree of oblateness of the core. Numerical simulations have shown that for about 10 ms after core bounce the newly formed proto-neutron star (PNS) undergoes a series of oscillations driven by the excitation of axisymmetric fluid modes. Shortly after, the GW signal becomes stochastic as a result of the growth of nonaxisymmetric hydrodynamical instabilities. One to three CCSN events are expected to occur in the Milky Way per century [5,6] and less than 10% are likely to be from fast-rotating progenitors [7].

To date, all confident GW signals observed by the Advanced Laser Interferometer Gravitational Wave Observatory (LIGO) and Advanced Virgo detectors correspond to coalescing compact binaries (CBCs) [8–11]. The most recent optically-targeted searches for GWs from CCSN events, using data from the first three observing runs of LIGO and Virgo, found no evidence of significant candidates [5,12]. As the sensitivity of the detectors increases, the probability of detecting GW signals from CCSN events will also increase, especially within our Galaxy and nearby satellite galaxies [5,12–17]. The successful detection of GW signals from CCSN events may help clarify the underlying physical processes occurring in the cores of massive stars during their gravitational collapse along with providing a brand new channel to infer properties of the source, allowing to probe the properties of the PNS, the nuclear EOS, the rotation of the core, and the explosion mechanism [17–32].

The detection of GW signals from CBCs is achieved through matched filtering [33]. This requires template waveform models to be precomputed with a faithful representation of true GWs. On the contrary, by their stochastic nature, CCSN GW signals are unmodeled, requiring a completely different approach for their detection. In this case, a coherent time-frequency analysis of the data in a network of detectors is used [12,34]. Correspondingly, the estimation of the source parameters in the case of CBC signals is done through Bayesian inference. This method uses a large number of waveforms (or "approximants") that cover a wide parameter space and applies stochastic sampling techniques to evaluate their likelihood functions. Being computationally expensive, the Bayesian approach is, in general, not the optimal method to perform parameter inference of CCSN GW signals for several reasons. On the one hand, postbounce signals rapidly become stochastic a few milliseconds after core bounce. On the other hand, the amount of CCSN waveforms available is severely limited by the computational cost of the simulations and by the impossibility to build the waveforms using the same standard approximations employed for compact binaries. For the fairly brief "deterministic" part of the early bounce signal, matched-filtering analysis or Bayesian model

selection can be used to infer source properties (see, e.g., [18,29]). We also note that the dominant features of CCSN waveforms can be extracted with principal component analysis (PCA), where a mapping between their measured eigenvectors and the physical parameters of the progenitor star can be created for third-generation detectors [24]. Moreover, for close enough sources, it is also possible to distinguish neutrino-driven CCSN events from magneto-rotational CCSN events using PCA [19,30,35–37].

An alternative to the methods mentioned before are machine learning (ML) techniques which have been shown to be significantly less time-consuming than Bayesian approaches in performing parameter estimation of CBC signals [38–44]. The potential of ML for GW data analysis is becoming increasingly relevant, and the number and scope of applications are already important (see [45–48] for recent reviews). Those include, e.g., detection methods for CBC signals [49–51], signal quality improvements [52,53], waveform generation [54–56], and noise transient simulations in detectors [57,58]. In the context of CCSN GW signals the application of ML techniques is also an active field of research [28,30,32,59–63]. The potential of convolutional neural networks (CNNs) to detect CCSN GW signals was first shown by [59] using phenomenological waveforms injected in Gaussian noise. CNNs were also employed by [60] with CCSN GW signals injected in detector noise. At a false alarm probability of 10%, neutrino-driven explosions at 10 kpc yield an expected true alarm probability of 55% (76%) for current (future) LIGO-Virgo-KAGRA (LVK) detectors, while for magneto-rotational explosions at 50 kpc the corresponding values increase to 84% (92%). Approaches based on learned dictionaries were first applied in [28] for CCSN signal classification, obtaining 85% true classifications on signals with a SNR from 15 to 20 (see also [30]). In [61] a mini-inception residual network (ResNet) for time-frequency images (also implementing convolutional layers) was used to improve the detectability of CCSN GW signals with the LVK pipeline coherent wave burst [34]. For a dataset comprising phenomenological CCSN waveforms in background noise from O2, an efficiency of around 80% was obtained for SNR ~ 16. Recently [32] have used ML to investigate if it is possible to infer the iron core mass from the bounce and early ring-down GW signal of rapidly rotating CCSN models, and [62] have compared the performance of different CNNs and long short-term memory networks for multilabel classification of CCSN simulated signals and noise transients using real data. CNNs have also been used by [63] to classify GW from CCSN events using spectrograms from numerical simulations injected onto real noise data from O3. We also note the recent work by [30] which presents a comprehensive comparison of different methods (Bayesian model selection, dictionary learning, and CNNs) to determine the explosion mechanism from a GW CCSN detection using

up-to-date waveforms from simulations (including, in particular, new three-dimensional (3D) long-duration magneto-rotational CCSN waveforms that cover the full explosion phase).

In this work, we present a deep learning (DL) approach for classification and parameter estimation using GWs from rapidly rotating CCSN events. We test residual CNN algorithms [64,65] using datasets built from the numerically generated CCSN waveforms computed by [20], which are injected into noise from the Advanced LIGO and Advanced Virgo detectors. We initially test these algorithms with spectrograms for classification as a proof of concept, building on previous group results on CBCs [49]. Additionally, we also test the same algorithms for classification and parameter estimation using time series. A different set of waveforms for rapidly rotating CCSN events, developed by [32], is used to assess the generality of our approach. The tests in the time domain follow closely the recent work of [29] and perform our inference on two key parameters, the GW strain amplitude (Δh) multiplied by the luminosity distance (D), $D \cdot \Delta h$, and the maximum (peak) frequency, f_{peak}, attained at core bounce. As discussed in [29], the results from [20] for a $12 M_\odot$ progenitor and a large group of equations of state show that when the ratio of rotational-kinetic energy to gravitational potential energy, $T/|W|$, is below 0.06 (i.e., for slowly rotating cores), the first parameter is proportional to $T/|W|$, and the second one is proportional to the square root of the central density, $\sqrt{\rho_c}$. These relationships allow, in principle, the inference of PNS properties from the GW detection of a rotating CCSN event. In [29] inference on these two parameters was conducted using Bayesian model selection employing a master waveform template built from the waveform catalog of [20]. Our approach, based on DL, can thus be regarded as complementary to the Bayesian approach of [29]. As we show in this work, for rotating Galactic CCSN events, the inference of these two parameters using neural networks can be successfully achieved, yielding results within the 1σ band from the expected (true) values.

This paper is organized as follows: In Sec. II we describe the CCSN waveforms we select to train our models. In Sec. III we discuss a first method in which the analysis of the signals is performed using spectrograms, focusing on signal classification only. Section IV presents a second, more complete, method. Here, DL methods are applied to time series, and we discuss results for both signal classification and parameter inference. Finally, our conclusions are outlined in Sec. V.

II. WAVEFORM SELECTION

Our models are trained with waveforms from the Richers et al. catalog [20]. This catalog is composed of 1824 numerical waveforms obtained from the collapse of a $12 M_\odot$ progenitor, using 18 different equations of state

and covering a parameter space of 98 rotation profiles. The simulations from [20] focus on the bounce signal, including the collapse phase and the postbounce evolution up to ~ 50 ms after bounce. The bounce signal has been extensively studied in previous works [18,20,66–71], and it is considered the key signature for the presence of rotation in CCSN events. As mentioned before, our study will focus only on the part of the signal corresponding to the bounce and the following 10 ms.

The waveforms were obtained through a combination of 1D simulations of the collapse-phase deleptonization with the code GR1D [72] and 2D core-collapse simulations with the CoCoNuT Code [73]. To produce the collapse-phase deleptonization, Richers et al. used an approximation proposed by Liebendörfer [74] where the electron fraction (Y_e) is parametrized as a function of density (ρ) only from spherically symmetric (1D) nonrotating general relativity hydrodynamic simulations. This approximation was also used for the rotating case, as electron captures and neutrino interactions with matter are local and depend on the density in the collapse phase. In this case, the rotational flattening of the collapsing core can be considered relatively small. The axisymmetric (2D) CCSN simulations were performed assuming the conformal flat condition approximation for the spacetime metric, and by forcing the initial model to rotate with constant angular velocity according to the rotation law

$$\Omega(r) = \Omega_0 \left[1 + \left(\frac{r}{A} \right)^2 \right]^{-1}, \tag{1}$$

where $\Omega(r)$ is the angular velocity, r is the distance from the axis of rotation, and Ω_0 and A are free parameters that determine the rotational speed/energy of the model and the distribution of angular momentum.

Following [29], from the Richers et al. catalog [20] we only consider waveforms with $\Omega_0 \geq 3.0$ rad s^{-1}. This is done to avoid the numerical noise present in models with slow rotation rates, which gets amplified when performing the normalization of the signals. Moreover, we do not use any of the simulations that do not collapse within the first second (see Table III of [20]). Considering the above constraints only 999 waveforms remain from the catalog. While the injections on the datasets used for the analyses with spectrograms employ the full catalog of 1824 waveforms, datasets created for the analyses with time series only apply the selected 999 waveforms.

The waveforms assume an optimally oriented source. Therefore, for an axisymmetric system there is no cross-polarization ($h_\times$), and only the plus-polarization component (h_+) is different from zero, i.e., $h_\times = 0$ and $h_+ = h_+^{\text{opt}} \sin^2 \theta$. Here, θ is the inclination angle between the rotational axis of the core and the line of sight of the observer. We fix this angle to $\pi/2$ rad, simplifying the expression above to $h_+ = h_+^{\text{opt}}$. Each waveform on the

catalog provides several physical parameters, from which we used the strain in units of distance ($D \cdot h_+^{\text{opt}}$), the timestamps ($t - t_{\text{b}}$), the peak frequency (f_{peak}), the maximum rotational velocity Ω_0, and the GW strain amplitude (Δh) multiplied by the luminosity distance (D), ($D \cdot \Delta h$).

III. ANALYSIS WITH SPECTROGRAMS

A. Dataset generation and neural network details

Our first analysis of the waveforms is based on time-frequency representations of the signals or spectrograms. These diagrams represent the signal amplitude in pixel intensity, with time on the x axis and frequency on the y axis. The use of spectrograms is not arbitrary. Since the bounce and postbounce phases of CCSN events involve many frequencies, different patterns may be found in the time-frequency representation (produced by splitting the input time series into shorter sections and taking Fourier/wavelet transforms of these sections) when compared with the time-strain representation. These patterns, being contained in a short time interval and having many frequencies, are seen in a spectrogram as an intense narrow vertical band, as shown in Fig. 1 (right).

In order to train and validate a neural network (NN) it is necessary to have a large enough data bank. To perform the classification test, we created a dataset consisting of images of only noise and others with injected signals. For both signal and noise spectrograms, we randomly selected time series segments of actual detector noise from O2 (corresponding to August 2017), with a duration of 4096 ms, without any specific criteria. We normalized the intensity of each frequency using a whitening procedure to account for variations in detector sensitivity across the frequency range. Our analysis considers frequencies between 20 and 1000 Hz, where the LIGO and Virgo detectors are most sensitive. We also notched out fixed frequencies affected by known artifacts, specifically 60, 120, and 240 Hz for the LIGO detectors and 50, 100, and 200 Hz for the Virgo detector, corresponding to the U.S. and European power grid frequency and their first harmonics, respectively. The

sampling frequency was set to 16384 Hz. High-resolution time-frequency maps with a final time duration of 200 ms were constructed using the Q-transform method from GWpy [75], with its default parameters. For the signal spectrograms, the only difference was the injection of a waveform at a random time position into a section of actual detector noise before whitening. We utilize all the 1824 waveforms from the Richers *et al.* catalog [20], described in Sec. II, assuming a constant sky position, fixing the distance to the source to 20 kpc and the right ascension, declination, and polarization angle to zero.

The resulting images have dimensions of 256×256 pixels, where the frequencies and time are distributed linearly on their respective axis. Since we use three detectors, the images are coded with RGB colors where red, green, and blue correspond to the spectrograms of LIGO-Hanford, LIGO-Livingston, and Virgo detectors, respectively. In addition, the intensity of each color varies between 0 and 255, so that when injecting a very intense signal, the normalization makes the noise almost imperceptible. In the left panel of Fig. 1 a RGB image of only real background noise is shown, while the right panel displays a RGB image with an injected CCSN GW signal. In the latter, the signal stands out over the LIGO-Hanford and LIGO-Livingston noise. It shows two narrow stripes with a rapid variation in frequency, associated with the burst, and two subsequent features appearing shortly after, connected with the oscillations of the PNS. The noise is barely noticeable in the green and red channels, due to the normalization, while in the blue channel, the intensity of the injections is much lower than the noise (hence, the normalization makes the noise stand out). Moreover, a delay can be observed between the detection of the two interferometers.

A bank of 10^4 images was generated, where half of them are only noise (labeled as "background") and the other half have CCSN GW signals injected into noise (labeled as "signal"). This dataset was divided into two: a training set (85%) and a validation set (15%). In addition, the SNR of each signal was evaluated, serving as a filter to eliminate weak signals (SNR < 5) that would otherwise only confuse our NN. We always consider the network SNR amplitude of the signals, which combines the individual SNR for each GW interferometer in the network, LIGO-Hanford, LIGO-Livingston, and Virgo, defined as

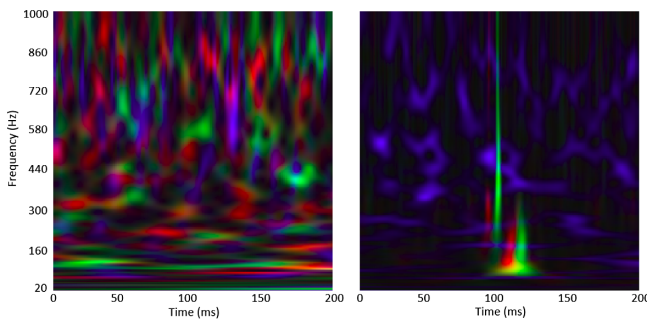$$\text{network SNR} = \sqrt{\sum_{i=1}^{3} \text{SNR}_i^2}. \qquad (2)$$

FIG. 1. RGB images of only real noise (left) and with an injected GW into real noise conditions (right) where the Hanford, Livingston, and Virgo interferometer spectrograms are represented in red, green, and blue, respectively.

The individual SNR_i is found by calculating the cross-correlation matrix between the postinjected strain and the injected template after projecting into the corresponding detector. Both strains are treated with whitening, bandpass, and notch filtering, as explained before. The maximum
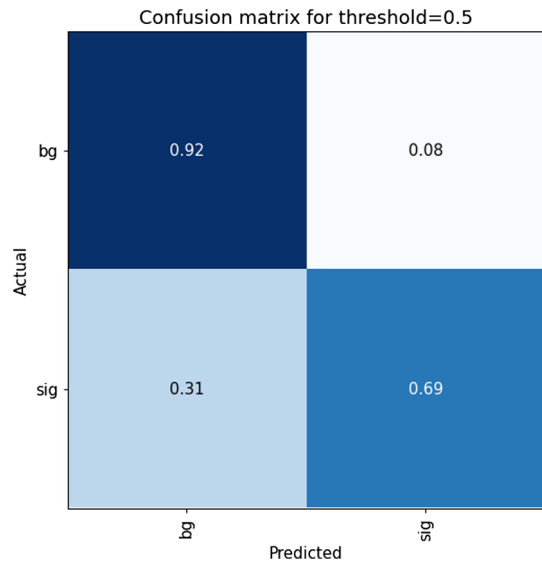
FIG. 2. Confusion matrix for the classification test with the 13th epoch weights showing the fractions of strains predicted as background noise or signal versus their actual labels.

value of this cross-correlation matrix is considered as the SNR_i.

As our study is based on images, we work with CNNs. These must be capable of differentiating between background and signal. We use a pretrained ResNet101 model which implements an architecture with a residual CNN 101 layers deep [76]. We load the ResNet101 model pretrained with over one million images from the ImageNet database [77]. The pretrained network can classify images into 1000 object categories. As a result, the network has learned feature-rich representations for a wide range of images, although we will only need it to distinguish between two classes, background and signal. To accomplish this, we need to adapt the network by changing its last layer of 1000 neurons to one with only 2 of them. For the generation of the datasets, functions from the PYTHON [78] libraries SciPy [79], PyCBC [80], and GWpy [75] were used. We also used the libraries FAST.AI [81] for the classification tests.

B. Classification results with spectrograms

Our classification test aims at distinguishing the dataset elements where a CCSN GW signal been injected from those where only detector background noise is present. For a training of 25 epochs, the network found the minimum error in the 13th epoch. The model obtained had an accuracy[1] of 82%, providing the confusion matrix displayed in Fig. 2. This figure shows that 92% of the background images are correctly classified, with a false

---

[1]The accuracy is defined as $\frac{TP+TN}{TP+TN+FP+FN}$, where TP (TN) corresponds to the correctly classified signals (background) and FP (FN) corresponds to the backgrounds (signals) that are misclassified.
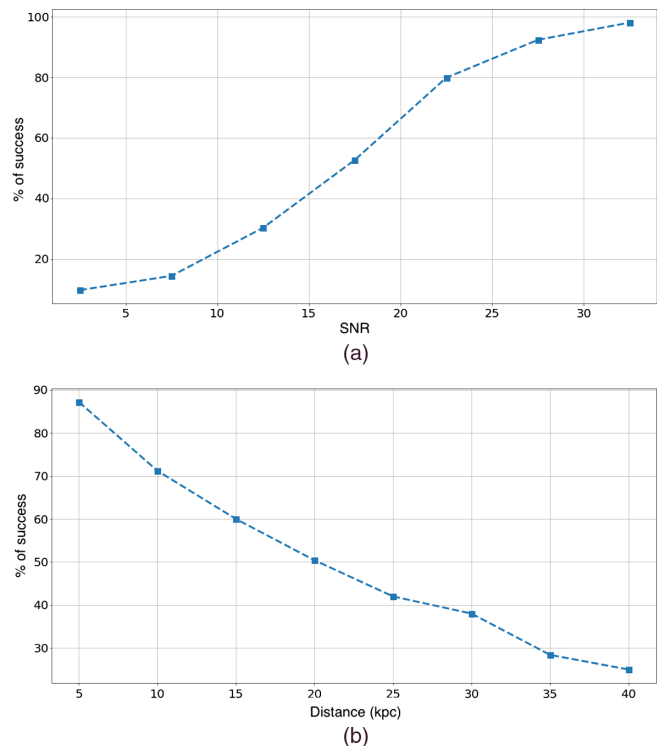

FIG. 3. Percentage of success in the validation of our network to detect a CCSN GW signal as a function of SNR (a) and distance (b).

positive rate (FPR) of 8%. While the model only classifies 69% of the signals correctly, a precision of 90% is obtained. These results show that our model detects signals with high confidence, even if it only detects a modest amount of GW signals.

C. Testing the classification model

We further tested our model by performing additional studies with two different datasets sorted by SNR and by distance to the source. One dataset consists of 1500 images arranged in groups differentiated by five SNR units, with fixed distance and sky position, similar to the datasets used for training. The other dataset consists of 4000 images with distances ranging between 5 and 40 kpc, keeping a fixed sky position and without any limits on the SNR. Figure 3(a) shows the relationship between the percentage of success of the network in detecting signals and the SNR of the injected GW. It is interesting to highlight that an accuracy of 95% and a precision of 93% is reached for SNR > 30. Although the network was trained to detect a GW generated at a distance of 20 kpc, it can also be used to infer GW signals at different distances. In Fig. 3(b), we observe that at shorter distances, the percentage of success is higher, as expected, decreasing as the source gets farther away. We observe a discrepancy for a distance of 20 kpc with respect to the value obtained previously during validation. This is due to the fact that when generating images for training and

validation, those with SNR < 5 were eliminated, as mentioned above, a condition that we did not apply here.

IV. ANALYSIS WITH TIME SERIES

A. Dataset generation

In our second method, we apply DL techniques in the time domain. While significant patterns may be more easily recognized in the time-frequency representation, some information might be lost during the Q-transform process, which would not occur when using time series directly. In this case, we only use the 999 selected waveforms from the Richers *et al.* catalog, described in Sec. II. For each of the selected waveforms, we scale the strain $D \cdot h_+^{\text{opt}}$ to a randomly chosen distance of the source between 5 and 20 kpc (i.e., a CCSN that occurs in our Galaxy). For the background noise we use publicly available data from both LIGO (Hanford and Livingston) and Virgo, with an initial Global Positioning System (GPS) time $t_{\text{GPS}} = 1253326755$ s, which corresponds to the end of September 2019, i.e., around the middle of the O3 run. We define a random time $t_0 \in [6, 1750]$ s and select a segment of the noise time series with a window $[t_0 - 5, t_0 + 10]$ s for the background sample. The waveforms were down-sampled to match the noise sampling frequency. In this analysis we used a sampling frequency of 4096 Hz, allowing faster performance without losing information. Each waveform was projected into each detector with random values for the angles defining the sky position (declination, right ascension, and polarization angle). Then, the projected waveforms are injected into noise samples for each detector, having the time of bounce of the signal (t_b) randomly set in the interval $[t_0, t_0 + 0.80]$ s, for the specific case of datasets with one-second time window. For completeness, we also consider datasets of 0.50 and 0.25 s with t_b in the range $[t_0, t_0 + 0.40]$ s and $[t_0, t_0 + 0.20]$ s, respectively.

We perform whitening on the resulting strains (both for the background and signal classes) through inverse spectrum truncation using the amplitude spectral density of the three detectors' noise. We then apply a bandpass filter from 20 to 1000 Hz and notch filters at the individual frequencies of 60, 120, and 240 Hz for the two LIGO strains and 50, 100, and 200 Hz for the Virgo strains. We finally apply a crop on the strains to get the expected time window (1.00, 0.50, and 0.25 s). At this point, a selection is made on the resulting whitened background section to exclude high-intensity glitches (such as blips). After whitening, the maximum absolute amplitude is around 1.5×10^{-21}, so we only allow segments with a maximum absolute amplitude below 2.7×10^{-21}.

For the classification and parameter estimation tests in the time domain, we use datasets of 10^4 elements, where each element is composed of three time series, one for each detector. Examples of these time series are given in Figs. 4
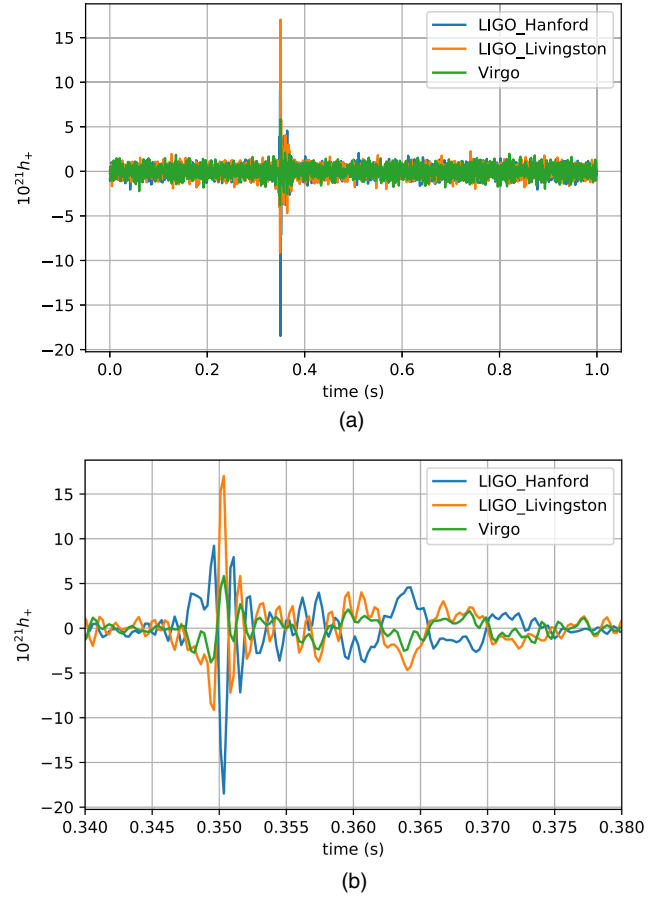

FIG. 4. Example of a dataset's element labeled as signal, with a waveform injected into noise from O3a, with a time window of 1 s (a) and 0.04 s (b).

and 5, corresponding to one element of the signal and background samples, respectively. Table I characterizes the datasets used for classification and parameter estimation with time series. For the generation of the datasets, functions from the PYTHON [78] libraries SciPy [79], PyCBC [80], and GWpy [75] were used. We also used the libraries FAST.AI [81] and TSAI [82] for the classification and parameter estimation tests, described in the next sections.

B. Deep neural network:
Architecture and methodology

Time-domain GW signals of CCSN events are tested, both for classification and parameter estimation, using a residual convolutional neural network (ResCNN). This network is an implementation of the CNN with residual learning blocks, proposed in [65]. It has six convolutional layers and uses batch normalization techniques and different activation functions. The most used activation function is the rectified linear unit (ReLU). Other versions of this activation function are also used in the architecture, i.e., leaky ReLU (LReLU), parametric ReLU (PReLU), and
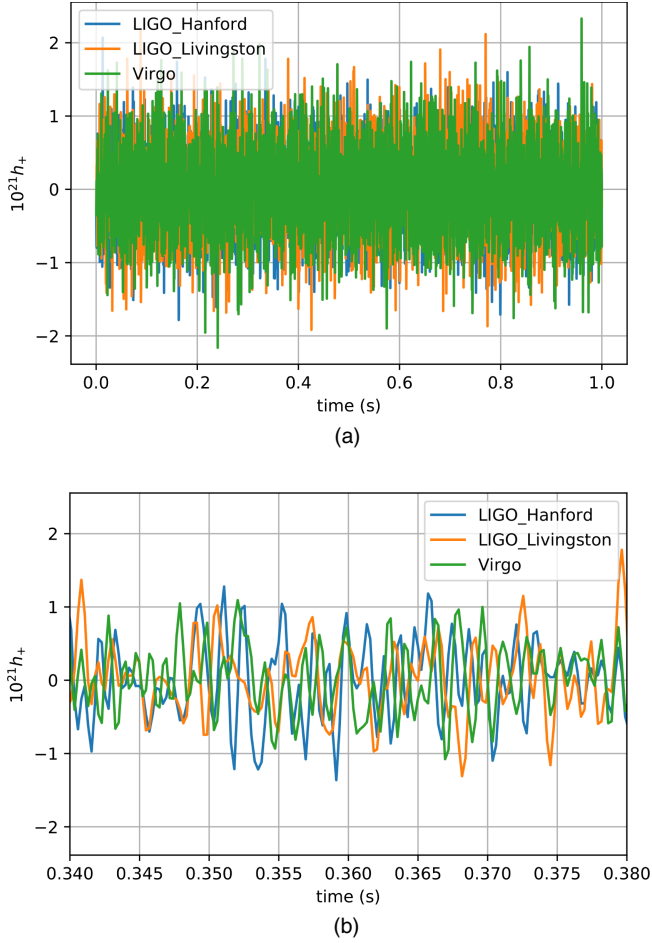
FIG. 5. Example of a dataset's element labeled as background, with only noise from O3a, with a time window of 1 s (a) and 0.04 s (b).
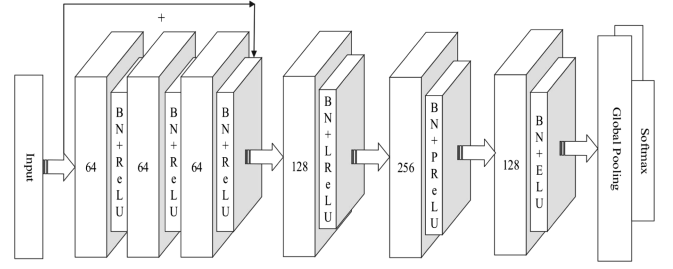

FIG. 6. Representation of the ResCNN's architecture as presented in [65].

architecture, the usual fully connected layer at the end of the network was replaced by a global average pooling and a final softmax layer, so that the output of the network is a probability distribution of its own predictions. The architecture of this network is represented in Fig. 6.

We use the function `fit_one_cycle` to perform the training of the network, defining the maximum learning rate (LR) and the weight decay. This function of FASTAI internally uses the usual `fit` function but applies the 1cycle policy for better results [83]. The maximum LR and weight decay used should be adapted to each dataset and architecture [84]. In our case, to find the best values for our datasets we used another FASTAI function called `lr_find` based on the LR range test [85]. We verified that for our datasets the best results obtained were for weight decays and maximum LR of the order of 10^{-3}.

For our parameter estimation computations, before giving any dataset to the network, we perform a standard normalization of the signals' parameters. To find the predictions of the network in both classification and parameter estimation, we use Monte Carlo (MC) dropout [86], passing each signal through the network 100 times. The prediction of the network is the mean value of all the 100 evaluations. Finally, we apply PYTHON's PCA function from SCIKIT-LEARN [87] on the two-dimensional distributions of predicted versus true values, obtained on the parameter estimation tests, according to what follows. Although PCA is typically used for dimensionality reduction by identifying the most relevant features in datasets, we do not use this capability of PCA here. Instead, we use the PYTHON function solely to perform singular value decomposition (SVD) on the calibration plots, comparing predicted values to true values of $D \cdot \Delta h$ and f_{peak}. We extract the two orthogonal eigenvectors to calculate the slope, and with the singular values we obtain the standard deviation of the distributions, σ.

C. Classification results with time series

A classification test aims at obtaining a model that best allows to distinguish elements of a dataset where a CCSN GW signal has been injected from those of only background noise. Our best results are achieved

exponential linear unit (ELU). It is desirable to use different activation functions in different layers to achieve better performance [65]. As the analysis of the time series needs the precise location of the signal features, the pooling operation was removed. Another reason for this choice is the fact that pooling reduces the number of inputs given to the next layer, limiting the information available. For this

TABLE I. Description of the datasets used to perform classification using time series and parameter estimation.

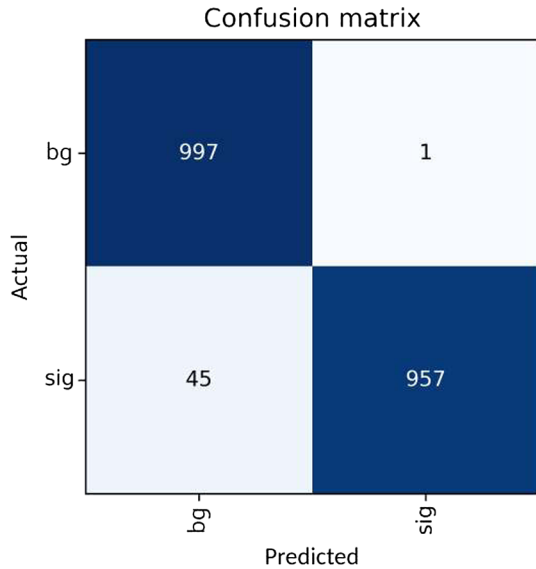| Parameters	Classification	Parameter estimation
SNR > 5	Training set: 80%	Training set: 70%
progenitor's mass $= 12 M_\odot$	Validation set: 20%	Validation set: 30%
distance $\in [5, 20]$ kpc		
inclination $= \frac{\pi}{2}$		
declination $\in [-\pi, \pi]$		
right ascension $\in [0, 2\pi]$		
polarization $\in [0, 2\pi]$		

FIG. 7. Confusion matrix for the classification test with 12 epochs giving the number of strains predicted as background or signal versus their assigned labels.

for a 12-epoch training and considering a 0.50 threshold, which led to a 0.09 minimum validation loss. This model has an accuracy of 98%, providing the confusion matrix shown in Fig. 7. Only 45 actual signals were predicted as background, which corresponds to 4.5% of all GW signals present in the validation set. In addition, only one element labeled as background was misclassified as a signal, giving a FPR of 0.10%. Figure 8 displays the receiver operating characteristic (ROC) curve of this classification model, with a 0.99 area under the curve.

To better understand the results of the misclassified 45 signals (false negatives), Fig. 9 shows the distribution of the model's scores for each of those signals. In this plot we show our two key CCSN parameters, $D \cdot \Delta h$ (top panel)
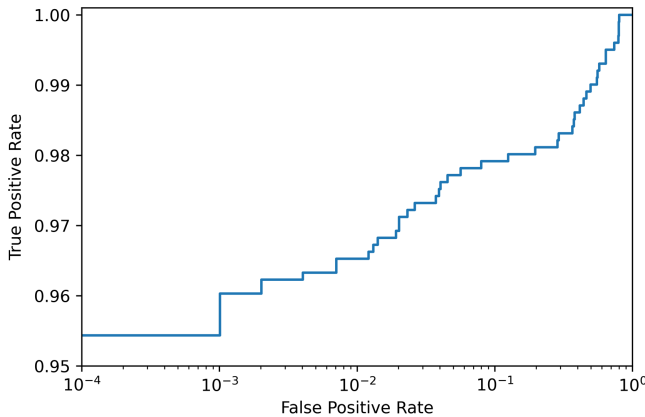

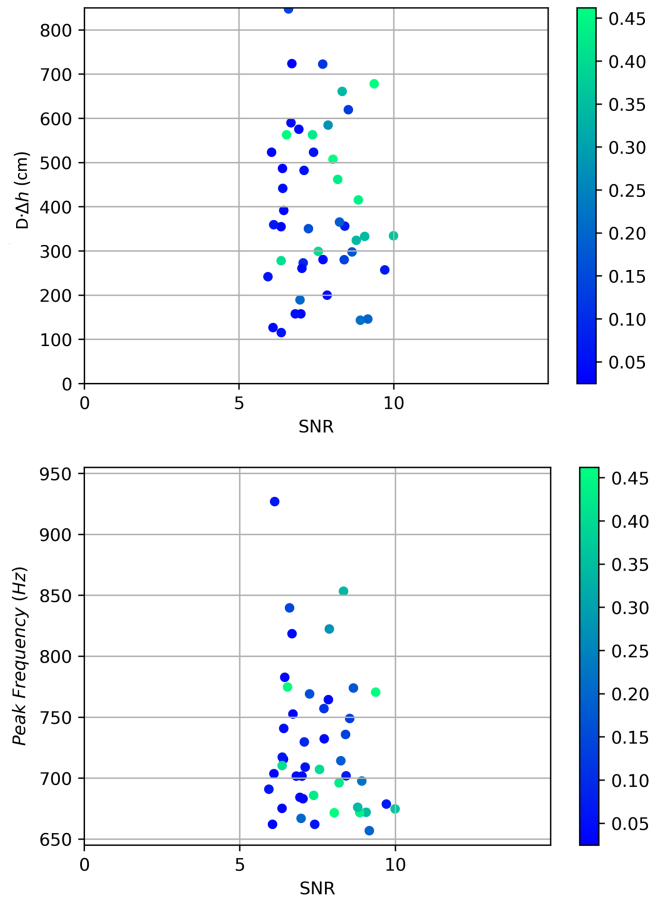FIG. 8. ROC curve of the classification test with 12 epochs.

FIG. 9. Values of $D \cdot \Delta h$ (top panel) and f_{peak} (bottom panel) as a function of the SNR for the 45 misclassified signals on the validation set. The color bar represents the score given by the model to each element.

and f_{peak} (bottom panel), as a function of the SNR. The different colors of the points represent the prediction scores given by the model. All misclassified signals have SNR ≤ 10, which is expected since in those cases it is difficult to distinguish the signal from the random peaks of the noise.

For completeness we have built a spectrograms dataset corresponding to the O3a data used here, to test the network applied in Sec. III B to spectrograms. As expected, when changing the background noise of the spectrograms to O3a, the accuracy of the network goes up to 95%, reflecting the better noise conditions of O3a when compared to O2.

D. Parameter inference results with time series

A parameter inference test was performed to predict the values of the GW strain amplitude (Δh) multiplied by the luminosity distance (D), $D \cdot \Delta h$ (the difference between the highest and lowest points in the bounce signal normalized to the distance) and the peak frequency, f_{peak} (the
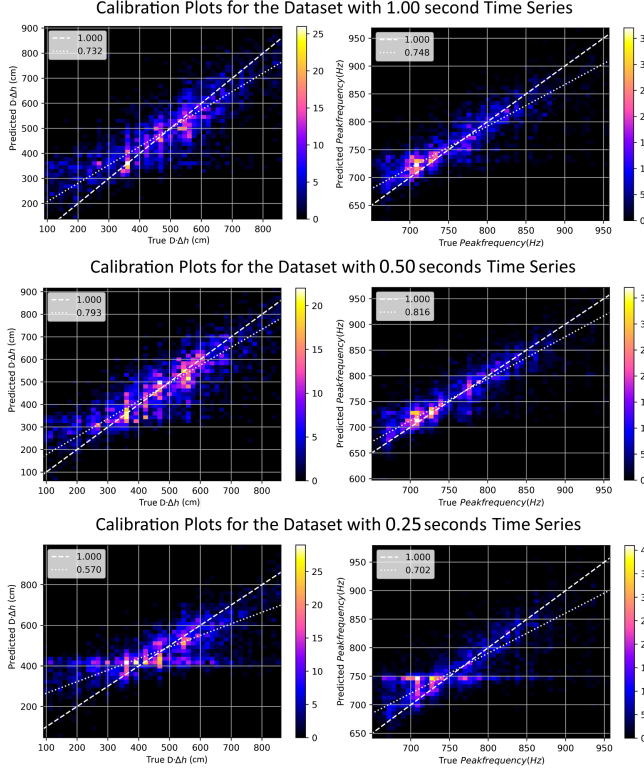
FIG. 10.   Calibration plots of predicted vs true $D \cdot \Delta h$ (left) and predicted vs true f_{peak} (right) with SNR ≥ 5 for the different datasets (corresponding to different time windows). The white dashed line represents the ideal case, where the prediction value is equal to the true value (slope is 1), and the white dotted line represents the slope of the distribution calculated with the PCA function. The specific values of the slopes are indicated in the legends.

FIG. 11.   Calibration plots of predicted vs true $D \cdot \Delta h$ (left) and predicted vs true f_{peak} (right) with SNR ≥ 15 for the different datasets. As in Fig. 10 the white dashed line represents the ideal case (slope 1), and the white dotted line represents the slope of the distribution calculated with the PCA function.

highest frequency measured in the first 6 ms after bounce). As discussed above (see also [29]) these two parameters are the most relevant ones to characterize the waveform at bounce when $T/|W| < 0.06$, as $D \cdot \Delta h \propto T/|W|$ and $f_{\mathrm{peak}} \propto \sqrt{\rho_c}$ for a large group of equations of state [20].

Three datasets of 10^4 elements each were analyzed for different choices of time windows, i.e. 1.00, 0.50, and 0.25 s, as mentioned previously. For all the parameter estimation tests, we trained the network for 30 epochs setting the LR to 2×10^{-3} and the weight decay to 10^{-3}. As expected, there is a relation between the size of the time window used and the training time needed for each epoch. For the 1.00 s dataset, each epoch took around 90 s to complete. By decreasing the window to 0.50 s, the time needed for each epoch also decreased to around 50 s. And lastly, for the 0.25 s dataset, each epoch took around 27 s to complete. As we decrease the time window of the dataset, the training time also decreases, almost in a linear way.

Figure 10 shows the calibration plots for the three datasets, considering SNR ≥ 5. These plots represent

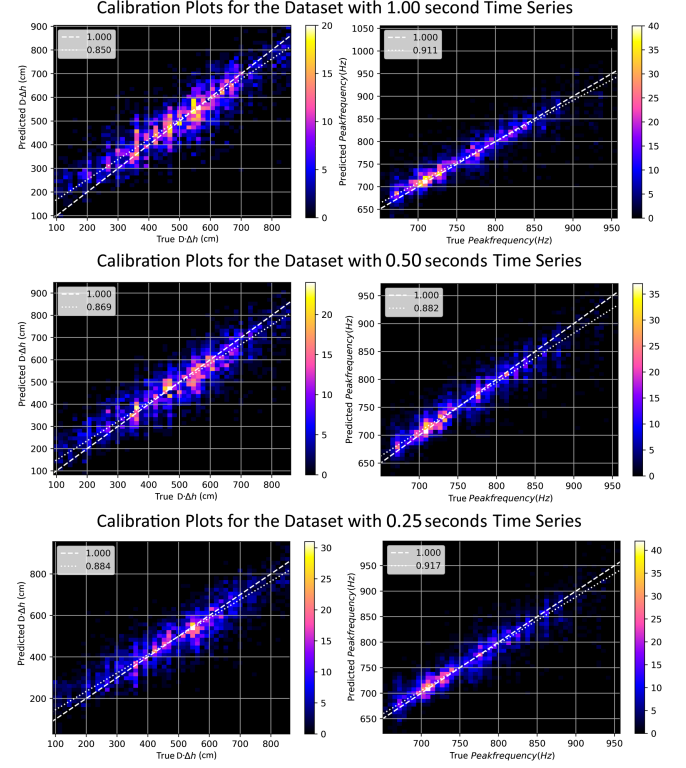predicted values versus true values for $D \cdot \Delta h$ (left) and f_{peak} (right). Two white lines are present in each plot. The dotted lines represent the slope of each distribution, which should be close to the ideal slope, 1.00, represented by the white dashed line. The best results are obtained for the 0.50 s dataset (middle panel in the figure). Comparing the calibration plots for the two parameters, our models yield more precise predictions for f_{peak} than for $D \cdot \Delta h$. Moreover, low values of $D \cdot \Delta h$ and f_{peak} generally tend to be overestimated.

It is worth noticing that horizontal scattered lines are visible in Fig. 10 for all the distributions of the calibration plots. In order to understand this feature we generated new datasets with a minimum SNR of 15, consistent with other studies which place constraints on the minimum SNR estimated to detect GW signals from CCSN events [14]. The new results for the parameter estimation of $D \cdot \Delta h$ and f_{peak} on the three different datasets are shown in Fig. 11. With this choice of minimum SNR the horizontal scattered lines disappear. We also see an improvement in the values of the slopes, with the best results corresponding to the 0.25 s dataset. In this case, for the $D \cdot \Delta h$ distribution, the slope is 0.884 ± 0.008, and the standard deviation is $\sigma_{D \cdot \Delta h} = (52.6 \pm 10.7)$ cm. For the f_{peak} distribution, the
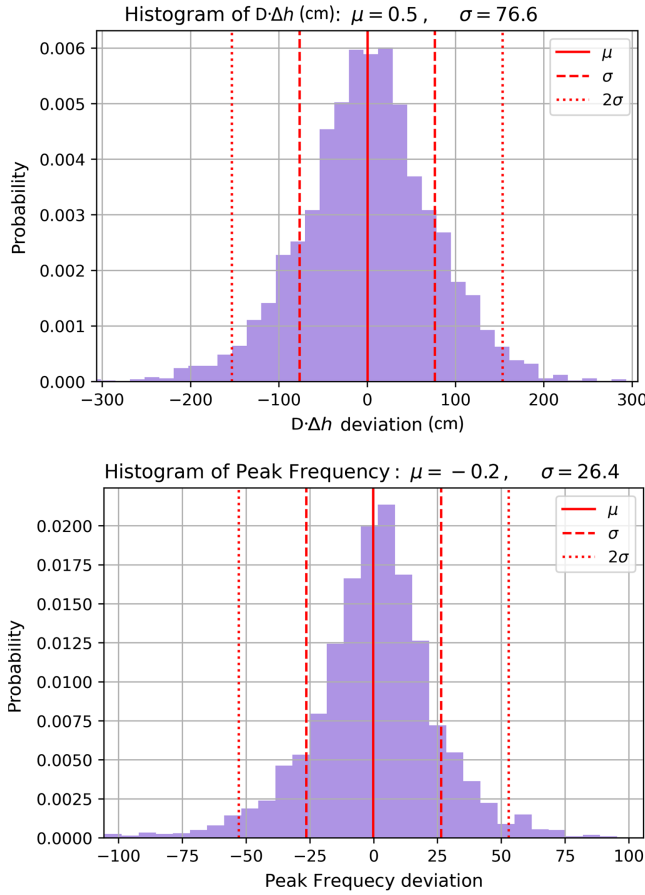
FIG. 12. Probability distribution of the deviations of $D \cdot \Delta h$ (top panel) and f_{peak} (bottom panel), i.e., the difference between the labeled value and the predicted value, for the 0.25 s dataset and a minimum SNR of 15.

results yield a slope of 0.917 ± 0.006 and a standard deviation of $\sigma_{f_{\text{peak}}} = (18.3 \pm 3.9)$ Hz.

The probability distributions of the difference between true and predicted values of $D \cdot \Delta h$ and f_{peak} are shown in the two plots of Fig. 12 for the 0.25 s dataset with minimum SNR of 15. For $D \cdot \Delta h$, a mean value of $\mu_{[D \cdot \Delta h_{\text{true}} - D \cdot \Delta h_{\text{pred}}]} = 0.5 \pm 76.6$ cm is obtained. As for the f_{peak} case, the mean value obtained is $\mu_{[f_{\text{peak,true}} - f_{\text{peak,pred}}]} = -0.2 \pm 26.4$ (Hz). We can clearly see that the inference of f_{peak} is much more accurate than $D \cdot \Delta h$. Taking into account that in our datasets $D \cdot \Delta h \in [100, 850]$ cm and $f_{\text{peak}} \in [650, 950]$ Hz, we can consider the mean values and standard deviations obtained quite reasonable.

E. Testing the network with a different catalog

We now discuss the performance of our network with a different set of CCSN signals. In the absence of an actual discovery of a CCSN GW signal to test our best model, we resort to employing waveforms from a different catalog within the same parameter space of the datasets used for training. The new waveforms were selected from the catalog created recently by Mitra *et al.* [32,88]. This catalog contains 404 waveforms of progenitors with masses ranging from 12 to $40 M_{\odot}$, albeit only for a single EOS, SFHo [89]. While training our network with the Richers *et al.* catalog [20] we only considered $12 M_{\odot}$ progenitors. Therefore, we do the same for this additional test. The CCSN simulations of [32,88] also employ the general relativistic hydrodynamics code CoCoNuT [73], as in Richers' simulations. While the simulations extend up to 25 ms after bounce, we only consider the part of the signals in the first 10 s after bounce in order to keep consistency between training and testing.

We selected 30 waveforms of this new catalog and, for each one of them, generated a dataset with 100 elements as we did before. We fixed the time window to 0.25 s and the minimum SNR to 15, which corresponds to the conditions that led to our best model. Moreover, we allowed the model to vary its response by using MC dropout. By performing signal injections in different noise conditions as well as changes in the model response, we obtain a more realistic implementation of a true GW detection. For a given dataset, prediction values were determined for each element and, with the collection of the 100 predicted values, the mean value μ and the standard deviation σ were calculated for both parameters, $D \cdot \Delta h$ and f_{peak}.

The results of the parameter estimation test with this new catalog are reported in Table II. The name of the waveforms in this table encodes the characteristics of the signal: s12 signifies that the mass of the progenitor star is $12 M_{\odot}$; A1, A3, A4, and A5 correspond to the degree of differential rotation of the precollapse star [see Eq. (1)] of 300, 634, 1268, and 10000 km, respectively, and the remainder of the string name gives the central angular velocity in rad s^{-1}. Most of our predictions are within 1σ from the true values, confirming that our model would be able to infer physical parameters of a Galactic CCSN GW signal in more realistic conditions. For $D \cdot \Delta h$ only four waveforms are outside of this interval but all are covered by the 2σ band. In the case of the f_{peak} predictions, only seven waveforms have true values outside of the 1σ interval, but all are within 2σ.

V. CONCLUSIONS

The first detection of a GW signal from a CCSN is highly anticipated as it can provide important information about the physical processes occurring during the gravitational collapse of massive stars. In this paper we have discussed a method to extract information through the analysis of simulated CCSN waveforms. In particular, we have developed DL techniques to perform classification and parameter inference of rapidly rotating CCSN events using the information encoded in the early (nonstochastic) part of their gravitational waveforms. To do so we have trained

TABLE II.　Parameter estimation results for the 30 selected waveforms from the Mitra *et al.* catalog [32,88]. The models are grouped by the degree of differential rotation A, corresponding from top to bottom to 300 km (A1), 634 km (A3), 1268 km (A4), and 10000 km (A5). For each waveform we report the *prediction*, for both $D \cdot \Delta h$ and f_{peak}, in the form $\mu \pm \sigma$ and the true value in brackets. When the true values are within 1σ of the predictions they are written in boldface.

Model A1

Waveform	$D \cdot \Delta h$ (cm)	f_{peak} (Hz)
s12A1005.5	422 ± 50 **[374]**	755 ± 12 [770]
s12A1006	446 ± 54 **[449]**	752 ± 12 **[764]**
s12A1006.5	501 ± 59 **[533]**	762 ± 12 [783]
s12A1007	591 ± 76 **[611]**	790 ± 25 [826]
s12A1007.5	676 ± 57 **[678]**	827 ± 26 **[843]**
s12A1008	730 ± 49 **[747]**	848 ± 16 [863]
s12A1008.5	770 ± 45 **[799]**	883 ± 17 [986]
s12A1009	757 ± 45 [832]	892 ± 30 [923]

Model A3

Waveform	$D \cdot \Delta h$ (cm)	f_{peak} (Hz)
s12A3003.5	449 ± 58 [361]	765 ± 11 [772]
s12A3004	493 ± 63 **[464]**	764 ± 13 [773]
s12A3004.5	543 ± 61 **[564]**	763 ± 18 [780]
s12A3005	647 ± 56 **[643]**	787 ± 19 [798]
s12A3005.5	648 ± 39 **[666]**	808 ± 17 [816]
s12A3006	637 ± 36 **[629]**	831 ± 15 [841]
s12A3006.5	596 ± 31 **[583]**	846 ± 13 [858]
s12A3007	561 ± 29 **[560]**	851 ± 14 [862]
s12A3007.5	512 ± 36 **[520]**	831 ± 19 [859]

Model A4

Waveform	$D \cdot \Delta h$ (cm)	f_{peak} (Hz)
s12A4003	445 ± 59 [354]	761 ± 12 [768]
s12A4003.5	484 ± 58 **[462]**	759 ± 12 [765]
s12A4004	538 ± 61 **[551]**	759 ± 15 [770]
s12A4004.5	597 ± 52 **[589]**	775 ± 14 [786]
s12A4005	570 ± 33 **[559]**	790 ± 16 [802]
s12A4005.5	520 ± 35 **[506]**	811 ± 14 [816]
s12A4006	477 ± 28 **[468]**	808 ± 12 [820]
s12A4006.5	442 ± 22 **[440]**	798 ± 15 [817]

Model A5

Waveform	$D \cdot \Delta h$ (cm)	f_{peak} (Hz)
s12A5003	463 ± 62 [387]	759 ± 12 [764]
s12A5003.5	506 ± 68 [494]	760 ± 16 [764]
s12A5004	550 ± 61 **[552]**	766 ± 14 [775]
s12A5004.5	560 ± 42 **[544]**	780 ± 14 [797]
s12A5005	532 ± 38 **[506]**	801 ± 15 [813]

NNs on a catalog of numerically generated signals [20] injected into real LIGO-Virgo background noise from O2 and O3a, both in the form of spectrograms (time-frequency diagrams) and time series. The results presented in this work provide further support to the usefulness of DL methods for GW data analysis. In the specific context of CCSN, the methods discussed here seem worthy of being implemented in CCSN detection (classification) and parameter estimation pipelines.

Following previous work on CBC GW signals [49], we have first presented a classification test using a ResNet101 on a dataset composed of 10^4 spectrograms. CCSN GW signals were injected into real noise from O2 of the LIGO Hanford, LIGO Livingston, and Virgo detectors making up half of the dataset. All signals have a fixed distance of 20 kpc, a fixed sky position, and a duration of 200 ms. The other half of the dataset contained only background noise spectrograms, which had to be separated by the network's model. The model obtained after training accomplished an accuracy of 82% and a precision of 90%. We have also analyzed the influence of SNR and distance to the source on the network's performance. For SNR > 30 the accuracy of the model becomes 95%. In addition, we attain more than 50% of success on the classification only for distances below 20 kpc. While there is room for improvement, we see that using spectrograms in DL shows great promise, with results aligning with the values achieved in [49] for CBCs in O2 noise. The percentage of false positives (and false negatives) is increased due to the numerous glitches in O2 and the lower sensitivity of the detectors at that time. This issue can be minimized using background noise from more recent observing runs. As expected, when changing the background noise of the spectrograms to O3a, the accuracy of the network goes up to 95%, reflecting the better noise conditions of O3a when compared to O2.

For the second and main part of our study, we have considered time series. This classification test was performed with a ResCNN on a dataset composed of 10^4 time series where CCSN GW signals had been injected into 1-second-long sections of real noise from O3a of the two LIGO detectors and the Virgo detector. For this method, we have variable distance, from 5 to 20 kpc, and random sky position. All the tests performed with time series have been done with residual CNNs applied to datasets composed of numerically generated signals from different waveforms. The classification test was accomplished with a 10^4 dataset consisting of equal parts of elements with only background noise from O3a and elements of injected signals. The network has provided a model with an accuracy of 98% with only one false positive. These results confirm the efficiency of these networks regarding classification and show that, even with a small dataset, high values of accuracy can be obtained. By further inspecting the false negatives we noticed a correlation with the SNR of the signals, as all of the misclassifications occurred for

elements with SNR < 10. These results show the excellent performance of deep learning techniques for detection, with the time series method reflecting slightly better results than the values achieved with spectrograms in O3a.

To perform parameter estimation we have used three datasets composed only of injected signals from the catalog of [20]. These datasets differ in their time duration: 1.00, 0.50, and 0.25 s. Considering a minimum SNR value of 15, the best results are obtained with the 0.25 s dataset. By performing SVD, we have found that the slope of the distribution of the $D \cdot \Delta h$ calibration plot is 0.884 ± 0.008 and the standard deviation $\sigma_{D \cdot \Delta h, 15} = (52.6 \pm 10.7)$ cm. For the peak frequency, the value of the slope obtained is 0.917 ± 0.006 with $\sigma_{f_{\text{peak}}, 15} = (18.3 \pm 3.9)$ Hz standard deviation. As a final test, we have assessed our best model with 30 additional waveforms from a new catalog of rapidly rotating CCSN models, computed recently by [32]. For each waveform, we have created a dataset composed of 100 elements with 0.25 s duration and SNR ≥ 15. The results obtained have shown that most of the true values are within 1σ of the prediction distribution, confirming that our model might be able to infer parameters of fast-rotating Galactic CCSN GW signals.

While the results reported in this work show the potential of deep learning for inferring properties of rotational CCSN events, they can nonetheless be improved. An obvious improvement would be accomplished by enlarging the datasets. The datasets used for the training of our models only have 10^4 elements, which is considered a small size for the kind of classification and regression tests with neural networks. However, enlarging the datasets would require having available more waveforms from CCSN simulations, which are computationally costly. We note that we are currently working on the computation of new fast-rotating models for different progenitor masses and expect to present an updated study of our findings elsewhere. In addition, in this study, we have used background noise from the existing network of GW detectors (namely Advanced LIGO and Advanced Virgo). As current detectors are improved and third-generation detectors become operational, the sensitivity will improve and the background noise will be reduced, which will lead to higher SNR values and easier recognition of CCSN GW signals.

ACKNOWLEDGMENTS

[1] E. Abdikamalov, G. Pagliaroli, and D. Radice, *Gravitational Waves from Core-Collapse Supernovae* (Springer Nature Singapore, Singapore, 2022), ISBN 978-981-16-4306-4, 10.1007/978-981-16-4306-4_21.

[2] H.-T. Janka, in *Handbook of Supernovae*, edited by A. W. Alsabti and P. Murdin (Springer Cham, Garching, 2017), p. 1095.

[3] A. Heger, S. E. Woosley, and H. C. Spruit, Astrophys. J. **626,** 350 (2005).

[4] M. Obergaulinger and M. Á. Aloy, Mon. Not. R. Astron. Soc. **492,** 4613 (2020).

[5] B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese (LIGO Scientific Collaboration and Virgo Collaboration), Phys. Rev. D **101,** 084002 (2020).

[6] S. E. Gossan, P. Sutton, A. Stuver, M. Zanolin, K. Gill, and C. D. Ott, Phys. Rev. D **93,** 042002 (2016).

[7] R. Raynaud, P. Cerdá-Durán, and J. Guilet, Mon. Not. R. Astron. Soc. **509,** 3410 (2022).

[8] B. P. Abbott *et al.*, Phys. Rev. X **9,** 031040 (2019).

[9] R. Abbott *et al.*, Phys. Rev. X **11,** 021053 (2021).

[10] R. Abbott *et al.*, Phys. Rev. D **109,** 022001 (2024).

[11] The LIGO Scientific Collaboration, the Virgo Collaboration, and the KAGRA Collaboration, Phys. Rev. X **13,** 041039 (2023).

[12] M. J. Szczepańczyk, Y. Zheng, J. M. Antelis, M. Benjamin, M.-A. Bizouard, A. Casallas-Lagos, P. Cerdá-Durán, D. Davis, D. Gondek-Rosińska, S. Klimenko *et al.*, arXiv:2305 .16146.

[13] B. P. Abbott, R. Abbott *et al.* (The LIGO Scientific Collaboration, Kagra Collaboration, and VIRGO Collaboration), Living Rev. Relativity **21,** 3 (2018).

[14] M. J. Szczepańczyk, J. M. Antelis, M. Benjamin, M. Cavaglià, D. Gondek-Rosińska, T. Hansen, S. Klimenko, M. D. Morales, C. Moreno, S. Mukherjee *et al.*, Phys. Rev. D **104,** 102002 (2021).

[15] A. Mezzacappa, P. Marronetti, R. E. Landfield, E. J. Lentz, R. D. Murphy, W. R. Hix, J. A. Harris, S. W. Bruenn, J. M. Blondin, O. E. Bronson Messer *et al.*, Phys. Rev. D **107,** 043008 (2023).

[16] A. Casallas Lagos, J. M. Antelis, C. Moreno, M. Zanolin, A. Mezzacappa, and M. J. Szczepańczyk, Phys. Rev. D **108,** 084027 (2023).

[17] T. Bruel, M.-A. Bizouard, M. Obergaulinger, P. Maturana-Russel, A. Torres-Forné, P. Cerdá-Durán, N. Christensen, J. A. Font, and R. Meyer, Phys. Rev. D **107,** 083029 (2023).

[18] E. Abdikamalov, S. Gossan, A. M. DeMaio, and C. D. Ott, Phys. Rev. D **90,** 044001 (2014).

[19] J. Powell, S. E. Gossan, J. Logue, and I. S. Heng, Phys. Rev. D **94,** 123012 (2016).

[20] S. Richers, C. D. Ott, E. Abdikamalov, E. O'Connor, and C. Sullivan, Phys. Rev. D **95,** 063019 (2017).

[21] V. Morozova, D. Radice, A. Burrows, and D. Vartanyan, Astrophys. J. **861,** 10 (2018).

[22] A. Torres-Forné, P. Cerdá-Durán, M. Obergaulinger, B. Müller, and J. A. Font, Phys. Rev. Lett. **123,** 051102 (2019).

[23] M.-A. Bizouard, P. Maturana-Russel, A. Torres-Forné, M. Obergaulinger, P. Cerdá-Durán, N. Christensen, J. A. Font, and R. Meyer, Phys. Rev. D **103,** 063006 (2021).

[24] C. Afle and D. A. Brown, Phys. Rev. D **103,** 023005 (2021).

[25] M. A. Pajkos, M. L. Warren, S. M. Couch, E. P. O'Connor, and K.-C. Pan, Astrophys. J. **914,** 80 (2021).

[26] H. Sotani, T. Takiwaki, and H. Togashi, Phys. Rev. D **104,** 123009 (2021).

[27] O. E. Andersen, S. Zha, A. da Silva Schneider, A. Betranhandy, S. M. Couch, and E. P. O'Connor, Astrophys. J. **923,** 201 (2021).

[28] A. Saiz-Pérez, A. Torres-Forné, and J. A. Font, Mon. Not. R. Astron. Soc. **512,** 3815 (2022).

[29] C. Pastor-Marcos, P. Cerdá-Durán, D. Walker, A. Torres-Forné, E. Abdikamalov, S. Richers, and J. A. Font, Phys. Rev. D **109,** 063028 (2024).

[30] J. Powell, A. Iess, M. Llorens-Monteagudo, M. Obergaulinger, B. Müller, A. Torres-Forné, E. Cuoco, and J. A. Font, Phys. Rev. D **109,** 063019 (2024).

[31] N. E. Wolfe, C. Fröhlich, J. M. Miller, A. Torres-Forné, and P. Cerdá-Durán, Astrophys. J. **954,** 161 (2023).

[32] A. Mitra, B. Shukirgaliyev, Y. S. Abylkairov, and E. Abdikamalov, Mon. Not. R. Astron. Soc. **520,** 2473 (2023).

[33] B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham *et al.* (LIGO Scientific Collaboration and Virgo Collaboration), Classical Quantum Gravity **37,** 055002 (2020).

[34] M. Drago, S. Klimenko, C. Lazzaro *et al.*, SoftwareX **14,** 100678 (2021).

[35] C. Röver, M.-A. Bizouard, N. Christensen, H. Dimmelmeier, I. S. Heng, and R. Meyer, Phys. Rev. D **80,** 102004 (2009).

[36] J. Logue, C. D. Ott, I. S. Heng, P. Kalmus, and J. H. C. Scargill, Phys. Rev. D **86**, 044023 (2012).

[37] J. Powell, M. Szczepanczyk, and I. S. Heng, Phys. Rev. D **96**, 123013 (2017).

[38] S. R. Green, C. Simpson, and J. Gair, Phys. Rev. D **102**, 104057 (2020).

[39] M. Dax, S. R. Green, J. Gair, J. H. Macke, A. Buonanno, and B. Schölkopf, Phys. Rev. Lett. **127**, 241103 (2021).

[40] M. J. Williams, J. Veitch, and C. Messenger, Phys. Rev. D **103**, 103006 (2021).

[41] J. Bayley, C. Messenger, and G. Woan, Phys. Rev. D **106**, 083022 (2022).

[42] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, Nat. Phys. **18**, 112 (2022).

[43] M. Dax, S. R. Green, J. Gair, M. Pürrer, J. Wildberger, J. H. Macke, A. Buonanno, and B. Schölkopf, Phys. Rev. Lett. **130**, 171403 (2023).

[44] U. Bhardwaj, J. Alvey, B. K. Miller, S. Nissanke, and C. Weniger, Phys. Rev. D **108**, 042004 (2023).

[45] E. A. Huerta, G. Allen, I. Andreoni, J. M. Antelis, E. Bachelet, G. B. Berriman, F. B. Bianco, R. Biswas, M. Carrasco Kind, K. Chard *et al.*, Nat. Rev. Phys. **1**, 600 (2019).

[46] E. Cuoco, J. Powell, M. Cavaglià, K. Ackley, M. Bejger, C. Chatterjee, M. Coughlin, S. Coughlin, P. Easter, R. Essick *et al.*, Mach. Learn. Sci. Tech. **2**, 011002 (2021).

[47] T. Zhao, R. Shi, Y. Zhou, Z. Cao, and Z. Ren, arXiv:2311.15585.

[48] N. Stergioulas, arXiv:2401.07406.

[49] J. D. Álvares, J. A. Font, F. F Freitas, O. G. Freitas, A. P. Morais, S. Nunes, A. Onofre, and A. Torres-Forné, Classical Quantum Gravity **38**, 155010 (2021).

[50] V. Boudart and M. Fays, in *Proceedings of the 2022 IEEE International Conference on Big Data (Big Data)* (IEEE, Osaka, 2022), pp. 6599–6601, 10.1109/BigData55660.2022.10020896.

[51] M. B. Schäfer, O. c. v. Zelenka, A. H. Nitz, H. Wang, S. Wu, Z.-K. Guo, Z. Cao, Z. Ren, P. Nousi, N. Stergioulas *et al.*, Phys. Rev. D **107**, 023021 (2023).

[52] A. Torres-Forné, A. Marquina, J. A. Font, and J. M. Ibáñez, Phys. Rev. D **94**, 124040 (2016).

[53] A. Torres-Forné, E. Cuoco, J. A. Font, and A. Marquina, Phys. Rev. D **102**, 023011 (2020).

[54] C.-H. Liao and F.-L. Lin, Phys. Rev. D **103**, 124051 (2021).

[55] S. Schmidt, M. Breschi, R. Gamba, G. Pagano, P. Rettegno, G. Riemenschneider, S. Bernuzzi, A. Nagar, and W. Del Pozzo, Phys. Rev. D **103**, 043020 (2021).

[56] J. Tissino, G. Carullo, M. Breschi, R. Gamba, S. Schmidt, and S. Bernuzzi, Phys. Rev. D **107**, 084037 (2023).

[57] M. Lopez, V. Boudart, K. Buijsman, A. Reza, and S. Caudill, Phys. Rev. D **106**, 023027 (2022).

[58] M. Lopez, V. Boudart, S. Schmidt, and S. Caudill, arXiv:2205.09204.

[59] P. Astone, P. Cerdá-Durán, I. Di Palma, M. Drago, F. Muciaccia, C. Palomba, and F. Ricci, Phys. Rev. D **98**, 122002 (2018).

[60] M. L. Chan, I. S. Heng, and C. Messenger, Phys. Rev. D **102**, 043022 (2020).

[61] M. López, I. Di Palma, M. Drago, P. Cerdá-Durán, and F. Ricci, Phys. Rev. D **103**, 063011 (2021).

[62] A. Iess, E. Cuoco, F. Morawski, C. Nicolaou, and O. Lahav, Astron. Astrophys. **669**, A42 (2023).

[63] S. Sasaoka, N. Koyama, D. Dominguez, Y. Sakai, K. Somiya, Y. Omae, and H. Takahashi, Phys. Rev. D **108**, 123033 (2023).

[64] K. He, X. Zhang, S. Ren, and J. Sun, arXiv:1512.03385.

[65] X. Zou, Z. Wang, Q. Li, and W. Sheng, Neurocomputing Variable Star Bull. **367**, 39 (2019).

[66] T. Zwerger and E. Mueller, Astron. Astrophys. **320**, 209 (1997), https://adsabs.harvard.edu/full/1997A%26A...320.209Z.

[67] H. Dimmelmeier, J. A. Font, and E. Müller, Astron. Astrophys. **393**, 523 (2002).

[68] H. Dimmelmeier, C. D. Ott, A. Marek, and H. T. Janka, Phys. Rev. D **78**, 064056 (2008).

[69] C. D. Ott, E. Abdikamalov, E. O'Connor, C. Reisswig, R. Haas, P. Kalmus, S. Drasco, A. Burrows, and E. Schnetter, Phys. Rev. D **86**, 024026 (2012).

[70] J. Fuller, H. Klion, E. Abdikamalov, and C. D. Ott, Mon. Not. R. Astron. Soc. **450**, 414 (2015).

[71] M. C. Edwards, Phys. Rev. D **103**, 024025 (2021).

[72] E. O'Connor, Astrophys. J. Suppl. Ser. **219**, 24 (2015).

[73] H. Dimmelmeier, J. Novak, J. A. Font, J. M. Ibáñez, and E. Müller, Phys. Rev. D **71**, 064023 (2005).

[74] M. Liebendörfer, Astrophys. J. **633**, 1042 (2005).

[75] GWpy, https://gwpy.github.io, accessed in 14-04-2023.

[76] K. He, X. Zhang, S. Ren, and J. Sun, in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, Las Vegas, 2016), pp. 770–778, 10.1109/CVPR.2016.90.

[77] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, in *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Miami, 2009), pp. 248–255, 10.1109/CVPR.2009.5206848.

[78] PYTHON, https://www.python.org, accessed in 14-04-2023.

[79] SciPy, https://scipy.org, accessed in 14-04-2023.

[80] PyCBC, https://pycbc.org, accessed in 14-04-2023.

[81] FAST.AI, https://www.fast.ai, accessed in 14-04-2023.

[82] TSAI: State-of-the-art deep learning library for time series and sequences, https://timeseriesai.github.io/tsai/, accessed in 14-04-2023.

[83] L. N. Smith and N. Topin, arXiv:1708.07120.

[84] L. N. Smith, arXiv:1803.09820.

[85] L. N. Smith, arXiv:1506.01186.

[86] Y. Gal and Z. Ghahramani, arXiv:1506.02142.

[87] SCIKIT-LEARN: Machine learning in Python, https://scikit-learn.org/stable/, accessed in 14-04-2023.

[88] Exploring supernova gravitational waves with machine learning, https://zenodo.org/record/7090935, accessed in 14-09-2023.

[89] A. W. Steiner, M. Hempel, and T. Fischer, Astrophys. J. **774**, 17 (2013).