

Important Notice to Authors

No further publication processing will occur until we receive your response to this proof.

Attached is a PDF proof of your forthcoming article in *Physical Review D*. The article accession code is DY12328.

Please note that as part of the production process, APS converts all articles, regardless of their original source, into standardized XML that in turn is used to create the PDF and online versions of the article as well as to populate third-party systems such as Portico, Crossref, and Web of Science. We share our authors' high expectations for the fidelity of the conversion into XML and for the accuracy and appearance of the final, formatted PDF. This process works exceptionally well for the vast majority of articles; however, please check carefully all key elements of your PDF proof, particularly any equations or tables.

Figures submitted electronically as separate files containing color appear in color in the online journal.

However, all figures will appear as grayscale images in the print journal unless the color figure charges have been paid in advance, in accordance with our policy for color in print (<https://journals.aps.org/authors/color-figures-print>).

Specific Questions and Comments to Address for This Paper

The numbered items below correspond to numbers in the margin of the proof pages pinpointing the source of the question and/or comment. The numbers will be removed from the margins prior to publication.


Q: Please check that all references include complete titles.

- 1** Second proof: Please carefully confirm that all first-proof corrections were addressed and that changes were made accurately.
- 2** Second proof: Please check and approve that the paper is ready to be published in its current form.

Titles in References

The editors now encourage insertion of article titles in references to journal articles and e-prints. This format is optional, but if chosen, authors should provide titles for *all* eligible references. If article titles remain missing from eligible references, the production team will remove the existing titles at final proof stage.

ORCIDiDs

Please follow any ORCID links () after the authors' names and verify that they point to the appropriate record for each author.

Funding Information

Information about an article's funding sources is now submitted to Crossref to help you comply with current or future funding agency mandates. Crossref's Open Funder Registry (<https://www.crossref.org/services/funder-registry/>) is the definitive registry of funding agencies. Please ensure that your acknowledgments include all sources of funding for your article following any requirements of your funding sources. Where possible, please include grant and award ids. Please carefully check the following funder information we have already extracted from your article and ensure its accuracy and completeness:

- Deutsche Forschungsgemeinschaft, FundRef ID <http://dx.doi.org/10.13039/501100001659> (DE/Federal Republic of Germany)
- H2020 European Research Council, FundRef ID <http://dx.doi.org/10.13039/100010663> (EU/European Union)
- National Science Foundation, FundRef ID <http://dx.doi.org/10.13039/1000000001> (US/United States)

- Centre National de la Recherche Scientifique, FundRef ID <http://dx.doi.org/10.13039/501100004794> (FR/Republic of France)
- Istituto Nazionale di Fisica Nucleare, FundRef ID <http://dx.doi.org/10.13039/501100004007> (IT/Repubblica Italiana)
- Dutch Nikhef

Other Items to Check

- Please note that the original manuscript has been converted to XML prior to the creation of the PDF proof, as described above. Please carefully check all key elements of the paper, particularly the equations and tabular data.
- Title: Please check; be mindful that the title may have been changed during the peer-review process.
- Author list: Please make sure all authors are presented, in the appropriate order, and that all names are spelled correctly.
- Please make sure you have inserted a byline footnote containing the email address for the corresponding author, if desired. Please note that this is not inserted automatically by this journal.
- Affiliations: Please check to be sure the institution names are spelled correctly and attributed to the appropriate author(s).
- Receipt date: Please confirm accuracy.
- Acknowledgments: Please be sure to appropriately acknowledge all funding sources.
- References: Please check to ensure that titles are given as appropriate.
- Hyphenation: Please note hyphens may have been inserted in word pairs that function as adjectives when they occur before a noun, as in “x-ray diffraction,” “4-mm-long gas cell,” and “*R*-matrix theory.” However, hyphens are deleted from word pairs when they are not used as adjectives before nouns, as in “emission by x rays,” “was 4 mm in length,” and “the *R* matrix is tested.”
Note also that Physical Review follows U.S. English guidelines in that hyphens are not used after prefixes or before suffixes: superresolution, quasiequilibrium, nanoprecipitates, resonancelike, clockwise.
- Please check that your figures are accurate and sized properly. Make sure all labeling is sufficiently legible. Figure quality in this proof is representative of the quality to be used in the online journal. To achieve manageable file size for online delivery, some compression and downsampling of figures may have occurred. Fine details may have become somewhat fuzzy, especially in color figures. The print journal uses files of higher resolution and therefore details may be sharper in print. Figures to be published in color online will appear in color on these proofs if viewed on a color monitor or printed on a color printer.
- Overall, please proofread the entire *formatted* article very carefully. The redlined PDF should be used as a guide to see changes that were made during copyediting. However, note that some changes to math and/or layout may not be indicated.

Ways to Respond

- **Web:** If you accessed this proof online, follow the instructions on the web page to submit corrections.
- **Email:** Send corrections to aps-robot@luminad.com. Include the accession code DY12328 in the subject line.
- **Fax:** Return this proof with corrections to +1.855.808.3897.

If You Need to Call Us

You may leave a voicemail message at +1.855.808.3897. Please reference the accession code and the first author of your article in your voicemail message. We will respond to you via email.

Machine learning gravitational waves from binary black hole mergers

Stefano Schmidt^{1,*} Matteo Breschi² Rossella Gamba² Giulia Pagano¹ Piero Rettegno^{3,4}
Gunnar Riemenschneider^{3,4} Sebastiano Bernuzzi² Alessandro Nagar^{4,5} and Walter Del Pozzo¹

¹*Dipartimento di Fisica Università di Pisa, and INFN Sezione di Pisa, Pisa I-56127, Italy*

²*Theoretisch-Physikalisches Institut, Friedrich-Schiller-Universität Jena, 07743, Jena, Germany*

³*Dipartimento di Fisica, Università di Torino, via P. Giuria 1, 10125 Torino, Italy*

⁴*INFN Sezione di Torino, Via P. Giuria 1, 10125 Torino, Italy*

⁵*IHES, 91440, Bures-sur-Yvette, France*

(Received 5 November 2020; accepted 19 January 2021)

We apply machine learning methods to build a time-domain model for gravitational waveforms from binary black hole mergers, called MLGW. The dimensionality of the problem is handled by representing the waveform's amplitude and phase using a principal component analysis. We train MLGW on about $\mathcal{O}(10^3)$ TEOBResumS and SEOBNRv4 effective-one-body waveforms with mass ratios $q \in [1, 20]$ and aligned dimensionless spins $s \in [-0.80, 0.95]$. The resulting models are faithful to the training sets at the $\sim 10^{-3}$ level (averaged on the parameter space). The speed up for a single waveform generation is a factor of 10–50 (depending on the binary mass and initial frequency) for TEOBResumS and approximately an order of magnitude more for SEOBNRv4. Furthermore, MLGW provides a closed form expression for the waveform and its gradient with respect to the orbital parameters; such an information might be useful for future improvements in gravitational wave (GW) data analysis. As demonstration of the capabilities of MLGW to perform a full parameter estimation, we reanalyze the public data from the first GW transient catalog (GWTC-1). We find broadly consistent results with previous analyses at a fraction of the cost, although the analysis with spin-aligned waveforms gives systematic larger values of the effective spins with respect to previous analyses with precessing waveforms. Since the generation time does not depend on the length of the signal, our model is particularly suitable for the analysis of the long signals that are expected to be detected by third-generation detectors. Future applications include the analysis of waveform systematics and model selection in parameter estimation.

DOI:

I. INTRODUCTION

The detection of gravitational waves (GWs) from compact binary coalescence (CBC) has been possible thanks to the joint effort of a number of different fields of expertise, all joining together to achieve the sophisticated detection process. GW data analysis concerns the detection of a GW signal hidden in the raw detector output (*matched filtering*) and subsequently the inference of its physical properties (*parameter estimation*). In order to accomplish its goal, GW data analysis relies on the availability of waveform (WF) templates to compare with the detector output. To accurately explore the posterior distribution for the parameters defining a CBC, state-of-the-art parameter estimation (PE) algorithms [1,2] can require the generation of as many as 10^7 waveform templates. It is therefore paramount for the waveform generation to be as fast as possible. At the same time, because of the extreme sensitivity to phase differences in the likelihood function, the templates must

retain a high degree of accuracy to avoid biases in the posterior exploration.

Many efforts have been devoted to numerically solve Einstein equations for two coalescing objects and to predict the gravitational radiation emitted [3–5]. As solving the full equations is still extremely computationally challenging, the LIGO-Virgo Collaboration relies on approximate analytical models. These can be broadly categorized in three families: (i) effective-one-body (EOB) [6] waveform models [7–10]; (ii) phenomenological models [11–13]; (iii) numerical relativity (NR) surrogates [14–18].

EOB models are the waveform approximants that incorporate the largest amount of analytical information. They compute the GW signal by solving Hamilton's equations and accurately predict the GW signal from early inspiral phase up to the final ringdown. The underlying relative dynamics is informed by (or calibrated to) NR simulations via a few parameters that take into account in an effective way yet uncalculated high-order corrections to the orbital or spin-orbit sector of the Hamiltonian. Similarly, the merger and ringdown parts of the waveform are also

*stefanoschmidt1995@gmail.com

informed by NR simulations.¹ Because of the numerical integrations involved to solve Hamilton's equations, they tend to be accurate but sometimes slow to generate; see, however, [21] for a more efficient approach to obtain the dynamics.

The phenomenological waveforms are based on the post-Newtonian formalism and then calibrated to EOB waveforms and numerical relativity. They usually tend to be faster than EOB models and for this reason are largely employed in data analysis.

Many efforts have been devoted to the task of speeding up the generation of GW signals from EOB families. For example, one leads to the development of *surrogate models*. Surrogate models are constructed starting from some decomposition in the waveform space followed by efficient interpolation to avoid any numerical integration [22–26]. Being fast to generate, they are routinely employed in GW data analysis.

A machine learning model is a promising alternative to accelerate the waveform generation of state-of-the-art models. Machine learning (ML) is a branch of statistics that is devoted to reproduce patterns read from data. An ML algorithm needs very little human input, and, by automatically solving an optimization problem, it is able to choose the best-performing element among a large class (i.e., the *model*) of parametric solutions. This is the so-called training procedure. The ML flexibility in modeling data and reproducing trends is appealing: With a proper model choice and with an appropriate training procedure, we can hope to have a reliable, fast to execute generator of GW waveforms while retaining the accuracy necessary for robust parameter estimation. ML procedures have already been successfully exploited for speeding up the WF generation [18,27] and for signal detection and/or parameter estimation [28–34]. A comparative study of different regression methods for the task of generating WFs is performed in Ref. [35].

In this work, we explore the application of ML to WF generation and we build an ML model, called MLGW, that allows one to reproduce waveforms from any EOB model for binary black hole (BBH) coalescence. We demonstrate that such a ML-based model can generate GW signals significantly faster than the original model, matching the performances of a reduced order modeling (ROM) [22–24,36]. At the same time, it shows a good agreement with the train model, at the 10^{-3} level² when averaged on the whole parameter space. For simplicity, we focus here only on the dominant $\ell = |m| = 2$ quadrupole waveform and we consider only the *aligned spin* case.

¹One should remember that this step is *a priori* not necessary in the theoretical construction of the model [6,19,20].

²As specified later, the mismatch is always computed with a flat power spectral density.

EOB models are the ideal candidate for training our model: They are moderately fast to run, and, although they are *checked* (and NR informed) only on the limited part of the parameter space covered by NR simulations, they are also typically able to robustly generate waveforms for a large set of parameters outside the NR-information domain (e.g., large mass ratios and spins). This is the case of the most recent avatar of the spin-aligned TEOBResumS [37] model, that incorporates subdominant waveform modes, TEOBResumS_SM [8,38].³ TEOBResumS_SM is NR faithful over the largest set of spin-aligned NR waveforms available today (595 datasets) [8], with maximum EOB or NR unfaithfulness always below 0.5%. Here, to reduce the generation time of the training set and to be more conservative, we slightly downgrade the performance of TEOBResumS considering its version *without* iteration⁴ on the next-to-quasicircular parameters [8]. The other EOB-based model in use is SEOBNRv4 [23,41], largely employed by the LIGO-Virgo Collaboration. This model was calibrated to NR so as to get maximum EOB and NR unfaithfulness at most of 1%. However, differently from TEOB [37], this model in its native implementation is computationally too slow for parameter estimation purposes and it is necessary to use its ROM version, SEOBNRv4_ROM [23].

We use both TEOBResumS_SM (in the following, simply TEOBResumS) and SEOBNRv4 to train different two different versions of MLGW, respectively, MLGW-TEOBResumS and MLGW-SEOBNRv4. As a relevant physical application, we use both MLGW-TEOBResumS and MLGW-SEOBNRv4 to provide two new, and independent, analyses of the ten BBH coalescence events collected in the O1 and O2 LIGO-Virgo observing runs [42]. The outcome of the analyses using the two models are largely consistent among themselves. In addition, the so-obtained physical properties of the ten BBH systems are compatible with previous analyses published in Ref. [42], though obtained using different, notably spin-precessing, waveform models.

On top of the specific application discussed here, our ML waveform model could also be used directly to speed up GW data analysis. Furthermore, since the time required to generate a WF does not depend on the signal time length but only on the number of grid points which the WF is evaluated at, the applicability of our approach goes far beyond the LIGO/Virgo physics scenario. In particular, we think about the forthcoming Einstein Telescope, that will be sensitive to very long stellar-mass inspirals from 5 Hz or to extreme mass ratio inspirals as laser interferometer space antenna sources. In this context, the problem of WF fast

³This model used several hundreds of the available SXS simulations to inform a highly accurate description of the postmerger-ringdown phase [39] but only around 40 datasets to improve the behavior of the analytical EOB dynamics.

⁴This slightly worsens the EOB or NR performance that in any case remains below 1% except for ~ 40 outliers that still do not exceed the 3% threshold and are mostly below the 2% level [40].

generation will be more pressing, and our approach, provided a suitable waveform model for training, might be essential for detection and parameter estimation.

The paper is organized as follows. In Sec. II, we briefly set the notation and the core of the ML problem we solve; in Sec. III, we describe our model in details. Section IV is devoted to validate the model and to assess its accuracy and speed of execution; Sec. V holds our analysis of the GWTC-1 transient catalog, while Sec. VI collects some final remarks and future prospects of our work.

II. CONVENTIONS SETUP

A binary black hole system is parametrized by a vector $\mathbf{\Theta} = (m_1, m_2, \mathbf{s}_1, \mathbf{s}_2)$, where m_i are the BH masses and $\mathbf{s}_i \equiv \mathbf{S}_i/m_i^2 \leq 1$ are the *dimensionless* spin. We call them the *orbital parameters*. We use the convention $m_1 \geq m_2$, and we denote the total mass as $M \equiv m_1 + m_2$ and the mass ratio as $q \equiv m_1/m_2$. In what follows, we will focus on the case in which spins \mathbf{s}_1 and \mathbf{s}_2 are *aligned* with the orbital angular momentum. Let d_L be the luminosity distance and ι and φ_0 the polar angle (inclination) and the azimuthal angle, respectively, of the orbital plane. A GW is parametrized as [see Eq. (II.6) in Ref. [43]]

$$h(t; d_L, \iota, \varphi_0, \mathbf{\Theta}) = h_+ + ih_\times$$

$$= \frac{G}{c^2} \frac{M}{d_L} \sum_{\ell=2}^{\infty} \sum_{m=-\ell}^{\ell} \overset{\leftrightarrow}{[-2]} Y_{\ell m}(\iota, \varphi_0)$$

$$\times H_{\ell m}(t/M; \tilde{\mathbf{\Theta}}), \quad (1)$$

where $\overset{\leftrightarrow}{[-2]} Y_{\ell m}(\iota, \varphi_0)$ are the spin-2 spherical harmonics. Once written as a function of the dimensionless time t/M , the quantities $H_{\ell m}$ depend only on the variables $\tilde{\mathbf{\Theta}} = (q, s_1, s_2)$, and we are considering here only $\ell = |m| = 2$. Since the dependence on the two angles and on the distance and total mass is known, for convenience we fix their value to $\iota = \varphi_0 = 0$, $d_L = 1$ Mpc, and $M = 20 M_\odot$ so to only work with waveforms $h_{\text{FIT}}(t; \mathbf{\Theta}) = h(t; d_L = 1 \text{ Mpc}, \iota = \varphi_0 = 0, M = 20 M_\odot, \tilde{\mathbf{\Theta}})$:

$$h_{\text{FIT}}(t; \mathbf{\Theta}) \equiv 9.6 \times 10^{-19} \overset{\leftrightarrow}{[-2]} Y_{22}(0, 0) H_{22}(t/20 M_\odot; \tilde{\mathbf{\Theta}})$$

$$= 6 \times 10^{-19} H_{22}(t/(20 M_\odot); \tilde{\mathbf{\Theta}}). \quad (2)$$

Finally, we express h_{FIT} in terms of its amplitude and phase⁵:

$$h_{\text{FIT}}(t; \mathbf{\Theta}) = A(t; \tilde{\mathbf{\Theta}}) e^{i\phi(t; \tilde{\mathbf{\Theta}})}. \quad (3)$$

We may also write $f_{\tilde{\mathbf{\Theta}}}(t)$ to denote a function $f(t; \tilde{\mathbf{\Theta}})$ of time with parametric dependence on $\mathbf{\Theta}$. In what follows, f

stands as a placeholder for any of the functions $A_{\tilde{\mathbf{\Theta}}}(t)$ and $\phi_{\tilde{\mathbf{\Theta}}}(t)$. With this definition, the full waveform can be expressed as

$$h(t, d_L, \iota, \varphi_0; \mathbf{\Theta}) = \frac{M}{20 M_\odot} \frac{1 \text{ Mpc}}{d_L}$$

$$\times \left\{ \frac{1 + \cos^2 \iota}{2} A_{\tilde{\mathbf{\Theta}}}(t_M) \cos[\phi_{\tilde{\mathbf{\Theta}}}(t_M) + 2\varphi_0] \right.$$

$$\left. + i \cos \iota A_{\tilde{\mathbf{\Theta}}}(t_M) \sin[\phi_{\tilde{\mathbf{\Theta}}}(t_M) + 2\varphi_0] \right\}, \quad (4)$$

where $t_M = t \frac{M}{20 M_\odot}$. Note that, in the equation above, we split the real and the imaginary parts of h and we used the relation $\overset{\leftrightarrow}{[-2]} Y_{2\pm 2}(\iota, \varphi_0) = \sqrt{\frac{5}{64\pi}} (1 \pm \cos \iota)^2 e^{\pm i 2\varphi_0}$. As a constant translation of $\phi_{\tilde{\mathbf{\Theta}}}$ can be absorbed in the definition of φ_0 and does not affect the physics, we choose the convention that $\phi_{\tilde{\mathbf{\Theta}}} = 0$ when the amplitude $A_{\tilde{\mathbf{\Theta}}}$ has a maximum.

III. MLGW

The goal of the present work is to provide an accurate machine learning model which outputs the functions $A(t; \tilde{\mathbf{\Theta}})$ and $\phi(t; \tilde{\mathbf{\Theta}})$ [Eqs. (2) and (3)], as generated by the state-of-the-art time-domain WF models. More formally, we seek an ML model that reliably reproduces the following map:

$$(q, s_1, s_2) \mapsto A_{(q, s_1, s_2)}(t), \quad (5)$$

$$(q, s_1, s_2) \mapsto \phi_{(q, s_1, s_2)}(t). \quad (6) \quad 223$$

In the context of ML, our task reduces to performing two regressions from $\tilde{\mathbf{\Theta}}$ to the amplitude and phase of the WF. A *regression* is a statistical method to infer the relationship between a set of “independent variables” and a set of “dependent variables.” A *model* consists in a functional form for such relation, usually with many free parameters to be specified. By looking at the data, one should be able to make a proper choice for their value.

In order to be able to perform each regression, several steps are required.

(A) *Setting a time grid.*—Each WF must be represented on a discrete time grid, which allows for efficient and reliable reconstruction on an arbitrary, user-given, grid. After this operation, the functions $A(t)$ and $\phi(t)$ are represented as vectors.⁶

(B) *Creating a dataset of WFs.*—A large number of WFs must be generated on the chosen time grid for a

⁵Note that we adopt a nonstandard sign convention for the phase.

⁶In ML jargon, this procedure is called preprocessing and aims to create a standard representation for all the data available (in our case the WFs).

different number of orbital parameters (q, s_1, s_2) .
This will form the training set for the model.

(C) *Reducing the dimensionality of a WF.*—In order to make the regression feasible, we build a low-dimensional representation of the WF. This operation must be invertible: Once a low-dimensional representation is given, one should be able to reconstruct the higher-dimensional WF.

(D) *Learning a regression.*—We train a model to perform the regression from (q, s_1, s_2) to the low-dimensional representation of the WF.

We discuss these points in detail in what follows.

A. The time grid

Each function $f(t)$ to fit (i.e., amplitude and phase) must be represented by its values $\mathbf{f} \in \mathbb{R}^D$ on a discrete grid of D points $\mathbf{t} \in \mathbb{R}^D$. It is convenient to work in a grid of (dimensionless) reduced time $\tau \equiv \mathbf{t}/M$. The time grid is chosen with the convention that at $\tau = 0$ the function $A(t; \mathbf{\theta})$ (i.e., the amplitude of the 22 mode) has a peak. Once a time grid is set, the vector \mathbf{f} is defined as follows:

$$\mathbf{f}(\tilde{\mathbf{\theta}})_i = f_{\tilde{\mathbf{\theta}}}(\tau_i), \quad i = 1, \dots, D. \quad (7)$$

The value of f at an arbitrary time must be found by interpolation, and, to make the interpolation effective, we introduce a grid adapted to the function's variation. Clearly, an equally spaced grid over times is not the best choice, since the amplitude has a very narrow peak at $\tau = 0$. A good solution is to build the τ grid τ as

$$\tau_i = \text{sgn} \tilde{\tau}_i \times (|\tilde{\tau}_i|)^{1/\alpha}, \quad i = 1, \dots, D, \quad (8)$$

where $\tilde{\tau}_i$ are D equally spaced points in the range of interest and we call α *distortion parameter*. This choice ensures that more points are accumulated around the peak of amplitude. As the phase has a rather regular behavior, it is not important to tune the time grid on it. For this reason, a single grid for amplitude and phase, tuned on the amplitude, is used.

The length of the time grid determines the maximum length of the WFs that the model can generate. Let us define $\tau_{\min} = -\tau_0 > 0$ as the starting point of the grid; thus, each WF starts at a time $\tau_{\min}M$ before the merger. Note that τ_{\min} is an important hyperparameter, set by the user, which strongly impacts on the model applicability. The minimum frequency in the signal as a function of M , q , and τ_{\min} is given approximately⁷ by

$$f_{\min} = 151 \text{ Hz} \left(\frac{(1+q)^2}{q} \right)^{3/8} \left(\frac{M_{\odot}}{M} \right) \left(\frac{1}{\tau_{\min}} \frac{s}{M_{\odot}} \right)^{3/8}. \quad (9)$$

⁷The expression is approximate, because it is obtained within a Newtonian framework and does not consider spin effects. Nevertheless, it gives a useful estimation of the range of the applicability of the model.

B. Dataset creation

As in any ML method, we must create a dataset before training a model. In our case, the dataset consists in a matrix $X \in \mathbf{Mat}(N, 3 + 2D)$ of N waveforms, which has the following form:

$$X_i = [q, s_1, s_2, A_{\tilde{\mathbf{\theta}}}^T, \phi_{\tilde{\mathbf{\theta}}}^T], \quad (10)$$

where X_i denotes the i th row of the dataset matrix.

The dataset is filled with parameters $\tilde{\mathbf{\theta}}$ randomly drawn from an uniform distribution in the domain of interest \mathcal{P} : $\tilde{\mathbf{\theta}}_i \sim \text{Unif}(\mathcal{P})$. As stressed above, any time-domain EOB waveform model is suitable for such purpose. Indeed we employed successfully both⁸ TEOBResumS and SEOBNRv4. The output of the training model must be interpolated to the chosen time grid.

It is important to ensure that all waves have zero phase at a constant time point \tilde{t} : This is crucial to obtain a continuous dependence of the phase components on the orbital parameters. As model performances are not seen to depend on the choice of \tilde{t} , we arbitrarily set $\tilde{t} = 0$. The range \mathcal{P} of masses and spins covered by the model, as well as the starting point of the grid τ_{\min} , can be freely chosen by the user, depending on their needs.

C. Dimensionality reduction

Once we are able to represent waveforms, a regressions $\tilde{\mathbf{\theta}} \mapsto A_{\tilde{\mathbf{\theta}}}, \phi_{\tilde{\mathbf{\theta}}} \in \mathbb{R}^D$ is unfeasible, as the dimension of the target space (i.e., the dimension D of the time grid) is too large. Luckily, the elements of A, ϕ are strongly correlated with each other: The independent amount of information, required to fully reconstruct the wave, can be stored in a low-dimensional vector. A number of ML techniques to perform such a task are available. Among them, principal component analysis (PCA) (see Chap. 12 in Ref. [45]) was found to be particularly effective.

The basic idea behind PCA is to seek a *linear relation* between high-dimensional and low-dimensional data: High-dimensional data ($\in \mathbb{R}^D$) are projected onto a K -dimensional subspace, by means of an orthogonal projection. A theorem (see Sec. 12.2.1 in Ref. [45]) guarantees that, for zero mean data, the generators of subspace are the (orthonormal) first K eigenvectors of the empirical covariance matrix $\Sigma \in \mathbf{Mat}(D, D)$. The eigenvectors are also called principal components (PCs) of the data. Thus, the projection matrix $H \in \mathbf{Mat}(K, D)$ holds in each row the PCs and each high-dimensional point can

⁸For completeness, we have also computed a MLGW mode using SEOBNRv2_opt [44], a spin-aligned model that was optimized with respect to the original SEOBNRv2 [41] so to improve its computational efficiency.

be effectively expressed as a linear combination of the K PCs.⁹

A PCA model is trained with the dataset Eq. (10): It represents an (approximate) bijective map between the high-dimensional WF $\mathbf{f} = \mathbf{A}_{\tilde{\mathbf{g}}}, \phi_{\tilde{\mathbf{g}}} \in \mathbb{R}^D$ and the low-dimensional representation $\mathbf{g} = \mathbf{g}_A, \mathbf{g}_\phi \in \mathbb{R}^K$. The relation takes the following form:

$$\mathbf{g} = H(\mathbf{f} - \boldsymbol{\mu}), \quad (11)$$

$$\mathbf{f} = H^T \mathbf{g} + \boldsymbol{\mu}, \quad (12)$$

where $\boldsymbol{\mu}$ is the empirical mean vector $\boldsymbol{\mu} = \frac{1}{N} \sum_{i=1}^N \mathbf{f}_i \in \mathbb{R}^D$ and the matrix H is computed from the empirical covariance $\Sigma = \frac{1}{N} \sum_{i=1}^N (\mathbf{f}_i - \boldsymbol{\mu})(\mathbf{f}_i - \boldsymbol{\mu})^T$.

D. Regression

Once a dimensional reduction (and reconstruction) scheme is available, we want to perform the regression

$$\tilde{\mathbf{g}} \mapsto \mathbf{g}(\tilde{\mathbf{g}}). \quad (13)$$

A number of ML models are available for this purpose. The model mixture of experts (MoE) [46] (see Chap. 11 in Ref. [45]) is found to be a good compromise between simplicity and flexibility.

MoE performs the following 1D regression:

$$y(\mathbf{x}) = \sum_{l=1}^L (W^T \mathbf{x})_l \cdot \mathcal{S}(V^T \mathbf{x})_l, \quad (14)$$

where \mathcal{S} is the *softmax function*:

$$\mathcal{S}(V^T \mathbf{x})_l = \frac{e^{(V^T \mathbf{x})_l}}{\sum_{l'=1}^L e^{(V^T \mathbf{x})_{l'}}}, \quad (15)$$

$\mathbf{x} \in \mathbb{R}^{\tilde{M}}$, and $V, W \in \mathbf{Mat}(\tilde{M}, L)$. The meaning of Eq. (14) is clear: The output is a weighted combination of L linear regressions $(W^T \mathbf{x})_l$ (called *experts*); each expert performs a reliable regression in a small region of the space. The softmax function (in this context also called *gating function*) switches on the expert contributions whenever this is required. MoE is usually fitted with the expectation maximization algorithm, which iteratively sets the W and V by refining a lower bound to the log-likelihood of the model.

Linear regression is a very simple model, often inadequate to model a complex relation. A simple trick to

improve its performance is called *basis functions expansion*. It consists in the replacement

$$\mathbf{x} \rightarrow \boldsymbol{\xi}(\mathbf{x}) = [\xi_1(\mathbf{x}), \dots, \xi_M(\mathbf{x})]^T. \quad (16)$$

Thus, each expert becomes a nonlinear regression of the input \mathbf{x} . A careful choice of basis functions can really make a difference in fit performances, and it must be done at validation time, by comparing performances of different models.

The user must choose the number L of experts and the basis functions features $\boldsymbol{\xi}(\tilde{\mathbf{g}}) \in \mathbb{R}^M$ to use. Including in the ξ_i every monomial up to third or fourth order in the three variables ($\log q$, s_1 , and s_2) seems a good working choice for our model. The choice of working with the variable $\log q$ rather than q is based on validation results. Heuristically, it prevents the values of the data features from varying too much within the range of interest, thus yielding more stable numerical performance.

As MoE model deals with single-dimensional outputs, a single independent regression must be performed for each component g_k of $\mathbf{g} \in \mathbb{R}^K$.¹⁰ In general, a regression will be a collection of MoE weights $\{W^{(k)}, V^{(k)} \in \mathbf{Mat}(M, L_k)\}_{k=0}^K$, where index k labels different regressions for each PC.

E. Summary

The model has the following explicit form:

$$\begin{aligned} \text{model: } \mathcal{P} \subset \mathbb{R}^3 \rightarrow \mathbb{R}^K \rightarrow \mathbb{R}^D \\ \tilde{\mathbf{g}} \mapsto \mathbf{g}(\tilde{\mathbf{g}}) = \begin{pmatrix} \sum_{l=1}^{L_1} (W^{(1)T} \boldsymbol{\xi})_l \cdot \mathcal{S}(V^{(1)T} \boldsymbol{\xi})_l \\ \vdots \\ \sum_{l=1}^{L_K} (W^{(K)T} \boldsymbol{\xi})_l \cdot \mathcal{S}(V^{(K)T} \boldsymbol{\xi})_l \end{pmatrix} \\ \mapsto \mathbf{f}(\tilde{\mathbf{g}}) = H^T \mathbf{g}(\tilde{\mathbf{g}}) + \boldsymbol{\mu}, \end{aligned} \quad (17)$$

where $\boldsymbol{\xi}(\tilde{\mathbf{g}}) \in \mathbb{R}^M$ are the chosen basis function for the regression and $\mathcal{S}(\cdot)_k$ is the softmax function Eq. (15). Two relations of the same type must be fitted, one for the amplitude, the other for the phase.

Once weights are set properly, the expression provides an estimation for the waveform h_{FIT} in Eq. (2). The complete WF $h(t; m_1, m_2, s_1, s_2, d_L, \iota, \varphi_0)$ is computed with Eq. (4). The model can extrapolate outside the range of train orbital parameters, without a guarantee of reliable results.

Note that Eq. (17) can be used to compute a closed form expression for the gradients of the waveform with respect to the orbital parameters. Such calculations are included in the released version of MLGW.

⁹For this reason, PCA can also be seen as a perturbative expansion of a high-dimensional observation. A more reliable reconstruction can be achieved by adding more and more PCs, each of which is less important than the previous.

¹⁰This is not a great limitation, because, due to orthogonality of PCs, each g_j is independent from the other: We do not miss correlation among different regressions.

IV. MODEL PERFORMANCE

We now discuss some validation tests on our model. We first study how its performance depends on the choice of hyperparameters. Second, we assess the model accuracy and its limitations. Finally, we measure the speed up provided by our model as compared with the training EOB model. For our tests, we train our model with TEOBResumS. Very similar results are obtained for a model trained on SEOBNRv4.

As is common, we measure the similarity between two waves by means of the *optimal mismatch*:

$$\tilde{\mathcal{F}}[h_1, h_2] = 1 - \frac{\langle h_1, h_2 \rangle}{\sqrt{\langle h_1, h_1 \rangle \langle h_2, h_2 \rangle}}, \quad (18)$$

where, as usual, we defined the *Wiener product* as

$$\langle h_1, h_2 \rangle = 4 \int_0^\infty df \frac{\tilde{h}_1^*(f) \tilde{h}_2(f)}{S_n(f)}. \quad (19)$$

In the equation above, $S_n(f)$ is the detector noise curve, the \tilde{h} denotes the Fourier transform of the strain h , and the $*$ denotes complex conjugation. In what follows, we always use a flat noise curve (i.e., constant power spectral density for the detector noise).

A. Validation

Wherever relevant, we will employ a dataset with 5800 waveforms generated in the domain $\mathcal{P} = [1, 20] \times [-0.8, 0.95] \times [-0.8, 0.95]$, with $\tau_{\min} = 1.0 \text{ s}/M_\odot$. The results here refer to MLGW-TEOBResumS, and similar results are obtained for MLGW-SEOBNRv4.

1. Dataset generation parameters

We first evaluate the impact of number of grid points N_{grid} and distortion parameter α [see Eq. (8)]. Let $\mathbf{f}_{N_{\text{grid}}, \alpha}$ be the wave stored in a dataset where τ_{\min} and \mathcal{P} are fixed as above. We compare it with the output of the EOB model \mathbf{f}_{EOB} . We then vary N_{grid} and α and report the resulting mismatch $\mathcal{F}[\mathbf{f}_{\text{EOB}}, \mathbf{f}_{N_{\text{grid}}, \alpha}]$ in Fig. 1.

As expected, we note that, by increasing the number of grid points, the mismatch decreases. Furthermore, using more than $\sim 10^3$ grid points does not bring any improvement to mismatch. In this case, the result is dominated by numerical errors in the interpolations, and it provides a lower bound for the performances of the fit. A careful choice of α provides a remarkable improvement when N_{grid} is small. For a high number of grid points, different values of α yield almost equivalent results. A good setting for dataset hyperparameters might be $N_{\text{grid}} \simeq 3 \times 10^3$ and $\alpha \in (0.3, 0.5)$.

2. MoE parameters

We focus only on setting the number of experts N_{exp} for each component model and the basis functions $\xi_i(\tilde{\theta})$ to use

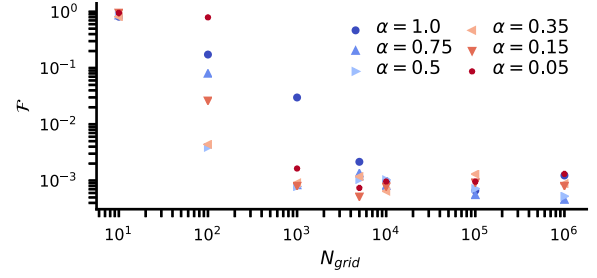


FIG. 1. Average mismatch between waves $\mathbf{f}_{N_{\text{grid}}, \alpha}$, as saved in the training dataset, and raw waves from EOB model, as a function of time grid size N_{grid} . Each series refers to a different values of α . Clearly, $N_{\text{grid}} \simeq 3 \times 10^3$ and $\alpha \in (0.3, 0.5)$ is a good choice for the dataset hyperparameters.

in the regression. Other parameters, related to the details of the training procedure, will not be considered here.

Figure 2 presents our results. We fitted a model for amplitude (or phase) for different configurations of expert number N_{exp} and polynomial basis function. By the label “ n th order,” we mean that, in the basis function expansion, every monomial up to n th order is used. We report with a color bar the value of the mismatch F between test and reconstructed WFs. The MoE models for each component share the same number of experts N_{exp} . The test mismatch for the fitted amplitude (phase) is computed by using the test phase (amplitude) in the reconstructed wave.

As a general trend, fit performance improves whenever the model complexity (i.e., number of fittable parameters) increases. In general, we note that adding more features is

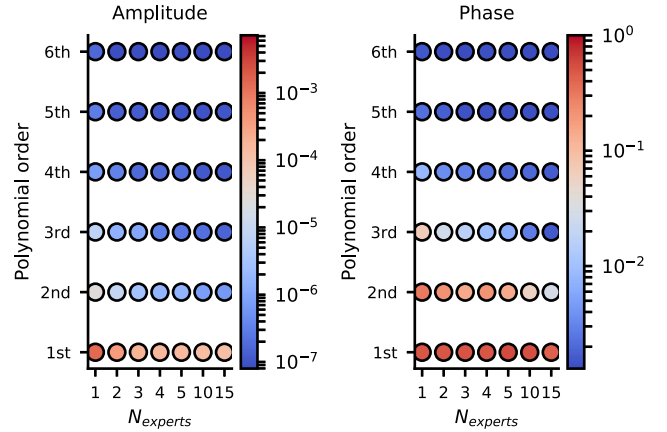


FIG. 2. Validation results for a fit of the MoE model. Each point corresponds to an MoE regression for the amplitude (left) and phase (right), with a different value of expert number N_{exp} and order of polynomial basis function. The amplitude and phase are represented with five and four PCs, respectively. In the color bar, we represent the mismatch on test waves: It is obtained by reconstructing test waves with fitted amplitude (phase) and test phase (amplitude). A model with four experts and with a fourth-order polynomial provides good balance between simplicity and accuracy.

more effective than employing the number of experts. However, the model performance does not improve indefinitely: As we see in Fig. 2, many “complex” models show similar performance, regardless their complexity. A model with four experts and fourth-order polynomial regression is the “simplest” of such models, and, thus, it should be deemed as the best choice.

3. Choosing the number of PCs

Of course, the accuracy of the reconstruction of the low-dimensional representation depends on the number K of principal components considered: The more PCs are used, the best accuracy can be achieved. However, in practice, due to errors in the MoE regression, one cannot reduce the reconstruction mismatch arbitrarily. Indeed, at high PC order, the relations to fit become noisy and the regression becomes less accurate, eventually washing out any improvement brought by a higher number of PCs. For this reason, one should choose the number of PCs while checking MoE performance.

In Fig. 3, we report a numerical study of this. We plot the reconstruction mismatch as a function of the number of PCs considered. We consider separately the amplitude and the phase. In one series, we reconstruct the wave using true values of PCs: The mismatch is a measure of PCA accuracy. In the other, we reconstruct a wave using values for PCs as guessed by MoE regression: This is a measure of accuracy of both PCA and regression. For the first two PCs, the regression is accurate enough for reproducing the PCA accuracy. On the other hand, any regression beyond the third or fourth PCA component does not give any improvement to the MoE mismatch: The noise in the relation of high-order PCs is too high for a regression to be performed.

In the PCA, we include every PC which yields improvement in MoE mismatch. For our model, $K = 5$ (4) is a good choice for amplitude (phase). Of course, this strongly depends on the regression model: The more precise the model is, the more PCs can be included. However, no model can increase its accuracy indefinitely, because every

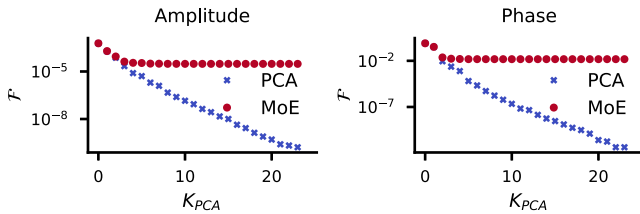


FIG. 3. Test mismatch as a function of the number of PCs used in the low-dimensional representation. Label “PCA” refers to waves reconstructed with PCA only; points with label “MoE” are reconstructed after a MoE regression. Data refers to the amplitude (left panel) and phase (right panel). MoE model is chosen to be the optimal one, with four experts and a fourth-order polynomial.

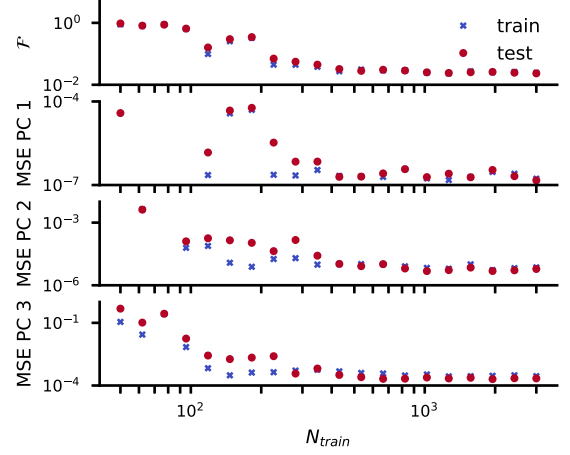


FIG. 4. Train and test error for MoE fit of four PCs of phase, as a function of the number of training points. We report train and test reconstruction mismatch (top) and MSE for the first three PCs (below). MoE model employs four experts and a fourth-order polynomial for a basis function expansion. Test mismatch are obtained using test amplitude to reconstruct the waveform; this is not a great limitation, as any error in phase reconstruction dominates the overall mismatch.

training set has an intrinsic noise level, due to numerical error and to the approximations in the underlying physical model.

4. Choosing the number of training points

The choice of the number of training points N_{train} must trade between accuracy and speed of execution. Too many training points will make the training slow, while too few training points will yield a poor model, which does not generalize the data (*underfitting*). In the choice of number of training points, the comparison between train and test error will provide important information on how the model is able to generalize the trend. In Fig. 4, we report the train and test value of mismatch and mean squared error (MSE) of the first three PCs as a function of the number of training points. Data refers to a MoE model fitted for four PCs of the phase dataset, with four experts and a fourth-order polynomial. As N_{train} increases, we see a steady decrease of the errors, until a plateau is reached. Since for a reasonably high number of training points ($N_{\text{train}} \gtrsim 50$) the train and test errors are close to each other, we note that overfitting is not a problem. For $N_{\text{train}} \gtrsim 800$, the trend stabilizes and increasing training points does not affect much model performance. In the present model, setting $N_{\text{train}} \simeq 3000$ is a good choice.¹¹

¹¹As compared with standard neural networks, which routinely employ $O(10^5)$ points datasets, this is an incredibly low amount of data. This is due to the fact that MoE is a simple model with a few number of parameters: Few data are enough for learning a reliable relation.

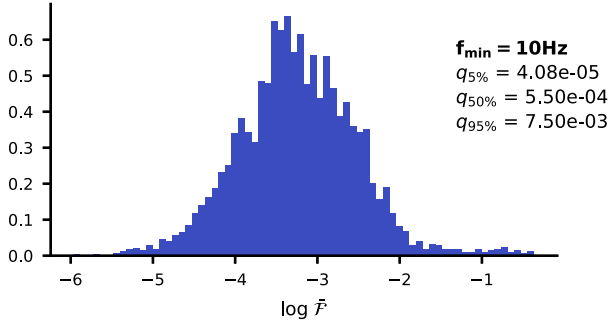


FIG. 5. Logarithm of mismatch between TEOBResumS and MLGW-TEOBResumS, computed on $N = 4000$ test waveforms. Each WF is generated with random masses and spins and with a starting frequency of 10 Hz. The median value $q_{50\%}$ and the positions $q_{5\%}$ and $q_{95\%}$ of the fifth and 95th percentile are reported.

B. Accuracy

We compute the mismatch between MLGW and the underlying training model (TEOBResumS) for a large number of WFs, and we report our results in the histogram in Fig. 5. The mismatch distribution has a median mismatch $\mathcal{F}_m = 5.5 \times 10^{-4}$.

When comparing our results with the discrepancies between state-of-the-art EOB waveforms and NR waveforms (see for comparison [8,23]), we see that we obtain a distribution with similar median value but longer tails (i.e., more WFs with either a very small or a very large mismatch). However, our results are reported with a flat noise curve, whereas the other discrepancies are quoted with the Advanced LIGO power spectral density: This might be a plausible explanation for the observed differences.

To understand better model performances, it is interesting to display the accuracy as a function of the orbital parameters $\Theta = (q, M, s_1, s_2)$. We generate waves for randomly chosen values of $\Theta = (q, M, s_1, s_2)$, and, for each wave, we measure test mismatch \mathcal{F} and MSE on the reconstruction of the first PC for the phase. The latter is useful to test the accuracy of the fit alone, before wave reconstruction. The results are reported in Fig. 6.

The model shows poor performances ($\mathcal{F} \sim 10^{-3}$) for $q \simeq 1$ and for high (positive) values of $s_1 + s_2$. By looking at the top line in Fig. 6, we note that the MSE does not depend on M , as expected since the dependence on M is inserted analytically in the model. In the center line in Fig. 6, as long as the s_1 dependence is considered, the most striking feature is the inverse correlation of mismatch and MSE for the first phase PC. This means that, being nonleading, spin contributions are not important for the first PC but become dominant at higher order of PCs. Indeed, the values of the first PC are well correlated with mismatch in the case of q . See [47] for a closely related discussion on PCA components and its dependence on physical parameters. In the third row in Fig. 6, the inverse correlation between the mismatch and MSE can be noted again.

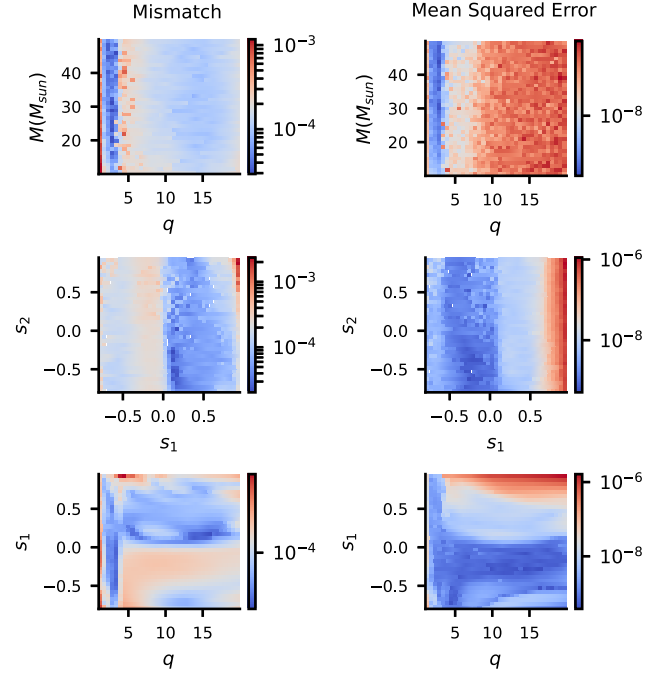


FIG. 6. To compare TEOBResumS and MLGW-TEOBResumS, we report test mismatch (left column) and MSE (right column) for the first PC of the phase, as a function of masses and spins. The histograms hold 145061 waveforms, with randomly drawn parameters. Each WF starts 8 s before merger. Apart from poor performances for $q \simeq 1$ and for high (positive) values of $s_1 + s_2$, the model performance does not depend much on the input parameters.

C. Run-time analysis

We now assess the time performances of our model. We are interested to make comparisons between MLGW and both training models as well as with SEOBNRv4_ROM.

1. Comparison with TEOBResumS and SEOBNRv4

When dealing with a real detection scenario, we are often interested in generating a WF which starts from a given frequency f_{\min} , which is usually set by the detector sensitivity window. Thus, it is crucial to measure the speed up that our model can provide in performing such task. We define the speed up \mathcal{S} as the ratio between the run-time of the benchmark model and the run-time of MLGW to produce a waveform starting from a given f_{\min} . Each waveform is produced with constant total mass $M = 100 M_{\odot}$ and random parameters; the WF is sampled at $f_{\text{sam}} = 2048$ Hz. We consider the two cases with $f_{\min} = 5$ Hz and $f_{\min} = 20$ Hz. The first choice refers to the hypothetical lower bound for the sensitivity of the Einstein Telescope (ET), while the second is close to that of Advanced-LIGO/Virgo. In Fig. 7, we report the histogram of the measured speed up values for both TEOBResumS and SEOBNRv4.

We see that in both cases a substantial speed up is achieved. The speed up is higher for longer WFs, making

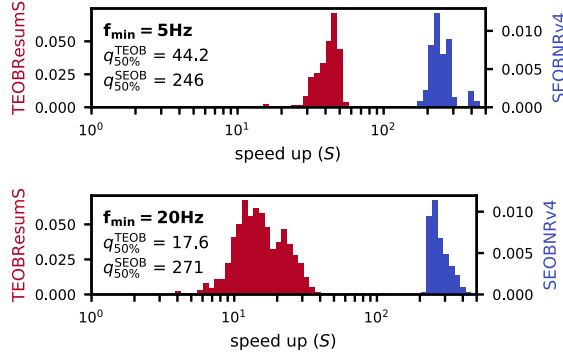


FIG. 7. Speed up given by MLGW-TEOBResumS (dark-red online) and MLGW-SEOBNRv4 (blue online), as compared with their respective native implementation. Because of the computational cost, we use $N = 500$ test waveforms for SEOBNRv4 and $N = 2000$ waveforms for TEOBResumS. Each WF is generated with random physical parameters and has a minimum frequency of 5 (top panel) and 20 Hz (bottom panel). We set a constant total mass $M = 100 M_{\odot}$ and the sampling rate $f_{\text{sam}} = 2048$ Hz. Median values $q_{50\%}^{\text{TEOB}}$ and $q_{50\%}^{\text{SEOB}}$ for the two models are also reported.

our model particularly convenient for advanced detectors, with a larger sensitivity window. This is clearly understood: A longer WF requires more computation for a EOB model, while roughly the same amount of work is done by MLGW. Furthermore, we note that the speed up with MLGW-SEOBNRv4 is around 10 times higher than that of MLGW-TEOBResumS.

2. Comparison with SEOBNRv4_ROM

Let us turn now to discuss a performance comparison with SEOBNRv4_ROM, that is currently considered state of the art for the WF generation time. We note that MLGW-SEOBNRv4 and MLGW-TEOBResumS are completely equivalent from the point of view of the generation time for a WF, so that we simply refer to the model as MLGW here and below. The measured speed up of MLGW with respect to SEOBNRv4_ROM is illustrated in Fig. 8. The comparison is made as above with a sampling rate $f_{\text{sam}} = 2048$ Hz for two different starting frequency $f_{\text{min}} = 5$ Hz and $f_{\text{min}} = 20$ Hz. As the ROM model yields WFs in the frequency domain, in the run-time evaluation we also included a fast Fourier transform (FFT) of the time-domain WF of MLGW. This ensures that we are evaluating the two models at the same conditions. Interestingly, the time taken by the FFT (in the NumPy implementation) is similar to that required to generate a WF. Thus for a WF in FD, our model cannot be substantially faster, due to the limitation imposed by the FFT.¹²

¹²Actually, the operations required by the FFT take the most of the time. In fact, before the FFT the waveform is evaluated on a dense equally spaced grid: As can be seen in Table I, such an operation can be very expensive.

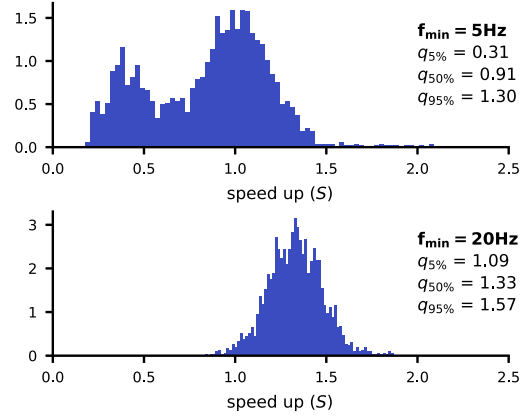


FIG. 8. Speed up of MLGW with respect to SEOBNRv4_ROM, computed on $N = 2000$ test waveforms. Each WF is generated with random physical parameters and has a minimum frequency of 5 (top panel) and 20 Hz (bottom panel). We set a constant total mass $M = 100 M_{\odot}$ and the sampling rate $f_{\text{sam}} = 2048$ Hz. The median value $q_{50\%}$ and the positions $q_{5\%}$ and $q_{95\%}$ of the fifth and 95th percentile are reported.

We note that the performances are quite similar to each other. If a lower starting frequency is chosen, MLGW is slightly outperformed. Perhaps, this can be cured by fitting a model in the frequency domain: In this case, the FFT would not be required anymore, resulting in a large speed up in the execution time. It is important to stress that MLGW is written in pure PYTHON, while SEOBNRv4_ROM is coded in C. In fact, a PYTHON code could be easily accelerated (i.e., parallelized, run on GPUs, etc.) with dedicated libraries, thus allowing one to push the code performance further.

3. Profiling

It is interesting to have a knowledge of the time spent by MLGW in each stage of the WF generation procedure. We generate 100 waves with random physical parameters, and we measure the CPU time spent to execute each basic task. In Table I, we compare the results for two values of N_{grid} . We see that the cost of generating the raw WF does not depend on the number of grid points. On the other hand, the interpolation and the postprocessing depend on N_{grid} , and their cost grows dramatically as the user requires more and more points. It is important to stress that the latter two tasks are slow only because they deal with a large number of points. Indeed, they perform trivial and “quick” operations, and their execution relies on well-optimized NumPy routines. If such a number of data points is required, very little space is left for speed up.

V. APPLICATION TO GWTC-1

We use the implementation of our MLGW-TEOBResumS and MLGW-SEOBNRv4 models to provide a new, and independent, measure of the properties of the GW sources collected in GWTC-1, the first catalog of detected GW

TABLE I. Time taken (averaged on multiple runs) by different stages of the generation of 100 waveforms; data refers to two different values of N_{grid} . “Generation of raw WF” refers to the computation of the strain h_{FIT} as produced by MLGW. “Interpolation to the user grid” evaluates the WF on the grid chosen by the user. “Postprocessing” labels the computation performed to include the dependence on d_L , t , and φ_0 .

Task (for 100 WFs)	CPU time (ms)	
	$N_{\text{grid}} = 10^3$	$N_{\text{grid}} = 10^5$
Generation of raw WF	6.9 (46.9%)	7 (1.6%)
Interpolation to the user grid	4.5 (30.6%)	194 (45.3%)
Postprocessing	1.7 (11.6%)	206 (48.1%)
Total	14.7 (100%)	428 (100.0%)

sources [42], corresponding to the first two observing runs of the LIGO and Virgo detectors. The GWTC-1 catalog consists of ten BBH systems and a binary neutron star (BNS) system, GW170817. Since the waveform models we considered for the training concern only spin-aligned BBH waveforms, we do not analyze GW170817, but we focus only on the ten BBH systems. We trained MLGW in the range $\mathcal{P} = [1, 20] \times [-0.8, 0.95] \times [-0.8, 0.95]$, and we set $\tau_{\text{min}} = 4 \text{ s}/M_{\odot}$. Our parameter estimation algorithm is gwmodel [48], a publicly available infrastructure written in a mixture of PYTHON and CYTHON that serves as interface for the parallel nested sampling implementation cpnest [49]. The analysis of each BBH system is set up as follows: We choose a total of 2000 live points, four parallel Markov chain Monte Carlo chains with a maximum length of 5000 steps to ensure that each successive sample is independent of the previous. These settings yield an average of ~ 15000 posterior samples and evidence calculations that are accurate to the first decimal digit. For each BBH system, we choose prior distributions as described in the GWTC-1 release paper [42]. Finally, and critically, to ensure that our results can be compared fairly to published ones, we employ the power spectral densities released as part of GWTC-1. No calibration uncertainty model is assumed for these runs.

Table II summarize the results with MLGW-TEOBResumS and with MLGW-SEOBNRv4. The table exclusively reports summary statistics for the intrinsic parameters. All mass parameters quoted are in the source frame. The redshift of each BBH is estimated from its luminosity distance posterior and converted into a redshift by assuming the cosmological parameters given in Ref. [50]. The second part of Table II also lists, for convenience, the results published in the GWTC-1 catalog paper [42]. In addition, posteriors for the individual masses, final masses, and spins and dimensionless spin magnitudes are shown in Figs. 9–11 for both the models.

A few observations are in order. First of all, our results, obtained with both models, are extremely similar to what published by the LVK Collaboration. This is reassuring, as

TABLE II. Summary table for the inferred intrinsic parameters from MLGW with TEOBResumS and SEOBNRv4 and the released GWTC-1 credible intervals. All mass parameters quoted are computed in the source frame; see the text for details of the calculation. For GWTC-1 we report results from Table III of Ref. [42]. These results were obtained by averaging together the outcomes of the *precessing* SEOBNRv3 [51] and *MRPhenomPv2* waveform models. The uncertainties correspond to the 90% credible intervals. The results with TEOBResumS and SEOBNRv4 are very consistent with each other, and they are slightly larger than the published LVC ones obtained using different waveform models. Note that the inverse mass ratio $1/q$ is not listed in Ref. [42].

Event	MLGW-TEOBResumS						MLGW-SEOBNRv4						GWTC-1					
	m_1/M_\odot	m_2/M_\odot	\mathcal{M}/M_\odot	$1/q$	χ_{eff}	m_1/M_\odot	m_2/M_\odot	\mathcal{M}/M_\odot	$1/q$	χ_{eff}	m_1/M_\odot	m_2/M_\odot	\mathcal{M}/M_\odot	m_1/M_\odot	m_2/M_\odot	\mathcal{M}/M_\odot	χ_{eff}	
GW150914	36.36 ^{+4.72} _{-2.64}	32.64 ^{+2.93} _{-4.44}	29.87 ^{+1.95} _{-1.50}	0.91 ^{+0.08} _{-0.21}	0.14 ^{+0.10} _{-0.10}	36.09 ^{+4.89} _{-2.58}	32.55 ^{+2.80} _{-4.37}	29.70 ^{+1.95} _{-1.36}	0.91 ^{+0.08} _{-0.21}	0.10 ^{+0.09} _{-0.08}	35.6 ^{+4.7} _{-3.1}	30.6 ^{+3.0} _{-4.4}	28.6 ^{+1.7} _{-1.5}	35.6 ^{+4.7} _{-3.1}	30.6 ^{+3.0} _{-4.4}	28.6 ^{+1.7} _{-1.5}	-0.01 ^{+0.12} _{-0.13}	
GW151012	34.51 ^{+21.37} _{-14.46}	11.67 ^{+6.92} _{-4.46}	16.86 ^{+3.01} _{-2.68}	0.33 ^{+0.56} _{-0.19}	0.53 ^{+0.20} _{-0.33}	32.54 ^{+20.08} _{-12.12}	12.18 ^{+4.50} _{-2.84}	16.96 ^{+2.67} _{-2.84}	0.37 ^{+0.50} _{-0.21}	0.53 ^{+0.19} _{-0.32}	23.2 ^{+14.9} _{-5.5}	13.6 ^{+4.1} _{-2.8}	15.2 ^{+2.1} _{-1.2}	23.2 ^{+14.9} _{-5.5}	13.6 ^{+4.1} _{-2.8}	15.2 ^{+2.1} _{-1.2}	0.05 ^{+0.32} _{-0.20}	
GW151226	16.44 ^{+12.15} _{-5.52}	6.38 ^{+2.86} _{-2.18}	8.72 ^{+0.45} _{-0.27}	0.39 ^{+0.45} _{-0.24}	0.32 ^{+0.24} _{-0.14}	16.35 ^{+12.60} _{-5.61}	6.36 ^{+2.98} _{-2.22}	8.69 ^{+0.41} _{-0.27}	0.39 ^{+0.48} _{-0.25}	0.31 ^{+0.24} _{-0.15}	13.7 ^{+8.8} _{-3.2}	7.7 ^{+2.2} _{-2.5}	8.9 ^{+0.3} _{-0.3}	13.7 ^{+8.8} _{-3.2}	7.7 ^{+2.2} _{-2.5}	8.9 ^{+0.3} _{-0.3}	0.18 ^{+0.20} _{-0.12}	
GW170104	31.16 ^{+10.55} _{-4.77}	22.69 ^{+4.70} _{-6.91}	22.83 ^{+2.64} _{-2.06}	0.74 ^{+0.23} _{-0.35}	0.23 ^{+0.15} _{-0.15}	30.45 ^{+10.49} _{-4.56}	22.82 ^{+4.43} _{-7.00}	22.64 ^{+2.51} _{-1.89}	0.76 ^{+0.22} _{-0.37}	0.19 ^{+0.15} _{-0.14}	30.8 ^{+5.6} _{-5.6}	20.0 ^{+4.9} _{-4.6}	21.4 ^{+2.2} _{-1.8}	30.8 ^{+5.6} _{-5.6}	20.0 ^{+4.9} _{-4.6}	21.4 ^{+2.2} _{-1.8}	-0.04 ^{+0.17} _{-0.21}	
GW170608	15.45 ^{+7.60} _{-3.05}	5.61 ^{+2.32} _{-1.50}	7.90 ^{+0.25} _{-0.17}	0.36 ^{+0.40} _{-0.18}	0.25 ^{+0.20} _{-0.17}	15.53 ^{+8.17} _{-5.25}	5.58 ^{+1.57} _{-1.57}	7.89 ^{+0.25} _{-0.18}	0.36 ^{+0.42} _{-0.19}	0.24 ^{+0.21} _{-0.18}	11.0 ^{+5.5} _{-1.7}	7.6 ^{+1.4} _{-2.2}	7.9 ^{+0.2} _{-0.2}	11.0 ^{+5.5} _{-1.7}	7.6 ^{+1.4} _{-2.2}	7.9 ^{+0.2} _{-0.2}	0.03 ^{+0.19} _{-0.07}	
GW170729	50.04 ^{+13.97} _{-9.98}	34.78 ^{+9.78} _{-9.73}	35.73 ^{+7.08} _{-5.08}	0.71 ^{+0.26} _{-0.28}	0.51 ^{+0.18} _{-0.23}	48.51 ^{+14.22} _{-9.62}	34.80 ^{+9.26} _{-9.01}	35.34 ^{+6.88} _{-4.89}	0.73 ^{+0.24} _{-0.27}	0.48 ^{+0.20} _{-0.24}	50.2 ^{+16.2} _{-10.2}	34.0 ^{+9.1} _{-10.0}	35.4 ^{+6.5} _{-4.8}	50.2 ^{+16.2} _{-10.2}	34.0 ^{+9.1} _{-10.0}	35.4 ^{+6.5} _{-4.8}	0.37 ^{+0.21} _{-0.25}	
GW170809	35.35 ^{+8.88} _{-5.43}	25.17 ^{+4.81} _{-5.91}	25.69 ^{+2.35} _{-1.74}	0.72 ^{+0.25} _{-0.27}	0.24 ^{+0.16} _{-0.14}	34.57 ^{+7.91} _{-5.12}	25.23 ^{+4.49} _{-3.47}	25.42 ^{+2.41} _{-1.62}	0.73 ^{+0.23} _{-0.26}	0.20 ^{+0.16} _{-0.13}	35.0 ^{+8.3} _{-5.9}	23.8 ^{+5.1} _{-5.2}	24.9 ^{+2.1} _{-1.7}	35.0 ^{+8.3} _{-5.9}	23.8 ^{+5.1} _{-5.2}	24.9 ^{+2.1} _{-1.7}	0.08 ^{+0.17} _{-0.17}	
GW170814	31.35 ^{+10.70} _{-3.48}	25.24 ^{+3.16} _{-6.40}	24.40 ^{+1.50} _{-1.29}	0.81 ^{+0.17} _{-0.36}	0.19 ^{+0.12} _{-0.11}	31.53 ^{+11.14} _{-3.73}	24.87 ^{+3.29} _{-3.11}	24.25 ^{+1.46} _{-1.31}	0.79 ^{+0.18} _{-0.36}	0.15 ^{+0.11} _{-0.10}	30.6 ^{+5.6} _{-3.0}	25.2 ^{+2.8} _{-4.0}	24.1 ^{+1.4} _{-1.1}	30.6 ^{+5.6} _{-3.0}	25.2 ^{+2.8} _{-4.0}	24.1 ^{+1.4} _{-1.1}	0.06 ^{+0.12} _{-0.12}	
GW170818	34.62 ^{+5.18} _{-5.18}	27.09 ^{+7.80} _{-2.77}	26.34 ^{+4.03} _{-2.77}	0.80 ^{+0.18} _{-0.37}	0.28 ^{+0.20} _{-0.19}	34.57 ^{+11.40} _{-5.52}	26.94 ^{+5.65} _{-7.67}	26.21 ^{+4.32} _{-2.82}	0.80 ^{+0.18} _{-0.36}	0.25 ^{+0.20} _{-0.17}	35.4 ^{+7.5} _{-4.7}	26.7 ^{+4.3} _{-5.2}	26.5 ^{+2.1} _{-1.7}	35.4 ^{+7.5} _{-4.7}	26.7 ^{+4.3} _{-5.2}	26.5 ^{+2.1} _{-1.7}	-0.09 ^{+0.18} _{-0.21}	
GW170823	40.69 ^{+10.21} _{-6.68}	31.17 ^{+7.13} _{-8.09}	30.63 ^{+5.01} _{-3.75}	0.78 ^{+0.19} _{-0.30}	0.31 ^{+0.18} _{-0.20}	40.66 ^{+10.11} _{-6.67}	30.91 ^{+6.72} _{-7.44}	30.52 ^{+5.09} _{-3.59}	0.77 ^{+0.20} _{-0.27}	0.28 ^{+0.19} _{-0.18}	39.5 ^{+11.2} _{-6.7}	29.0 ^{+6.7} _{-7.8}	29.2 ^{+4.6} _{-3.6}	39.5 ^{+11.2} _{-6.7}	29.0 ^{+6.7} _{-7.8}	29.2 ^{+4.6} _{-3.6}	0.09 ^{+0.22} _{-0.26}	

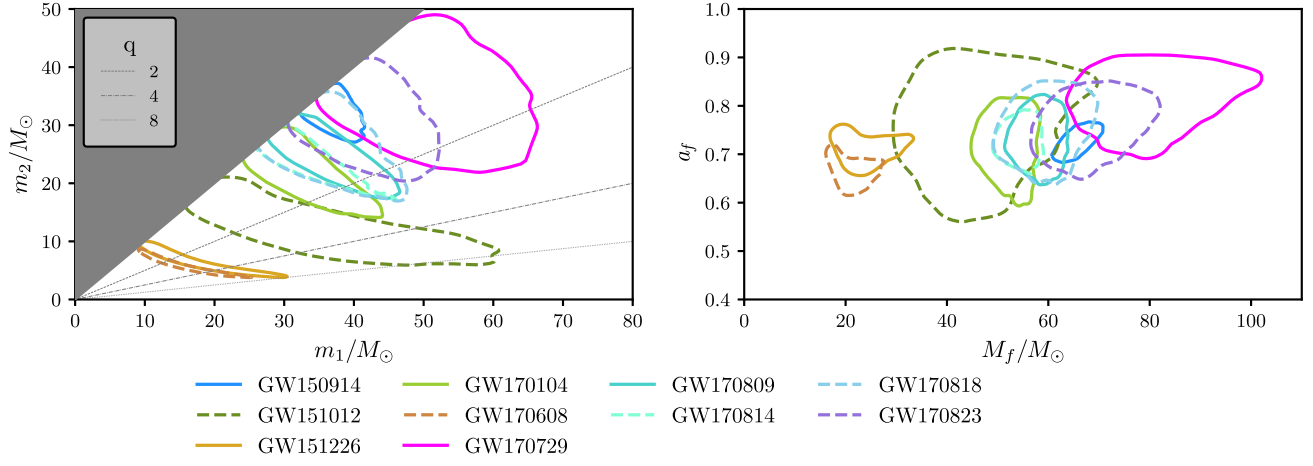


FIG. 9. Posterior probability densities of the component masses and final masses and spins of all the BBH system in GWTC-1 obtained using MLGW trained with the TEOBResums [8] spin-aligned waveform model. The contours enclose the 90% credible regions. Left panel: Source-frame component masses m_1 and m_2 . We use the convention $m_1 \geq m_2$ which produces the sharp cut in the two-dimensional (m_1, m_2) distribution (shaded region). Lines of constant mass ratio $q \equiv m_1/m_2$ are shown for $q = \{2, 4, 8\}$. Right panel: The mass M_f and dimensionless spin magnitude a_f of the final black holes. The figure is consistent with, though different from, Fig. 4 of Ref. [42].

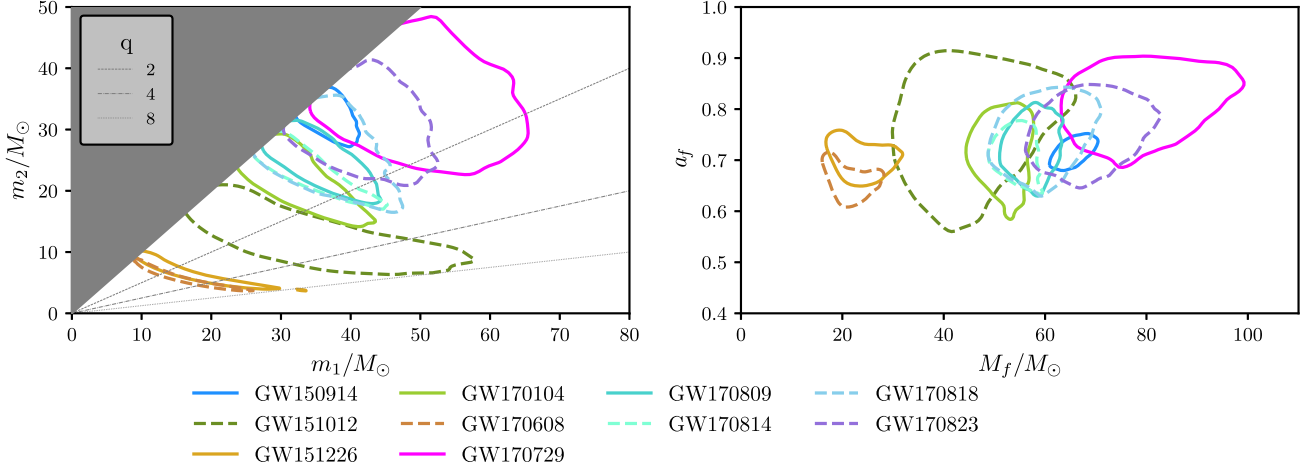


FIG. 10. Posterior probability densities of the component masses and final masses and spins of all the BBH system in GWTC-1 obtained using MLGW trained with the SEOBNRv4 [23] waveform model. The contours enclose the 90% credible regions. Left panel: Source-frame component masses m_1 and m_2 . We use the convention $m_1 \geq m_2$ which produces the sharp cut in the two-dimensional (m_1, m_2) distribution (shaded region). Lines of constant mass ratio $q \equiv m_1/m_2$ are shown for $q = \{2, 4, 8\}$. Right panel: The mass M_f and dimensionless spin magnitude a_f of the final black holes. The differences with Fig. 9 above are practically negligible.

it validates both the WF model hereby presented as well as the data analysis scheme and sampler implemented.¹³ There are, however, differences that are worth mentioning. The most striking one is that *both* MLGW models tend to recover slightly larger masses and effective spin variable χ_{eff} than what published in Ref. [42]. The reason for this discrepancy is probably related to the fact that Ref. [42] does not use spin-aligned waveform models but rather relies the analysis

¹³However, a full validation of the algorithm is presented in Ref. [48].

on the precessing models SEOBNRv3 and IMRPhenomPv2. Although the differences are, in general, negligible, still they highlight the differences in the physical input of the waveform approximants. By contrast, it is remarkable the excellent agreement between the two waveform models, although the physical input and the analytical structure of the two models are rather different, especially in the spin sector [52]. In this respect, we also note in passing that the spin posteriors of GW151012 have most of the support in the region when s_1 and s_2 are nonzero. This also reflects in the rather large value of $\chi_{\text{eff}} \sim 0.53$, about one order of magnitude larger than the result of Ref. [42].

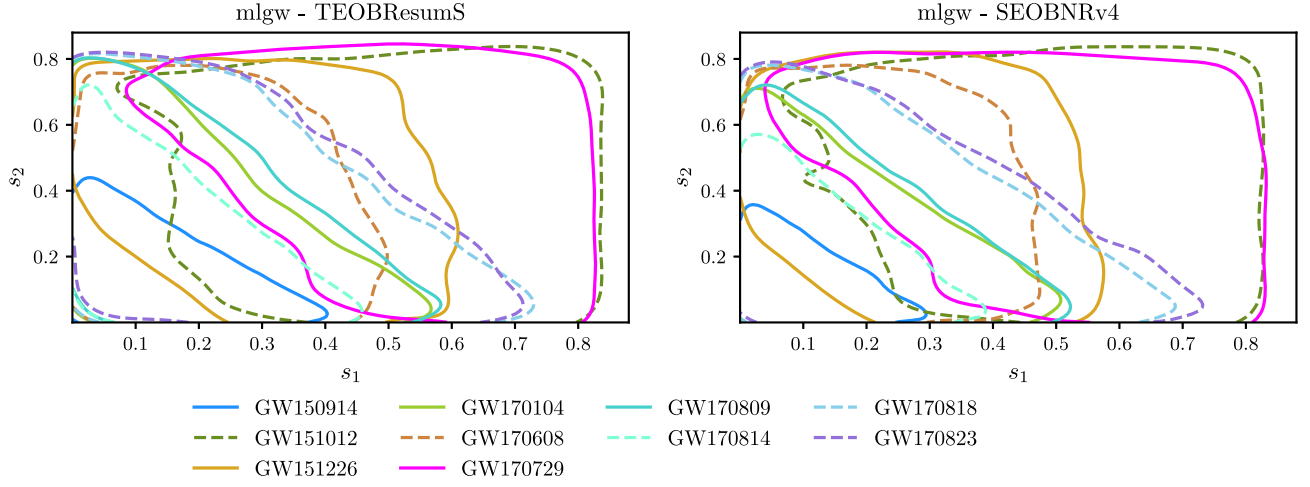


FIG. 11. Posterior probability densities of the component, dimensionless, spins for all the BBH systems in GWTC-1 obtained using MLGW-TEOBResumS [8] or MLGW-SEOBNRv4 [23]. The contours enclose the 90% credible regions.

VI. FINAL REMARKS AND FUTURE PROSPECTS

We built a ready-to-use machine learning model which generates the (dominant quadrupole) time-domain gravitational wave signal from a binary black hole coalescence in the nonprecessing case. The code is released as the package MLGW, which is publicly available at Ref. [53] and can be installed with the command `pip install MLGW`. The model consists of a PCA model to reduce the dimensionality of the $\ell = |m| = 2$ mode (decomposed in amplitude and phase). A regression (a MoE model) is performed to infer a relation from the orbital parameters to the low-dimensional representation of the WF.

It is important to stress that our model is very simple (i.e., it has a very small number of trainable parameters and it is not expensive to train) and flexible (i.e., it works for a large range of parameters and for long waveforms). In Ref. [27], an ML model for GW generation is built, with similar performances. However, the reduced dimension space is considerably larger [$O(200)$] than ours [$O(10)$]. In Ref. [18], the low-dimensional space has a similar dimension $O(30)$ and the ML model achieves a similar performance in the time execution (when computed on a CPU). They manage to achieve a better accuracy $O(2 \times 10^{-5})$, but they generate significantly shorter WFs ($f_{\min} = 15$ Hz for $M = 60 M_{\odot}$ against $f_{\min} \simeq 2.5$ Hz for MLGW) and need a larger training time [$O(6$ h) against $O(6$ min)].

Remarkably, we discovered that a PCA is able to reproduce a high-dimensional wave using a small number of variables. On the other hand, the MoE model is currently the “bottleneck” of the model accuracy. For this reason, we explored several alternative regression methods, including neural networks, but none of them showed dramatically better performances: Perhaps much more computational power and a larger training set are required to improve any better.

Despite this, our model shows excellent agreement with the underlying training set. At test time, the median

mismatch is $\mathcal{F}_m \sim 5 \times 10^{-4}$. Furthermore, a single WF generations takes 0.1–5 ms (depending mostly on the number of grid points required by the user), which is a factor of ~ 40 faster than TEOBResumS and ~ 250 faster than SEOBNRv4. Interestingly, MLGW matches the performances of a ROM, which is currently close to the state-of-the-art for quick generation of waveforms.

The model outputs WFs in the time domain. Of course, a similar approach can be applied to WFs in the frequency domain: This might further speed up the parameter estimation, as the FFT would not be required. A future update to include WFs in the frequency domain is in program.

Our ML framework allows for several generalizations, which might build a more accurate WF generator. First of all, it is quite straightforward to include higher-order modes in the WF computation. Different regressions, each for each mode, might be done as we already did for the $\ell = m = 2$ mode. A future update of MLGW along this direction is currently under way. Second, also the precession effects might be included in the model. The precession dynamics could be inserted as a single spin parameter s_P [54], and the WF dependence on s_P can be fitted together with the other orbital parameters.

Furthermore, our model could be trained on the publicly available NR waveforms catalogs (see, e.g., [3,4,55]), and it would provide the best generalization of the numerical waveform, dispensing with the EOB models altogether. Unfortunately, at the moment there are too few NR waveforms [$O(10^2)$] available to perform a reliable training: As discussed above in Sec. IV A, at least $O(5 \times 10^3)$ waveforms are needed—the improvement shall wait until enough NR waveforms are available. Moreover, NR waveforms are too short to be used as they are, and an extension (e.g., by hybridization with EOB waveforms) toward the early inspiral is needed to compute any kind of NR-based ML model.

Lastly, we expect our ML approach to work for every kind of source for which a training set of waveforms is available. Machine learning models to generate WFs might be crucial in the future, where signals from a number of different sources are expected to be detected. In that scenario, a parameter estimation must be able to detect among different sources, and this will require a lot of computational work. Speed up will be more pressing.

Our work opens up interesting opportunities in GW data analysis (searches and parameter estimation), both because of its speed and because of the closed form expression for the WF.

Because of its speed, MLGW could be employed for a systematic comparison between different waveform models, directly on data. By training (and the training procedure is also quick) MLGW with different waveform models, it will be possible to compare their predictions on several observed events. This could allow one to detect systematic biases or to prefer a model over another by means of Bayesian model selection (i.e., by comparing different model evidences). We started this program by analyzing GWTC-1 with MLGW-TEOBResumS and MLGW-SEOBNRv4 and highlighted some differences in the predictions as compared to the published results (see Sec. V). Future work might repeat such an analysis on other EOB models or with more observations.

Furthermore, as shown in Fig. 7, the model is most useful whenever a long waveform is required: In such a case, the speed up gets even more substantial. This is crucial for the detection of low-frequency signals, as is the case for ET. The analysis of such signals can be performed in the same time required to deal with shorter signals: It will become feasible, even with a small amount of resources and *without any loss of WF quality*.

A closed form expression for the gradients of the waveform with respect to the orbital parameters (already included in the MLGW package) could give an advantage on the parameter estimation procedure by using the Hamiltonian Monte Carlo (HM). HM [56,57] is a variant

of Markov chain Monte Carlo, which employs the gradient of the likelihood (dependent on the gradient of the waveform) to perform an effective sampling of the posterior distribution. The sampling chain converges faster to the steady state by “finding quickly” the high-density regions, thus offering a speed up of the PE.

Another option, so far never explored, is to use the gradients of the WF for a fast exploration of the likelihood landscape. With any gradient-based optimizer, it should be easy to jump to a *local* maximum of the likelihood. Such information might be helpful to reliably locate a *global* maximum of the likelihood. Such ability could speed up the searches as well as the parameter estimation.

In conclusion, we presented MLGW, an off-the-shelf machine learning model for gravitational waves signals from BBHs. We demonstrated that MLGW is fast, accurate, and easy to train and to use. We anticipate that MLGW will enable studies hitherto unfeasible due to the lack of fast and easy to use models.

ACKNOWLEDGMENTS

R. G. acknowledges support from the Deutsche Forschungsgemeinschaft (DFG) under Grant No. 406116891 within the Research Training Group RTG 2522/1. M. B. and S. B. acknowledge support by the EU H2020 under ERC Starting Grant, No. BinGraSp-714626. M. B. acknowledges support from the Deutsche Forschungsgemeinschaft (DFG) under Grant No. 406116891 within the Research Training Group RTG 2522/1. This research has made use of data, software, and/or web tools obtained from the Gravitational Wave Open Science Center, a service of LIGO Laboratory, the LIGO Scientific Collaboration, and the Virgo Collaboration. LIGO is funded by the U.S. National Science Foundation. Virgo is funded by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale della Fisica Nucleare (INFN), and the Dutch Nikhef, with contributions by Polish and Hungarian institutes.

- [1] J. Aasi *et al.*, Parameter estimation for compact binary coalescence signals with the first generation gravitational-wave detector network, *Phys. Rev. D* **88**, 062001 (2013).
- [2] J. Veitch *et al.*, Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library, *Phys. Rev. D* **91**, 042003 (2015).
- [3] M. Boyle *et al.*, The SXS Collaboration catalog of binary black hole simulations, *Classical Quantum Gravity* **36**, 195006 (2019).

- [4] J. Healy, C. O. Lousto, J. Lange, R. O’Shaughnessy, Y. Zlochower, and M. Campanelli, Second RIT binary black hole simulations catalog and its application to gravitational waves parameter estimation, *Phys. Rev. D* **100**, 024021 (2019).
- [5] J. Healy and C. O. Lousto, The third RIT binary black hole simulations catalog, *Phys. Rev. D* **102**, 104018 (2020).
- [6] A. Buonanno and T. Damour, Transition from inspiral to plunge in binary black hole coalescences, *Phys. Rev. D* **62**, 064015 (2000).

- [7] T. Damour and A. Nagar, An improved analytical description of inspiralling and coalescing black-hole binaries, *Phys. Rev. D* **79**, 081503 (2009).
- [8] A. Nagar, G. Riemenschneider, G. Pratten, P. Retegno, and F. Messina, A multipolar effective one body waveform model for spin-aligned black hole binaries, *Phys. Rev. D* **102**, 024077 (2020).
- [9] D. Chiamarello and A. Nagar, Faithful analytical effective-one-body waveform model for spin-aligned, moderately eccentric, coalescing black hole binaries, *Phys. Rev. D* **101**, 101501 (2020).
- [10] S. Ossokine *et al.*, Multipolar effective-one-body waveforms for precessing binary black holes: Construction and validation, *Phys. Rev. D* **102**, 044055 (2020).
- [11] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Pürrer, X. Jiménez Forteza, and A. Bohé, Frequency-domain gravitational waves from nonprecessing black-hole binaries. II. A phenomenological model for the advanced detector era, *Phys. Rev. D* **93**, 044007 (2016).
- [12] G. Pratten *et al.*, Let's twist again: Computationally efficient models for the dominant and sub-dominant harmonic modes of precessing binary black holes, [arXiv:2004.06503](https://arxiv.org/abs/2004.06503).
- [13] H. Estellés, A. Ramos-Buades, S. Husa, C. García-Quirós, M. Colleoni, L. Haegel, and R. Jaume, IMRPhenomTP: A phenomenological time domain model for dominant quadrupole gravitational wave signal of coalescing binary black holes, [arXiv:2004.08302](https://arxiv.org/abs/2004.08302).
- [14] V. Varma, S.E. Field, M.A. Scheel, J. Blackman, L.E. Kidder, and H.P. Pfeiffer, Surrogate model of hybridized numerical relativity binary black hole waveforms, *Phys. Rev. D* **99**, 064045 (2019).
- [15] D. Williams, I.S. Heng, J. Gair, J.A. Clark, and B. Khamesra, A precessing numerical relativity waveform surrogate model for binary black holes: A Gaussian process regression approach, *Phys. Rev. D* **101**, 063011 (2020).
- [16] V. Varma, S.E. Field, M.A. Scheel, J. Blackman, D. Gerosa, L.C. Stein, L.E. Kidder, and H.P. Pfeiffer, Surrogate models for precessing binary black hole simulations with unequal masses, *Phys. Rev. Research* **1**, 033015 (2019).
- [17] N.E. Rifat, S.E. Field, G. Khanna, and V. Varma, Surrogate model for gravitational wave signals from comparable and large-mass-ratio black hole binaries, *Phys. Rev. D* **101**, 081502 (2020).
- [18] S. Khan and R. Green, Gravitational-wave surrogate models powered by artificial neural networks: The ANN-Sur for waveform generation, [arXiv:2008.12932](https://arxiv.org/abs/2008.12932).
- [19] T. Damour and A. Nagar, Faithful effective-one-body waveforms of small-mass-ratio coalescing black-hole binaries, *Phys. Rev. D* **76**, 064028 (2007).
- [20] T. Damour and A. Nagar, Effective one body description of tidal effects in inspiralling compact binaries, *Phys. Rev. D* **81**, 084016 (2010).
- [21] A. Nagar and P. Retegno, Efficient effective one body time-domain gravitational waveforms, *Phys. Rev. D* **99**, 021501 (2019).
- [22] M. Pürrer, Frequency domain reduced order model of aligned-spin effective-one-body waveforms with generic mass-ratios and spins, *Phys. Rev. D* **93**, 064041 (2016).
- [23] A. Bohé *et al.*, Improved effective-one-body model of spinning, nonprecessing binary black holes for the era of gravitational-wave astrophysics with advanced detectors, *Phys. Rev. D* **95**, 044028 (2017).
- [24] M. Pürrer, R. Smith, S. Field, P. Cañizares, V. Raymond, J. Gair, and M. Hannam, Accelerating parameter estimation of gravitational waves from black hole binaries with reduced order quadratures, in *Proceedings of the 14th Marcel Grossmann Meeting on Recent Developments in Theoretical and Experimental General Relativity, Astrophysics, and Relativistic Field Theories* (2017), Vol. 2, pp. 2015–2018.
- [25] B.D. Lackey, M. Pürrer, A. Taracchini, and S. Marsat, Surrogate model for an aligned-spin effective one body waveform model of binary neutron star inspirals using Gaussian process regression, *Phys. Rev. D* **100**, 024002 (2019).
- [26] R. Cotesta, S. Marsat, and M. Pürrer, Frequency domain reduced order model of aligned-spin effective-one-body waveforms with higher-order modes, *Phys. Rev. D* **101**, 124040 (2020).
- [27] A. J. Chua, C. R. Galley, and M. Vallisneri, Reduced-Order Modeling with Artificial Neurons for Gravitational-Wave Inference, *Phys. Rev. Lett.* **122**, 211101 (2019).
- [28] H. Gabbard, M. Williams, F. Hayes, and C. Messenger, Matching Matched Filtering with Deep Networks for Gravitational-Wave Astronomy, *Phys. Rev. Lett.* **120**, 141103 (2018).
- [29] D. George and E. Huerta, Deep learning for real-time gravitational wave detection and parameter estimation: Results with advanced LIGO data, *Phys. Lett. B* **778**, 64 (2018).
- [30] D. George and E. Huerta, Deep learning for real-time gravitational wave detection and parameter estimation with LIGO data, in *NIPS Summer School 2017 (Neural Information Processing Systems, Long Beach, CA, 2017)*.
- [31] A. Rebei, E. Huerta, S. Wang, S. Habib, R. Haas, D. Johnson, and D. George, Fusing numerical relativity and deep learning to detect higher-order multipole waveforms from eccentric binary black hole mergers, *Phys. Rev. D* **100**, 044025 (2019).
- [32] C. Chatterjee, L. Wen, K. Vinsen, M. Kovalam, and A. Datta, Using deep learning to localize gravitational wave sources, *Phys. Rev. D* **100**, 103025 (2019).
- [33] K.W. Wong, K.K. Ng, and E. Berti, Gravitational-wave signal-to-noise interpolation via neural networks, [arXiv:2007.10350](https://arxiv.org/abs/2007.10350).
- [34] A. Khan, E. Huerta, and A. Das, Physics-inspired deep learning to characterize the signal manifold of quasi-circular, spinning, non-precessing binary black hole mergers, *Phys. Lett. B* **808**, 135628 (2020).
- [35] Y. Setyawati, M. Pürrer, and F. Ohme, Regression methods in waveform modeling: A comparative study, *Classical Quantum Gravity* **37**, 075012 (2020).
- [36] S.E. Field, C.R. Galley, J.S. Hesthaven, J. Kaye, and M. Tiglio, Fast Prediction and Evaluation of Gravitational Waveforms Using Surrogate Models, *Phys. Rev. X* **4**, 031006 (2014).
- [37] A. Nagar *et al.*, Time-domain effective-one-body gravitational waveforms for coalescing compact binaries with nonprecessing spins, tides and self-spin effects, *Phys. Rev. D* **98**, 104052 (2018).

- 999 [38] A. Nagar, G. Pratten, G. Riemenschneider, and R. Gamba, A
1000 multipolar effective one body model for non-spinning black
1001 hole binaries, *Phys. Rev. D* **101**, 024041 (2020).
1002 [39] T. Damour and A. Nagar, A new analytic representation of
1003 the ringdown waveform of coalescing spinning black hole
1004 **1** binaries, *Phys. Rev. D* **90**, 024054 (2014).
1005 [40] G. Riemenschneider, P. Retegno, and A. Nagar (to be
1006 **2** published).
1007 [41] A. Taracchini, A. Buonanno, Y. Pan, T. Hinderer, M. Boyle
1008 *et al.*, Effective-one-body model for black-hole binaries
1009 with generic mass ratios and spins, *Phys. Rev. D* **89**, 061502
1010 (2014).
1011 [42] B. P. Abbott *et al.*, GWTC-1: A Gravitational-Wave Tran-
1012 sient Catalog of Compact Binary Mergers Observed by
1013 LIGO and Virgo during the First and Second Observing
1014 Runs, *Phys. Rev. X* **9**, 031040 (2019).
1015 [43] P. Ajith, M. Boyle, D. A. Brown, S. Fairhurst, M. Hannam,
1016 I. Hinder, S. Husa, B. Krishnan, R. A. Mercer, F. Ohme,
1017 C. D. Ott, J. S. Read, L. Santamaria, and J. T. Whelan,
1018 Data formats for numerical relativity waves, [arXiv:0709](https://arxiv.org/abs/0709.0093)
1019 [.0093](https://arxiv.org/abs/0709.0093).
1020 [44] C. Devine, Z. B. Etienne, and S. T. McWilliams, Optimizing
1021 spinning time-domain gravitational waveforms for ad-
1022 vanced LIGO data analysis, *Classical Quantum Gravity*
1023 **33**, 125025 (2016).
1024 [45] K. Murphy, *Machine Learning: A Probabilistic Perspective*,
1025 Adaptive Computation and Machine Learning Series (MIT
1026 Press, Cambridge, MA, 2012).
1027 [46] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton,
1028 Adaptive mixtures of local experts, *Neural Comput.* **3**, 79
1029 (1991).
[47] F. Ohme, A. B. Nielsen, D. Keppel, and A. Lundgren,
Statistical and systematic errors for gravitational-wave
inspiral signals: A principal component analysis, *Phys.*
Rev. D **88**, 042002 (2013).
[48] G. Pagano and W. D. Pozzo (to be published).
[49] J. Veitch, W. D. Pozzo, Cody, and M. Pitkin, ed1d1a8d,
johnveitch/cpnest: Minor optimisation (2017).
[50] N. Aghanim *et al.*, Planck 2018 results. VI. Cosmological
parameters, *Astron. Astrophys.* **641**, A6 (2020).
[51] S. Babak, A. Taracchini, and A. Buonanno, Validating the
effective-one-body model of spinning, precessing binary
black holes against numerical relativity, *Phys. Rev. D* **95**,
024010 (2017).
[52] P. Retegno, F. Martinetti, A. Nagar, D. Bini, G. Riemensch-
neider, and T. Damour, Comparing Effective One Body
Hamiltonians for spin-aligned coalescing binaries, *Phys.*
Rev. D **101**, 104027 (2020).
[53] <https://pypi.org/project/mlgw/>
[54] P. Schmidt, F. Ohme, and M. Hannam, Towards models of
gravitational waveforms from generic binaries: Modelling
precession effects with a single effective precession param-
eter, *Phys. Rev. D* **91**, 024043 (2015).
[55] A. H. Mroue, M. A. Scheel, B. Szilagyi, H. P. Pfeiffer, M.
Boyle *et al.*, A Catalog of 174 Binary Black-Hole Simu-
lations for Gravitational-Wave Astronomy, *Phys. Rev. Lett.*
111, 241104 (2013).
[56] M. Betancourt, A conceptual introduction to Hamiltonian
Monte Carlo, [arXiv:1701.02434](https://arxiv.org/abs/1701.02434).
[57] E. K. Porter and J. Carré, A Hamiltonian Monte-Carlo
method for Bayesian inference of supermassive black hole
binaries, *Classical Quantum Gravity* **31**, 145004 (2014).