Accelerating multimodal gravitational waveforms from precessing compact binaries with artificial neural networks

Lucy M. Thomas, Geraint Pratten, and Patricia Schmidt

Phys. Rev. D **106**, 104029 — Published 15 November 2022

DOI: 10.1103/PhysRevD.106.104029

Accelerating multimodal gravitational waveforms from precessing compact binaries with artificial neural networks

Lucy M. Thomas [1,*] Geraint Pratten [1,†] and Patricia Schmidt [1,‡]

[1] *School of Physics and Astronomy and Institute for Gravitational Wave Astronomy, University of Birmingham, Edgbaston, Birmingham, B15 2TT, United Kingdom*

(Dated: October 5, 2022)

Gravitational waves from the coalescences of black hole and neutron stars afford us the unique opportunity to determine the sources' properties, such as their masses and spins, with unprecedented accuracy. To do so, however, theoretical models of the emitted signal that are i) extremely accurate and ii) computationally highly efficient are necessary. The inclusion of more detailed physics such as higher-order multipoles and relativistic spin-induced orbital precession increases the complexity and hence also computational cost of waveform models, which presents a severe bottleneck to the parameter inference problem. A popular method to generate waveforms more efficiently is to build a fast surrogate model of a slower one. In this paper, we show that traditional surrogate modelling methods combined with artificial neural networks can be used to build a computationally highly efficient while still accurate emulation of multipolar time-domain waveform models of precessing binary black holes. We apply this method to the state-of-the-art waveform model `SEOBNRv4PHM` and find significant computational improvements: On a traditional CPU, the typical generation of a single waveform using our neural network surrogate `SEOBNN_v4PHM_4dq2` takes 18ms for a binary black hole with a total mass of $44M_\odot$ when generated from 20Hz. In comparison to `SEOBNRv4PHM` itself, this amounts to an improvement in computational efficiency by two orders of magnitude. Utilising additional GPU acceleration, we find that this speed-up can be increased further with the generation of batches of waveforms simultaneously. Even without additional GPU acceleration, this dramatic decrease in waveform generation cost can reduce the inference timescale from weeks to hours.

I. INTRODUCTION

Since the first observing run of the currently operating ground-based gravitational-wave (GW) detector network consisting of Advanced LIGO [1] and Advanced Virgo [2], and soon also KAGRA [3], detections of GWs from more than 90 compact binary inspirals and mergers consisting of black holes and neutron stars [4–7] have been announced. These observations have a transformative impact on our understanding of the properties of black holes and neutron stars, allowing us to determine their mass and spin distributions [8] and to put them into their astrophysical context.

The inference of the source properties is predicated on the availability of accurate theoretical models of the emitted GW signal through inspiral, merger and ringdown (IMR). Recent years have seen much progress in the improvement of such waveform models through the inclusion of higher-order multipoles and spin-induced precession of the orbital plane. These improvements come, however, at the expense of computational efficiency. Fast model evaluation speeds are a necessary requirement for key analysis such as Bayesian inference where on average $10^6 - 10^8$ such model evaluations are needed to obtain well-sampled posterior probability distributions. Different strategies are employed to make waveform models computationally more efficient, some prominent ones include: Phenomenological ansatzes [9]; reduced-order surrogate modelling [10–16]; dimensional reduction [17, 18]; and most recently the incorporation of machine learning into reduced-order models [19]. Alternatively, routes to either accelerate the likelihood evaluation directly [20–27] or to perform likelihood-free inference [28, 29] have also been developed.

In this work we combine reduced-order modelling with the power of artificial neural networks (ANNs) to build a computationally vastly more efficient surrogate model of the state-of-the-art inspiral-merger-ringdown (IMR) waveform model `SEOBNRv4PHM` [30] that includes both spin-induced orbital precession [31] and higher-order modes beyond the quadrupole emission. While the efficacy of this approach has previously been demonstrated for the quadrupole ($(2,2)$-) mode of aligned-spin binary black holes (BBHs) [19, 32], here we demonstrate its feasibility for the multimodal, precessing case. To achieve this, we decompose the `SEOBNRv4PHM` waveform model into eight components that describe the modes in a non-inertial, co-prcessing coordinate frame and three components that encode the precession dynamics. Using a combination of traditional surrogate modelling steps and neural networks to produce parameter fits, we build a fast surrogate model for each component. Using extensive optimisation we determine an optimal network for each component, which allows us to speed up the model evaluation by a factor of a few hundred on average on a CPU and even further on a GPU, demonstrating the efficacy of this approach for state-of-art multimodal waveforms

* lthomas@star.sr.bham.ac.uk
† G.Pratten@bham.ac.uk
‡ P.Schmidt@bham.ac.uk

with precession.

The paper is organised as follows: First in Sec. II we introduce the methodology behind this work, including a brief overview of surrogate waveform modelling in Sec. II A, details of precessing waveform decomposition in Sec. II B, and details of the mismatch metrics we use to assess the accuracy of our model in Sec. II C. We then detail the construction of the model in Sec. III, first describing the training data upon which the model is built in Sec. III A, the reduced basis and empirical interpolant construction in Sec. III B, and the neural networks for each of the model components in Sec. III C, both coprecessing modes and Euler angles, putting these together to describe the full surrogate model construction in Sec. III D. We then discuss the completed model in Sec. IV, both the accuracy of the waveforms generated in Sec. IV A and the timing of the model evaluation in Sec. IV B. Finally, we summarise the model construction and results in Sec. V, also discussing caveats and further work.

Throughout this paper we use geometric units, $G = c = 1$, unless stated otherwise.

II. METHODOLOGY

A. Surrogate Modelling

Surrogate models are fast, accurate approximations to an underlying (slower) model, over a chosen parameter space region. Therefore, their range of validity in parameter space is limited to the region over which they are constructed, the training space, plus an extrapolation region over which the model has been tested and shown to be accurate to within some tolerance. Recent examples of surrogate models for waveforms from coalescing compact binaries include Numerical Relativity (NR) and NR-hybrid surrogate models [13, 14, 16, 33–35], surrogates for the aligned-spin effective-one-body (EOB) model `SEOBNRv4` [36] using artificial neural networks [19, 32] and a machine learning emulation of a different EOB model, TEOBResumS [37, 38].

In the following, we will provide a brief outline of the main steps for building a surrogate model. For a more complete explanation we refer the reader to e.g. [11].

The process of building a surrogate model may begin with building a *reduced basis*, which enables us to represent any arbitrary function, e.g. a time-domain waveform $h(t, \vec{\lambda})$ with intrinsic parameters $\vec{\lambda}$, within the discrete training space $\mathcal{T}_M = \{\vec{\lambda}_i\}_{i=1}^{M} \subseteq \mathcal{T} = \{\vec{\lambda}_i\}_{i=1}^{\infty}$ as a linear combination of an n-dimensional orthonormal basis $\{\hat{e}_i(t)\}_{i=1}^{n}$ and projection coefficients $\{c_n(\vec{\lambda})\}_{i=1}^{n}$,

$$h(t, \vec{\lambda}) \approx \sum_{i=1}^{n} c_i(\vec{\lambda})\hat{e}_i(t), \tag{1}$$

with $n \leq M$. The reduced basis is constructed recursively using a greedy algorithm [10, 11] until all waveforms in the training space $\mathcal{T}_M$ are represented by the basis to within a certain tolerance σ, which is related to the representation error ϵ by

$$\max \epsilon = \max \left\| h(t; \vec{\lambda}) - \sum_{i=1}^{n} c_i(\vec{\lambda})\hat{e}_i(t) \right\|^2 \leq \sigma, \tag{2}$$

where $||*||$ denotes the L^2-norm, which we compute via the Chebyshev-Gauss quadrature rule. To achieve this, at each step the waveform with the largest representation error using the current basis is chosen, orthogonalised with respect to all current basis elements, and normalised, before being added to the basis as the next basis element. The greedy algorithm stops once Eq (2) is fulfilled or if the waveform with the largest representation error is already a basis element. The latter is an indication that the training space $\mathcal{T}_M$ is sampled too coarsely to achieve the desired accuracy σ. If the discrete training space is sampled sufficiently densely, then the reduced basis representation allows us to approximate *any* waveform in the entire parameter space $\mathcal{T}$.

After the basis has been constructed, we proceed to build an *empirical interpolant* (EI) using the empirical interpolation method [39, 40], which allows us to reconstruct each waveform $h(t; \vec{\lambda})$ for $\vec{\lambda} \in \mathcal{T}_M$ to within a high accuracy, only using information at certain (sparse) time nodes $\{T_i\}_{i=1}^{n}$. These carefully selected empirical times or nodes are determined exclusively by the reduced basis waveforms, and the number of time nodes will be equal to the number of waveforms within the reduced basis:

$$\mathrm{EI}[h](t; \vec{\lambda}) = \sum_{j=1}^{n} B_j(t) h(T_j; \vec{\lambda}), \tag{3}$$

$$= \sum_{i=1}^{n}\sum_{j=1}^{n} \hat{e}_i(t)(V^{-1})_{ij} h(T_j; \vec{\lambda}), \tag{4}$$

where $(V)_{ij} = (\hat{e}_i(T_j))$ is the interpolation matrix.

The final step for building a surrogate model is to perform a parameter space fit which allows us to predict waveforms at the empirical times $\{T_i\}_{i=1}^{n}$ for arbitrary parameters $\vec{\lambda} \in \mathcal{T}$ based on the greedy points $\{\vec{\lambda}_i\}_{i=1}^{n}$ selected to construct the reduced basis. This requires us to fit $h(t; \vec{\lambda})$ across the parameter space at each empirical node such that

$$h(T_i; \vec{\lambda}) \approx A_i(\vec{\lambda}) e^{i\phi_i(\vec{\lambda})}, \tag{5}$$

where A_i and ϕ_i are the amplitude and phase at the i-th empirical node. The $2n$-functions that determine the parameter space fits can be determined by different means, for example via traditional fitting functions such as splines or polynomials [10, 13, 16] or by using machine learning algorithms such artificial neural networks [19, 32] or Gaussian processes [33, 34, 41, 42]. In this work, we will follow Ref. [19] and use ANNs to determine the fitting coefficients $A_i(\vec{\lambda})$ and $\phi_i(\vec{\lambda})$. The final surrogate

model for a waveform $h(t;\vec{\lambda})$ is then given by

$$h^S(t;\vec{\lambda}) \equiv \sum_{i=1}^{n}\sum_{j=1}^{n}(V^{-1})_{ji}\hat{e}_j(t)A_i(\vec{\lambda})e^{-i\phi_i(\vec{\lambda})}. \quad (6)$$

We note that this prescription applies to generic functions up to the parameter space fits Eq. (5), whose RHS decomposition depends on the function that is being modelled. In subsequent sections, we will follow this approach for individual waveform modes decomposed into amplitude and phase as well as angle functions.

To construct the reduced bases and empirical interpolants we use the publicly available `Python` package `RomPy` [11, 43].

B. Waveform Decomposition

Binary black holes on quasi-spherical orbits span a seven-dimensional (intrinsic) parameter space characterised by the mass ratio $q = m_1/m_2 \geq 1$ and the (dimensionless) spin angular momenta $\vec{\chi}_1$ and $\vec{\chi}_2$. If the spin angular momenta are misaligned with the direction of the instantaneous orbital angular momentum $\hat{L}(t)$, then spin-induced precession occurs [31, 44]. This causes the orbital plane to change its spatial orientation as the binary inspirals due to GW emission. This more complex two-body dynamics leads to amplitude and phase modulations of the emitted GW signal $h(t;\vec{\lambda})$ and is also a source of the excitation of higher-order multipoles, $h_{\ell m}$, in the radiation field, which must be included to accurately describe the GW signal:

$$h(t;\vec{\lambda};\theta,\varphi) = \sum_{\ell=2}^{\infty}\sum_{m=-\ell}^{\ell} h_{\ell m}(t;\vec{\lambda})\,{}^{-2}Y_{\ell m}(\theta,\varphi), \quad (7)$$

where (θ,φ) denote the angles on the unit sphere. Due to the increased complexity, modelling the signal from precessing BBHs is a challenging task but is accomplished as follows [31, 45–47]: The GW modes from precessing binaries, $h^P_{\ell m}(t;\vec{\lambda})$, can be conveniently decomposed into a simpler carrier signal corresponding to a non-inertial coprecessing observer, $h^{\rm co-prec}_{\ell m}(t;\vec{\lambda})$, and a time-dependent rotation operator $\mathbf{R}$ which encodes the orbital precession dynamics, i.e.

$$h^P_{\ell m}(t;\vec{\lambda}) = \sum_{m'=-\ell}^{\ell} \mathbf{R}_{\ell m m'}(t;\vec{\lambda})h^{\rm co-prec}_{\ell m'}(t;\vec{\lambda}), \quad (8)$$

where $\vec{\lambda}$ denotes the binary's intrinsic parameters.

As a first approximation, the coprecessing waveform modes can be approximated by aligned-spin modes [46, 47]. This simplifying approximation is made in many of the state-of-the-art waveform models [30, 48, 49] and is a known source of modelling errors [50]. Importantly, this approximation assumes a conjugate symmetry between the $+m$ and $-m$ modes, which no longer holds in the case of precessing binaries [47, 51]. The waveform model we emulate here, `SEOBNRv4PHM` [30, 52], contains the $(2,\pm2),(2,\pm1),(3,\pm3),(4,\pm4)$ and $(5,\pm5)$ coprecessing modes defined in a time-dependent coordinate frame that tracks $\hat{L}(t)$ (L-frame), and assumes conjugate mode symmetry, i.e.,

$$h^{\rm co-prec}_{\ell,-m}(t;\vec{\lambda}) = (-1)^{\ell}h^{\rm co-prec*}_{\ell m}(t;\vec{\lambda}). \quad (9)$$

Therefore, we only model the positive $m-$modes in the coprecessing frame and obtain the $-m$-modes via Eq. (9). The coprecessing waveform modes are then further decomposed into amplitude and phase,

$$h^{\rm co-prec}_{\ell m}(t;\vec{\lambda}) = A_{\ell m}(t;\vec{\lambda})e^{i\phi_{\ell m}(t;\vec{\lambda})}. \quad (10)$$

For the rotation operator we will use its SO(3) representation and model the three Euler angles $\alpha(t;\vec{\lambda}), \beta(t;\vec{\lambda})$ and $\gamma(t;\vec{\lambda})$ in an inertial Cartesian coordinate frame that is aligned with the total angular momentum at the the initial time t_0, i.e. $\mathbf{J}(t_0) = \hat{z}$, as shown in Fig. 1, henceforth referred to as the J-frame.

Utilising this decomposition, we build (i) ANN surrogates for the amplitude and phase of each coprecessing positive $m-$mode contained in `SEONBNRv4PHM` (see Sec. III C 1) and (ii) ANN surrogates for the three Euler angles (see Sec. III C 2).

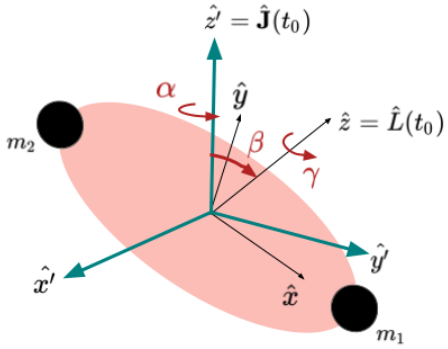

FIG. 1. Definition of the inertial J-frame and the Euler angles. The three Euler angles α, β and γ define the rotation from the L-frame where the z-component is parallel to the orbital angular momentum of the binary $\hat{z} = \hat{L}(t_0)$, to the J-frame where it is parallel to the total angular momentum $\hat{z}' = \hat{\mathbf{J}}(t_0)$ at the start time of the waveform t_0. The angle γ is defined by the minimal rotation condition, $\dot{\gamma} = -\dot{\alpha}\cos\beta$ [45, 53].

C. Mismatch

To determine the agreement between the input waveform model `SEOBNRv4PHM` and its ANN emulation, we will

use the mismatch as the metric to assess the accuracy of the various elements in the surrogate modelling process.

To quantify the agreement between two coprecessing modes/Euler angles/polarisations, we will employ the (frequency-domain, noise-weighted) inner product or match $\mathcal{M}_f$ optimised over a time and phase shift, given as

$$\mathcal{M}_f(h_1,h_2) = \max_{t_c,\phi_0} \frac{\langle h_1,h_2\rangle}{\sqrt{\langle h_1,h_1\rangle\langle h_2,h_2\rangle}}, \tag{11}$$

where the inner product is defined as

$$\langle h_1,h_2\rangle = 4\Re\int_{f_{\rm min}}^{f_{\rm max}} \frac{\tilde{h}_1(f)\tilde{h}_2^*(f)}{S_n(|f|)}\,df, \tag{12}$$

with $S_n(f)$ the one-sided power spectral density (PSD) of the detector noise, $\tilde{h}$ indicates the Fourier transform of h, and $\tilde{h}^*$ the complex conjugate. The *mismatch* can now be defined as

$$\bar{\mathcal{M}}_f(h_1,h_2) \equiv 1 - \mathcal{M}_f(h_1,h_2). \tag{13}$$

We will also find it convenient to introduce a normalized waveform $\hat{h} = h/\sqrt{\langle h,h\rangle}$.

When using a white-noise PSD, i.e. independent of frequency, it is convenient to define a time-domain overlap

$$\langle h_1,h_2\rangle_t = \Re\int_{t_{\rm min}}^{t_{\rm max}} h_1(t)h_2^*(t)\,dt, \tag{14}$$

with an inherited norm $\|h\|_t^2 = \langle h,h\rangle_t$. We can then define an analogous time domain match as

$$\mathcal{M}_t(h_1,h_2) = \frac{\langle h_1,h_2\rangle_t}{\|h_1\|_t\,\|h_2\|_t}, \tag{15}$$

and the associated *mismatch* as $\bar{\mathcal{M}}_t = 1-\mathcal{M}_t$.

III. MODEL

In this section we describe the construction of our surrogate model. As described in Sec. II B, we model the coprecessing modes and Euler angles separately. We detail the training dataset upon which the model is built in Sec. III A, as well as the validation and test datasets. In Sec. III B we describe the construction of the reduced bases and empirical interpolants for each component, and in Secs. III C 1 and III C 2 we describe the neutral network architecture and training for the coprecessing modes and Euler angles respectively, assessing the accuracy of each component. We will then consider the accuracy and speed of the complete model to produce a fully precessing signal in Sec. IV.

A. Training, Validation and Testing Data

1. *Waveforms*

Our waveform training dataset consists of 2×10^5 multipolar `SEOBNRv4PHM` waveforms with mass ratios $q\in[1,2]$ and arbitrarily oriented spin on the primary black hole with magnitude $|\chi_1|\leq 0.8$[1]; the secondary black hole is nonspinning. Both the coprecessing waveform modes and the time-dependent Euler angles are obtained directly from the `SEOBNRv4PHM` implementation in the public LIGO Algorithm Library `LAL` [54], mitigating the need to perform any additional post-processing.

We first randomly sample 199,226 binaries from the reduced parameter space, drawing the parameters from distributions uniform in mass ratio q, uniform in spin magnitude $|\chi_1|$ and isotropic in spin orientation (θ_1,ϕ_1). We supplement these binaries with an additional 774 systematically chosen points to accurately sample the boundaries. The parameters of these systematically chosen binaries are listed in Tab. II C and a visualisation of the training set can be found in Fig. 11 in Appendix V.

We initially generate waveforms such that the $(2,2)$-mode of a binary with a total mass of $60\,M_\odot$ starts from an initial frequency of 4Hz. In geometrised units this corresponds to an approximate length of $\sim 2\times10^5 M$ before merger, though the duration varies due to mass ratio and inspiral spin [55, 56]. The modes are then aligned such that the peak of the quadrature of all modes occurs at $t=0M$. Each waveform is first generated on a uniform time grid with a time spacing $\Delta t = 0.1M$, and then reinterpolated onto a non-uniform grid which is 20 times coarser in the early inspiral, but retains the $0.1M$ spacing in the later-inspiral, merger and ringdown. The waveforms are of varying length pre- and post-merger, and so have different time grids of the same resolution, but are required to be of equal length and evaluated upon the same times in order to build the reduced bases. As a data preprocessing step we choose the waveform with the shortest length pre-merger, and reinterpolate all waveforms onto this common time grid. We then choose the waveform with the shortest post-merger signal and truncate all waveforms such that the final time matches that of the shortest. We note that this truncation is less than $10M$ for all waveforms, and contains a negligible amount of the ringdown signal in all cases. For computational reasons, we truncate all waveforms to be of length 10^4M pre-merger. Therefore all waveforms start at 10^4M before the peak, and include $110M$ of post-merger signal. We note that due to this truncation, the spin parameters are specified at the initial time $t\sim -2\times10^5M$ and not at

[1] We note that the spin orientation is defined relative to the orbital angular momentum $\hat{L}(t_0)$ at the initial time. Decomposed in Cartesian coordinates the spin vector is given by $\{\chi_{1x},\chi_{1y},\chi_{1z}\}$, where $\chi_{1z} = \vec{\chi}_1\cdot\hat{L}(t_0)$.

Training data subset		Number of binaries	q	$\lvert\chi_1\rvert$	θ_1 [rad]	ϕ_1 [rad]
Systematically sampled	Nonspinning	6	$[1, 1.2, 1.4, 1.6, 1.8, 2]$	$[0]$	-	-
	Spin-aligned	48	$[1, 1.2, 1.4, 1.6, 1.8, 2]$	$[0.2, 0.4, 0.6, 0.8]$	$[0, \pi]$	-
	Precessing	720	$[1, 1.2, 1.4, 1.6, 1.8, 2]$	$[0.2, 0.4, 0.6, 0.8]$	$[\pi/6, \pi/3, \pi/2, 2\pi/3, 5\pi/6]$	$[0, \pi/3, 2\pi/3, \pi, 4\pi/3, 5\pi/3]$
Randomly sampled		$199,226$	$U[1, 2]$	$U[0, 0.8]$	$U[0, \pi]$	$U[0, 2\pi]$

TABLE I. Parameters of the $200,000$ binaries which span our training dataset. The training space is split into two sectors: (i) a systematically sampled subset which is included to ensure coverage of the parameter space boundaries; (ii) a random but uniformly sampled subset. All spin parameters are quoted at a reference frequency of 4Hz for the $(2,2)$-mode for a binary with a total mass of $60M_\odot$

the start of the waveforms. Since the spins in precessing binaries evolve with time, it is necessary to define the reference time or frequency at which they are defined. Being able to do this for some arbitrary time/frequency requires either code (see e.g. [57]) or additional NNs that track the spin evolution. We leave building neural networks for the spin evolution for future work. The truncated waveforms are then re-interpolated onto a uniform grid with spacing $\Delta t = 1M$ in order to build the reduced bases and empirical interpolants, as we found that the finer $0.1M$ spacing was not required.

We note that when constructing the models for the coprecessing odd-m mode amplitudes and phases, not all of the 2×10^5 training waveforms are used. We first remove training points where there is very little spin or mass asymmetry in the system, as we expect the odd-m amplitudes to be small and therefore noisy in the true `SEOBNRv4PHM` data. We impose a cut of $q > 1.01$, $\chi_{1z} > 10^{-2}$, which removes 109 points from the training set. Next, we remove any training data which show signs of (unphysical) discontinuities in the phase, possibly due to next-to-quasi-circular corrections in the `SEOBNRv4PHM` data. For the coprecessing $(2,1)$-mode, this amounts to $11,091$ points, and 198 for the $(3,3)$-mode. Therefore, for the $(2,1)$-mode amplitude and phase, the total training dataset is $188,800$ waveforms, whereas for the $(3,3)$-mode it is $199,693$.

To validate our neural networks as they train, we also produce a validation dataset of 10^4 waveforms which covers the same parameter space as the training set. We sample this validation set uniformly is mass ratio, primary spin magnitude, spin tilt and azimuthal angles. All preprocessing steps for the validation data are the same as for the training data: we interpolate these waveforms onto the same common time grid with an equivalent spacing. For the coprecessing odd-m modes, we remove 4 points with little asymmetry, 553 which show signs of phase discontinuity in the $(2,1)$-mode phase, and 9 in the $(3,3)$-mode. This equates to a validation set size of $9,443$ for the coprecessing $(2,1)$-mode, and $9,987$ for the $(3,3)$-mode.

Lastly, we also produce a separate test dataset of 10^4 waveforms in exactly the same way as the validation set, which is completely independent and unseen by the neural networks. Of this dataset, 3 points are removed for the coprecessing odd-m modes due to little symmetry, 672 due to discontinuities in the $(2,1)$-phase, and 13 due to the $(3,3)$-phase. Therefore for the $(2,1)$-mode amplitude and phase, the test set is of size 9325, and for the $(3,3)$-mode it is $9,984$.

2. *Euler Angles*

For the Euler angles, we use the same dataset of 2×10^5 waveforms as described above. However, as the Euler angles become ill defined in the non-precessing limit, we restrict our training data to only those binaries with an initial in-plane spin magnitude $|\chi_{1,\perp}| = \sqrt{\chi_{1x}^2 + \chi_{1y}^2} > 10^{-3}$. In contrast to above, we decompose this initial dataset into a training dataset of 1.8×10^5 binaries and a validation dataset of $18,634$ binaries. As no hyperparameter optimization was performed on the Euler angle networks, the validation dataset is never used to train the network or to inform the network hyperparameters. We therefore treat the validation dataset as being effectively independent. The data conditioning is identical to the procedure described above for the waveform modes, with the Euler angles being evaluated on a uniform grid with spacing $\Delta t = 1M$ and a length of $10^4 M$.

B. Reduced Basis and Empirical Interpolant

We construct our reduced bases and empirical interpolants following the algorithm described in Sec.II A. We separate each coprecessing mode into its constituent amplitude and phase, and construct a reduced basis, empirical interpolant and neural network for each compo-

nent. We also construct a reduced basis and empirical interpolant for each Euler angle separately, leading to a total of 11 different components to make up the full precessing signal[2]. When discussing the construction and evaluation of these models, we use the following terminology: $\mathbf{X}$ describes the input parameters of the model, i.e. the four intrinsic parameters of the binary $\mathbf{X} = \vec{\lambda} = \{q, \chi_{1x}, \chi_{1y}, \chi_{1z}\}$; $\mathbf{Y}$ is an n-dimensional vector that denotes the fitting coefficients, for example the mode amplitudes in Eq. (6).

We choose to condition the data before building our reduced bases as we found this to be beneficial for the neural network performance: For the coprecessing modes we use a `scikit-learn` [58] `Standard` scaler on the $\mathbf{X}$ data and a `MinMax` scaler on the $\mathbf{Y}$ data for the phases as we found that without scaling the greedy algorithm for the coprecessing $(2,1)$- and $(3,3)$-mode phases was unable to converge and produce a reduced basis to within the greedy tolerance accuracy. We also remove the initial phase at time $t = -10,000M$, such that all phase data begin at zero. We note that we do not explicitly model these initial phases, and leave this to future work.

In contrast, we find no major benefit to scaling the $\mathbf{X}$ data for the Euler angles and the amplitude $\mathbf{Y}$ data for the coprecessing modes. For α and γ, we apply a `MinMax` scaler to them$\mathbf{Y}$ data but we do not apply any preprocessing to the $\mathbf{Y}$ data for β. A summary of the data conditioning can be found in Tab. III C.

To build the reduced bases, we use an absolute greedy error tolerance of $\sigma = 10^{-6}$ for all components of the coprecessing modes, except for the phases of the $(2,1)$ and $(3,3)$-modes. For the $(2,1)$-mode, we decreased the greedy tolerance to 10^{-8} as we found a significant tail of poor mismatches against the reduced basis representation with a tolerance of 10^{-6}. Conversely, for the $(3,3)$-mode, we reduced the tolerance to 10^{-3} in order to achieve a reduced basis of manageable size. The tolerances and the sizes of the resulting reduced bases (and therefore the number of empirical interpolation nodes) are given in the fourth and fifth column of Tab. III B.

To assess the accuracy of the coprecessing (ℓ, m)-modes reconstructed from their reduced basis representations in amplitude and phase, we compute frequency-domain white noise mode-by-mode mismatches $\bar{\mathcal{M}}_f$, defined by Eq. (11) against the original `SEOBNRv4PHM` data. Columns 6-9 of Tab. III B show the maximum and median mismatch across the full training and validation datasets for each coprecessing mode, noting that the validation data is not used in the construction of the reduced bases. Generally, we find that the odd-m modes are less accurately represented than the even-m modes and that that their bases sizes are larger. This is perhaps not too surprising as the odd-m modes are (i) subdominant and (ii) contain more structure, therefore requiring more basis elements to achieve the same representation accuracy [59].

Similarly, we compute time-domain mismatches $\bar{\mathcal{M}}_t$, defined by Eq. (15) between the original `SEOBNRv4PHM` data Euler angles, and those reconstructed from the reduced basis projections. We do this across both the training and validation datasets, and state the median and maximum values for each dataset in columns 5-8 in the bottom half of Tab. III B. We see that for both the coprecessing modes and the Euler angles, the median mismatch across both datasets is comparable to the greedy tolerance used to create the reduced basis (for the mode mismatches, it is limited by whichever greedy tolerance is larger, amplitude or phase).

Lastly, the similar mismatches for the coprecessing modes across both the training dataset, which was used to construct the bases, and the validation dataset, which was previously unseen, suggests that the reduced bases are large enough to accurately represent waveforms across our chosen parameter space. We note that for the Euler angles, the mismatches (both median and maximum) over the validation dataset can be up to an order of magnitude smaller than over the training dataset. This suggests that the validation dataset is not large enough to accurately represent the full distribution over the entire parameter space, especially for the β angle which is typically much flatter than either α or γ.

C. Parameter space fits with ANNs

We now describe the architecture, training and optimization of our neural networks for the fitting coefficients of the coprecessing modes each decomposed into amplitude and phase and Euler angles, and discuss the achieved accuracy for each of component separately. We build the neural network for each model component using `Tensorflow` [60] and `Keras` [61]. Specifically, we use the `Sequential` model with fully-connected `Dense` layers. A summary of the final neural network architectures for each coprecessing mode and the Euler angles is given in Tab. III C. As an example, a graphical representation of the neural network architecture for the coprecessing $(2,2)$-mode phase is shown in Fig. 2. The neural network is shown by the red and teal rectangles, where the red ones represent the four fully-connected hidden layers, each with 320 neurons for this component and a Softplus activation function, and the teal ones show the input and output layers: 4 neurons for the intrinsic parameters $\mathbf{X}$, and 29 for the output layer as this is the number of empirical time nodes T_i for this component. The output can then be reinterpolated onto the full uniform time grid, and inverse scaled to produce the full coprecessing $(2,2)$-mode phase ϕ_{22}. For this particular component we have applied scaling to the $\mathbf{X}$ and $\mathbf{Y}$ data, as shown by the blue rectangles. For all ANNs we use 4 input neurons, but the detailed architecture is adapted for each

[2] The odd-m modes are obtained via conjugation and hence do not need to be modelled separately but are included in the full precessing signal.

(ℓ, m)	Component	Training Set Size	Greedy Tolerance	Basis Size	$\bar{\mathcal{M}}_f^{\max,\text{train}}$	$\bar{\mathcal{M}}_f^{\text{median,train}}$	$\bar{\mathcal{M}}_f^{\max,\text{val}}$	$\bar{\mathcal{M}}_f^{\text{median,val}}$
$(2,2)$	Amplitude	$200,000$	10^{-6}	23	1.1×10^{-5}	5.0×10^{-7}	1.4×10^{-5}	6.0×10^{-7}
$(2,2)$	Phase		10^{-6}	29				
$(2,1)$	Amplitude	$188,800$	10^{-6}	26	3.2×10^{-3}	1.8×10^{-6}	3.4×10^{-4}	1.7×10^{-6}
$(2,1)$	Phase		10^{-8}	40				
$(3,3)$	Amplitude	$199,693$	10^{-6}	26	1.8×10^{-2}	3.0×10^{-4}	2.7×10^{-2}	3.1×10^{-4}
$(3,3)$	Phase		10^{-3}	46				
$(4,4)$	Amplitude	$200,000$	10^{-6}	4	1.6×10^{-4}	4.0×10^{-5}	1.9×10^{-4}	4.1×10^{-5}
$(4,4)$	Phase		10^{-6}	29				
Euler Angle					$\bar{\mathcal{M}}_t^{\max,\text{train}}$	$\bar{\mathcal{M}}_t^{\text{median,train}}$	$\bar{\mathcal{M}}_t^{\max,\text{val}}$	$\bar{\mathcal{M}}_t^{\text{median,val}}$
α		180,000	7×10^{-9}	18	3.0×10^{-5}	1.4×10^{-9}	2.6×10^{-6}	1.4×10^{-9}
β		180,000	6×10^{-7}	19	4.1×10^{-6}	1.3×10^{-6}	4.4×10^{-6}	1.3×10^{-7}
γ		180,000	7×10^{-9}	18	4.2×10^{-5}	1.5×10^{-9}	1.0×10^{-6}	1.5×10^{-9}

TABLE II. Greedy tolerances, reduced basis sizes and the maximum and median training space mismatches for the amplitude and phase of each mode. For phases, MinMax scaling was used on the Y data. For both amplitudes and phases, standard scaling was used on the X data. For the Euler angles, only MinMax scaling was used on the Y-data for α and γ and we use the time-domain mismatch $\bar{\mathcal{M}}_t$ as our metric.

	(ℓ, m)- Amplitude & Phase	Euler Angles
X-data conditioning	`Standard`	None
Y-data conditioning	None (Amplitude) `MinMax` (Phase)	`MinMax` (α, γ) None (β)
Number of input neurons	4	4
Number of layers	4	9
Neurons per layer	320	128
Optimiser	Adam	AdaMax
Activation function	Softplus	Softplus
Mini-batch size	64	512
Number of training epochs	$10,000$	5000

TABLE III. Details of the final neural network architecture for each component.

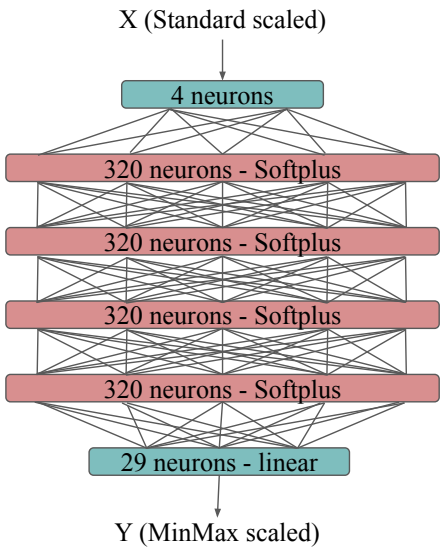

FIG. 2. Graphical representation of the ANN architecture for the coprecessing $(2,2)$-mode phase $\phi_{22}(t;\vec{\lambda})$, as an example. This neural network takes in the `Standard` scaled intrinsic binary parameters $\mathbf{X}$ as input, and outputs the `MinMax` scaled $\mathbf{Y}$, a prediction of the coprecessing $(2,2)$-mode phase at the empirical time nodes. This output vector may then be reinterpolated onto the full uniform time grid using the empirical interpolant, and inverse `MinMax` scaled to produce the full coprecessing mode phase ϕ_{22}.

component.

The size of neural network differs between the coprecessing modes and Euler angles, as shown in Tab. III C. Additionally, the coprecessing amplitudes will not undergo inverse `MinMax` scaling as we did not scale the amplitude training data in our model construction, and for the Euler angles the $\mathbf{X}$ data is not scaled. Lastly, we note that the size of the neural network output layer will vary, as it is equal to the number of empirical time nodes for each model component.

1. *Coprecessing Modes*

When training our coprecessing mode neural networks, we use a mean squared error (MSE) loss function. This quantity can be computed over either the training dataset

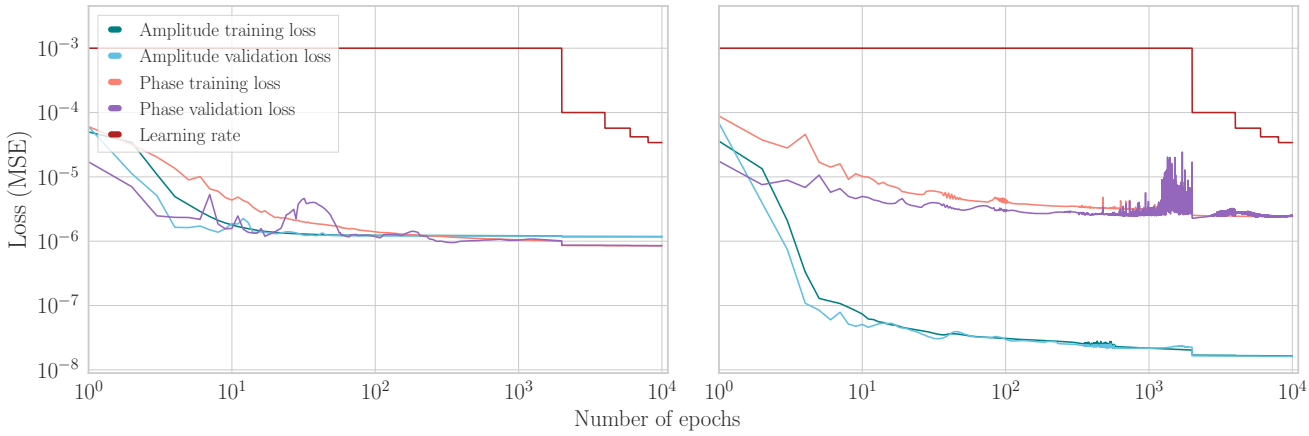

FIG. 3. Training and validation losses for the $(2,2)$- (left) and $(2,1)$-mode (right), for both amplitude and phase. The loss shown is the mean squared error (MSE) as a function of training epochs. Also shown in both panels is the learning rate (red), which changes as a function of epoch as defined by Eq. (17).

to monitor training progress, or the validation dataset as a control to check for over- or under-fitting. It is defined as

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left|\vec{y}_i^{\,true} - \vec{y}_i^{\,pred}\right|^2, \tag{16}$$

where $\vec{y}_i^{\,pred}$ is the output from the neural network, $\vec{y}_i^{\,true}$ is the true `SEOBNRv4PHM` data, scaled accordingly if required and at the appropriate empirical time nodes, and N is the number of points in either the training or validation set.

The final neural network architecture for the coprecessing modes is determined through optimisation via the hyperparameter sampling package `Hyperopt` [62]. We parse choices for neural network hyperparameters, as well as a maximum number of neural network evaluations. The package then creates and trains neural networks with hyperparameters drawn from these choices, and returns the best performing hyperparameters based on a metric, which we specify to be the validation loss upon completion of training. The choices for optimisable hyperparameters are as follows: activation function (Relu [63, 64], Elu [65], Tanh, Softmax, Softplus, Softsign [66]); optimiser (Adam [67], Nadam [68], Adamax [67], Adadelta [69]), number of training epochs (1000, 2000, 10000); and mini-batch size (32, 64, 128). We refer the reader to [70] for a systematic overview of activation functions and [71] for an overview of gradient descent optimization algorithms. We also compare three sizes of neural network: 4 layers with 320 neurons per layer; 9 layers with 128 neurons per layer; and 4 layers where the number of neurons per layer is the next largest power of two from the reduced basis size. We find slightly improved performance with shallower, wider architectures, and so use the 4 layer, 320 neurons per layer architecture for the final networks. We also do not use dropout in our final configurations as we find this can create a lack of stability in training leading to higher mismatches. Our final optimal neural network architecture is detailed in Tab. III C.

Additionally, we use an adaptive learning rate as in Ref. [19] in order to achieve faster convergence and prevent overshooting of the optimal trained weights. Our learning rate takes the form

$$\tau_i = (\tau_{\rm init} - \tau_{\rm final})/(1 + R\lfloor i/\Delta i\rfloor) + \tau_{\rm final}, \tag{17}$$

where τ_i is the learning rate at epoch i, the initial learning rate $\tau_{\rm init} = 10^{-3}$, the final learning rate $\tau_{\rm init} = 10^{-5}$, the decay rate $R = 10$, and our training epoch interval $\Delta i = 2000$. Thus our learning rate exhibits step-wise changes, decreasing every $2,000$ epochs. We use a mean squared error (MSE) as the loss metric, losses on both the training and validation datasets for the $(2,2)$ and $(2,1)$-mode amplitudes and phases are shown in Fig. 3, as well as the variable learning rate. We see that for the $(2,2)$-mode (left panel), both the amplitude and phase loss plateau around 10^{-6} after ~ 100 epochs of training, and for the $(2,1)$-mode (right panel) the phase reaches a similar loss plateau as the $(2,2)$-mode, however, the amplitude continues to improve to a loss value of $\sim 10^{-8}$. We also note that for all components, the training and validation losses are very comparable – a sign that we are neither over- nor under-fitting in our training procedure. The training and validation losses for the $(3,3)$- and $(4,4)$-modes are shown in Fig. 12 in Appendix V.

2. *Euler Angles*

In contrast to the architecture used for the coprecessing modes, for the Euler angles we use a network that is narrower and deeper consisting of 9 layers with 128 neurons per layer. We found that the Softplus activation function coupled with the Adamax optimizer produced robust results at the desired level of accuracy, though

we did not perform the more exhaustive hyperparameter optimization used in the construction of the networks for the coprecessing modes. The networks are trained for 5000 epochs using a mean squared error loss function, as defined in Eq. (16). For the learning rate, we use an initial value of 10^{-2} and use an adaptive scheme that reduces the learning rate when the loss has stopped improving, as implemented by the `ReduceLROnPlateau` callback in `Keras` . We found no significant improvement when exploring the use of dropout regularization or L^p regularizers[3], so do not include them in the final model.

In addition to the default network above, we also constructed a neural network for the residuals between the input empirical interpolation coefficients and the default neural network predictions (see also [32]), $\tilde{y}_k = y_k^{\rm true} - y_k^{\rm pred}$. This allows us to reconstruct the empirical interpolation coefficients using a two step procedure: we first evaluate the default neural network then we correct for any residual errors using the second network. However, we found this gave no noticeable improvement in accuracy. Due to the additional computational cost associated to the network evaluation, we opt not to use the residuals approach in the final model.

D. Complete Surrogate Model

Once the reduced bases and empirical interpolants are built and the neural networks have been trained, we have a total of 11 surrogate models for the different components that constitute the complete precessing model, `SEOBNN_v4PHM_4dq2`: The four coprecessing modes split into amplitudes and phases, and the three Euler angles. In Fig. 4 we show an example for a fiducial binary with parameters $\vec{\lambda} = \{q, \chi_{1x}, \chi_{1y}, \chi_{1z}\} = \{1.86, 0.045, -0.283, 0.274\}$, i.e. a moderately precessing binary with a moderate unequal mass ratio. We note that this particular binary was not in our training or validation datasets. The top left panel shows the mode amplitudes as predicted by the surrogate for each coprecessing mode, the top right panel the corresponding phases. The `SEOBNRv4PHM` data are shown by the dashed curves in all panels. The middle panel shows the final surrogate models for the Euler angles. We note the excellent agreement between the true data and predictions, including around merger at $t = 0M$. In the bottom panel we show the time-domain strain (Eq. (7)) obtained by combining the surrogate models (plus the conjugate modes) following the description in Eq. (8). We note, however, that we do not explicitly model the relative phase offsets between the coprecessing modes, which were incorporated manually from the true `SEOBNRv4PHM` data in the construction of the precessing strain. We leave the modelling of these relative phase offsets to future work.

Having seen the excellent agreement between prediction and true `SEOBNRv4PHM` data for a single fiducial binary, we now quantify the accuracy the surrogate models for each component across the parameter space.

For each coprecessing mode we compute white noise frequency-domain mismatches $\bar{\mathcal{M}}_f$ between the true `SEOBNRv4PHM` coprecessing waveform modes and the surrogate predictions for the test dataset, which consists of 10^4 waveforms that were not part of our training space $\mathcal{T}_M$ (see Sec. III A). We limit our mismatch integration to start at $f_{\rm min} = 20$ Hz and fix the total mass to $44M_\odot$, which completely covers also the longest waveforms in our test set. The mismatch result for each of the four coprecessing modes is shown in in Fig. 5. For each of the four coprecessing modes we find that the bulk of mismatches is less than 10^{-2} or 1%, with 4.6% greater than this value for the $(2,1)$-mode, 0.8% for the $(3,3)$-mode, and 2.6% for the $(4,4)$-mode. For the $(2,2)$-mode we find that it is less than 10^{-3} with only 3.3% of mismatches greater than this, with a median mismatch of $\sim 3 \times 10^{-4}$. We find comparable performance for each of the three higher modes considered, with a median mismatch of $\sim 10^{-3}$, however we do note that there are tails of higher mismatches in the odd m-modes. Histograms of the mismatches for the coprecessing $(2,2)$ and $(2,1)$-modes at different total masses can be found in Fig. 14 in App. V. We find almost identical results for the $(2,2)$-mode, and find small improvement for the $(2,1)$-mode as the total mass is increased. Therefore, the results in Fig. 5 represents the worst case scenario.

To see where in parameter space the worst mismatches lie, particularly the high mismatch tails in the odd m-modes, we take the worst 5% for each coprecessing mode and plot them in the space of mass ratio q against $\chi_{1\perp}$, with the $(2,2)$- and $(2,1)$-modes shown in Fig. 6, and the $(3,3)$- and $(4,4)$-modes in Fig. 13 in App. V. We see that for the $(2,2)$- and $(4,4)$-modes, the highest mismatches lie broadly evenly across the parameter space, although with fewer high mismatches at low in-plane spin values. For the odd m-modes, however, the worst mismatches lie close to equal mass and at low in-plane spin values. In this region of parameter space, we expect the odd m-modes to be heavily suppressed, and so training data may be considerably more noisy, therefore leading to worse mismatches. It also means that when combining the modes into a full precessing strain, the contribution of these modes to the full signal is diminished and so will not have as much impact on the accuracy of the full waveform.

For the Euler angles, we use time-domain mismatches, see Eq. (15), as the main metric to quantify the accuracy of the surrogate prediction. We show the mismatches between the `SEOBNRv4PHM` data and the surrogate models for the Euler angles in Fig. 7. We also demonstrate that the accuracy of the residual surrogate model outlined in Sec. III C offers no noticeable benefit with mismatches in

[3] The L^p norm is defined by $\|L\|_p = \sum_n (|x_n|^p)^{1/p}$ and we applied the regularization penalty to both the kernel and bias using the `L1L2 Class` in `Keras` .

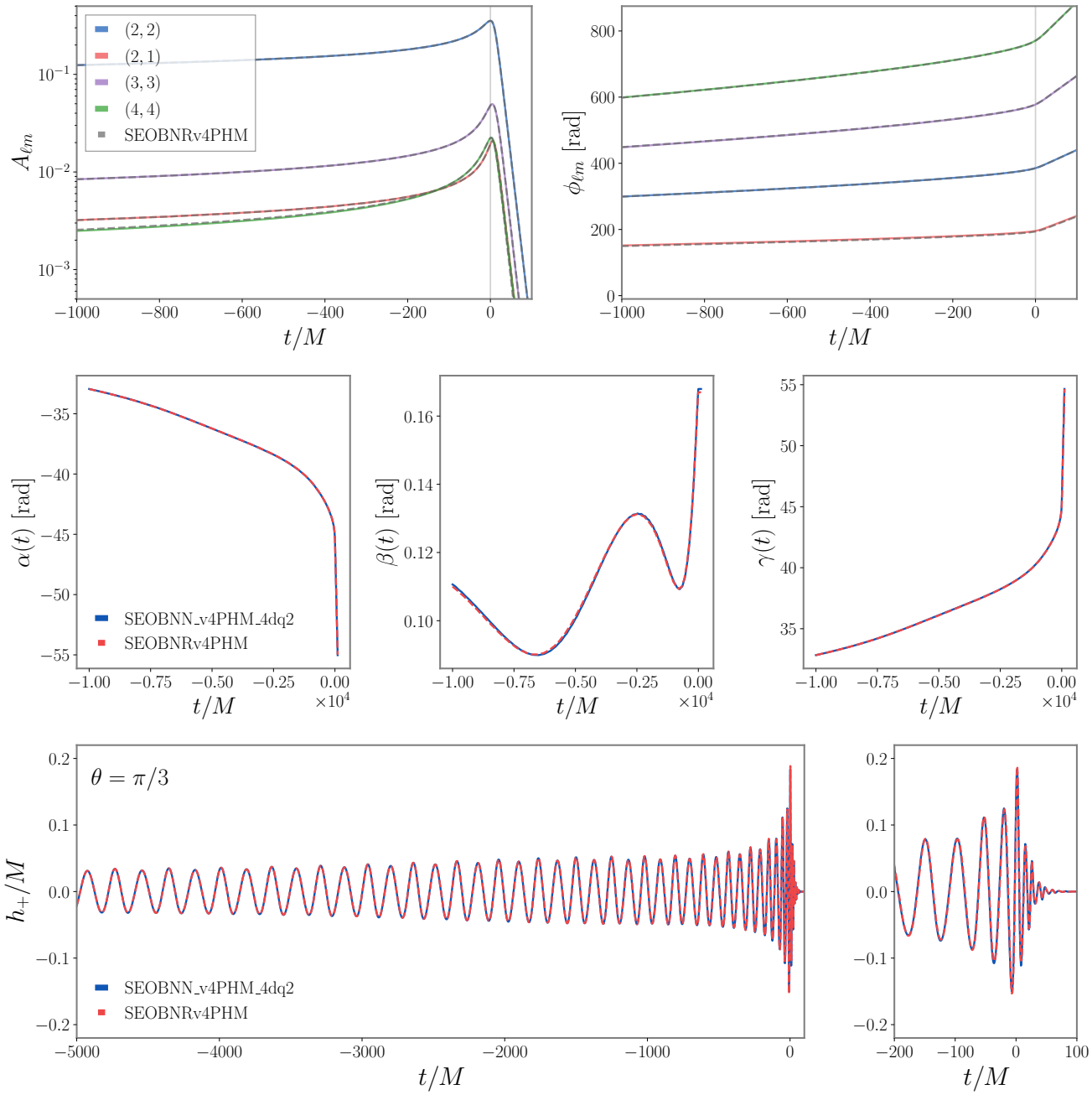

FIG. 4. Top panel: Comparison of the coprecessing mode amplitudes (left) and phases (right) predicted by the surrogates (solid coloured lines) against the `SEOBNRv4PHM` data (dashed grey)for a fiducial binary with parameters $\{q, \chi_{1x}, \chi_{1y}, \chi_{1z}\} = \{1.86, 0.045, -0.283, 0.274\}$, where the Cartesian spin parameters are specified at a $(2,2)$-mode reference frequency of 4 Hz. The merger at $t = 0M$ is indicated by the grey vertical line. Middle panel: Comparison of the Euler angles predicted by the neural network (blue) against the `SEOBNRv4PHM` data (red) for the fiducial binary. Bottom panel: The time-domain strain in the J-frame for our fiducial binary at an inclination of $\theta = \pi/3$. We include all modes up to $\ell \leq 4$.

broad agreement with our default model.

Finally, whilst we find it convenient to work with the $SO(3)$ representation of the Euler angles, an appealing alternative approach is to parameterize the rotation group by a set of unit quaternions [53, 72]. Fundamentally, the quaternions still describe the time-dependent rotation of the frame but are endowed with a number of beneficial mathematical properties, such as the singularities that can occur in the Euler angle formalism. For the reduced parameter space considered here, we found no

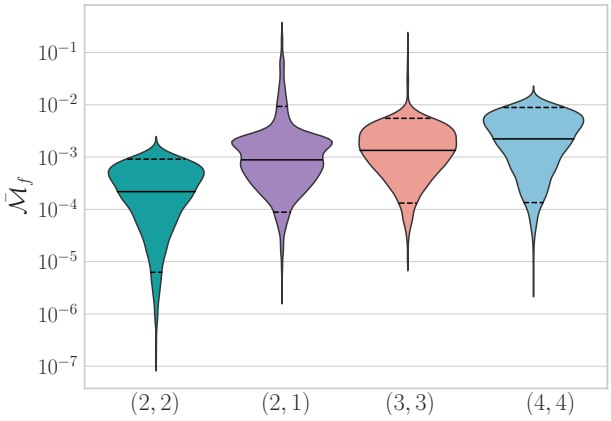

FIG. 5. White noise mismatches between the SEOBNRv4PHM-generated coprecessing frame mode data and and the neural network-predicted coprecessing mode, for each of the four modes across the 10,000 binary test set. Mismatch calculations start from an $f_{\rm min} = 20$Hz for a total mass of $44M_\odot$. Also shown for each coprecessing mode are the median mismatch (black) and 90% intervals (black dashed).

noticeable benefit to adopting the quaternion framework and opted to work with Euler angles out of simplicity. We leave a more detailed investigation of the wider parameter space to future work.

IV. MODEL EVALUATION

A. Waveform Accuracy

The observed GW signal from single-spin precessing binary black holes depends on 12 parameters: the component masses m_i, the dimensionless spin $\vec{\chi}_1(t)$, the direction from the source frame to the observer (ι, ϕ_0), the polarization ψ_0, time of arrival t, the luminosity distance d_L and the sky location (θ, ϕ). Here we neglect the sky location and write the real-valued detector response $h_r(t)$ as

$$h_r(t) = h_+(t)\cos(2\psi) + h_\times(t)\sin(2\psi). \tag{18}$$

where $h(t) = h_+(t) - ih_\times(t)$. We are now interested in validating the accuracy of our surrogate model, SEOBNN_v4PHM_4dq2, against the slow waveform model SEOBNRv4PHM. To do so, we calculate strain mismatches optimized over $\{\psi, \varphi, t\}$, as these quantities are not astrophysically relevant. We follow the approaches detailed in [17, 30, 49, 56] and numerically optimize over the phase ϕ and analytically maximize over the template polarization ψ and relative time shift t,

$$\mathcal{M}_\kappa(\phi_0^s, \psi_0^s) = \max_{t_0^h, \phi_0^h, \psi_0^h} \langle \hat{h}_r, \hat{s}_r(\phi_0^s, \psi_0^s)\rangle_f, \tag{19}$$

where h_r denotes the template waveform, generated by

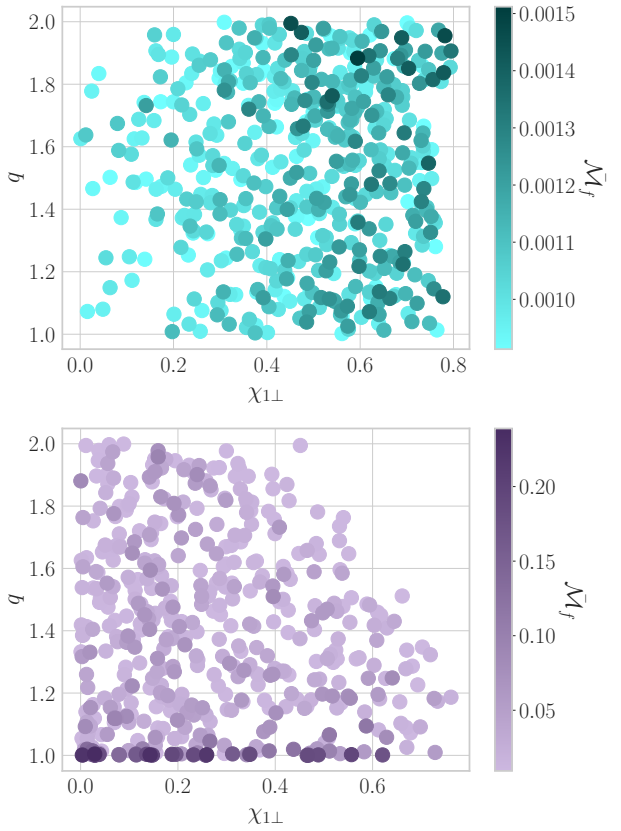

FIG. 6. Worst 5% of test dataset mismatches for the coprecessing (2, 2)- (top), (2, 1)-modes (bottom), shown in parameter space of mass ratio q against in-plane spin magnitude $|\chi_{1\perp}|$. We note that the highest mismatches for the (2, 2)-mode are scattered across much of this space, although with the worst mismatches at larger mass ratio and spin magnitude. In contrast, the worst mismatches for the (2, 1)-mode lie in the region close to equal-mass where there is less asymmetry in the system and so this particular mode is heavily suppressed.

our SEOBNN_v4PHM_4dq2 surrogate, and s_r is the signal waveform, taken to be SEOBNRv4PHM. We use the index κ to distinguish the match optimised over the polarisation angle from Eq. (11). Finally, we average the match by weighting each waveform (indexed by i) by its optimal signal-to-noise ratio ρ to account for the likelihood that the signal would have been detected. This allows us to define an orientation-averaged match as [56]

$$\mathcal{M}_w = \left(\frac{\sum_i \mathcal{M}_{\kappa,i}^3 \rho_i^3}{\sum_i \rho_i^3}\right)^{1/3}, \tag{20}$$

and the concomitant orientation-averaged mismatch $\bar{\mathcal{M}}_w = 1 - \mathcal{M}_w$. For the match calculation, we assume a lower cutoff frequency of 20Hz and use the projected PSD for Advanced LIGO in the upcoming fourth observing run (O4) [73], consisting of the Advanced LIGO and

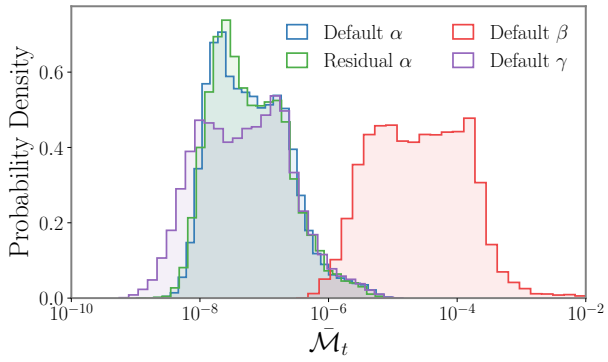

FIG. 7. Time-domain mismatches for the surrogate model for the Euler angles against the training data. For α, we show mismatches for the surrogate when using empirical coefficients predicted by the default network and when predicted by a two stage network that includes a fit to the residuals from the default network. We find no notable improvement in fitting the residuals.

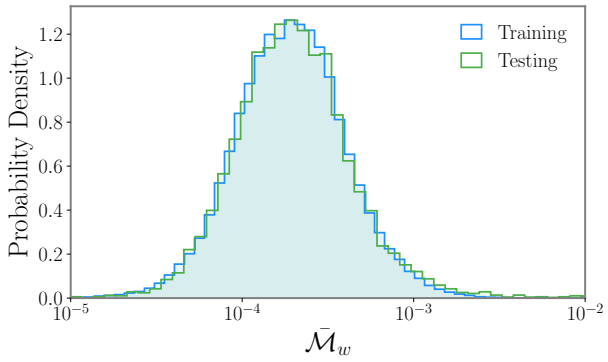

FIG. 8. Orientation averaged mismatches for `SEOBNRv4PHM` against `SEOBNN_v4PHM_4dq2` for all ≤ 4 modes in the J-frame. We show 9×10^4 binaries randomly drawn from the training (blue) dataset and 1×10^4 binaries randomly drawn from the independent testing (green) dataset, which the neural network has never been exposed to. We find excellent agreement irrespective of the dataset.

Virgo detectors as well as KAGRA. We take the masses to be uniformly distributed between $50M_\odot$ and $200M_\odot$ and the orientation angles to be isotropic on the unit sphere. The mass ratio, spin magnitude and spin orientation are as described in Table. II C. We reiterate that to construct the full precessing strain from our ANN waveform model, here we use the true `SEOBNRv4PHM` relative phase offsets between the coprecessing modes. The resulting mismatches are shown in Fig. 8 using all $\ell \leq 4$ modes in the inertial J-frame as per Eq. (8). We show mismatches against the training dataset, used to construct our ANN waveform model, and the independent testing dataset to which the model has never been exposed. For both datasets we find excellent agreement and find a median mismatch of 1.9×10^{-4}. The 5th and 95th percentiles for the mismatches against the training dataset are 5.8×10^{-5} and 6.5×10^{-4} respectively. The mismatch errors here are approximately an order of magnitude below the anticipated error of `SEOBNRv4PHM` against precessing numerical relativity simulations [30]. We find that the error of our model against the input data is competitive with the accuracy provided by other surrogate models, e.g. [13, 16].

B. Timing

In order to test the efficiency of our surrogate model, we developed two interfaces. The first interface is built exclusively within the `NumPy` framework. The second interface uses the `Tensorflow` framework to provide GPU acceleration. When run on a single CPU, we find broad parity between the computational efficiency of the two implementations. However, when run on a GPU, the implementation in `Tensorflow` allows for significant computational speedup, as discussed below.

The typical evaluation time for a single Euler angle surrogate model is on the order of 250μs. As a reminder, this includes the computational cost of producing a single prediction for the empirical interpolation coefficients from the neural network as well as the multiplication by the empirical interpolation matrix. The amplitude and phase surrogate models are marginally slower such that each waveform mode $h_{\ell m} = A_{\ell m} e^{-i\phi_{\ell m}}$ takes $\sim 925\mu$s to generate.

In required model components are constructed from 11 individual surrogate models: 3 Euler angles and 4 waveform modes plus their conjugates. To evaluate all 3 Euler angles takes $\sim 750\mu$s and to evaluate the 4 waveform mode surrogates takes ~ 3.7ms. Array conjugation is a significantly cheaper operation requiring only $\sim 10\mu$s per array. Evaluation of the waveform modes is the single most expensive operation in our model.

Next we need to evaluate the Wigner-D matrices, $D^{\ell}_{mm'}(\alpha, \beta, \gamma)$, in order to perform the time-dependent rotations. This is the second most expensive operation in `SEOBNN_v4PHM_4dq2`. In order to mitigate against the computational cost, we can perform a series of optimizations, such as pre-caching of numerical coefficients. This allows us to significantly reduce the cost of evaluating the Wigner-D matrices to ~ 5.5ms. Further optimization could be achieved through the use of interpolating non-uniform grids or pre-compilation in C. We leave such optimizations to the future. Performing the time-dependent rotations of the waveform modes from the L-frame to the J-frame is relatively efficient, requiring only ~ 2ms.

Altogether, we find that the typical waveform generation cost for a signal covering the surrogate length of $10,000M$ is on the order of 18ms on a single CPU with `SEOBNN_v4PHM_4dq2`. This is on average $\mathcal{O}(10^2)$ times

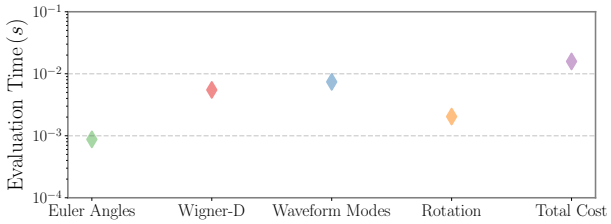

FIG. 9. Computational cost for each step in the waveform construction. A notable bottleneck is the computation of the Wigner-D matrices $D^{\ell}_{mm'}(\alpha, \beta, \gamma)$ over the full $10^4 M$ time grid.

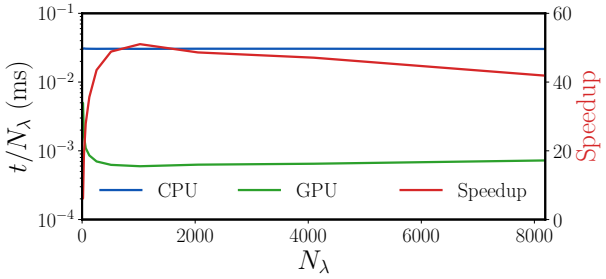

FIG. 10. Computational cost per binary for evaluating the 22-mode surrogate model over batches of N_λ binaries. We show both CPU and GPU timings as well as the overall speedup enabled by GPU acceleration.

faster than the underlying `SEOBNRv4PHM` model, which takes ~ 3000ms. In addition, it is also almost three times as fast as the surrogate model presented in [74], though the surrogate model presented here is twice as long in duration spanning $10^4 M$ compared to $5 \times 10^3 M$ in [74]. A notable caveat is that the surrogate model presented in [74] covers a significantly larger domain of the parameter space making any direct comparison difficult. Nevertheless, the preliminary model presented here suggests that reduced order models for precessing multipolar waveform models powered by neural networks are highly competitive relative to alternative strategies, even on a CPU. We show the typical timings for each element and for the entire waveform in Fig. 9. All CPU timings were generated using an Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz using the `NumPy` interface.

However, a significant benefit of reduced order models powered by the `Tensorflow` architecture is that they provide a convenient platform for GPU acceleration. In particular, GPU acceleration is most beneficial when generating batches of surrogate models, mitigating any overhead in the transfer of data between the CPU and the GPU. Evaluating the surrogate model for the 22-mode over a varying number of binaries, we find that GPU acceleration leads to a factor $\sim \mathcal{O}(30)$ speedup in surrogate generation cost relative to CPUs. For 8192 binary configurations, we find that on a CPU each surrogate model takes ~ 30ms compared to ~ 0.7ms on a GPU. For the CPU-GPU benchmarking, CPU timings were performed using an Intel(R) Xeon(R) CPU @ 2.30GHz and GPU timings were performed using an NVIDIA Tesla P100-PCIE-16GB. We show the comparative CPU and GPU timings in Fig. 10 along with the relative speedup provided by GPU acceleration.

V. DISCUSSION

As the number of observations of GW signals from BBH mergers is only set to increase with improving detector sensitivity, the availability of accurate, highly computationally efficient theoretical models is critical for future GW data analysis. Faster surrogate models of very accurate but slower underlying waveform models will prove beneficial, motivating a need for exploration of novel surrogate-building methods. In this paper, we have constructed a proof-of-concept time-domain surrogate model of `SEOBNRv4PHM`, which makes use of neural networks to perform parameter space fits. We follow the techniques used in Refs. [19, 28], extending them to precessing multipolar waveforms for the full inspiral-merger-ringdown signal. We decompose our surrogate model into two sections: coprecessing waveform modes defined in the non-inertial coprecessing L-frame which tracks the precessing motion of the binary, and the three Euler angles which represent the rotation between this frame and the J-frame. We consider four coprecessing modes: the dominant quadrupolar $(2, 2)$-mode, and three higher multipoles $(2, 1)$, $(3, 3)$ and $(4, 4)$, and each of these modes is then decomposed further into amplitude and phase. Therefore we model a total of 11 components. For each component, we construct a reduced basis and empirical interpolant, before performing parameter space fits using artificial neural networks.

We demonstrate that the performance of our surrogate `SEOBNN_v4PHM_4dq2` is highly competitive in comparison to alternative surrogate modelling strategies, producing waveforms with precessing strain mismatches $\sim \mathcal{O}\left(10^{-3} - 10^{-4}\right)$ against the true `SEOBNRv4PHM` data. We also show that this model is computationally efficient, producing waveforms on a CPU two orders of magnitude faster than the underlying `SEOBNRv4PHM` model, and almost three times as fast as the recently developed surrogate model [74] in the restricted intrinsic parameter space covered by our ANN model. We also note that our output waveforms are around two times longer than this surrogate, and that unlike the underlying `SEOBNRv4PHM` model, the evaluation time is independent upon the binary parameters. Additionally, we have shown that our surrogate model allows for an even more significant speed up in evaluation time when evaluating batches of waveforms simultaneously on GPUs.

As a proof of concept for neural network surrogates of precessing multipolar waveforms, our model is built

on a restricted parameter range of mass ratios $q \in [1, 2]$ and single precessing spins $|\chi_1| \leq 0.8$, $|\chi_2| = 0$. This multidimensional portion of the precessing BBH parameter space is a starting point for surrogates which utilise neural networks, though we do not envisage any imminent roadblocks to incorporating additional information in order to extend towards the full 7D intrinsic parameter space of double precessing spins, with more unequal mass ratios. We note, however, that the size of training dataset would need to be significantly larger to accurately represents the full range of waveforms in this larger parameter space. Additionally, any higher dimensional training dataset would need to be thoroughly checked for data quality across the parameter space, as we noted that even in our restricted parameter we faced issues of pathologies in the underlying waveform model, where the coprecessing mode phases became discontinuous in the inspiral, possibly due to inaccurate next-to-quasicircular corrections in `SEOBNRv4PHM`.

To explore how accurately our model can extrapolate outside the training range, we tested each coprecessing mode surrogate on $1,500$ single-spin binaries with mass ratio in $q \in (2, 4]$ or with primary spin magnitudes $0.8 < |\chi_1| \leq 0.99$ and computed mismatches against the true `SEOBNRv4PHM` data. We found that the extrapolation in spin magnitude is relatively smooth as long as the mass ratio is constrained to values that were in the original training space (i.e. $q \leq 2$), resulting in mismatches for each coprecessing mode approximately one order of magnitude worse than shown in Fig. 5. However, for binaries with $q > 2$, irrespective of the in-plane spin magnitude, each mode surrogate performs poorly. The same trends were observed for the Euler angles.

Additionally, we investigated whether our model is able to capture the behaviour of binaries with two spinning black holes by using the previously developed dimensional reduction mapping of [18]. To do so, we constructed $1,000$ double-spin binaries with parameters inside the training space, ensuring that the mapped spin magnitude was ≤ 0.8. We found that the coprecessing $(2, 2)$-mode is replicated with a mismatch accuracy of $\mathcal{O}(10^2 - 10^3)$, but that higher modes are less well reproduced.

When building our surrogate model, we explored several options to improve the accuracy of the coprecessing modes neural network fit. Before training the artificial neural networks for the coprecessing modes, we tried using principal component analysis on the reduced basis coefficient phase training data, to identify trend directions in the data which may be easier for the neural network to fit. Whilst this provided a small improvement in resulting mismatches for the $(2, 2)$-mode, it led to marginally worse results for the $(2, 1)$-mode and no noticeable difference in the $(3, 3)$ and $(4, 4)$-modes. We also attempted to improve the mismatches of our coprecessing modes by training the neural networks for longer than $10,000$ epochs. However, between $10,000$ and $100,000$ epochs, almost no improvements were seen in the loss values for both amplitudes and phases for all modes. Furthermore we tried training on the residual coprecessing phase, where the geometric mean has been subtracted to de-trend the phase data. We find this had no impact on either the reduced basis sizes or the accuracy with which we were able to train our artificial neural networks. For the Euler angles, we explored the possibility of training an additional neural network to model the residual error on the predicted α, but found no noticeable improvement.

In addition, we also explored the effect of different sizes of training data sets upon the accuracy of the coprecessing mode fits. We found that the reduced basis size and projection errors were insensitive to smaller training set sizes for sets above 100 waveforms, and similarly that the coprecessing mode mismatches for the $(2, 1)$-mode shown in Fig. 5 were virtually identical when the $(2, 1)$-phase was reconstructed on a random training subset of $10,000$. This suggests that our choice of training set size may have been conservative, and future models over this parameter space could attain similar accuracies with smaller training set sizes.

We have demonstrated the feasibility and efficacy of using neural networks as part of precessing multipolar IMR waveform surrogate models, and leave the extension to the full 7D precessing parameter space as well as the modelling of the spin evolution to further work. We suggest that with even further consideration given to neural network optimisation and data de-trending over the full 7D parameter space of generically precessing BBHs, this could prove a promising pathway towards accurate, efficient gravitational waveform surrogate model building.

ACKNOWLEDGMENTS

The authors thank Alberto Vecchio for useful discussions and Vijay Varma for comments on the manuscript. L.M.T. is supported by STFC, the School of Physics and Astronomy at the University of Birmingham and the Birmingham Institute for Gravitational Wave Astronomy. G.P. and P.S. acknowledge support from STFC grant ST/V005677/1. PS also acknowledges the Dutch Research Council (NWO) Veni Grant No. 680-47-460. GP gratefully acknowledges support from an NVIDIA Academic Hardware Grant. Computations were performed using the University of Birmingham's BlueBEAR HPC service, which provides a High Performance Computing service to the University's research community, as well as on resources provided by Supercomputing Wales, funded by STFC grants ST/I006285/1 and ST/V001167/1 supporting the UK Involvement in the Operation of Advanced LIGO. Some computations were also performed using Google Colaboratory. Part of this research was performed while L.M.T., G.P. and P.S. were visiting the Institute for Pure and Applied Mathematics (IPAM), which is supported by the National Science Foundation (Grant No. DMS-1925919). This manuscript has the LIGO document number P2200161.

APPENDIX

Here we first show in Fig. 11 the distribution of the 2×10^5 waveform training dataset, plotted in the space of primary spin components and coloured by number density. This dataset is made up of a systematically sampled subset of 774 points (whose parameters are specified in Tab. II C) in order to effectively sample the boundaries, as well as 199,226 randomly sampled binaries, uniform in mass ratio, spin magnitude, azimuthal and tilt angles.

Next we show in Fig. 12 the training and validation losses as defined by Eq. (16), for both the amplitude and phase of the coprecessing $(3,3)$- (left) and $(4,4)$ (right) -modes, over the course of the neural network training. We also plot the adaptive learning rate, defined by Eq. (17). We note that similarly to the $(2,1)$-mode amplitude as seen in Fig. 3, the amplitude losses for the $(3,3)$- and $(4,4)$-modes evolve to a minimum of around 10^{-8} at the end of training, and the phases to around 10^{-6}. Additionally, we see that the training and validation losses remain similar in magnitude throughout the training process, suggesting we are neither over- nor under-fitting.

We show in Fig. 13 the worst 5% of mismatches $\bar{\mathcal{M}}_f$ over the test dataset between the surrogate-predicted coprecessing waveform modes and the true `SEOBNRv4PHM` data, for the $(3,3)$- (left) and $(4,4)$- (right) modes, plotted over mass ratio q and in-plane spin magnitude $|\chi_{1\perp}|$ and coloured by mismatch. We see that for the $(3,3)$-mode, similarly to the $(2,1)$-mode in Fig. 6, the worst mismatches appear around equal mass and less in-plane spin where there is less asymmetry in the system and so these modes are heavily suppressed in the full precessing strain. In contrast, and similarly to the $(2,2)$-mode in Fig. 6, the $(4,4)$-mode exhibits lower mismatches overall, and more evenly spread across the parameter space, although the worst mismatches tend to be at more unequal mass ratios and larger in-plane spins.

Finally, we show in Fig. 14 the effect of changing the total mass of the binary upon the coprecessing mode mismatches shown in Fig. 5. We choose a representative sample of four total masses $M_{\rm tot} \in \{44, 65, 85, 125\} M_\odot$ and recompute mismatches in the same way as shown in Fig. 5, from a low frequency cutoff of 20 Hz each time, over the 10,000 binary test set, for each of the $(2,2)$- (left) and $(2,1)$-modes. We find that the change in total mass makes little difference in both cases, and that in fact a higher total mass than $44 M_\odot$ slightly improves the $(2,1)$-mode mismatches, as may be expected since the higher total mass is effectively a decrease in waveform length.

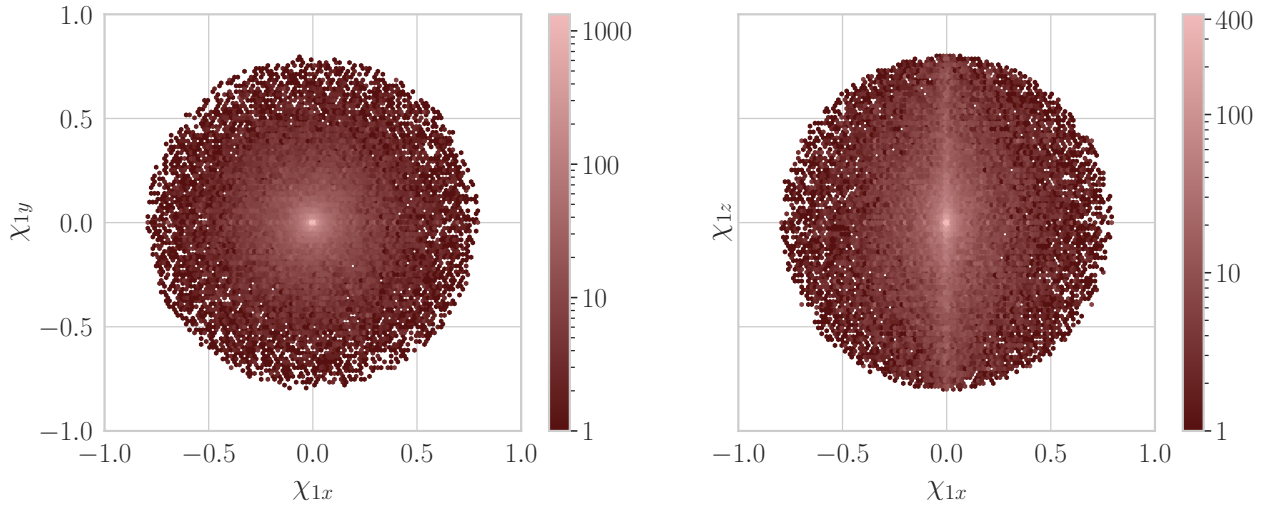

FIG. 11. Visualisation of the spin parameters of the entire training dataset coloured by number density.

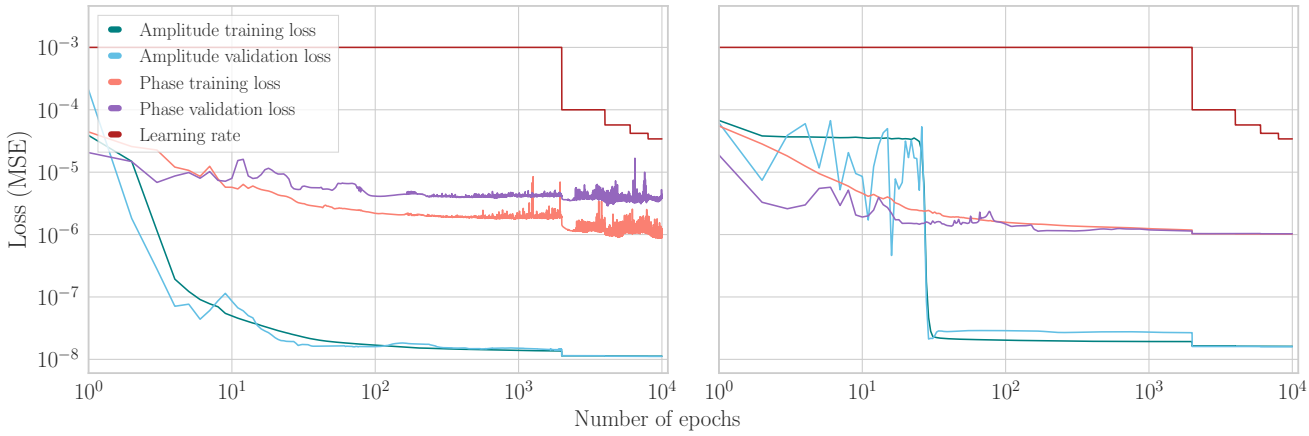

FIG. 12. Training and validation losses for the $(3,3)$ (left) and $(4,4)$ (right) -modes, both amplitude and phase. The loss shown is the mean squared error (MSE) as a function of epochs trained. Also shown in both panels is the learning rate, which changes as a function of epoch as defined by Eq. (17).

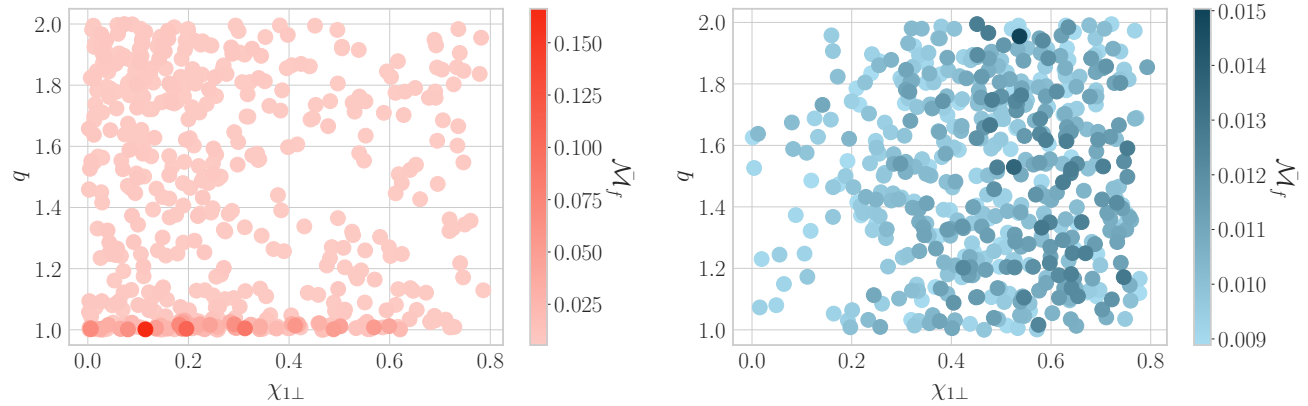

FIG. 13. Worst 5% of test dataset mismatches $\bar{\mathcal{M}}_f$ for the $(3,3)$- (left) and $(4,4)$ (right), modes, shown in parameter space of mass ratio and in-plane spin magnitude, and coloured by mismatch.

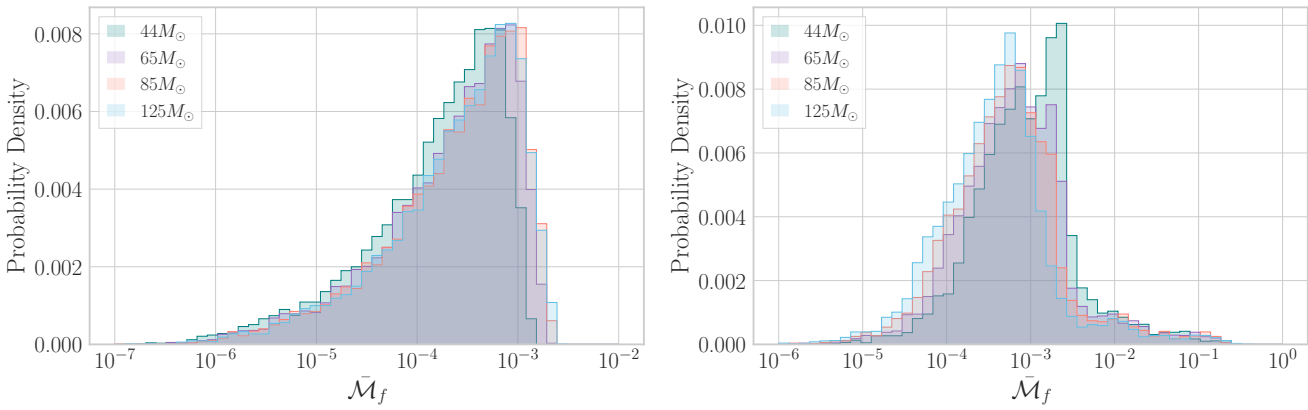

FIG. 14. White noise mismatches between the `SEOBNRv4PHM`-generated coprecessing frame mode data and and the neural network-predicted coprecessing mode, for the $(2,2)$ (left) and the $(2,1)$-mode (right), as computed in Fig. 5 but for a range of different total masses $M_{\rm tot} = [44, 65, 85, 125]\,M_\odot$. Mismatch calculations start from an $f_{\rm min} = 20$Hz and are computed across the $10,000$ binary test set.

[1] J. Aasi *et al.* (LIGO Scientific), "Advanced LIGO," Class. Quant. Grav. **32**, 074001 (2015), arXiv:1411.4547 [gr-qc].

[2] F. Acernese *et al.* (VIRGO), "Advanced Virgo: a second-generation interferometric gravitational wave detector," Class. Quant. Grav. **32**, 024001 (2015), arXiv:1408.3978 [gr-qc].

[3] T. Akutsu *et al.* (KAGRA), "Overview of KAGRA: Detector design and construction history," PTEP **2021**, 05A101 (2021), arXiv:2005.05574 [physics.ins-det].

[4] B. P. Abbott *et al.* (LIGO Scientific, Virgo), "GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs," Phys. Rev. X **9**, 031040 (2019), arXiv:1811.12907 [astro-ph.HE].

[5] R. Abbott *et al.* (LIGO Scientific, Virgo), "GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo During the First Half of the Third Observing Run," Phys. Rev. X **11**, 021053 (2021), arXiv:2010.14527 [gr-qc].

[6] R. Abbott *et al.* (LIGO Scientific, VIRGO), "GWTC-2.1: Deep Extended Catalog of Compact Binary Coalescences Observed by LIGO and Virgo During the First Half of the Third Observing Run," (2021), arXiv:2108.01045 [gr-qc].

[7] R. Abbott *et al.* (LIGO Scientific, VIRGO, KAGRA), "GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run," (2021), arXiv:2111.03606 [gr-qc].

[8] R. Abbott *et al.* (LIGO Scientific, VIRGO, KAGRA), "The population of merging compact binaries inferred using gravitational waves through GWTC-3," (2021), arXiv:2111.03634 [astro-ph.HE].

[9] Geraint Pratten, Sascha Husa, Cecilio Garcia-Quiros, Marta Colleoni, Antoni Ramos-Buades, Hector Estelles, and Rafel Jaume, "Setting the cornerstone for a family of models for gravitational waves from compact binaries: The dominant harmonic for nonprecessing quasi-circular black holes," Phys. Rev. D **102**, 064001 (2020), arXiv:2001.11412 [gr-qc].

[10] Scott E. Field, Chad R. Galley, Frank Herrmann, Jan S. Hesthaven, Evan Ochsner, and Manuel Tiglio, "Reduced basis catalogs for gravitational wave templates," Phys. Rev. Lett. **106**, 221102 (2011), arXiv:1101.3765 [gr-qc].

[11] Scott E. Field, Chad R. Galley, Jan S. Hesthaven, Jason Kaye, and Manuel Tiglio, "Fast prediction and evaluation of gravitational waveforms using surrogate models," Phys. Rev. X **4**, 031006 (2014), arXiv:1308.3565 [gr-qc].

[12] Michael Pürrer, "Frequency domain reduced order models for gravitational waves from aligned-spin compact binaries," Class. Quant. Grav. **31**, 195010 (2014), arXiv:1402.4146 [gr-qc].

[13] Jonathan Blackman, Scott E. Field, Chad R. Galley, Béla Szilágyi, Mark A. Scheel, Manuel Tiglio, and Daniel A. Hemberger, "Fast and Accurate Prediction of Numerical Relativity Waveforms from Binary Black Hole Coalescences Using Surrogate Models," Phys. Rev. Lett. **115**, 121102 (2015), arXiv:1502.07758 [gr-qc].

[14] Jonathan Blackman, Scott E. Field, Mark A. Scheel, Chad R. Galley, Daniel A. Hemberger, Patricia Schmidt, and Rory Smith, "A Surrogate Model of Gravitational Waveforms from Numerical Relativity Simulations of Precessing Binary Black Hole Mergers," Phys. Rev. D **95**, 104023 (2017), arXiv:1701.00550 [gr-qc].

[15] Chad R. Galley and Patricia Schmidt, "Fast and efficient evaluation of gravitational waveforms via reduced-order spline interpolation," (2016), arXiv:1611.07529 [gr-qc].

[16] Vijay Varma, Scott E. Field, Mark A. Scheel, Jonathan Blackman, Davide Gerosa, Leo C. Stein, Lawrence E. Kidder, and Harald P. Pfeiffer, "Surrogate models for precessing binary black hole simulations with unequal masses," Phys. Rev. Research. **1**, 033015 (2019), arXiv:1905.09300 [gr-qc].

[17] Patricia Schmidt, Frank Ohme, and Mark Hannam, "Towards models of gravitational waveforms from generic binaries II: Modelling precession effects with a single effective precession parameter," Phys. Rev. D **91**, 024043 (2015), arXiv:1408.1810 [gr-qc].

[18] Lucy M. Thomas, Patricia Schmidt, and Geraint Pratten, "A new effective precession spin for modelling multi-modal gravitational waveforms in the strong-field regime," (2020), arXiv:2012.02209 [gr-qc].

[19] Sebastian Khan and Rhys Green, "Gravitational-wave surrogate models powered by artificial neural networks," Phys. Rev. D **103**, 064015 (2021), arXiv:2008.12932 [gr-qc].

[20] Neil J. Cornish, "Fast Fisher Matrices and Lazy Likelihoods," (2010), arXiv:1007.4820 [gr-qc].

[21] Priscilla Canizares, Scott E. Field, Jonathan R. Gair, and Manuel Tiglio, "Gravitational wave parameter estimation with compressed likelihood evaluations," Phys. Rev. D **87**, 124005 (2013), arXiv:1304.0462 [gr-qc].

[22] Priscilla Canizares, Scott E. Field, Jonathan Gair, Vivien Raymond, Rory Smith, and Manuel Tiglio, "Accelerated gravitational-wave parameter estimation with reduced order modeling," Phys. Rev. Lett. **114**, 071104 (2015), arXiv:1404.6284 [gr-qc].

[23] Rory Smith, Scott E. Field, Kent Blackburn, Carl-Johan Haster, Michael Pürrer, Vivien Raymond, and Patricia Schmidt, "Fast and accurate inference on gravitational waves from precessing compact binaries," Phys. Rev. D **94**, 044031 (2016), arXiv:1604.08253 [gr-qc].

[24] Barak Zackay, Liang Dai, and Tejaswi Venumadhav, "Relative Binning and Fast Likelihood Evaluation for Gravitational Wave Parameter Estimation," (2018), arXiv:1806.08792 [astro-ph.IM].

[25] Neil J. Cornish, "Heterodyned likelihood for rapid gravitational wave parameter inference," Phys. Rev. D **104**, 104054 (2021), arXiv:2109.02728 [gr-qc].

[26] Nathaniel Leslie, Liang Dai, and Geraint Pratten, "Mode-by-mode relative binning: Fast likelihood estimation for gravitational waveforms with spin-orbit precession and multiple harmonics," Phys. Rev. D **104**, 123030 (2021), arXiv:2109.09872 [astro-ph.IM].

[27] Soichiro Morisaki, "Accelerating parameter estimation of gravitational waves from compact binary coalescence using adaptive frequency resolutions," Phys. Rev. D **104**, 044062 (2021), arXiv:2104.07813 [gr-qc].

[28] Alvin J. K. Chua and Michele Vallisneri, "Learning Bayesian posteriors with neural networks for gravitational-wave inference," Phys. Rev. Lett. **124**, 041102 (2020), arXiv:1909.05966 [gr-qc].

[29] Maximilian Dax, Stephen R. Green, Jonathan Gair, Jakob H. Macke, Alessandra Buonanno, and Bern-

hard Schölkopf, "Real-Time Gravitational Wave Science with Neural Posterior Estimation," Phys. Rev. Lett. **127**, 241103 (2021), arXiv:2106.12594 [gr-qc].
[30] Serguei Ossokine *et al.*, "Multipolar Effective-One-Body Waveforms for Precessing Binary Black Holes: Construction and Validation," Phys. Rev. D **102**, 044055 (2020), arXiv:2004.09442 [gr-qc].
[31] Theocharis A. Apostolatos, Curt Cutler, Gerald J. Sussman, and Kip S. Thorne, "Spin induced orbital precession and its modulation of the gravitational wave forms from merging binaries," Phys. Rev. D **49**, 6274–6297 (1994).
[32] Styliani-Christina Fragkouli, Paraskevi Nousi, Nikolaos Passalis, Panagiotis Iosif, Nikolaos Stergioulas, and Anastasios Tefas, "Deep Residual Error and Bag-of-Tricks Learning for Gravitational Wave Surrogate Modeling," (2022), arXiv:2203.08434 [astro-ph.IM].
[33] Vijay Varma, Scott E. Field, Mark A. Scheel, Jonathan Blackman, Lawrence E. Kidder, and Harald P. Pfeiffer, "Surrogate model of hybridized numerical relativity binary black hole waveforms," Phys. Rev. D **99**, 064045 (2019), arXiv:1812.07865 [gr-qc].
[34] Tousif Islam, Vijay Varma, Jackie Lodman, Scott E. Field, Gaurav Khanna, Mark A. Scheel, Harald P. Pfeiffer, Davide Gerosa, and Lawrence E. Kidder, "Eccentric binary black hole surrogate models for the gravitational waveform and remnant properties: comparable mass, nonspinning case," Phys. Rev. D **103**, 064022 (2021), arXiv:2101.11798 [gr-qc].
[35] Tousif Islam, Scott E. Field, Scott A. Hughes, Gaurav Khanna, Vijay Varma, Matthew Giesler, Mark A. Scheel, Lawrence E. Kidder, and Harald P. Pfeiffer, "Surrogate model for gravitational wave signals from non-spinning, comparable- to large-mass-ratio black hole binaries built on black hole perturbation theory waveforms calibrated to numerical relativity," (2022), arXiv:2204.01972 [gr-qc].
[36] Alejandro 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), arXiv:1611.03703 [gr-qc].
[37] Alessandro 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), arXiv:1806.01772 [gr-qc].
[38] Stefano Schmidt, Matteo Breschi, Rossella Gamba, Giulia Pagano, Piero Rettegno, Gunnar Riemenschneider, Sebastiano Bernuzzi, Alessandro Nagar, and Walter Del Pozzo, "Machine Learning Gravitational Waves from Binary Black Hole Mergers," Phys. Rev. D **103**, 043020 (2021), arXiv:2011.01958 [gr-qc].
[39] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T. Patera, "An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations," Comptes Rendus Mathematique **339**, 667–672 (2004).
[40] Yvon Maday, Ngoc Cuong Nguyen, Anthony T. Patera, and S. H. Pau, "A general multipurpose interpolation procedure: the magic points," Communications on Pure and Applied Analysis **8**, 383–404 (2009).
[41] Daniel Williams, Ik Siong Heng, Jonathan Gair, James A. Clark, and Bhavesh Khamesra, "Precessing numerical relativity waveform surrogate model for binary black holes: A Gaussian process regression approach," Phys. Rev. D **101**, 063011 (2020), arXiv:1903.09204 [gr-qc].
[42] Jooheon Yoo, Vijay Varma, Matthew Giesler, Mark A. Scheel, Carl-Johan Haster, Harald P. Pfeiffer, Lawrence E. Kidder, and Michael Boyle, "Targeted large mass ratio numerical relativity surrogate waveform model for GW190814," (2022), arXiv:2203.10109 [gr-qc].
[43] Galley, C. R., "Rompy," free software (GPL).
[44] Lawrence E. Kidder, "Coalescing binary systems of compact objects to postNewtonian 5/2 order. 5. Spin effects," Phys. Rev. D **52**, 821–847 (1995), arXiv:gr-qc/9506022.
[45] Alessandra Buonanno, Yan-bei Chen, and Michele Vallisneri, "Detecting gravitational waves from precessing binaries of spinning compact objects: Adiabatic limit," Phys. Rev. D **67**, 104025 (2003), [Erratum: Phys.Rev.D 74, 029904 (2006)], arXiv:gr-qc/0211087.
[46] Patricia Schmidt, Mark Hannam, Sascha Husa, and P. Ajith, "Tracking the precession of compact binaries from their gravitational-wave signal," Phys. Rev. D **84**, 024046 (2011), arXiv:1012.2879 [gr-qc].
[47] Patricia Schmidt, Mark Hannam, and Sascha Husa, "Towards models of gravitational waveforms from generic binaries: A simple approximate mapping between precessing and non-precessing inspiral signals," Phys. Rev. D **86**, 104063 (2012), arXiv:1207.3088 [gr-qc].
[48] Mark Hannam, Patricia Schmidt, Alejandro Bohé, Leïla Haegel, Sascha Husa, Frank Ohme, Geraint Pratten, and Michael Pürrer, "Simple Model of Complete Precessing Black-Hole-Binary Gravitational Waveforms," Phys. Rev. Lett. **113**, 151101 (2014), arXiv:1308.3271 [gr-qc].
[49] Geraint Pratten *et al.*, "Computationally efficient models for the dominant and subdominant harmonic modes of precessing binary black holes," Phys. Rev. D **103**, 104056 (2021), arXiv:2004.06503 [gr-qc].
[50] Antoni Ramos-Buades, Patricia Schmidt, Geraint Pratten, and Sascha Husa, "Validity of common modeling approximations for precessing binary black holes with higher-order modes," Phys. Rev. D **101**, 103014 (2020), arXiv:2001.10936 [gr-qc].
[51] Michael Boyle, Lawrence E. Kidder, Serguei Ossokine, and Harald P. Pfeiffer, "Gravitational-wave modes from precessing black-hole binaries," (2014), arXiv:1409.4431 [gr-qc].
[52] Roberto Cotesta, Alessandra Buonanno, Alejandro Bohé, Andrea Taracchini, Ian Hinder, and Serguei Ossokine, "Enriching the Symphony of Gravitational Waves from Binary Black Holes by Tuning Higher Harmonics," Phys. Rev. D **98**, 084028 (2018), arXiv:1803.10701 [gr-qc].
[53] Michael Boyle, "Angular velocity of gravitational radiation from precessing binaries and the corotating frame," Phys. Rev. D **87**, 104006 (2013), arXiv:1302.2919 [gr-qc].
[54] LIGO Scientific Collaboration, "LIGO Algorithm Library - LALSuite," free software (GPL) (2018).
[55] Etienne Racine, "Analysis of spin precession in binary black hole systems including quadrupole-monopole interaction," Phys. Rev. D **78**, 044021 (2008), arXiv:0803.1820 [gr-qc].
[56] Ian Harry, Stephen Privitera, Alejandro Bohé, and Alessandra Buonanno, "Searching for Gravitational Waves from Compact Binaries with Precessing Spins," Phys. Rev. D **94**, 024012 (2016), arXiv:1603.02444 [gr-qc].

[57] Nathan K. Johnson-McDaniel, Sumeet Kulkarni, and Anuradha Gupta, "Inferring spin tilts at formation from gravitational wave observations of binary black holes: Interfacing precession-averaged and orbit-averaged spin evolution," (2021), arXiv:2107.11902 [astro-ph.HE].

[58] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," Journal of Machine Learning Research **12**, 2825–2830 (2011).

[59] Chad. Galley and Patricia Schmidt, "Fast and efficient evaluation of gravitational waveforms via reduced-order spline interpolation," (2016), arXiv:1611.07529 [gr-qc].

[60] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," (2015), software available from tensorflow.org.

[61] François Chollet *et al.*, "Keras," `https://keras.io` (2015).

[62] James Bergstra, Daniel Yamins, and David Cox, "Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures," in *International conference on machine learning* (PMLR, 2013) pp. 115–123.

[63] Kunihiko Fukushima, "Visual feature extraction by a multilayered network of analog threshold elements," IEEE Transactions on Systems Science and Cybernetics **5**, 322–333 (1969).

[64] Xavier Glorot, Antoine Bordes, and Yoshua Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics* (JMLR Workshop and Conference Proceedings, 2011) pp. 315–323.

[65] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," (2015).

[66] D.L. Elliott and David L. Elliott, "A better activation function for artificial neural networks," (1993).

[67] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980 (2014).

[68] Timothy Dozat, "Incorporating nesterov momentum into adam," ICLR Workshop **1:2013-2016** (2016).

[69] Matthew D. Zeiler, "Adadelta: An adaptive learning rate method," (2012), 10.48550/ARXIV.1212.5701.

[70] Johannes Lederer, "Activation functions in artificial neural networks: A systematic overview," (2021).

[71] Sebastian Ruder, "An overview of gradient descent optimization algorithms," (2016).

[72] Michael Boyle, Robert Owen, and Harald P. Pfeiffer, "A geometric approach to the precession of compact binaries," Phys. Rev. D **84**, 124011 (2011), arXiv:1110.2965 [gr-qc].

[73] B. P. Abbott *et al.* (KAGRA, LIGO Scientific, Virgo, VIRGO), "Prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo and KAGRA," Living Rev. Rel. **21**, 3 (2018), arXiv:1304.0670 [gr-qc].

[74] Bhooshan Gadre, Michael Pürrer, Scott E. Field, Serguei Ossokine, and Vijay Varma, "A fully precessing higher-mode surrogate model of effective-one-body waveforms," (2022), arXiv:2203.00381 [gr-qc].