Interpretable AI forecasting for numerical relativity waveforms of quasicircular, spinning, nonprecessing binary black hole mergers

Asad Khan,[1,2,3,*] E. A. Huerta,[2,4,1] and Huihuo Zheng[5]

[1]*Department of Physics, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA*
[2]*Data Science and Learning Division, Argonne National Laboratory, Lemont, Illinois 60439, USA*
[3]*National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA*
[4]*Department of Computer Science, University of Chicago, Chicago, Illinois 60637, USA*
[5]*Leadership Computing Facility, Argonne National Laboratory, Lemont, Illinois 60439, USA*

(Received 22 October 2021; accepted 21 December 2021; published 6 January 2022)

We present a deep-learning artificial intelligence model (AI) that is capable of learning and forecasting the late-inspiral, merger and ringdown of numerical relativity waveforms that describe quasicircular, spinning, nonprecessing binary black hole mergers. We used the`NRHybSur3dq8` surrogate model to produce train, validation and test sets of $\ell = |m| = 2$ waveforms that cover the parameter space of binary black hole mergers with mass ratios $q \leq 8$ and individual spins $|s^z_{\{1,2\}}| \leq 0.8$. These waveforms cover the time range $t \in [-5000\ \mathrm{M}, 130\ \mathrm{M}]$, where $t = 0M$ marks the merger event, defined as the maximum value of the waveform amplitude. We harnessed the ThetaGPU supercomputer at the Argonne Leadership Computing Facility to train our AI model using a training set of 1.5 million waveforms. We used 16 NVIDIA DGX A100 nodes, each consisting of 8 NVIDIA A100 Tensor Core GPUs and 2 AMD Rome CPUs, to fully train our model within 3.5 h. Our findings show that artificial intelligence can accurately forecast the dynamical evolution of numerical relativity waveforms in the time range $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$. Sampling a test set of 190,000 waveforms, we find that the average overlap between target and predicted waveforms is $\gtrsim 99\%$ over the entire parameter space under consideration. We also combined scientific visualization and accelerated computing to identify what components of our model take in knowledge from the early and late-time waveform evolution to accurately forecast the latter part of numerical relativity waveforms. This work aims to accelerate the creation of scalable, computationally efficient and interpretable artificial intelligence models for gravitational wave astrophysics.

DOI: 10.1103/PhysRevD.105.024024

I. INTRODUCTION

The combination of artificial intelligence (AI) and innovative computing has led to novel, computationally efficient and scalable methodologies for gravitational wave detection [1–17], denoising [18–20], parameter estimation [21–26], rapid waveform production [27,28], and early warning systems for multimessenger sources [29–31], to mention a few. The convergence of AI, distributed computing and scientific data infrastructure has enabled the creation of production-scale, AI-driven frameworks for gravitational wave detection [32–34]. The fact that these advances have stemmed from prototypes to search for gravitational waves in advanced Laser Interferometer Gravitational Wave Observatory (LIGO) data [2] into production-scale AI frameworks that process advanced LIGO data in bulk [29–31] within just five years, and that these methodologies have been embraced and developed by multiple teams around the world, furnish evidence for the transformational, global impact of AI and innovative computing in gravitational wave astrophysics [35–37].

AI has also been harnessed to learn and describe multiscale and multiphysics phenomena, such as the physics of subgrid-scale ideal magnetohydrodynamics turbulence of 2D simulations of the magnetized Kelvin-Helmholtz instability [38]. The creation of AI surrogates is an active area of research that aims to improve the computational efficiency, scalability and accuracy of scientific software utilized in conjunction with high-performance computing (HPC) platforms to study and simulate complex phenomena [39,40]. It is in the spirit of this work that researchers have explored the ability of AI to forecast the nonlinear behavior of waveforms that describe the physics of quasicircular, nonspinning, binary black hole mergers [41].

In this study we quantify the ability of AI to learn and describe the highly dynamical, nonlinear behavior of numerical relativity waveforms that describe quasicircular,

*khan74@illinois.edu

spinning, nonprecessing binary black hole mergers. To do this, we have implemented a deep-learning AI model that takes as input time-series waveform data that describe the inspiral evolution, and then outputs time-series data that describe the late-inspiral, merger and ringdown of binary black holes that span systems with mass ratios $1 \leq q \leq 8$, and individual spins $s^z_{1,2} \in [-0.8, 0.8]$. To make apparent the size and complexity of this problem, the astute reader may notice that the amount of training data to address this problem in the context of nonspinning, quasicircular binary black hole mergers is of order $\sim 1.2 \times 10^4$ [41]. In stark contrast, addressing this problem in the context of quasicircular, spinning, nonprecessing binary black hole mergers requires a training dataset that contains over $\sim 1.5 \times 10^6$ modeled waveforms to densely sample this high-dimensional signal manifold. This amount of data is needed to capture the rich dynamics imprinted in the waveforms that describe these astrophysical systems. The strategy we have followed to tackle this computational grand challenge consists of combining AI and HPC to reduce time to insight, and to incorporate a number of methodologies to create our Transformer-based AI model, including positional encoding, multihead self-attention, multihead cross attention, layer normalization, and residual connections.

Furthermore, we acknowledge the importance of going beyond innovative algorithm design, and the confluence of AI and HPC to address these types of computational challenges. There is a pressing need to understand how AI models abstract knowledge from data and make predictions. Thus, we also showcase the use of scientific visualization and HPC to *interpret* and *understand* how various components of our AI model work together to make accurate predictions. Throughout this paper we use geometric units in which $G = c = 1$. In this convention, M sets the length scale of the scale-invariant black hole simulations, and corresponds to the total mass of the spacetime simulated. For instance, $M = 1\ M_\odot = 4.93 \times 10^{-6}$ s or $M = 1\ M_\odot = 1.48$ km. In this article we use M to describe time.

This article is organized as follows. Section II describes the datasets, neural network architecture and optimization methods used to create our AI model. We present and discuss our results in Sec. III. This section includes a detailed study of the forecasting capabilities of our AI model, as well as interpretability studies. Finally, we summarize our findings and outline future work in Sec. IV.

II. METHODS

Here we describe the waveform datasets used for this study, the key components of our AI model, and the approaches followed to train and optimize it.

A. Dataset

We consider inspiral-merger-ringdown waveforms that describe quasicircular, spinning, nonprecessing binary black hole mergers. We have produced training, test and validation waveform sets with the surrogate model `NRHybSur3dq8` [42]. Since the surrogate `NRHybSur3dq8` is trained with 104 numerical relativity waveforms in the parameter range $q \leq 8$ and $|s^z_i| \leq 0.8$, we restrict our datasets to lie within the same parameter span. Throughout this paper we use a geometric unit system in which $G = c = 1$.

We use $\ell = |m| = 2$ waveforms for this study that cover the time span $t \in [-5,000\ \mathrm{M}, 130\ \mathrm{M}]$ with the merger (amplitude peak of the signal) occurring at $t = 0M$. To accurately capture the dynamics of the waveform we sample it with a time step $\Delta t = 2$ M. We split each waveform into two segments, namely, the input consisting of the early inspiral phase covering the time span $t \in [-5,000\ \mathrm{M}, -100\ \mathrm{M}]$, and the target consisting of late-inspiral, merger and ringdown covering the time span $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$. We then train an AI model to forecast the target waveform segment when fed with the input waveform segment. An example waveform with the input and target segments is shown in Fig. 1.

The training set consists of ~ 1.5 million waveforms generated by sampling the mass ratio $q \in [1, 8]$ in steps of $\Delta q = 0.08$, and the individual spins $s^z_i \in [-0.8, 0.8]$ in steps of $\Delta s^z_i = 0.012$. The validation and test sets consist of $\sim 190,000$ waveforms each, and are generated by alternately sampling the intermediate values, i.e., by sampling q and s^z_i in steps of 0.16 and 0.024 to lie between training set values. We show a small slice of the parameter space to illustrate this sampling in Fig. 1.

B. Neural network architecture

The neural network we use for numerical relativity waveform forecasting is a slightly modified version of the `Transformer` model, originally proposed in the context of Natural Language Processing [43]. The fundamental operation in the `Transformer` model is the multihead scaled dot-product `Attention` mechanism. `Attention` can be thought of as a mapping between two sets; each element of the output set is a weighted average of all elements in the input set, where the weights are assigned according to some scoring function. This helps with context-aware memorization of long sequences. We briefly discuss the various components of the `Transformer` model below.

1. Scaled dot-product attention

Consider a set of n input vectors $\{x_1, x_2, x_3, \ldots, x_n\}$ and a set of t output vectors $\{h_1, h_2, h_3, \ldots, h_t\}$ in $\mathbb{R}^d$. Then according to scaled dot-product attention, the outputs are computed as follows:

$$h_i = \sum_j w_{ij} v_j, \tag{1}$$

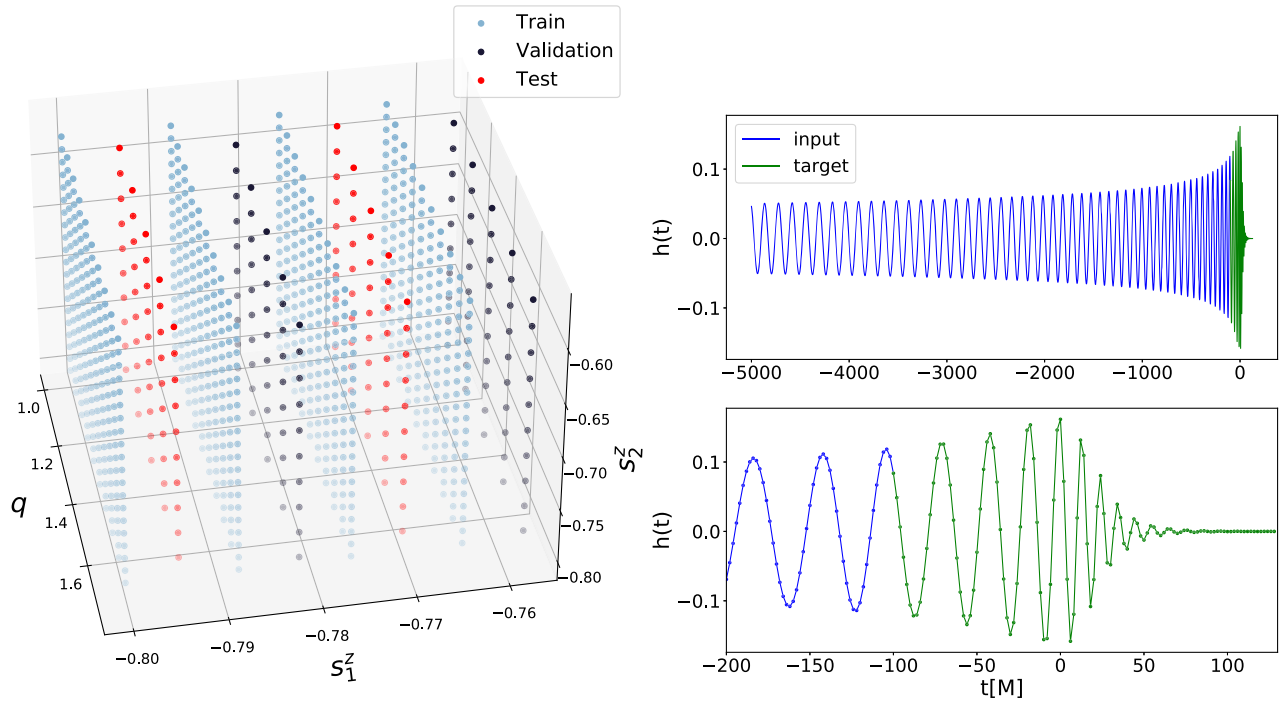

FIG. 1. (Left panel) Training, validation and test sets for the binary black hole 3D signal manifold $1 \leq q \leq 8$ and $s^z_{\{1,2\}} \in [-0.8, 0.8]$. 1.5M waveforms are used for the training set, and 190,00 waveforms for the test and validation sets. The sampling shown in this 3D representation for $q \in [1, 1.8)$ is mirrored throughout the parameter space under consideration. (Top-right panel) Sample waveform for a binary black hole with parameters $\{q, s^z_1, s^z_2\} = \{6.8, 0.718, 0.718\}$. Signals span the time window $t \in [-5000\ \mathrm{M}, 130\ \mathrm{M}]$ sampled with a time step $\Delta t = 2$ M. (Bottom-right panel) Input data to our AI model span the time window $t \leq -100M$, whereas $t \geq -100M$ represents the target time-series output.

where

$$w_{ij} = \mathtt{softmax}\left(\frac{q_i^T k_j}{\sqrt{d}}\right), \tag{2}$$

$$q_i = W_q x_i, \tag{3}$$

$$k_i = W_k x_i, \tag{4}$$

$$v_i = W_v x_i, \tag{5}$$

where W_q, W_k and W_v are three learnable weight matrices and each of the three vectors q_i, k_i, v_i (referred to as queries, keys and values) are linear transformations of the specific input x_i.

2. Self- and cross attention

Self-attention refers to applying the attention mechanism to relate different elements of a single set, i.e., queries, keys and values all correspond to the linear transformations of the same set of vectors $\{x_i\}$ as above. However, in cross attention the queries can come from a different set of vectors $\{y_i\}$, i.e., $q_i = W_q y_i$.

In our case, the set $\{x_1, x_2, x_3, \ldots, x_n\}$ corresponds to the input waveform segment and the set $\{y_1, y_2, y_3, \ldots, y_t\}$ corresponds to the target waveform segment. These are shown in blue and green respectively in the right panels of Fig. 1.

3. Multihead attention

Multihead attention simply refers to applying the attention operation several times in parallel to independently projected queries, keys and values, i.e., for n heads we would have n sets of the three matrices; W^i_q, W^i_k and W^i_v, $i \in \{1, 2, 3, \ldots, n\}$. To do this efficiently, the multihead attention module first splits the input vector x_i into n smaller chunks, and then computes the attention scores over each of the n subspaces in parallel.

4. Positional encoding

In our case, the inputs and output waveform segments are not sets but ordered time-series sequences. However, we can see from Eq. (1) that attention mechanism is permutation equivariant, i.e., it ignores the sequential nature of the input. In order to make the model sensitive to the sequential ordering of the data, we inject information about the absolute positioning of the time steps in the form of positional encoding (PE), i.e., some fixed function $f: \mathbb{N} \to \mathbb{R}^d$ to map the positions to real-valued vectors. Following

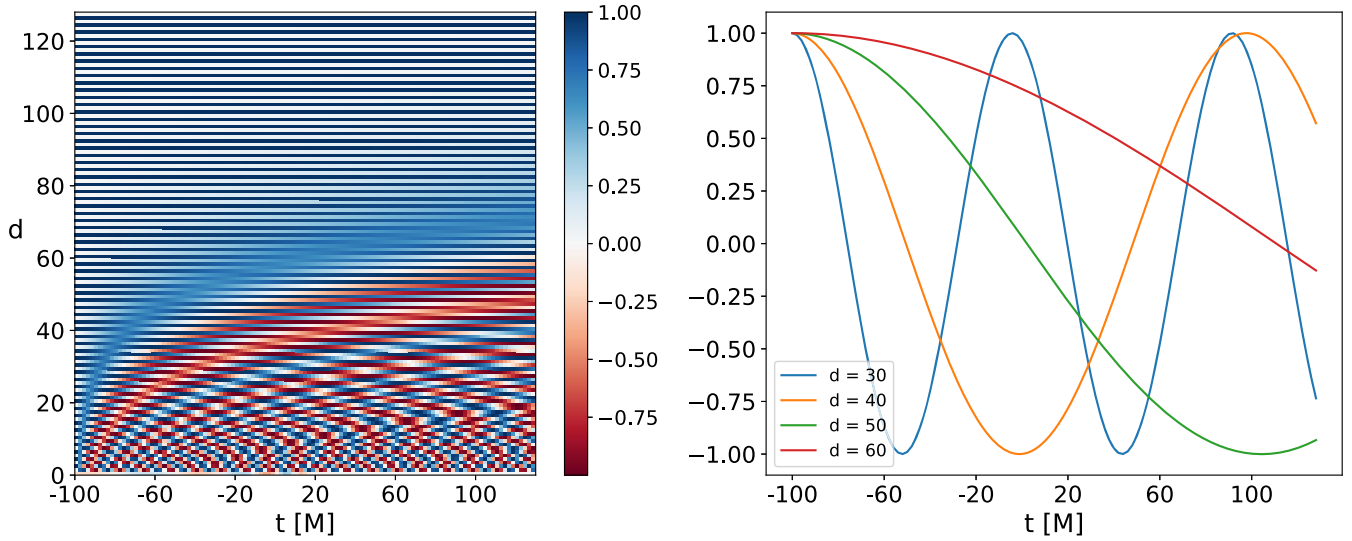

FIG. 2. (Left panel) Heatmap of the evaluation of the positional encodings—see Eqs. (6) and (7). These real-valued vectors are computed at each time stamp of the target waveform $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$—shown in the x axis, for each dimension d—shown in the y axis. (Right panel) Sample of encodings evaluated at several time stamps and dimensions. The encoding at each dimension d is a sinusoid of a different frequency.

the original `Transformer` paper, we compute the positional encodings as follows:

$$\mathrm{PE}(p, 2i) = \sin(p/10000^{2i/d}), \tag{6}$$

$$\mathrm{PE}(p, 2i+1) = \cos(p/10000^{2i/d}), \tag{7}$$

where p is the position and i is the dimension. A sample encoding for $d = 128$, used for the actual analysis conducted in this paper, is shown in Fig. 2. It is worth mentioning that the dimension d is a hyperparameter and has to be tuned for optimal performance.

At the fundamental level, the input to our model is one dimensional (a 1D wave). However, we transform these data from rank-1 to rank-2, i.e., from a sequence of real numbers of amplitude values $(h_1, h_2, , h_n)$ to a sequence of $d+1$ dimensional vectors $(\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n})$, where each $\mathbf{v_i} = [h_i, \mathrm{PE}(i, 1), \mathrm{PE}(i, 2), \mathrm{PE}(i, 3), \ldots, \mathrm{PE}(i, d)]$, and $\mathrm{PE}(i, n)$ is given by Eqs. (6) and (7).

We do this because we want the model to be aware of the time stamp of each amplitude value. One could in principle do this by inputting into the model a tuple (h_i, t_i) instead of just the sequence of amplitude values $(h_i,)$. However, positional encodings in the manner described above have historically worked much better.

5. Encoder and decoder modules

The `Transformer` model consists of an encoder module and a decoder module. The encoder takes in an input sequence $\{x_1, x_2, x_3, \ldots, x_n\}$, passes it through a multihead self-attention layer and a positionwise fully connected feed-forward network, mapping it to an attention-based latent vector representation $\{h_1, h_2, h_3, \ldots, h_n\}$. This latent representation is then passed to the decoder module, which outputs the desired target sequence $\{y_1, y_2, y_3, \ldots, y_t\}$. At each time step $t = i$ when the decoder is predicting y_i, it passes the thus-far generated output sequence $\{y_1, y_2, y_3, \ldots, y_{i-1}\}$ through a multihead self-attention layer and the latent vector representation $\{h_1, h_2, h_3, \ldots, h_n\}$ through a multihead cross-attention layer. The two are added together and passed through a positionwise fully connected feed-forward network and a final 1D convolutional layer to generate the next time step of the output sequence y_i in an autoregressive fashion.

Both the encoder and decoder modules also make use of layer normalization and residual connections. We refer the reader for a more in-depth discussion of the `Transformer` model to the original paper [43]. We summarize the architecture for our model in Fig. 3.

C. Training and optimization

As mentioned above, we first divide the waveforms into input segments corresponding to $t \in [-5000\ \mathrm{M}, -100\ \mathrm{M}]$, and target segments corresponding to $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$. We then concatenate both segments with their respective fixed positional encodings. In our experiments, we trained two models; one on only the plus-polarization waveforms and another on a dataset composed of equal number of plus- and cross-polarization waveforms. However, we did not find a significant difference in the performance between these two models, i.e. the model trained on only plus polarizations was just as good at generalizing to the cross polarizations as the model that was trained on both. Consequently, in this paper we report

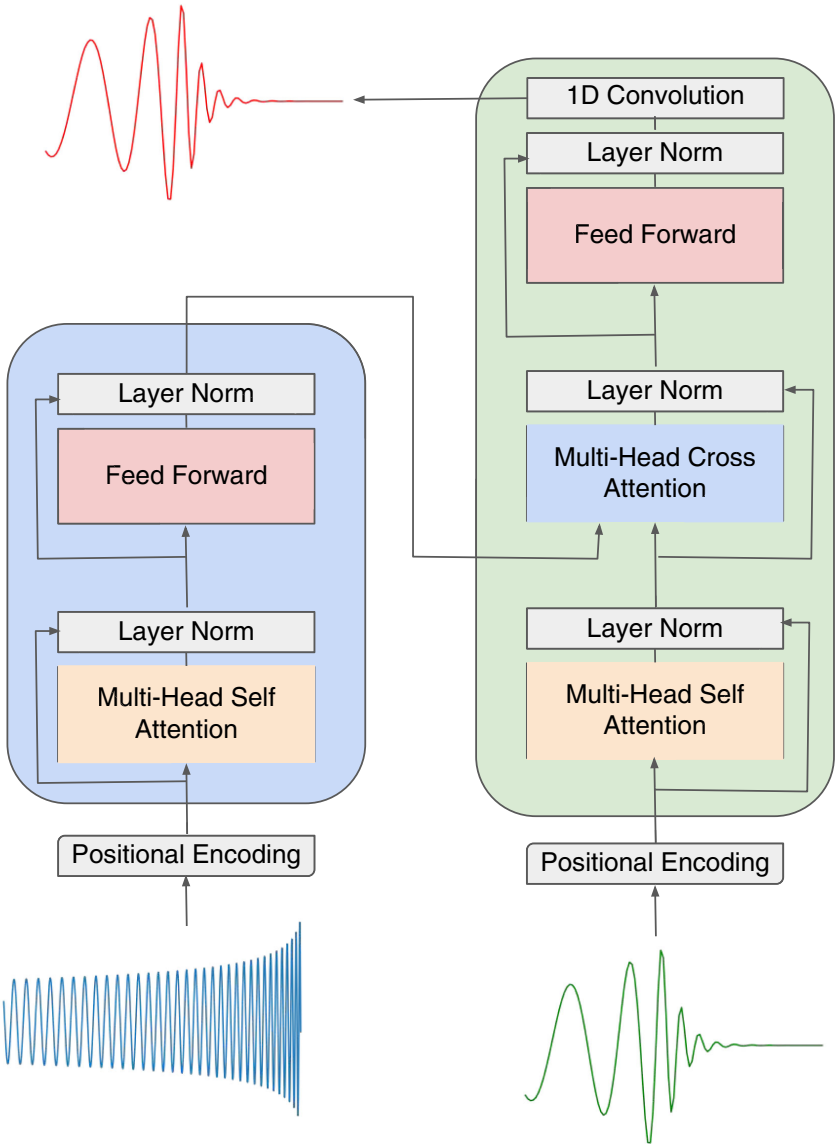

FIG. 3. Model architecture. Schematic representation of our AI model. During training we provide two input waveforms, namely, a premerger waveform that spans the time range $t \leq -100$ M—shown at the bottom left of the diagram, and a time-shift version of the target waveform that spans the time range $t \in [-101\ \mathrm{M}, 129\ \mathrm{M}]$—shown at the bottom right of the diagram. The output of this AI model, the target waveform that spans the range $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$, is shown at the top left of this diagram. At inference, we provide an input waveform—as indicated in the bottom left of the diagram. The model then outputs time samples up to time i, which are then passed as input—as shown in the bottom right panel of the figure—so that the model produces the following time samples up to time $i+1$. The final output is a waveform that covers the range $t \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$.

results for the model that was trained only on the plus polarization, but during inference it is used to predict both plus- and cross polarization. During training time we employ the `Teacher Forcing` methodology, i.e., we pass the input segment through the encoder, and a one-step time-shifted version of the target to the decoder. This means that true output is fed to the decoder for the next time-step prediction regardless of the predicted value at the current time step, which helps the model converge faster. A visual exposition of this methodology is presented in Appendix A.

We use mean-squared error between the predicted and the target series as the loss function, and use Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-07$ and learning_rate $=$ 0.001. During training we also monitor the loss on the validation set to prevent overfitting and to dynamically reduce the learning rate whenever the loss hits a plateau.

We trained our AI model using 16 NVIDIA DGX A100 nodes at the Argonne Leadership Computing Facility. Each node comprises eight NVIDIA A100 Tensor Core GPUs and two AMD Rome CPUs that provide 320 gigabytes of GPU memory. We used a batch size of 8 and trained the

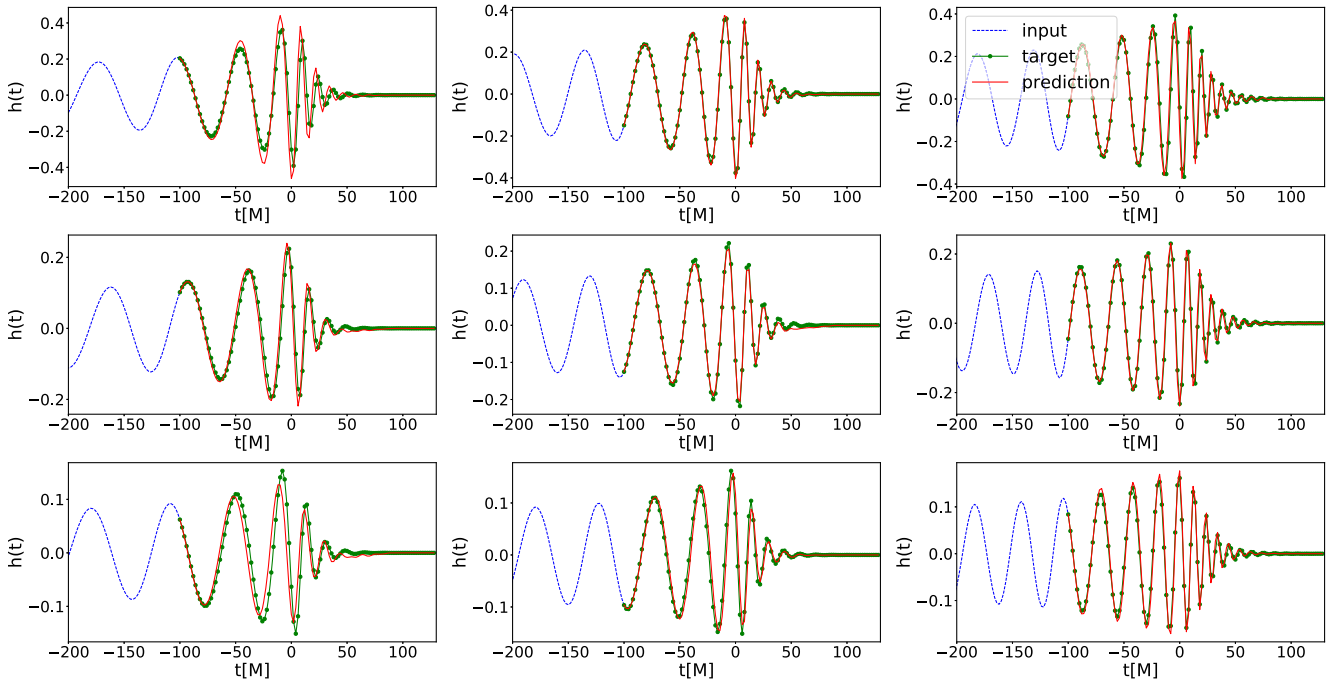

FIG. 4. Gallery of results. Sample input, target and predicted waveforms for binary black holes with mass ratios $q = \{1.04, 4.24, 6.80\}$, from top to bottom; and spins $s_1^z = s_2^z = \{-0.7, 0.0, 0.7\}$, from left to right. Notice the impact of individual spins in the dynamics of the systems, encompassing rapid (left column) and delayed plunges (right column). The model predicts the waveform evolution in the range $-100\ \mathrm{M} \leq t \leq 130\ \mathrm{M}$.

model for a total of 53 epochs, reaching convergence in 3.5 h.

III. RESULTS

During inference, we only feed the input segment to the model and let it recover the full target sequence autoregressively, i.e., to make the prediction at time step $t = i$, the decoder module is fed its own prediction from the previous time step $t = i - 1$. Our AI model outputs both the plus- and cross polarizations. We show a representative sample of target and predicted waveforms in Fig. 4. We have selected these cases to provide a visual representation of the rich dynamics captured by our AI model,

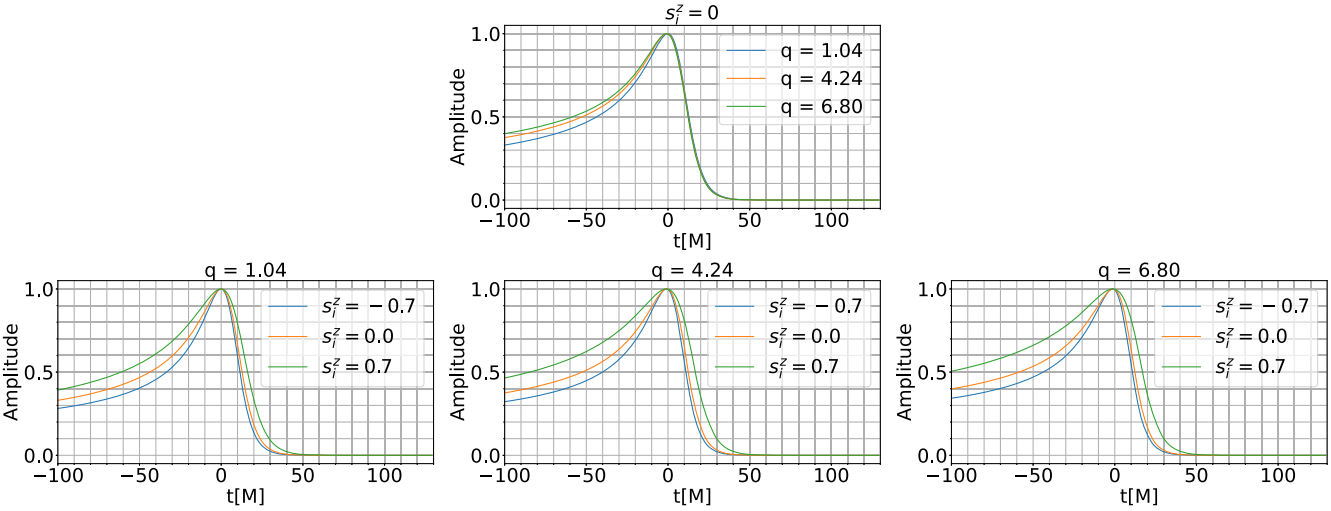

FIG. 5. (Top panel) Near-universal behavior in the dynamical evolution of quasicircular, nonspinning binary black hole mergers in the vicinity of merger ($t = 0$ M). (Bottom panels) Individual spins modulate the waveform amplitude, and drive binary black holes to merger in distinct ways, namely, rapid merger (left panel) and delayed merger (right panel). Notice the distinct features of the normalized amplitude for each system near merger. These subtle differences in waveform dynamics are highly nontrivial to capture by semianalytical waveform models, though AI can accurately learn and predict these properties in a data-driven fashion.

encompassing rapid plunges represented by black hole binaries whose components have negative spins (left column); nonspinning binary black holes (mid column); and systems that, on account of having binary components with positive spins and thus more angular momentum, complete more waveform cycles before plunge (right column).

To get a visual representation of the type of dynamics that our AI model needs to capture, we present in Fig. 5 the normalized waveform amplitude of the binary black hole systems considered in Fig, 4. Key points to extract from these results include:

(i) Top panel: Quasicircular, nonspinning, binary black holes display a well-known universal behavior in the vicinity of merger. These physical properties facilitate the training of AI models for these types of systems.

(ii) Bottom panels: We notice the role individual spins play in modulating the waveform amplitude, and driving the systems to merger. These physical properties are one of the most challenging features to capture for waveform modeling experts who aim to accurately describe the late-time evolution of spinning, nonprecessing binary black hole mergers. In this study we have demonstrated that AI may accomplish such a task in data-driven manner.

We have quantified the accuracy of our model's predictions by computing the overlap, $\mathcal{O}(h_t, h_p)$, between the target waveform h_t and the predicted waveform h_p:

$$\mathcal{O}(h_t, h_p) = \max_{t_c \phi_c}(\hat{h}_t|\hat{h}_p[t_c, \phi_c]),$$
$$\text{with} \quad \hat{h}_t = h_t(h_t|h_t)^{-1/2}, \tag{8}$$

where $\hat{h}_p[t_c, \phi_c]$ indicates that the normalized waveform $\hat{h}_p$ has been time- and phase shifted. Hence, the overlap $\mathcal{O}$ lies in [0, 1], reaching the maximum value of 1 for a perfect match. To visualize our findings, we first recast the parameter space (q, s_1^z, s_2^z) into symmetric mass ratio η and effective spin $\sigma_{\rm eff}$ using the relations

$$\eta = \frac{q}{(1+q)^2} \quad \text{and} \quad \sigma_{\rm eff} = \frac{q s_1^z + s_2^z}{1+q}. \tag{9}$$

Using these conventions, we present overlap calculations between the target and predicted waveforms for the entire test dataset in Fig. 6. To carry out these calculations, we used the plus- and cross polarizations of the target waveforms spanning the range $t \in [-5000\ \text{M}, 130\ \text{M}]$. Our target waveforms consist of input data spanning the range $t \in [-5000\ \text{M}, -100\ \text{M}]$ and complemented with our predicted waveforms that span the range $t \in [-100\ \text{M}, 130\ \text{M}]$. These calculations, presented in the top panels of Fig. 6, indicate that both the mean and median overlaps $\mathcal{O} > 0.99$, and that less than 10% of the test dataset has $\mathcal{O} < 0.98$. These outliers are localized at the edges of the parameter space, as shown in the top right panel of Fig. 6. In brief, our model predicts the late-inspiral, merger and ringdown waveform evolution in the time range $[-100\ \text{M}, 130\ \text{M}]$. Since we sampled waveforms with a time step of 2 M, this means that the model outputs 115 steps of waveform evolution.

We have also used our AI model to quantify the accuracy of its predictions from two additional initial times, namely $t = \{-80\ \text{M}, -60\ \text{M}\}$. In these cases, the model outputs 105 and 95 steps of waveform evolution, respectively. We present results for these cases in the mid and bottom panels of Fig. 6. The overlap distributions for these cases are such that

(i) $t = -80$ M: median and mean overlaps $\mathcal{O} > 0.994$, with less than 6.1% of the test dataset with $\mathcal{O} < 0.98$.

(ii) $t = -60$ M: median and mean overlaps $\mathcal{O} > 0.996$, with less than 2.4% of the test dataset with $\mathcal{O} < 0.98$.

We provide additional results that may be explored interactively in the website [44]. We see a progressive degradation in overlaps as we increase the target interval from $t \in [-60\ \text{M}, 130\ \text{M}]$ to $t \in [-100\ \text{M}, 130\ \text{M}]$. To explore the cause of this effect further, we trained three more models tasked with predicting only the segments $[-80\ \text{M}, 130\ \text{M}]$, $[-60\ \text{M}, 130\ \text{M}]$, and $[-50\ \text{M}, 130\ \text{M}]$ respectively. Let us call these models M80, M60, and M50 respectively. Then we noticed that the performance of M60 was slightly worse than M80 when predicting the same segment $[-60\ \text{M}, 130\ \text{M}]$, and so on. This hints at some of the loss in performance coming from the margin effect, i.e., -60 M is at the margin during training for M60 but not for M80, etc. However, these small variations are hard to quantify due to inherent stochasticity of training deep neural networks. But more importantly, the most significant degradation in performance came from increasing the prediction span from $[-60\ \text{M}, 130\ \text{M}]$ to $[-80\ \text{M}, 130\ \text{M}]$, and similarly from $[-80\ \text{M}, 130\ \text{M}]$ to $[-100\ \text{M}, 130\ \text{M}]$.

Interpretability: A nice side effect and a major advantage of the attention mechanism is that it enables us to visualize and try to interpret what is happening inside the model. Looking at Eqs (1) and (2) we notice that the coefficients w_{ij} form a $t \times n$ matrix A. The i*th* row of A consists of the attention scores over all the input vectors $\{x_1, x_2, x_3, \ldots, x_n\}$ when producing the ith output h_i, and hence each row sums up to 1. Therefore visualizing the ith row of matrix A shows which parts of the input $\{x_1, x_2, x_3, \ldots, x_n\}$ the model was "paying attention" to when generating the ith output h_i. Visualizing the whole matrix A then summarizes where the model was "looking at" when generating each time step of the output.

In this vein, we visualize the self-attention and cross-attention score matrices of the decoder module when

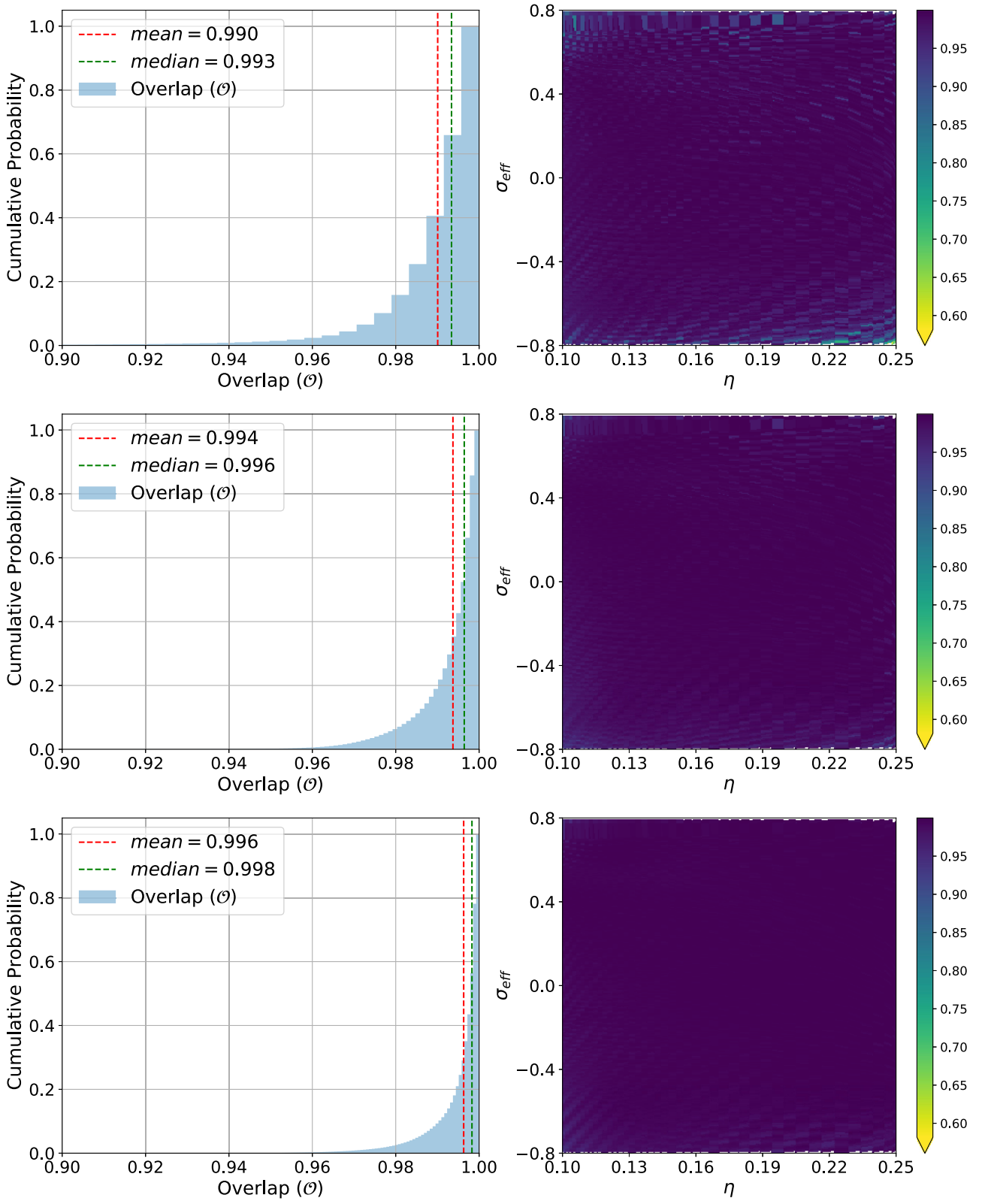

FIG. 6. (Left column) Cumulative distribution of overlaps between target and predicted waveforms. From top to bottom, we present results for our AI model predicting the waveform evolution from $t = \{-100\ \mathrm{M}, -80\ \mathrm{M}, -60\ \mathrm{M}\}$, respectively. (Right column) Heatmap of the overlap distribution over the entire test set. We present results in terms of the symmetric mass ratio and effective spin, $(\eta, \sigma_{\mathrm{eff}})$, as defined in Eq. (9).

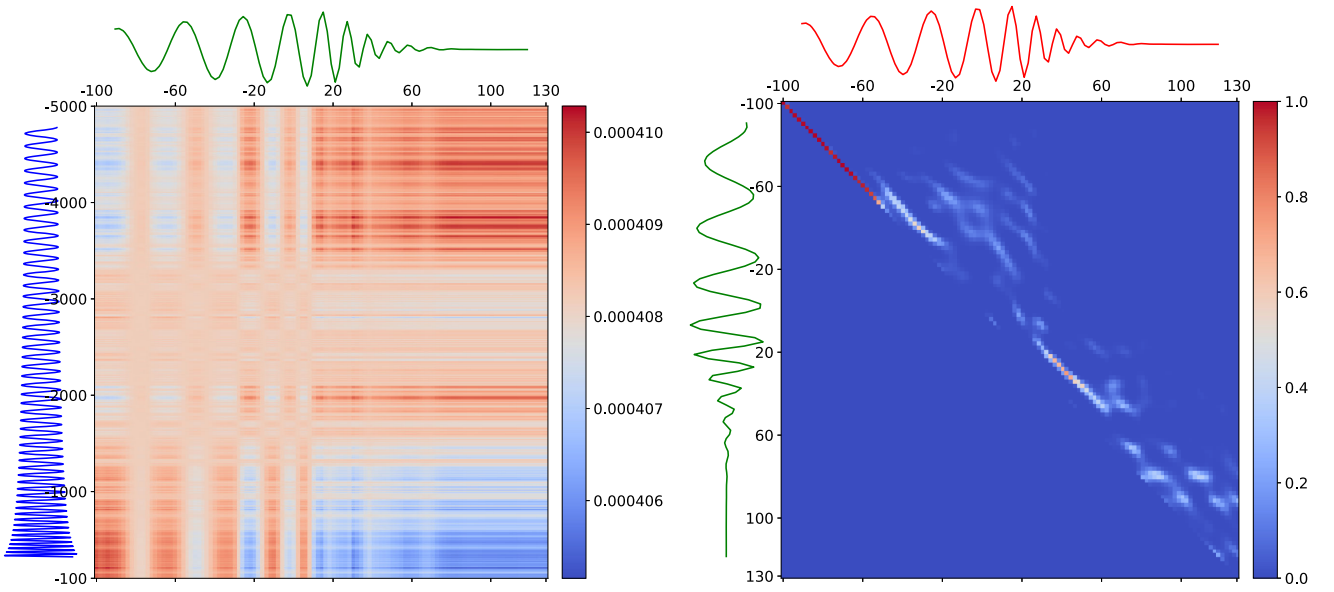

FIG. 7. (Left panel) Heatmap for one of the 12 *cross-attention* heads showing which parts of the input waveform (shown in blue on the left) the decoder is paying attention to when predicting the output at any particular time step (shown in green at the top). (Right panel) Heatmap showing one of the *self-attention* heads of the decoder.

generating the predictions for a sample waveform with parameters $\{q, s_1^z, s_2^z\} = \{6.8, 0.718, 0.574\}$ in Fig. 7. Therein we present results for one of the 12 attention heads from our model's decoder. We present additional results for the other attention heads in Appendix B.

The left panel of Fig. 7 shows the transpose of the cross-attention score matrix. Each column j shows which parts of the input waveform segment ($t \in [-5000\ \mathrm{M}, -100\ \mathrm{M}]$) the model was paying attention to when predicting the jth time step of the target waveform segment ($t_j \in [-100\ \mathrm{M}, 130\ \mathrm{M}]$). For reference, we also plot the input waveform segment and the predicted waveform segment to the left and top of the matrix, respectively. We see that for the late-inspiral and merger phases of the prediction, the model is paying a diffused form of attention to the whole input segment, occasionally flip flopping, i.e., paying more attention to the late inspiral rather than early inspiral and vice versa. However, when predicting the ringdown, all of the attention gets focused towards the early inspiral of the input segment.

The right panel of Fig. 7 shows the transpose of the self-attention matrix. Since predictions are generated autoregressively, the self-attention here is causal, i.e., at any given time step $t = j$, the model cannot pay attention to future time steps $t > j$. Consequently this matrix is upper triangular with a strong correlation between adjacent time steps, thus mostly diagonal.

The results in Fig. 7 provide a glimpse of the activity happening within our trained AI model that is responsible for accurate and reliable forecasting predictions. For the interested readers, we provide in the website [44] additional interactive results to enhance our intuition into how our AI model behaves for different astrophysical configurations.

IV. CONCLUSIONS

We have designed an AI model that is capable of learning and predicting the late-inspiral, merger and ringdown evolution of quasicircular, spinning, nonprecessing binary black hole mergers. The data-driven methodology used to create these AI tools demonstrates that AI can learn and accurately describe the plus- and cross polarizations of numerical relativity waveforms when we feed input signals that contain information up to $-100M$ before the merger event (defined as the amplitude peak of the waveform signal). We have also demonstrated that our AI model may forecast the waveform evolution starting at some other initial time t_i. In this study we presented quantitative results for the cases $t_i = -80$ M and $t_i = -60$ M. In all these cases, the mean and median overlap between target and predicted waveforms is $\mathcal{O} \geq 0.99$.

We have also explored visualizing several components in our AI model (i.e., the various attentions heads) that are responsible for data-driven decision-making and waveform forecasting. In particular, we generated visualizations to see which components of the input are responsible for the prediction of the premerger, merger and ringdown pieces of our predicted waveforms. We have made available an interactive website where users can explore these results in further detail for a variety of astrophysical systems. We expect that this approach persuades other researchers to go a step beyond and try to understand how AI models make predictions, and will help advance other efforts on creating interpretable AI models.

ACKNOWLEDGMENTS

A. K. and E. A. H. gratefully acknowledge National Science Foundation (NSF) Grants No. OAC-1931561 and No. OAC-1934757. This research used resources of the Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract No. DE-AC02-06CH11357. E. A. H. gratefully acknowledges the Innovative and Novel Computational Impact on Theory and Experiment project "Multi-Messenger Astrophysics at Extreme Scale in Summit" This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract No. DE-AC05-00OR22725. This work utilized resources supported by the NSF's Major Research Instrumentation program, the HAL cluster (Grant No. OAC-1725729), as well as the University of Illinois at Urbana-Champaign. We thank NVIDIA for their continued support.

APPENDIX A: TEACHER FORCING

Our model is designed to predict the waveform evolution in the time range $-100\ \mathrm{M} \leq t \leq 130\ \mathrm{M}$. During both training and inference, we compute the loss and quantify the performance of the model by comparing the entire predicted and ground-truth waveforms in the time segment $-100\ \mathrm{M} \leq t \leq 130\ \mathrm{M}$.

During both training and inference we feed the input waveform covering the time span $[-5000\ \mathrm{M}, -100\ \mathrm{M}]$ into

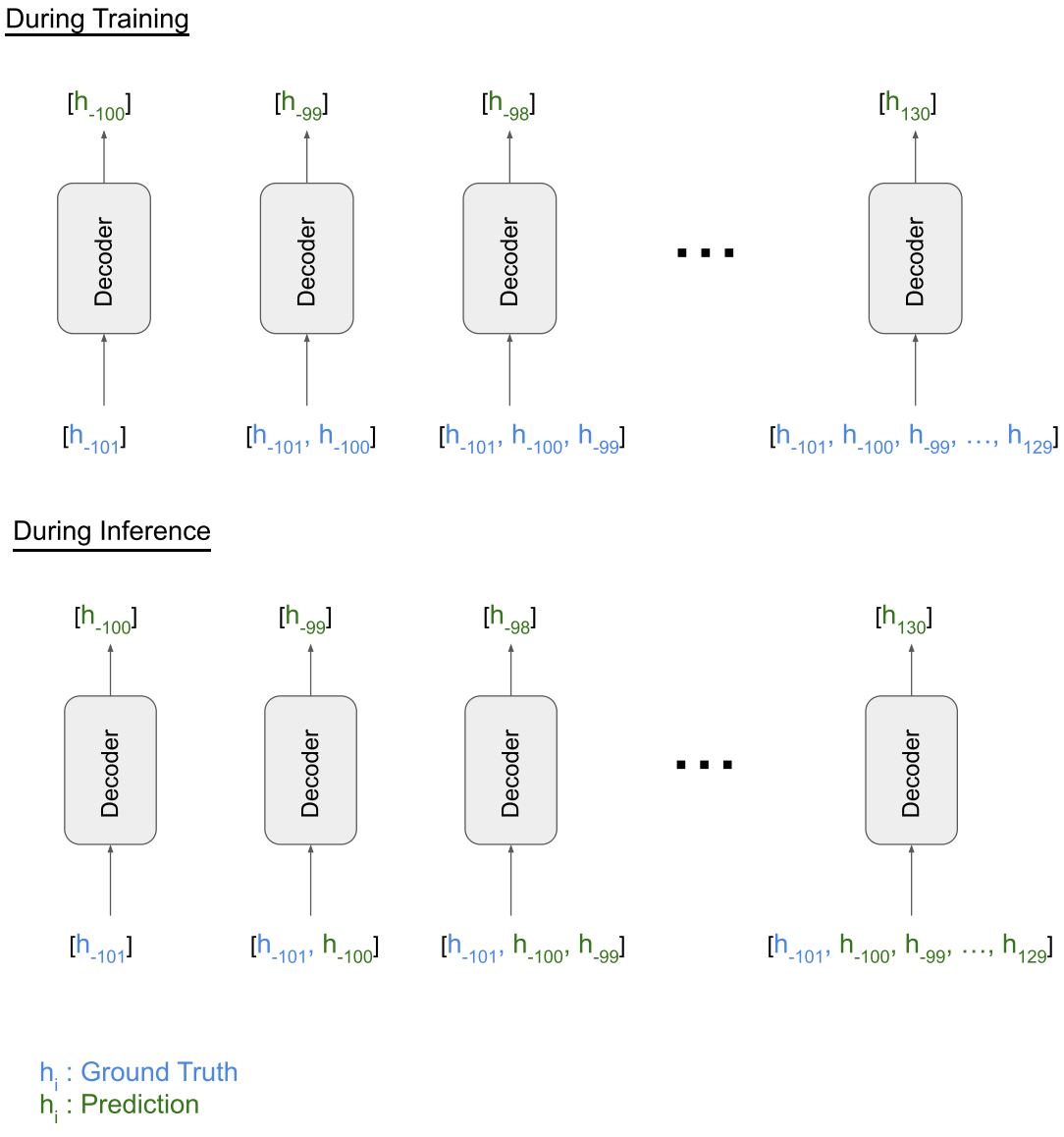

FIG. 8. A visual representation of Teacher Forcing approach. (Top panel) Teacher Forcing is used during training; at each time step the decoder is fed the ground-truth target values from the previous time steps. (Bottom panel) During inference, Teacher Forcing is turned off, and at each time step the decoder is fed its own predicted values from the previous time steps.

the encoder. Additionally, during training we also employ `Teacher Forcing`, whereby at each time step the decoder is fed the ground-truth target values from the previous time steps, as illustrated in the top panel of Fig. 8. This methodology results in a more stable training and helps the model converge faster. Finally, during inference we turn off `Teacher Forcing` and instead feed the decoder its own predictions from the previous time steps, as illustrated in the bottom panel of Fig. 8.

APPENDIX B: INTERPRETABILITY

We provide additional results for the 12 attention heads that our AI model utilizes for the forecasting of numerical relativity waveforms (Figs. 9 and 10). As in Fig. 7, we have produced these results for a binary black hole system with parameters $\{q, s_1^z, s_2^z\} = \{6.8, 0.718, 0.574\}$. For additional results, we refer readers to the interactive website [44].

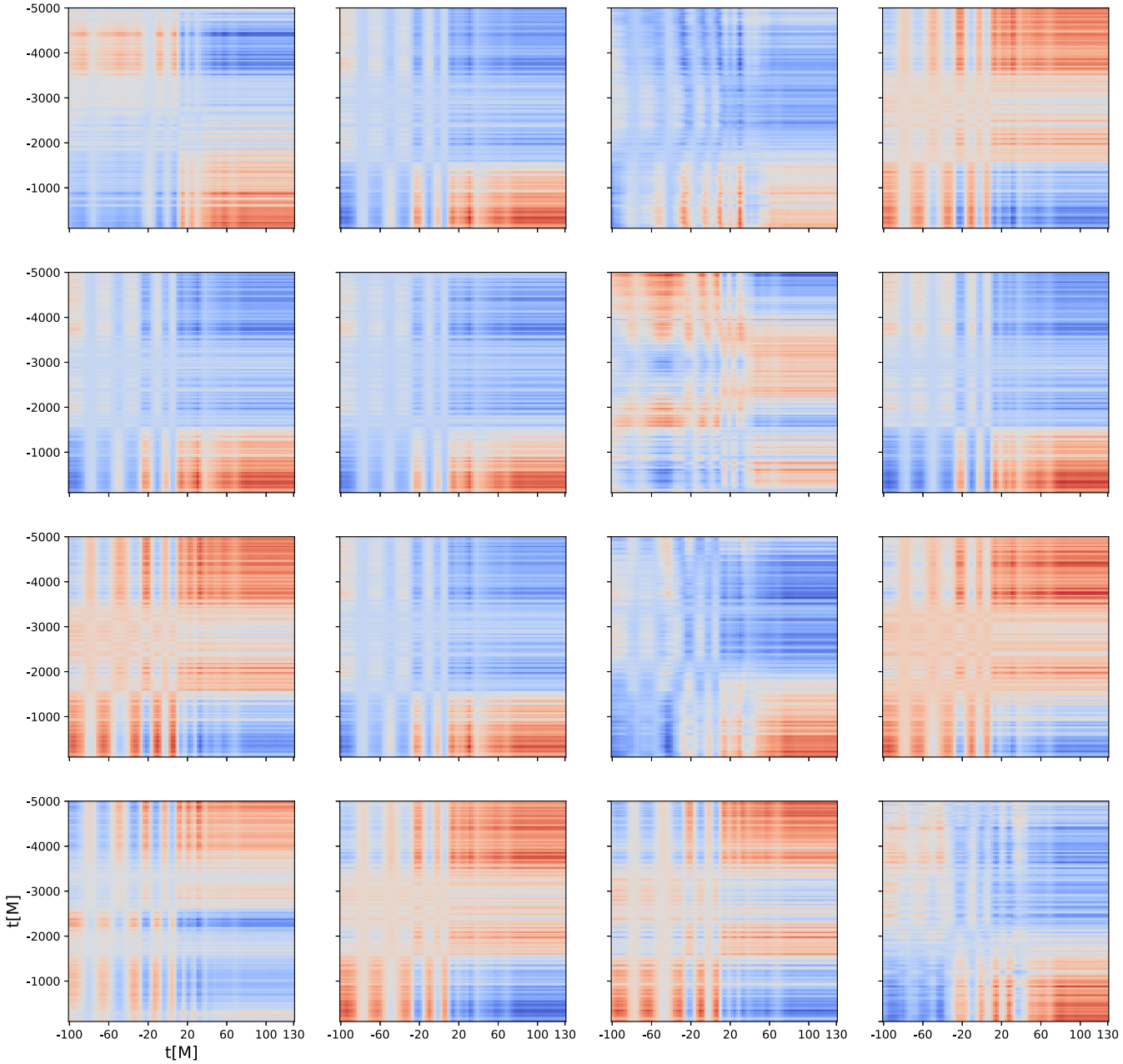

FIG. 9. Response of all cross-attention heads to a given input signal, indicating which parts of the input waveform signal are taken into account to forecast the late-inspiral, merger and ringdown evolution. This behavior is very consistent across the parameter space under consideration.

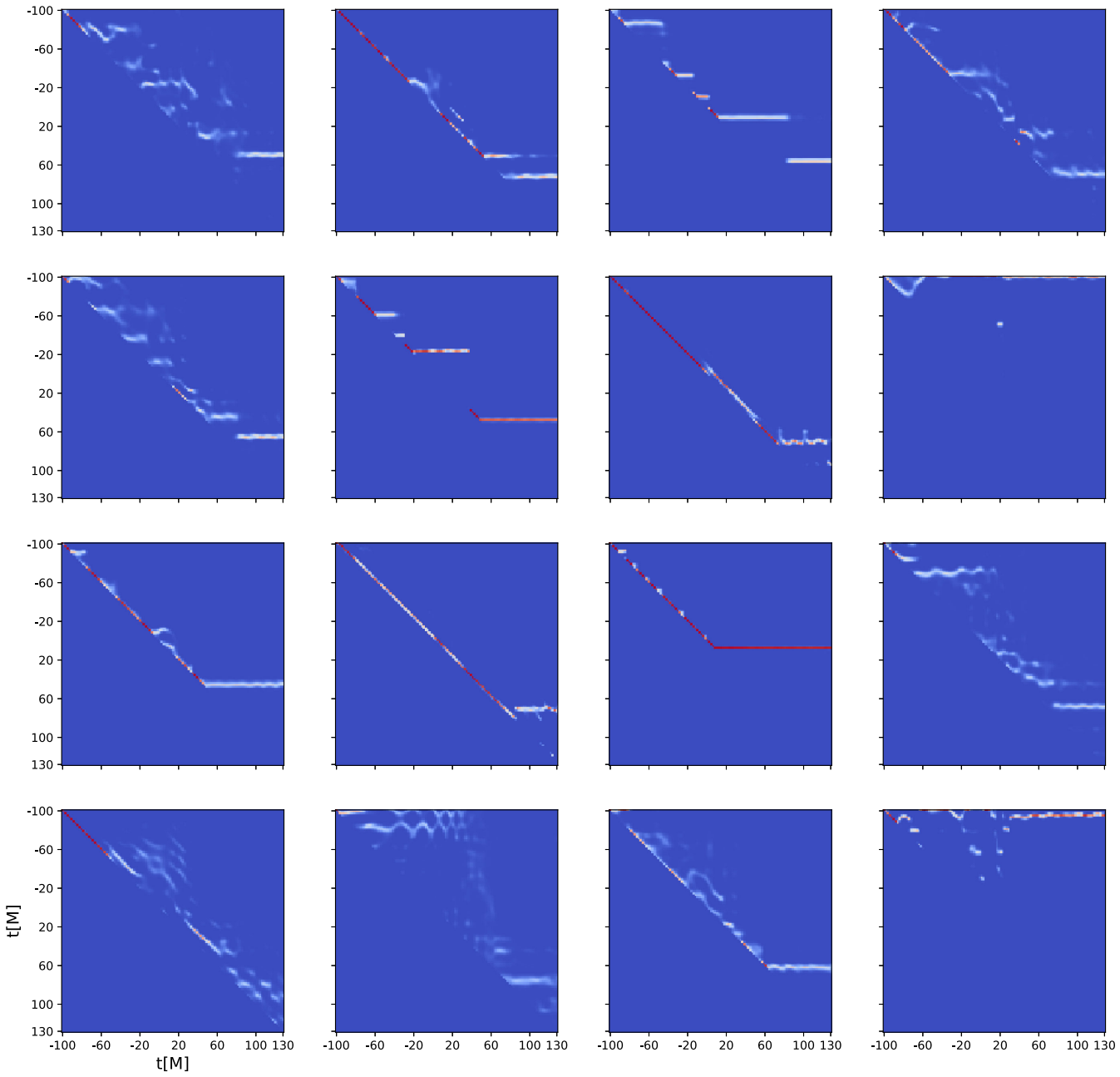

FIG. 10. As Fig. 9 but now for self-attention heads.

[1] D. George and E. A. Huerta, Deep neural networks to enable real-time multimessenger astrophysics, Phys. Rev. D **97**, 044039 (2018).

[2] 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).

[3] 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).

[4] V. Skliris, M. R. K. Norman, and P. J. Sutton, Real-time detection of unmodeled gravitational-wave transients using convolutional neural networks, arXiv:2009.14611.

[5] Y.-C. Lin and J.-H. P. Wu, Detection of gravitational waves using bayesian neural networks, Phys. Rev. D **103**, 063034 (2021).
[6] H. Wang, S. Wu, Z. Cao, X. Liu, and J.-Y. Zhu, Gravitational-wave signal recognition of LIGO data by deep learning, Phys. Rev. D **101**, 104003 (2020).
[7] X. Fan, J. Li, X. Li, Y. Zhong, and J. Cao, Applying deep neural networks to the detection and space parameter estimation of compact binary coalescence with a network of gravitational wave detectors, Sci. China Phys. Mech. Astron. **62**, 969512 (2019).
[8] X.-R. Li, G. Babu, W.-L. Yu, and X.-L. Fan, Some optimizations on detecting gravitational wave using convolutional neural network, Front. Phys. **15**, 54501 (2020).
[9] D. S. Deighan, S. E. Field, C. D. Capano, and G. Khanna, Genetic-algorithm-optimized neural networks for gravitational wave classification, arXiv:2010.04340.
[10] A. L. Miller *et al.*, How effective is machine learning to detect long transient gravitational waves from neutron stars in a real search?, Phys. Rev. D **100**, 062005 (2019).
[11] P. G. Krastev, Real-time detection of gravitational waves from binary neutron stars using artificial neural networks, Phys. Lett. B **803**, 135330 (2020).
[12] M. B. Schäfer, F. Ohme, and A. H. Nitz, Detection of gravitational-wave signals from binary neutron star mergers using machine learning, Phys. Rev. D **102**, 063015 (2020).
[13] C. Dreissigacker and R. Prix, Deep-learning continuous gravitational waves: Multiple detectors and realistic noise, Phys. Rev. D **102**, 022005 (2020).
[14] A. Rebei, E. A. 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).
[15] C. Dreissigacker, R. Sharma, C. Messenger, R. Zhao, and R. Prix, Deep-learning continuous gravitational waves, Phys. Rev. D **100**, 044009 (2019).
[16] B. Beheshtipour and M. A. Papa, Deep learning for clustering of continuous gravitational wave candidates, Phys. Rev. D **101**, 064009 (2020).
[17] M. B. Schäfer and A. H. Nitz, From one to many: A deep learning coincident gravitational-wave search, arXiv:2108.10715.
[18] H. Shen, D. George, E. A. Huerta, and Z. Zhao, Denoising gravitational waves with enhanced deep recurrent denoising auto-encoders, in *Proceedings of the ICASSP 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2019* (IEEE, Brighton, United Kingdom, 2019), pp. 3237–3241.
[19] W. Wei and E. A. Huerta, Gravitational wave denoising of binary black hole mergers with deep learning, Phys. Lett. B **800**, 135081 (2020).
[20] R. Ormiston, T. Nguyen, M. Coughlin, R. X. Adhikari, and E. Katsavounidis, Noise reduction in gravitational-wave data via deep learning, Phys. Rev. Research **2**, 033066 (2020).
[21] H. Shen, E. A. Huerta, E. O'Shea, P. Kumar, and Z. Zhao, Statistically-informed deep learning for gravitational wave parameter estimation, Mach. Learn. Sci. Tech. **3**, 015007 (2022).
[22] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy, Nat. Phys. (2021), 10.1038/s41567-021-01425-7.
[23] A. J. Chua and M. Vallisneri, Learning Bayesian Posteriors with Neural Networks for Gravitational-Wave Inference, Phys. Rev. Lett. **124**, 041102 (2020).
[24] S. R. Green, C. Simpson, and J. Gair, Gravitational-wave parameter estimation with autoregressive neural network flows, Phys. Rev. D **102**, 104057 (2020).
[25] S. R. Green and J. Gair, Complete parameter inference for GW150914 using deep learning, Mach. Learn. **2**, 03LT01 (2021).
[26] M. Dax, S. R. Green, J. Gair, J. H. Macke, A. Buonanno, and B. Schölkopf, Real-Time Gravitational-Wave Science with Neural Posterior Estimation, Phys. Rev. Lett. **127**, 241103 (2021).
[27] S. Khan and R. Green, Gravitational-wave surrogate models powered by artificial neural networks, Phys. Rev. D **103**, 064015 (2021).
[28] A. J. K. Chua, C. R. Galley, and M. Vallisneri, Reduced-Order Modeling with Artificial Neurons for Gravitational-Wave Inference, Phys. Rev. Lett. **122**, 211101 (2019).
[29] W. Wei, E. A. Huerta, M. Yun, N. Loutrel, M. A. Shaikh, P. Kumar, R. Haas, and V. Kindratenko, Deep learning with quantized neural networks for gravitational-wave forecasting of eccentric compact binary coalescence, Astrophys. J. **919**, 82 (2021).
[30] W. Wei and E. A. Huerta, Deep learning for gravitational wave forecasting of neutron star mergers, Phys. Lett. B **816**, 136185 (2021).
[31] H. Yu, R. X. Adhikari, R. Magee, S. Sachdev, and Y. Chen, Early warning of coalescing neutron-star and neutron-star-black-hole binaries from the nonstationary noise background using neural networks, Phys. Rev. D **104**, 062004 (2021).
[32] 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).
[33] W. Wei, A. Khan, E. A. Huerta, X. Huang, and M. Tian, Deep learning ensemble for real-time gravitational wave detection of spinning binary black hole mergers, Phys. Lett. B **812**, 136029 (2021).
[34] E. A. Huerta, A. Khan, X. Huang, M. Tian, M. Levental, R. Chard, W. Wei, M. Heflin, D. S. Katz, V. Kindratenko, D. Mu, B. Blaiszik, and I. Foster, Accelerated, scalable and reproducible AI-driven gravitational wave detection, Nat. Astron. **5**, 1062 (2021).
[35] E. A. Huerta *et al.*, Enabling real-time multi-messenger astrophysics discoveries with deep learning, Nat. Rev. Phys. **1**, 600 (2019).
[36] E. A. Huerta and Z. Zhao, *Advances in Machine and Deep Learning for Modeling and Real-Time Detection of Multi-messenger Sources* (Springer Singapore, Singapore, 2020), pp. 1–27.
[37] E. Cuoco *et al.*, Enhancing gravitational-wave science with machine learning, Mach. Learn. **2**, 011002 (2021).
[38] S. G. Rosofsky and E. A. Huerta, Artificial neural network subgrid models of 2D compressible magnetohydrodynamic turbulence, Phys. Rev. D **101**, 084024 (2020).

[39] Y. Zhu, N. Zabaras, P.-S. Koutsourelakis, and P. Perdikaris, Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data, J. Comput. Phys. **394**, 56 (2019).

[40] R. Anirudh, J. J. Thiagarajan, P.-T. Bremer, and B. K. Spears, Improved surrogates in inertial confinement fusion with manifold and cycle consistencies, Proc. Natl. Acad. Sci. U.S.A. **117**, 9741 (2020).

[41] J. Lee, S. H. Oh, K. Kim, G. Cho, J. J. Oh, E. J. Son, and H. M. Lee, Deep learning model on gravitational waveforms in merging and ringdown phases of binary black hole coalescences, Phys. Rev. D **103**, 123023 (2021).

[42] 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).

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, Attention is all you need, arXiv:1706.03762.

[44] A. Khan, E. A. Huerta, and H. Zheng, Interpretable AI forecasting for numerical relativity waveforms of quasi-circular, spinning, non-precessing binary black hole mergers, https://khanx169.github.io/gw_forecasting/interactive_results.html, 2021.