Generating gravitational waveform libraries of exotic compact binaries with deep learning

Felipe F. Freitas,[1,2,*] Carlos A. R. Herdeiro,[3,2,†] António P. Morais,[1,2,‡] António Onofre,[4,§]
Roman Pasechnik,[5,∥] Eugen Radu,[3,2,¶] Nicolas Sanchis-Gual,[3,2,6,**] and Rui Santos[7,8,††]

[1] *Departamento de Física, Universidade de Aveiro, Campus de Santiago, 3810-183 Aveiro, Portugal*
[2] *Centre for Research and Development in Mathematics and Applications (CIDMA),*
Campus de Santiago, 3810-183 Aveiro, Portugal
[3] *Departamento de Matemática, Universidade de Aveiro, Campus de Santiago, 3810-183 Aveiro, Portugal*
[4] *Centro de Física das Universidades do Minho e do Porto (CF-UM-UP), Universidade do Minho,*
4710-057 Braga, Portugal
[5] *Department of Astronomy and Theoretical Physics, Lund University, 221 00 Lund, Sweden*
[6] *Departamento de Astronomía y Astrofísica, Universitat de València,*
Dr. Moliner 50, 46100, Burjassot (València), Spain
[7] *ISEL—Instituto Superior de Engenharia de Lisboa,*
Instituto Politécnico de Lisboa 1959-007 Lisboa, Portugal
[8] *Centro de Física Teórica e Computacional, Faculdade de Ciências, Universidade de Lisboa,*
Campo Grande, Edifício C8 1749-016 Lisboa, Portugal

(Received 9 October 2023; accepted 6 May 2024; published 25 June 2024)

Current gravitational wave (GW) detections rely on the existence of libraries of theoretical waveforms. Consequently, finding new physics with GWs requires libraries of nonstandard models, which are computationally demanding. We discuss how deep learning frameworks can be used to generate new waveforms "learned" from a simulation dataset obtained, say, from numerical relativity simulations. Concretely, we use the WaveGAN architecture of a generative adversarial network (GAN). As a proof of concept we provide this neural network (NN) with a sample of (>500) waveforms from the collisions of exotic compact objects (Proca stars), obtained from numerical relativity simulations. Dividing the sample into a training and a validation set, we show that after a sufficiently large number of training epochs the NN can produce from 12% to 25% of the synthetic waveforms with an overlapping match of at least 95% with the ones from the validation set. We also demonstrate that a NN can be used to predict the overlapping match score, with 90% accuracy, of new synthetic samples. These are encouraging results for using GANs for data augmentation and interpolation in the context of GWs, to cover the full parameter space of, say, exotic compact binaries, without the need for intensive numerical relativity simulations.

DOI: 10.1103/PhysRevD.109.124059

I. INTRODUCTION

The advent of the gravitational wave (GW) era [1–3] opens new possibilities not only for relativistic astrophysics, cosmology, and strong gravity, but also for fundamental physics. Alongside the unveiling of the population of black holes and neutron stars in the Universe (see, e.g., [4]), it is possible that smoking guns about the nature of dark energy, dark matter, and even quantum gravity will emerge from this new channel. In fact, particular events have already established concrete illustrations of how dark energy models can be constrained (see, e.g., [5–7]) and dark matter could be identified (see, e.g., [8]).

The interpretation (and parameter estimation) of GW signals relies on matched filtering [9]. Therefore, libraries of theoretical templates are mandatory. The construction of such libraries is a nontrivial, time-consuming process. For the vanilla black hole binary problem in (vacuum) general relativity, the construction of full waveforms, including inspiral, merger, and ringdown, became under control after the numerical relativity breakthroughs of 2005—see the review in [10]. However, a dense scanning of the full parameter space (of the black hole binary problem) only with numerical relativity simulations is computationally impossible. Thus, a community effort drove the scanning of

[*] felipefreitas@ua.pt
[†] herdeiro@ua.pt
[‡] aapmorais@ua.pt
[§] Antonio.Onofre@cern.ch
[∥] roman.pasechnik@thep.lu.se
[¶] eugen.radu@ua.pt
[**] nicolas.sanchis@uv.es
[††] rasantos@fc.ul.pt

the parameter space using state of the art numerical relativity simulations, patched together with approximation methods for both the inspiral and the ringdown—see, e.g., [11–14], building approximants for the theoretical waveforms with generic parameters.

An underemphasized caveat of current GW interpretations is the degeneracy problem: can nonstandard waveforms fit the data better? Nonstandard means waveforms from exotic compact objects, which could either be non-Kerr black holes or horizonless compact objects. Moreover, such exotic compact objects could originate either from general relativity with matter sources or from modified gravity. The difficulty in tackling the above question is, however, the almost complete lack of alternative waveform libraries that can be compared with real events to determine whether the vanilla (that is, Kerr black holes and neutron star binaries) waveforms are indeed the ones selected within a larger library, when employing matched filtering and Bayesian analysis.

At the time of writing, the one nonstandard model of compact binaries for which there has been a more consistent and successful effort to produce waveforms is the case of bosonic (i.e., scalar [15–17] or vector [18–20]) stars. The dynamical evolution of these models is theoretically and technically under control [21] and presents a variety of motivations: bosonic stars emerge in sound physical models, can be dynamically robust [21,22], and have been put forward as “fuzzy” dark matter [23] lumps and black hole imitators, e.g., [24–26]. In the context of GWs, several studies of waveforms from collisions and binaries of bosonic stars have been reported, e.g., [27–30]. As an application to the ongoing detections, the massive GW event GW190521 [31] was shown to fit well a collision of two vector bosonic (also known as Proca) stars [8]. This effort relied on scanning a library of 89 Proca star collision waveforms (in the meantime enlarged to nearly 800 waveforms) [32]. Still, this only scratches the surface of the full parameter space of the model. As such, looking for efficient computational methodologies that can transform a coarse sampling of the parameter space into a dense coverage is of paramount importance.

The goal of this paper is to start an exploration of such a methodology using deep learning techniques. Moreover, the method can, in principle, be used for waveforms produced from generic nonstandard compact binaries. Thus, the Proca model explicitly discussed herein can be taken both as interesting in its own right, but simultaneously as a proof of concept of the application of the method, illustrating it but not exhausting it. To be concrete, we shall be making use of generative adversarial networks (GANs) [33], a particular class of deep learning frameworks.

GANs can be described as unsupervised methods for mapping low-dimensional latent vectors to high-dimensional data. In our case, this means mapping known waveforms, corresponding to a prior distribution, p_{model}, to a larger space of waveforms, the generated data distribution, p_{data}. In a nutshell, GANs are based on a game-theoretic scenario where we have two networks competing against each other. On the one hand, we have the *generator*, responsible for mapping the low dimensional vector $\mathbf{z}$ into the high dimensional samples we want to reproduce $\mathbf{x} = g(\mathbf{z}, \theta^g)$ (i.e., the waveforms in our case). Here, θ^g are the parameters from the generator network to be adjusted during the training phase. Competing against the generator we have, on the other hand, the *discriminator* network, whose sole purpose is to distinguish between samples drawn from the original dataset and samples drawn from the generator. The discriminator provides a probability, $d(\mathbf{x}; \theta^d) \in [0.0, 1.0]$, of a given sample $\mathbf{x}$ being real, as opposed to a fake one drawn from the generator model. Here, θ^d are the parameters from the discriminator network to be adjusted during the training phase.

The simplest way to describe the learning process of a GAN is a zero-sum game, in which a function $\mathcal{L}(\theta^g, \theta^d)$ determines the payoff of the discriminator. The generator receives $-\mathcal{L}(\theta^g, \theta^d)$ as its own payoff. During the training phase, each player attempts to maximize its own payoff, so that the generator is trained to maximize $\mathcal{L}(\theta^g, \theta^d)$, whereas the discriminator is trained to minimize it.

The original proposal [33] for the function $\mathcal{L}(\theta^g, \theta^d)$ is

$$\arg\min_g \max_d \mathcal{L}(\theta^g, \theta^d) = \mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}[\log d(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_{\mathrm{model}}}[\log(1 - d(g(z))], \quad (1.1)$$

where $\mathbb{E}_{\mathbf{x}\sim p_{\mathrm{data}}}$ and $\mathbb{E}_{\mathbf{z}\sim p_{\mathrm{model}}}$ are the expected values for a sample to be drawn from the data and the generator, respectively.

This drives the discriminator to learn to correctly classify samples as real or fake. Meanwhile, the generator attempts to fool the discriminator by producing fake samples with features as close as possible to the features from real samples. At convergence, the generator's samples are indistinguishable from the real ones, and the discriminator outputs a probability of 50% for every sample. The discriminator may be discarded or its parameters can be reused for other purposes later on.

GANs have shown great success in generating high-quality synthetic images [34–36] indistinguishable from real images. This has encouraged the use of GANs for synthetic data generation in broader contexts, in particular in high-energy physics, where in some instances the data generation can be a computational intensive task [37–40]. In this regard, while GANs were developed for image generation [33], there have been attempts to adapt this approach for other formats, such as tabular data [34], time series [41], video content augmentation [42], and audio synthesis [43–45].

In this article, we shall examine the potential of GANs to produce a larger waveform catalog from a limited dataset of the corresponding waveforms. We shall focus on the case of waveforms produced by Proca star binaries. For this purpose, we shall modify *WaveGAN* [45], a GAN initially designed to provide an unsupervised synthesis of raw-waveform audio, such that it could learn and produce Proca waveforms from an initial dataset obtained from numerical relativity simulations, i.e., standard audio. Dividing the sample into a training and a validation set, we show that after a sufficiently large number of training epochs the neural network (NN) can generate from 12% to 25% of the synthetic data with at least 95% overlapping match with reference samples from the validation set.[1]

This article is organized as follows. In Sec. II, we briefly review the Proca model of bosonic stars and describe the dataset and methodology explored in this study. We also discuss the issue of waveform normalization. In Sec. III, we describe the WaveGAN architecture and the training methodology. Then, in Sec. IV we discuss the evaluation methodology, i.e., how to assess the quality of the generated waveforms and use the trained discriminator architectures to predict the match score for new synthetic samples. Section V presents our results after applying the chosen architecture, training, and evaluation to the initial dataset. Finally, Sec. VI provides a final discussion on the approach proposed herein.

II. THE PROCA MODEL AND THE DATASET

The Proca stars, their dynamics, and the corresponding GWs will be considered in the simplest model: a complex, massive Proca field minimally coupled to Einstein's gravity. The action reads (with $c = 1 = G$)

$$\mathcal{S} = \int d^4x\sqrt{-g}\left[\frac{R}{16\pi} - \frac{1}{4}\mathcal{F}_{\alpha\beta}\bar{\mathcal{F}}^{\alpha\beta} - \frac{\mu^2}{2}\mathcal{A}_\alpha\bar{\mathcal{A}}^\alpha\right], \qquad (2.1)$$

where R is the Ricci scalar of the spacetime metric g, $\mathcal{A}$ is a complex four-potential, with the field strength $\mathcal{F}_{\alpha\beta} = \partial_\alpha\mathcal{A}_\beta - \partial_\beta\mathcal{A}_\alpha$, $\mu > 0$ corresponds to the mass of the Proca field, and the overbar denotes complex conjugation.

Spinning Proca stars (the fundamental solutions, in the stable branch [22]) can be labeled by their Arnowitt-Deser-Misner (ADM) mass, $M\mu$ or, alternatively, by their oscillating frequency ω/μ, both in units of the Proca field mass. In the following, for simplicity, we shall set $\mu = 1$ and label the solutions via M. The fundamental solutions in the stable branch have M and ω in the interval(s) [20]:

$$(M, \omega) \in ([0, 1.125], [0.469, 1]). \qquad (2.2)$$

Note that the upper (lower) limit in the M interval corresponds to the lower (upper) limit in the ω interval. The angular momentum of the solutions is determined by M. For the considered solutions the total angular momentum is in the range [20] $J \in [0, 1.259]$.

The collision of two Proca stars generates GWs. These are extracted via the Newman-Penrose (complex) scalar Ψ_4. Both the real $[\mathcal{R}(\Psi_4)]$ and imaginary parts of this scalar (corresponding to the two GW polarizations) can be decomposed into harmonics. The dominant GW modes, i.e., with higher amplitude, have harmonic indices $(l, m) = (2, 2)$ and $(l, m) = (2, 0)$. For simplicity, we shall consider only the $(l, m) = (2, 2)$ waveforms for each collision, focusing on the real part of the scalar (the "+" polarization). Each waveform is a time series for $r\mathcal{R}(\Psi_4)$, since Ψ_4 falls as $1/r$, with r being the distance to the source.

Our dataset consists of waveforms generated from the merger of two spinning Proca stars with aligned spin axes. These sorts of collisions were recently studied in [8,47]. Although the stars start from rest, due to frame dragging the binary describes an eccentric (rather than precisely head on) trajectory. The end point depends on the progenitor Proca stars. In the region of the parameter space explored here, the Proca star progenitors are sufficiently massive to trigger black hole formation after the merger.

The waveforms are generated from numerical evolutions using the Einstein Toolkit infrastructure [48–50], together with the CARPET package [51,52] for mesh refinement. The Proca evolution equations are solved via a modified Proca thorn [22,30,53,54] to include a complex field. We have performed numerical simulations of equal and unequal mass Proca stars. The initial data consists in the superposition of two equilibrium solutions separated by $D = 40/\mu$ [8], in geometrized units. This guarantees an admissible initial constraint violation. The equilibrium spinning Proca stars are numerically constructed using the solver FIDISOL/CADSOL for nonlinear partial differential equations of elliptic type, via a Newton Raphson method—see [18–20] for more details.

We divide our data into two sets:

(i) One set contains 98 waveforms generated from the merger of two equal mass Proca stars ($M_1 = M_2$). For each collision, we consider waveforms of the $(l, m) = (2, 2)$ mode.

(ii) The other dataset consists of 457 waveforms from the merger of two unequal mass Proca stars ($M_1 \neq M_2$), with the same $(l, m) = (2, 2)$ mode.

In Figs. 1 and 2, some samples for both datasets are illustrated, for the dominant quadrupolar mode $(l, m) = (2, 2)$.[2] In Fig. 3 we display the mass distribution for both datasets [equal mass case in Fig. 3(a) and different mass in Fig. 3(b)]. Both datasets are preprocessed to be sampled at

[1] We remark that using other data-driven methods like surrogate models can produce better matches for 1D spaces—see, e.g., [46]; but GANs offer a novel methodology with substantial room to improve.

[2] We remark that the junk radiation in these simulations is small and it is not causally connected to the physical gravitational-wave burst emitted during the collision. They are both separated by a retarded time interval of $\Delta u \sim 200$—see Fig. 16 in [55].

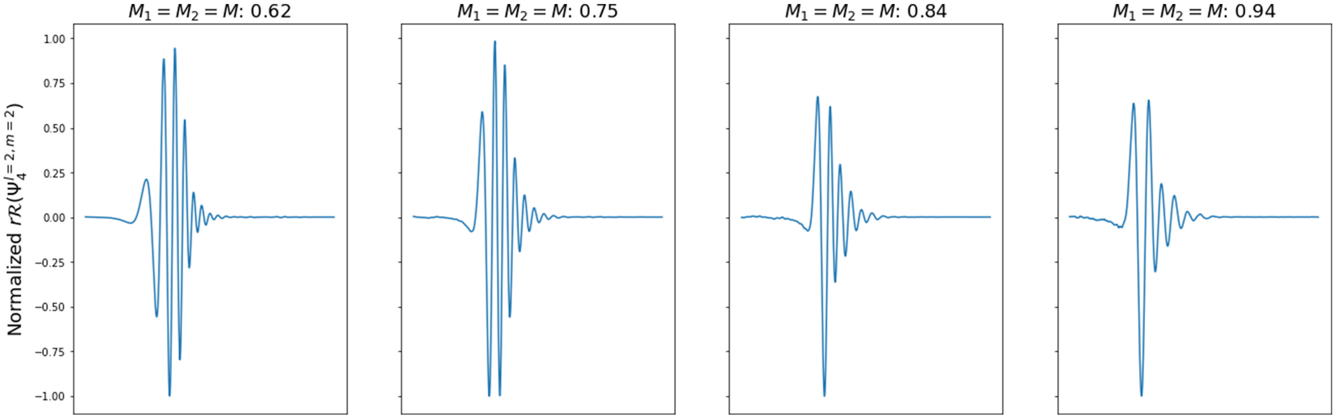


FIG. 1. Samples of $r\mathcal{R}(\Psi_4^{l=2,m=2})$ for the equal mass dataset ($M = M_1 = M_2$). The amplitude of $r\mathcal{R}(\Psi_4^{l=2,m=2})$ is normalized to be within the range of $[-1.0, 1.0]$.

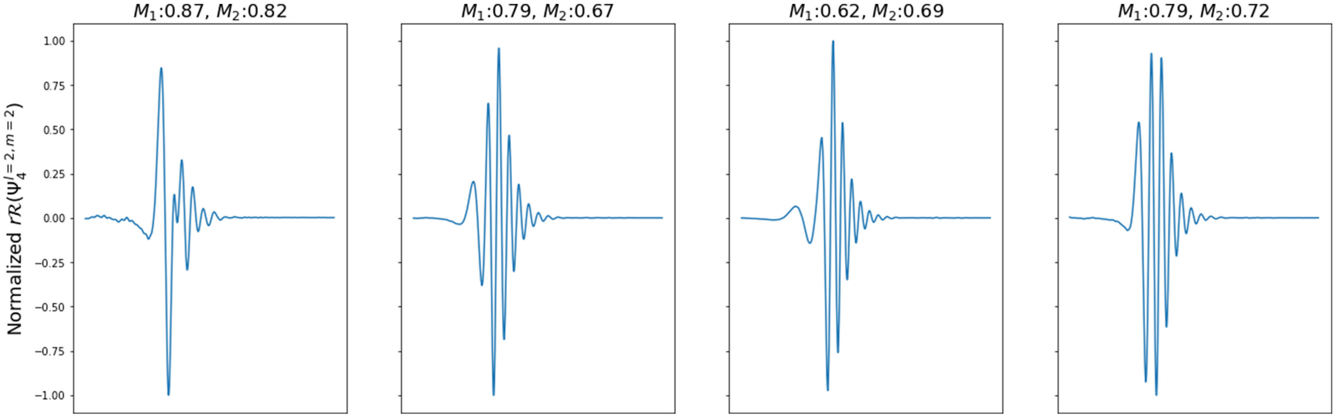


FIG. 2. Samples of $r\mathcal{R}(\Psi_4^{l=2,m=2})$ for the unequal mass dataset ($M_1 \neq M_2$). The amplitude of $r\mathcal{R}(\Psi_4^{l=2,m=2})$ is normalized to be within the range of $[-1.0, 1.0]$.

2048 Hz. Because of the feature that the dataset have different y ranges we need to normalize them. Having the samples scaled to a similar range helps to prevent or at least mitigate bias and to speed up the optimization process by preventing the model parameter weights to either vanish or explode [56]. We have tested different methods of scaling, including standard scaling,[3] Robust Scaler,[4] the min-max scaler,[5] and max-absolute scaler[6] which scale the features into the $[-1.0, 1.0]$ range without breaking the sparsity of the dataset. We have chosen to scale the datasets according to the max-absolute, since we want to preserve the sparsity of our dataset. It is important to mention that all features are scaled only after the train/validation split happens, to avoid any bias in the training procedure, and their true amplitude ranges are stored for a later use to transform back the normalized samples into their original values. Then each dataset is shuffled and split into training (80% of the total dataset) and validation (20% of the total dataset) datasets; these are then fed into the NN model, as further explained in the next section.

Each sample consist of a time series representing the real part of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$, together with the value of the mass—as shown in Fig. 1—and the feature scale.

III. MODEL ARCHITECTURE AND TRAINING METHODOLOGY

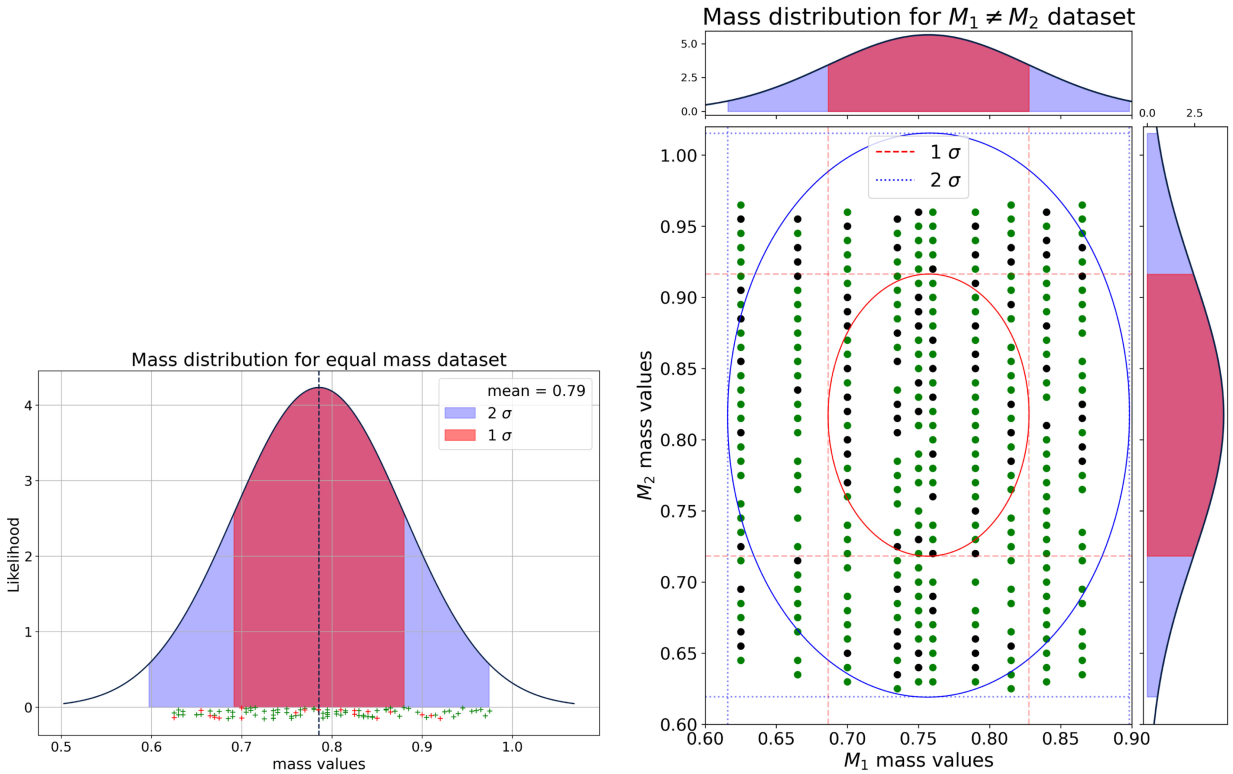
Inspired by the use GANs for audio generation, we employ WaveGAN [45] to generate new waveforms “learned” from our simulations dataset. Our purpose is

[3] $x' = \frac{x-\mu}{\sigma}$, where x' is the transformed feature, μ and σ are the features’ average and standard deviation from the dataset, respectively.

[4] $x' = \frac{x-\mu_{1/2}}{p75-p25}$, where x' is the transformed feature, $\mu_{1/2}$ is the median, and $(p75 - p25)$ is the interquartile range which is the difference between the 75th and 25th percentiles.

[5] $x'[a,b] = a + \frac{(x-\min x)(b-a)}{\max x-\min x}$, where x' is the transformed feature, a and b determine an arbitrary desired interval for the final feature range.

[6] $x' = \frac{x}{\max|x|}$.



(a) (b)

FIG. 3. Mass distribution for the equal (a) and unequal (b) mass dataset with one (red) and two (blue) sigmas. (a) The green and red crosses at the bottom show the sample values for the training and validation datasets, respectively. (b) The green and black dots display the sample values for the training and validation datasets, respectively, while the red and blue ellipses display the one and two sigmas regions for M_1 and M_2 distributions. (a) $M_1 = M_2$ dataset. The green (red) crosses represent the samples for the training (validation) set. (b) $M_1 \neq M_2$ dataset. The green (black) points represent the samples for the training (validation) set.

to test whether this method is useful for data augmentation and data interpolation to cover the full parameter space without the need for intensive computational simulations.

The WaveGAN architecture (see Table I) is based on deep convolutional GAN (DCGAN) [57] which popularized the usage of GANs for image synthesis. The DCGAN generator uses the *transposed convolution* operation to iteratively upsample low-resolution feature maps into high-resolution images. The WaveGAN uses a modified *transposed convolution* operation to widen its receptive fields.[7] We keep the longer one-dimensional filters of 25, as proposed in [45], however we set the number of layers to 4 and channels to 1 at first, and upsample by a factor of 4 at each layer. The discriminator network is modified in a similar fashion, using length-25 filters. The output length from the generator, as well as the input length to the discriminator, is set to 2048, to be the same length as the waveform samples.

The usual GANs generate samples similar to the ones learned in the training. However, this approach is not the most practical if one wants to produce synthetic samples from particular classes present in the dataset, i.e., augment

TABLE I. WaveGAN generator architecture.

Operation	Kernel size	Output shape
Input $z \sim \text{uniform}(-1, 1)$		$(n, 100)$
Dense 1	$(100, 256d)$	$(n, 256d)$
Reshape		$(n, 16, 16d)$
ReLU		$(n, 16, 16d)$
Trans Conv1D (stride = 4)	$(25, 16d, 8d)$	$(n, 64, 8d)$
ReLU		$(n, 64, 8d)$
Trans Conv1D (stride = 4)	$(25, 8d, 4d)$	$(n, 256, 4d)$
ReLU		$(n, 256, 4d)$
Trans Conv1D (stride = 4)	$(25, 4d, 2d)$	$(n, 1024, 2d)$
ReLU		$(n, 1024, 2d)$
Trans Conv1D (stride = 4)	$(25, 2d, d)$	$(n, 4096, d)$
ReLU		$(n, 4096, d)$
Trans Conv1D (stride = 4)	$(25, d, c)$	$(n, 2048, c)$
Tanh		$(n, 2048, c)$

[7] The receptive field in convolutional neural networks is the region of the input space that affects a particular unit of the network.

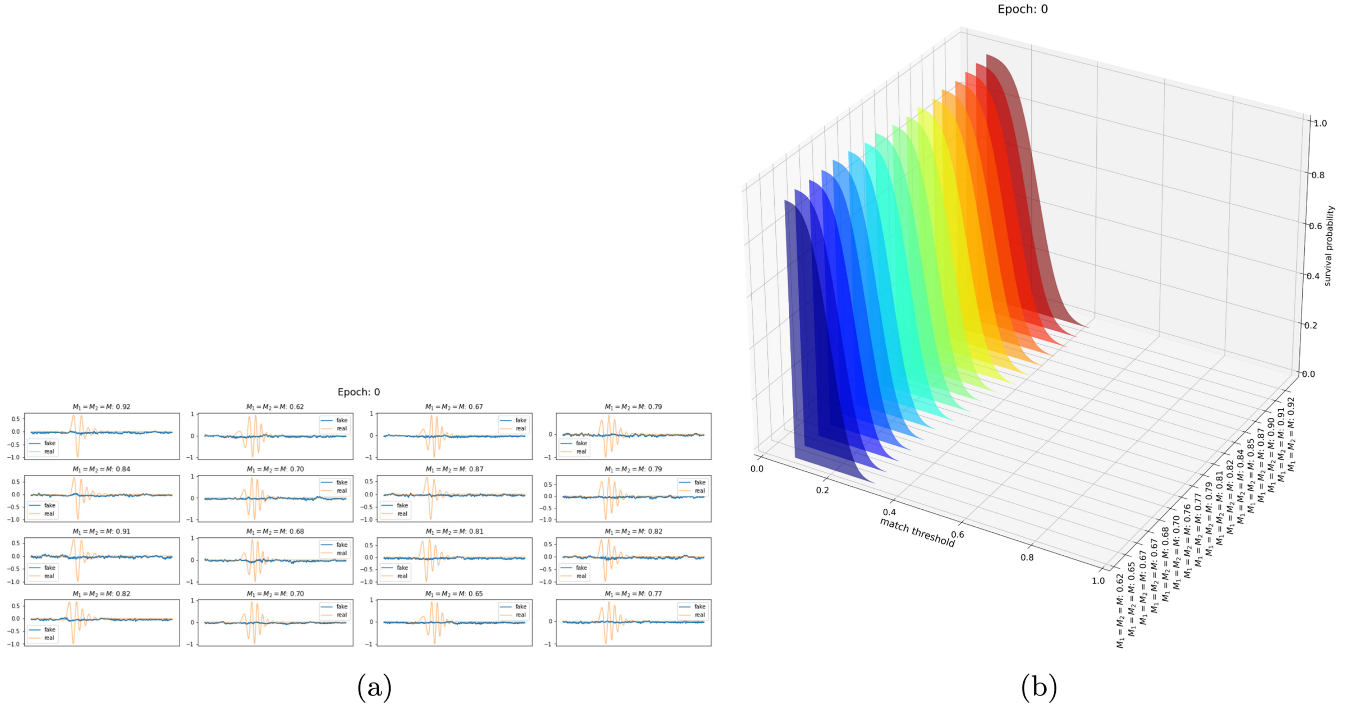


FIG. 4. Equal mass dataset, epoch 0. Evaluation of the quality of the generated samples compared to the real samples of $r\mathcal{R}(\Psi_4^{l=2,m=2})$ and probabilities to generate good quality samples. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $r\mathcal{R}(\Psi_4^{l=2,m=2})$. (b) Probabilities of generating samples for a given match accuracy for different mass parameters.

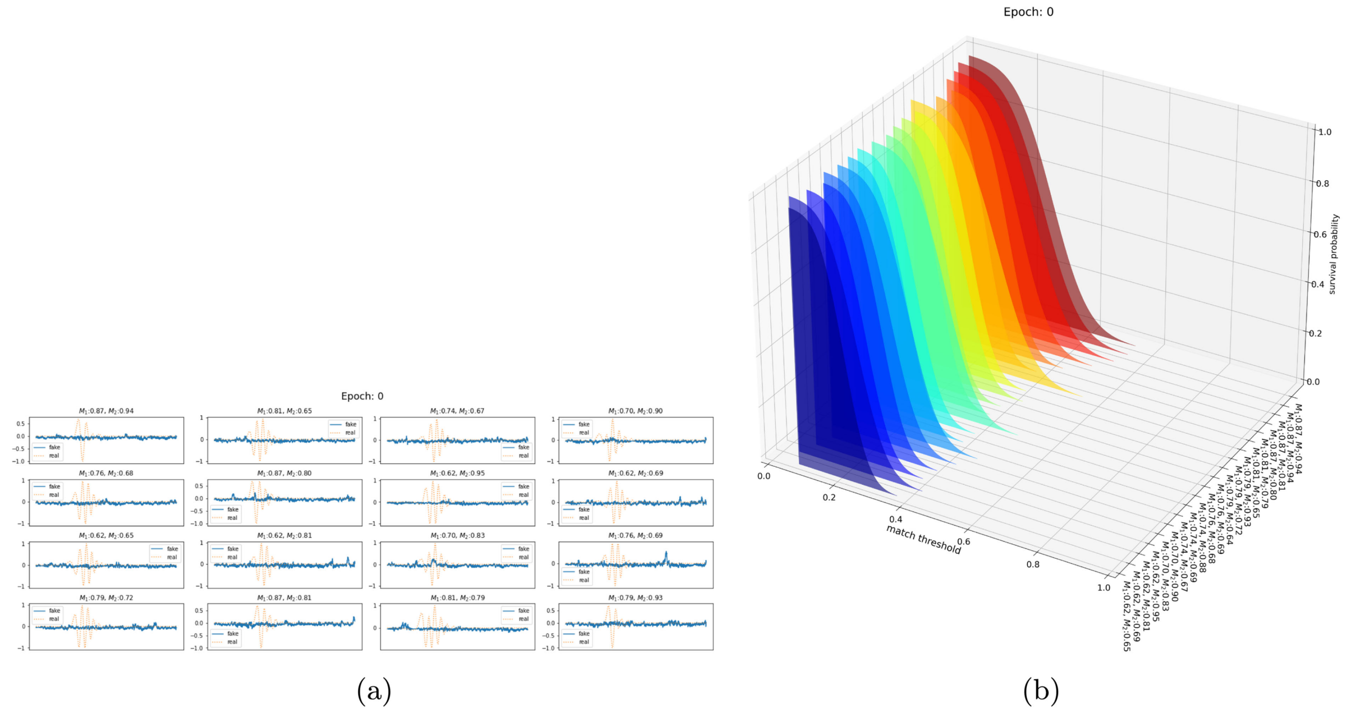
classes or generate samples to interpolate missing regions from the dataset. One way to overcome such a problem is to condition our generator and discriminator models. To promote a generator and discriminator to its conditional model forms, one must provide additional information about the training samples, which can be any kind of auxiliary information, such as class labels or, in our case, the mass value M for each sample waveform. We conditioned our WaveGAN using the values of the mass M as labels $\mathbf{y}$ and the feature scale $\max|\mathbf{x}_i|$ used to normalize the sample with the intent to restrict the model to generate samples within these parameter constraints. We can perform the conditioning by feeding $\mathbf{y}$ and $\max|\mathbf{x}_i|$ into both the discriminator and the generator. We use a similar approach as in [58]. To include the label $\mathbf{y}$ and $\max|\mathbf{x}_i|$, we scale the feature maps output from each hidden layer based on the conditioning representation; in our case we scale the feature maps by the mass and feature scale values provided by the sample labels. It is important to mention that this approach is applicable in our case due to the low variance of our labels. In order to deal with high variance labels the best approach is to either normalize them or encode it using a linear layer. Meanwhile, the scale factors $\max|\mathbf{x}_i|$ help us to constrain the amplitude scale of the synthetic samples, in a sense that when we produce the new samples their amplitude values will be within a region allowed by their physical parameters in the dataset. This is required in order to avoid producing samples which are not permitted by physics or artifacts that can be produced by such methods[8] [59].

Our WaveGAN is implemented in PyTorch [60], and we train our model for 1050 epochs using WGAN-GP [61] strategy, with Adam [62] as an optimizer, for both generator and discriminator, with learning rate of 10^{-4} for the generator and 3×10^{-4} for the discriminator. We train our networks using batches of size 32, while the validation set has batches of size 16, on a single GPU NVIDIA Tesla V100. As a first task, we set the generators and discriminators to one channel in order to generate the synthetic $r\Psi^{l=2,m=2}$ modes for the equal and different masses datasets. The results for equal and unequal mass datasets are presented in Sec. V.

IV. EVALUATION METHODOLOGY

The evaluation of generative models is an ongoing topic in the community [63]. Just as important as choosing the right strategy to train a generative model is selecting the right metric to evaluate the quality of the generated

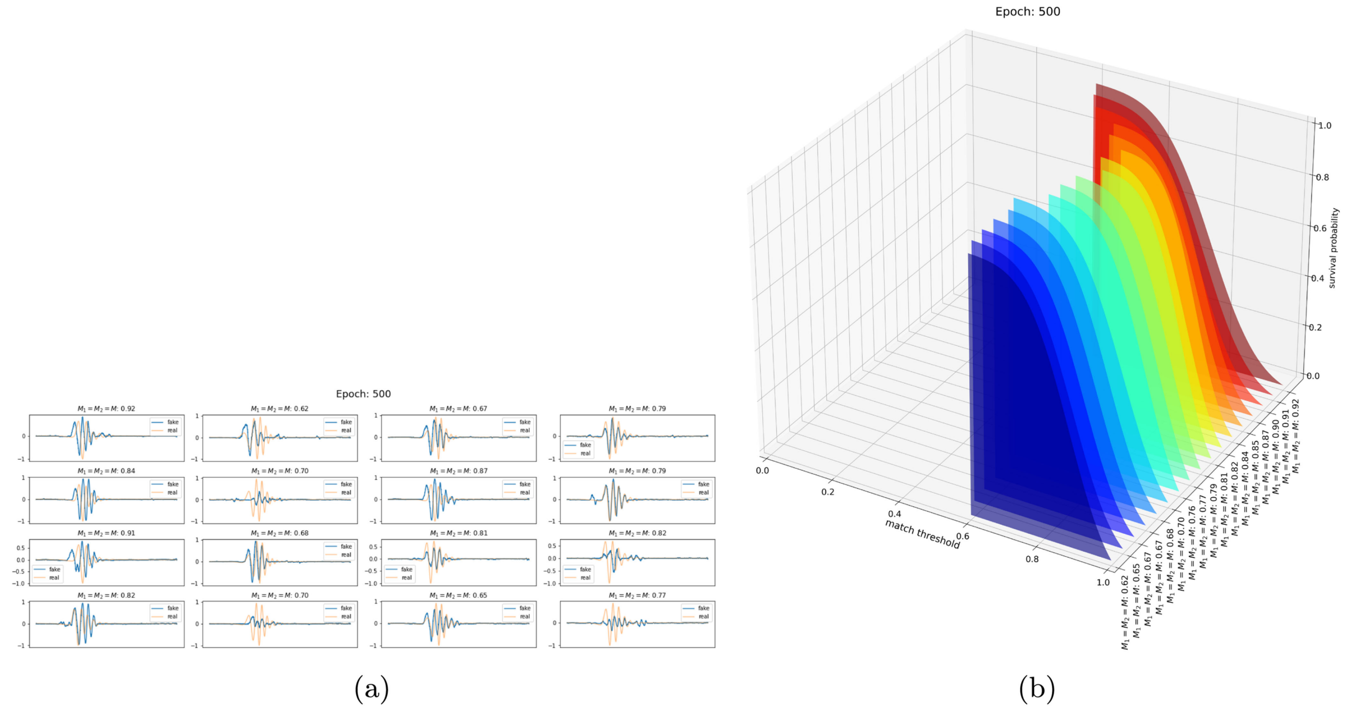
[8] We used the rescaling factor during the training to make the final neural network output give the results in the same scale as in the simulations (training data). For the new synthetic data, the model outputs a distribution around a credible region of expected value for the mass of the objects, leading to predictions for the scaling of new synthetic data. This is certainly an important point that deserves future refinement.



(a)

(b)

FIG. 5. Evaluation of the quality of the generated samples compared to the real samples of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$ and probabilities to generate good quality samples at epoch 0 with a different mass dataset. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $\Psi_4^{l=2,m=2}$. (b) Probabilities of generating samples for a given match accuracy, for different mass parameters.



(a)

(b)

FIG. 6. Evaluation of the quality of the generated samples compared to the real samples of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$ and probabilities to generate good quality samples at epoch 500. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $\Psi_4^{l=2,m=2}$. (b) Probabilities of generating samples for a given match accuracy for different mass parameters.

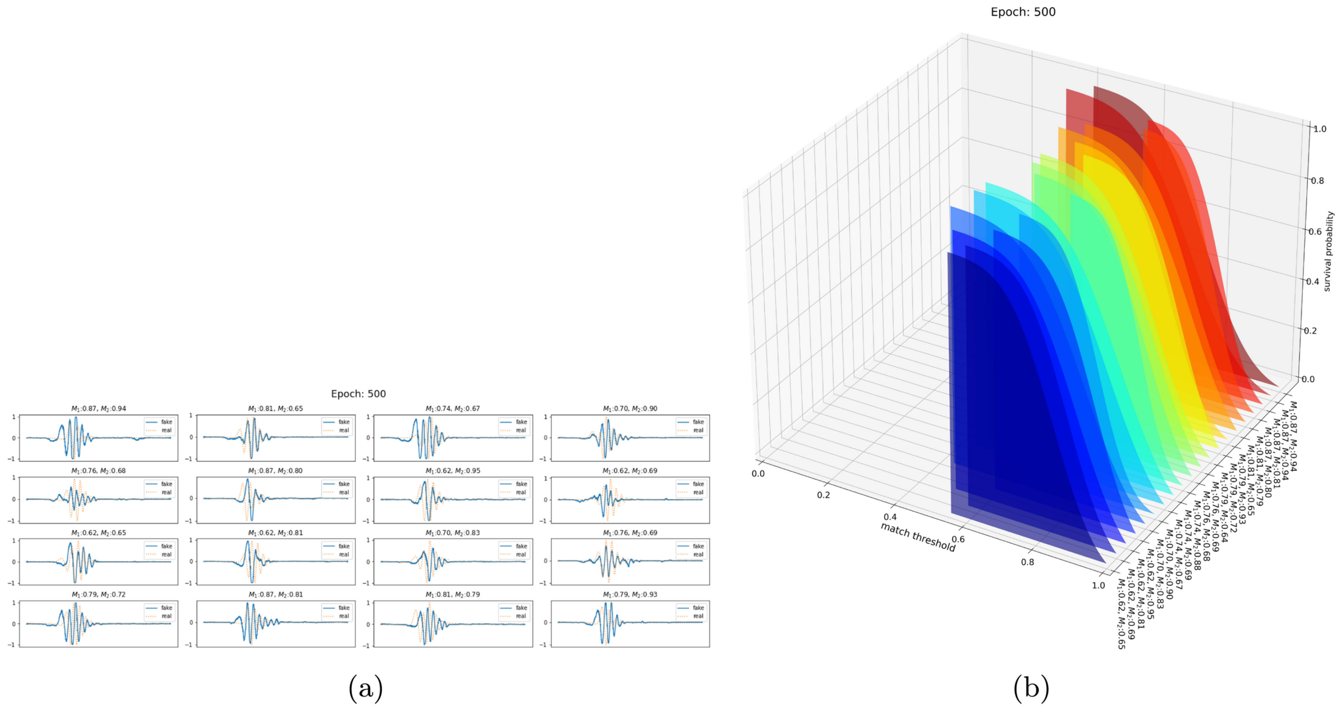


FIG. 7. Evaluation of the quality of the generated samples compared to the real samples of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$ and probabilities to generate good quality samples at epoch 500 with the different mass dataset. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $\Psi_4^{l=2,m=2}$. (b) Probabilities of generating samples for a given match accuracy for different mass parameters.

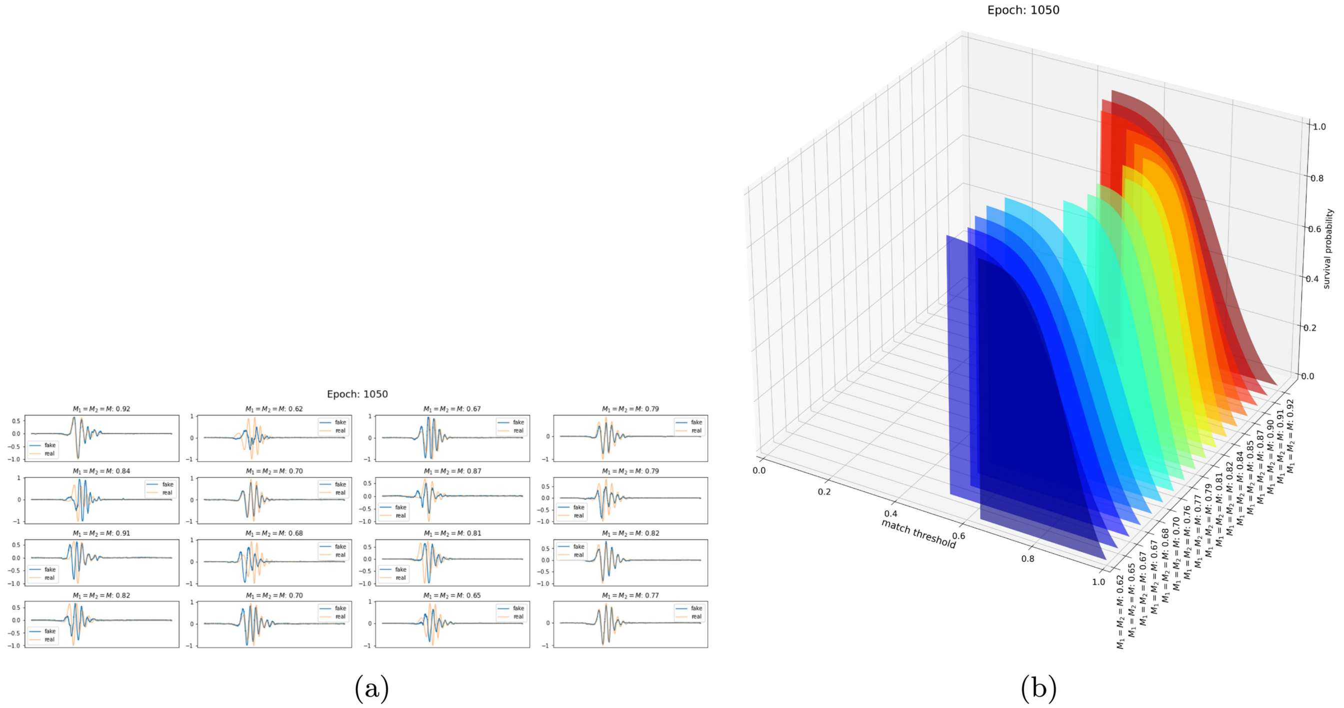
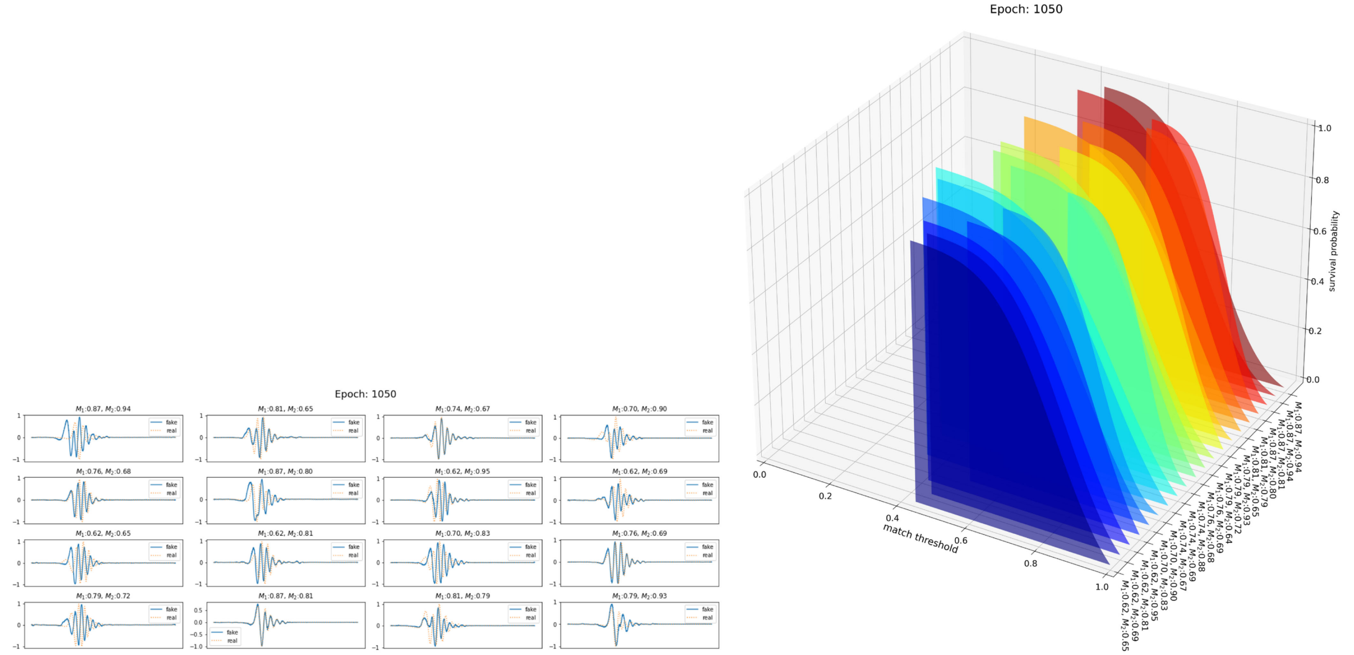


FIG. 8. Evaluation of the quality of the generated samples compared to the real samples of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$ and probabilities to generate good quality samples at epoch 1050. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $\Psi_4^{l=2,m=2}$. (b) Probabilities of generating samples for a given match accuracy for different mass parameters.



(a)

(b)

FIG. 9. Evaluation of the quality of the generated samples compared to the real samples of the Newman-Penrose scalar $\Psi_4^{l=2,m=2}$ and probabilities to generate good quality samples at epoch 1050 with the different mass dataset. (a) Real (yellow) and generated (blue) Newman-Penrose scalar $\Psi_4^{l=2,m=2}$. (b) Probabilities of generating samples for a given match accuracy for different mass parameters.

samples. A direct comparison between the synthetic samples and the real ones can be a useful diagnostic, often allowing us to build intuition of how the generative model is working, how it is failing and how it can be improved. However, qualitative as well quantitative analysis based on this approach can be misleading about the performance of the generator. In order to evaluate the quality, and therefore how trustworthy our generator is, we shall employ the following strategy. Using the PyCBC [64] matched filtering module, we estimate the overlap over time and phase between the synthetic and real samples for a given value of the parameters. We generate a set of 1000 synthetic samples for a given set of parameters, (M) for equal mass dataset or (M_1, M_2) for unequal mass dataset, and compute the overlapping match for each sample to the real equivalent samples. The overlapping match is computed with the real and synthetic normalized samples, so we can ensure that features generated for the synthetic samples are as close as possible to the expected real features. With these values, we estimate the probability of a generated sample to be above a certain threshold of match. In Figs. 4–9 we show the evolution of the match between synthetic and original samples throughout the training epochs of our NN model. To visualize the overlapping between the real and synthetic samples, we plot various samples from the equal and different mass datasets and select synthetic samples according to their overlapping matches to check against the real ones; these plots are shown in Figs. 10 and 11.

Using the overlapping match we build a separate dataset with synthetic samples and their respective match score. This dataset is further used to train another NN with the intent of predicting the match score for a given waveform sample. This new dataset consists of 85000 synthetic samples, and their matched scores are evaluated using the validation dataset for the equal and unequal mass

TABLE II. WaveGAN discriminator architecture.

Operation	Kernel size	Output shape
Input $\boldsymbol{x}$ or $G(z)$		$(n, 2048, c)$
Conv1D (stride = 4)	$(25, c, d)$	$(n, 4096, d)$
LReLU ($\alpha = 0.2$)		$(n, 4096, d)$
Phase shuffle ($n = 2$)		$(n, 4096, d)$
Conv1D (stride = 4)	$(25, d, 2d)$	$(n, 1024, 2d)$
LReLU ($\alpha = 0.2$)		$(n, 1024, 2d)$
Phase shuffle ($n = 2$)		$(n, 1024, 2d)$
Conv1D (stride = 4)	$(25, 2d, 4d)$	$(n, 256, 4d)$
LReLU ($\alpha = 0.2$)		$(n, 256, 4d)$
Phase shuffle ($n = 2$)		$(n, 256, 4d)$
Conv1D (stride = 4)	$(25, 4d, 8d)$	$(n, 64, 8d)$
LReLU ($\alpha = 0.2$)		$(n, 64, 8d)$
Phase shuffle ($n = 2$)		$(n, 64, 8d)$
Conv1D (stride = 4)	$(25, 8d, 16d)$	$(n, 16, 16d)$
LReLU ($\alpha = 0.2$)		$(n, 16, 16d)$
Reshape		$(n, 256d)$
Dense	$(256d, 1)$	$(n, 1)$

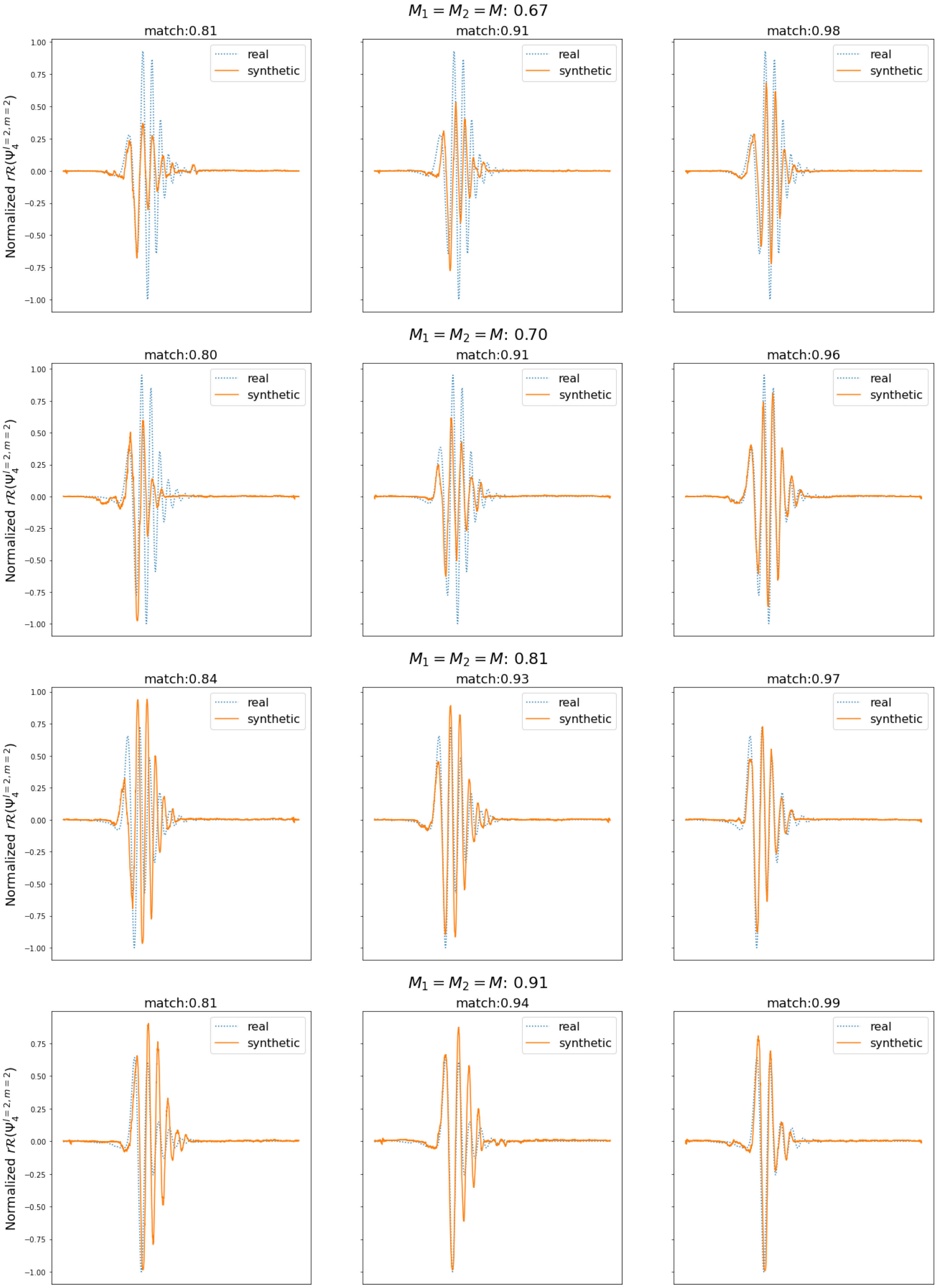


FIG. 10. Real (dotted blue) and generated (continuous yellow) $r\mathcal{R}(\Psi_4^{l=2,m=2})$ for overlapping matches in the regions $0.8 \le \text{match} \le 0.85$ (left column), $0.9 \le \text{match} \le 0.95$ (middle column) and $\text{match} > 0.95$ (right column) of the equal mass dataset.

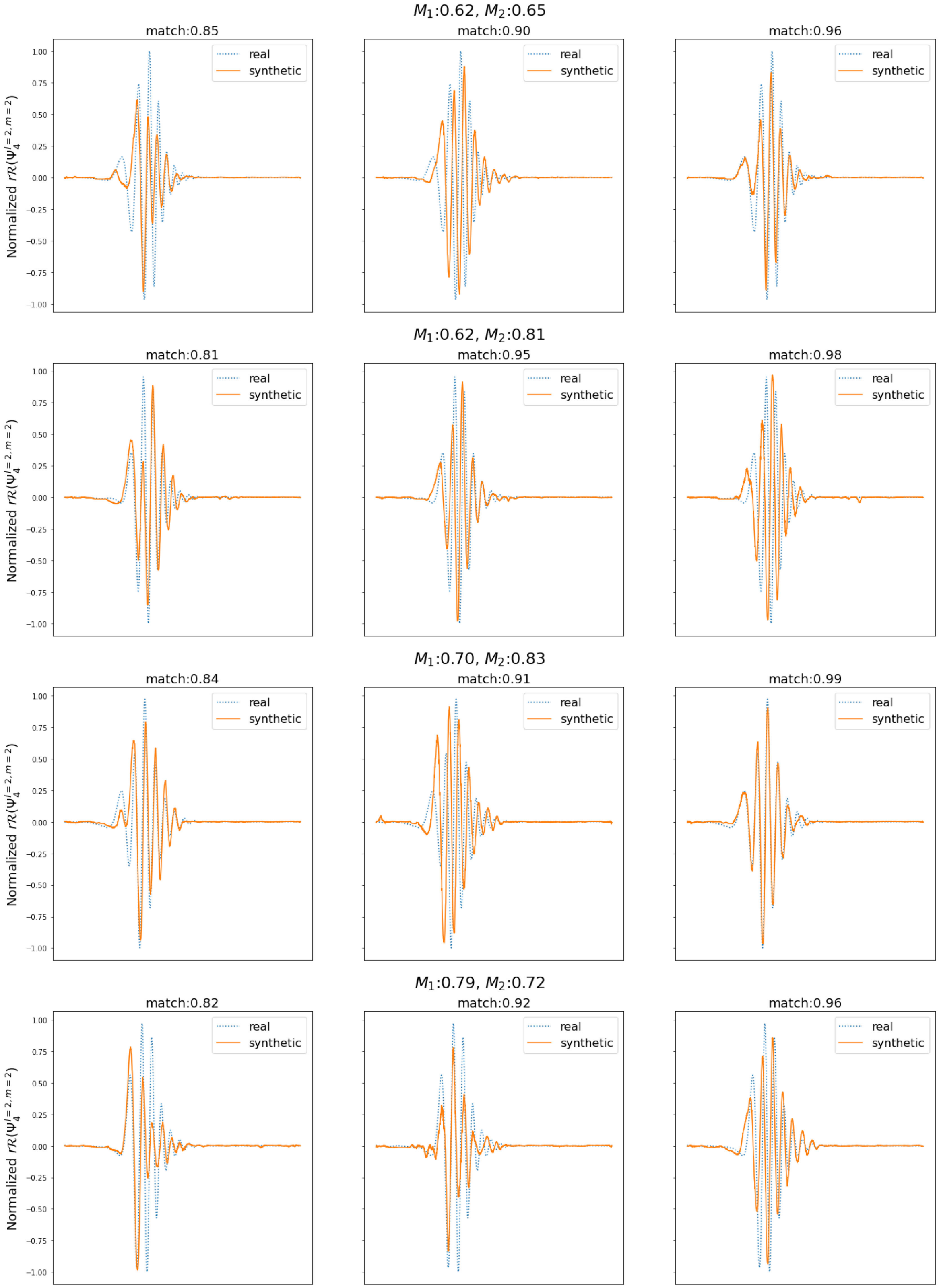


FIG. 11. Same as Fig. 11 but for the unequal mass dataset.

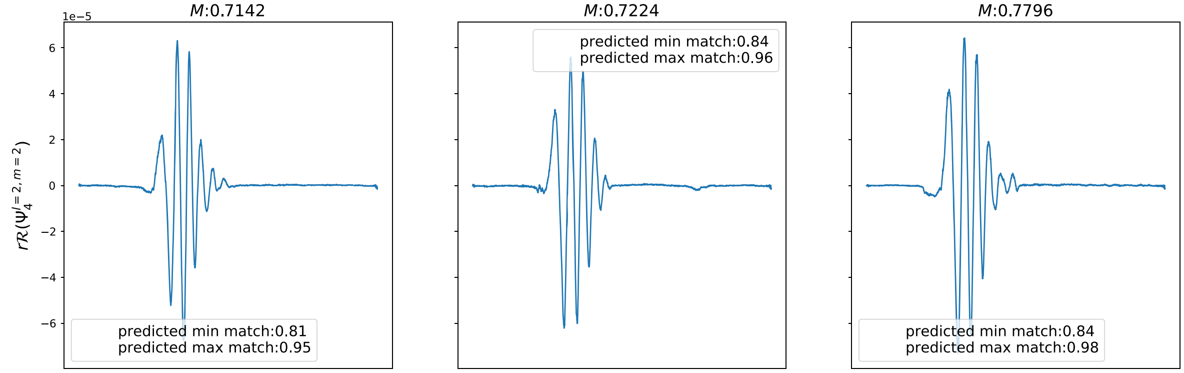


FIG. 12. Synthetic samples and their predicted matched samples for parameter values of M not present, but within the one sigma range from the mean, in the current equal mass dataset.

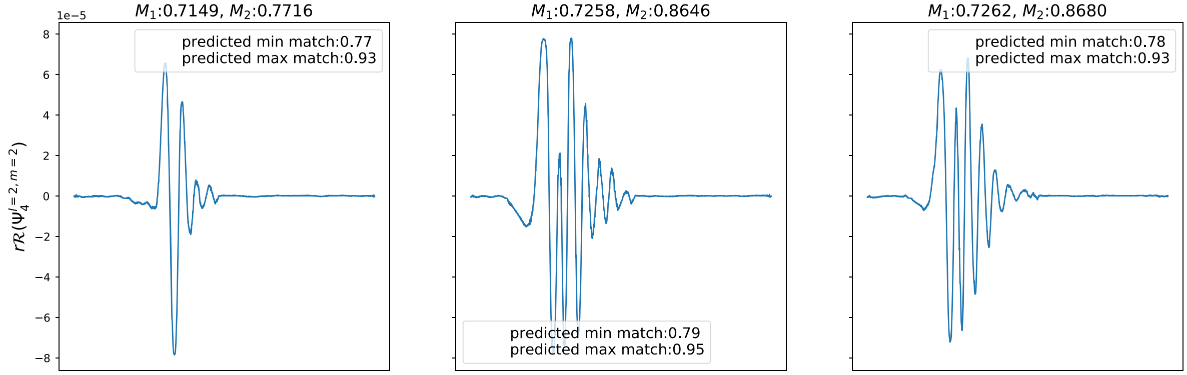


FIG. 13. Synthetic samples and their predicted matched samples for parameter values of M_1 and M_2 not present, but within the one sigma range from the mean, in the current unequal mass dataset.

dataset, each sample consisting of the normalized $r\Psi_4^{l=2,m=2}$ time series, the value of the match score for the synthetic sample when compared to the real one and the mass M parameter value. This dataset is again randomly shuffled and divided by 80% for training and 20% for validation dataset.

We use the discriminator architecture as in Table II, with the inclusion of dropout layers, with 0.2 probability of an element to be zeroed, in between each convolution layer; the dropout layers are included so we can further use the Monte Carlo dropout method [65] to estimate the predictions' uncertainties. We train this modified discriminator using the mean squared loss since the new task now—to predict the match score for a given sample—is similar to a regression problem. We also take advantage of the transfer learning method and use the weights from the trained discriminators to speed up the training of the new NN. We use the Adam optimizer and this time we train the model using the OneCycle [65] learning rate policy; the model is trained for 11 epochs with an initial learning rate $2e^{-3}$ and uses the root mean squared error as the main metric. The final model can predict the match values for a sample with an accuracy around 90%. Using the Monte Carlo dropout scheme we can further estimate the uncertainty of the model and determine the minimum and maximum predicted values of matches. Figures 12 and 13 display three synthetic samples generated from our model using M values which are not present in the original dataset, but within the one sigma range from the mean M value, and their minimum and maximum predicted match values.

V. RESULTS AND DISCUSSIONS

The results of our evaluation are shown in Tables III and IV. In Figs. 10 and 11 we sample waveforms and rank them according to their respective matches to their waveform reference, from the validation dataset. We are able to generate waveforms with a 95% to 99% match with the reference waveforms, despite the model never having had access to the samples from the validation dataset. From Tables III and IV, we see an interesting and understandable pattern: our model has a higher chance of producing samples which are closer to the expected ones whenever the sample parameters desired are close to the mean of the mass M parameters of the dataset, i.e., the region where we have more samples and, by and large, the model is

TABLE III. Probabilities for a generator to produce $r\mathcal{R}(\Psi_4^{l=2,m=2})$ with an overlapping match to the original one above 0.8 (second column), 0.9 (third column), and 0.95 (last column) for each mass ($M = M_1 = M_2$) and ω values of the validation equal mass dataset. The colors show where the values fall into the 1 and 2σ mass distribution in Fig. 3(a).

$(M_1 = M_2 = M)$	$P(0.8)$	$P(0.9)$	$P(0.95)$
0.62	0.82	0.38	0.16
0.65	0.67	0.28	0.12
0.67	0.70	0.31	0.13
0.67	0.67	0.29	0.12
0.68	0.66	0.28	0.11
0.70	0.70	0.32	0.13
0.76	0.89	0.47	0.20
0.77	0.93	0.52	0.22
0.79	0.96	0.57	0.24
0.81	0.97	0.60	0.24
0.82	0.97	0.55	0.21
0.84	0.97	0.54	0.20
0.85	0.90	0.44	0.17
0.87	0.79	0.34	0.13
0.90	0.61	0.23	0.09
0.91	0.59	0.22	0.09
0.92	0.52	0.19	0.07

TABLE IV. Same as Table III but for the unequal mass case providing each mass ($M_1 \neq M_2$) and values of the validation dataset. The colors show where the values fall into the 1 and 2σ mass distribution in Fig. 3(b).

M_1	M_2	$P(0.8)$	$P(0.9)$	$P(0.95)$
0.62	**0.65**	0.57	0.24	0.10
0.62	**0.69**	0.59	0.24	0.10
0.62	*0.81*	0.57	0.25	0.11
0.62	**0.95**	0.69	0.32	0.14
0.70	*0.83*	0.50	0.21	0.09
0.70	*0.90*	0.60	0.19	0.06
0.74	**0.67**	0.49	0.22	0.10
0.74	**0.69**	0.46	0.20	0.09
0.74	*0.88*	0.83	0.32	0.11
0.76	**0.68**	0.62	0.28	0.12
0.76	**0.69**	0.53	0.23	0.10
0.79	**0.64**	0.46	0.18	0.07
0.79	*0.72*	0.70	0.33	0.15
0.79	**0.93**	0.66	0.22	0.08
0.81	**0.65**	0.52	0.24	0.11
0.81	*0.79*	0.61	0.25	0.10
0.87	*0.80*	0.84	0.35	0.12
0.87	*0.81*	0.81	0.30	0.10
0.87	**0.94**	0.37	0.11	0.04
0.87	**0.94**	0.43	0.13	0.04

"learning" the underlying features which appear more often in the data. Nevertheless, in the parameter region away from the expected mean we still have a relatively high chance of producing samples with, at least, 90% match with the expected real samples. Such results show the potential of the method presented in this work. Moreover, one key aspect of such methodology that should be emphasized is the speed of generation of the samples. By using this technique we can generate 1000 synthetic samples in 188 ms,[9] assuming that we can have from 16% to 25% (depending on the mass parameter values) samples which are 95% similar to the expected real waveforms. In other words, we can quickly build a catalog of waveforms, to bridge the gaps within the parameter space. Such numbers show a clear advantage of the method to help the task of exploring the parameter space for the case of Proca star waveforms, or more generically, gravitational waveform generation in any model.

Although the methodology put forward here shows promising results, one should consider this proposal as a proof of concept, with necessary improvements yet to be done. The evaluation methodology using a NN can be improved by experimenting with different architectures for this task, and using Bayesian methods to estimate the error on the predicted match score. Additionally, the number of samples is a factor that heavily influences the quality of the synthetic sample; a rich dataset with not only more samples but samples with different sample rates can greatly improve the quality of the synthetic samples. New architecture models are already being tested and the Transformer [66] based architectures are showing promising results that will be explored in a future project.

VI. CONCLUSIONS

In this work, we have presented a unique application of a particular GAN architecture—WaveGan—in the context of unsupervised gravitational waveforms generation. The model presented here can generate hundreds of thousands of waveforms within very short time intervals, labeled by the physical parameters of the Proca stars that sourced them—with equal (M) or different (M_1, M_2) masses—with a high probability that from 12% to 25% of such "fake" samples are 95%, or higher, similar to the expected real ones.

We have also explored the use of the trained discriminator architectures to assist in the task of estimating the overlap score of synthetic samples, which can be used to select the synthetic waveforms which show closer features to the expected real ones. The methods presented here show that it is possible to use such techniques to accelerate the generation of the waveforms, in particular for the case of binaries of exotic compact objects.

In a future work we plan to extend and refine the method to produce samples with higher quality and automatically assign the overlapping match factor. We are also working on applying such methods in the generation of waveforms from core-collapse supernovae.

[9] The Benchmark was obtained in a Intel®Core™ i7-8750H CPU with an NVIDIA GeForce GTX 1070.

ACKNOWLEDGMENTS

This work is supported by the Center for Research and Development in Mathematics and Applications (CIDMA) through the Portuguese Foundation for Science and Technology (FCT—Fundação para a Ciência e a Tecnologia), References No. 10.54499/UIDB/04106/2020 and No. 10.54499/UIDP/04106/2020, and by national funds (OE), through FCT, I. P., in the scope of the framework contract foreseen in the numbers 4, 5, and 6 of the article 23, of the Decree-Law 57/2016, of August 29, changed by Law 57/2017, of July 19. This work is also supported by CFTC-UL through FCT, References No. 10.54499/UIDB/00618/2020 and No. 10.54499/UIDP/00618/2020. The authors gratefully acknowledge the computer resources at Artemisa, funded by the European Union ERDF and Comunitat Valenciana, as well as the technical support provided by the Instituto de Física Corpuscular, IFIC (CSIC-UV). We acknowledge support from Projects No. 10.54499/PTDC/FIS-OUT/28407/2017, No. 10.54499/PTDC/FIS-OUT/28407/2017, No. 10.54499/CERN/FIS-PAR/0021/2021, No. 10.54499/CERN/FIS-PAR/0027/2019, No. 10.54499/CERN/FIS-PAR/0002/2019, and No. 10.54499/PTDC/FIS-AST/3041/2020. This work has further been supported by the European Union's Horizon 2020 research and innovation (RISE) program H2020-MSCA-RISE-2017 Grant No. FunFiCO-777740 and the European Horizon Europe staff exchange (SE) programme HORIZON-MSCA2021-SE-01 Grant No. NewFunFiCO-101086251. N. S. G. acknowledges support from the Spanish Ministerio de Universidades, through a María Zambrano grant (ZA21-031) with reference UP2021-044, funded within the European Union-Next Generation EU, and from the Spanish Ministry of Science and Innovation via the Ramón y Cajal programme (Grant RYC2022-037424-I), funded by MCIN/AEI/10.13039/501100011033 and by "ESF Investing in your future". N. S. G. is further supported by the Spanish Agencia Estatal de Investigación (Grant PID2021-125485NB-C21) funded by MCIN/AEI/10.13039/501100011033 and ERDF A way of making Europe. This work is also supported by FCT under Contracts No. 10.54499/UIDB/00618/2020, No. 10.54499/UIDP/00618/2020, No. CERN/FISPAR/0002/2017, and No. 10.54499/CERN/FIS-PAR/0014/2019. This work has also been supported in part by the Swedish Research Council grant, Contract No. 2016-05996 and by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 668679).

[1] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), 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).

[2] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), 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).

[3] R. Abbott *et al.* (LIGO Scientific, Virgo, and KAGRA Collaborations), GWTC-3: Compact binary coalescences observed by LIGO and Virgo during the second part of the third observing run, Phys. Rev. X **13**, 041039 (2023).

[4] B. P. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), Binary black hole population properties inferred from the first and second observing runs of Advanced LIGO and Advanced Virgo, Astrophys. J. Lett. **882**, L24 (2019).

[5] P. Creminelli and F. Vernizzi, Dark energy after GW170817 and GRB170817A, Phys. Rev. Lett. **119**, 251302 (2017).

[6] T. Baker, E. Bellini, P. G. Ferreira, M. Lagos, J. Noller, and I. Sawicki, Strong constraints on cosmological gravity from GW170817 and GRB 170817A, Phys. Rev. Lett. **119**, 251301 (2017).

[7] J. M. Ezquiaga and M. Zumalacárregui, Dark energy after GW170817: Dead ends and the road ahead, Phys. Rev. Lett. **119**, 251304 (2017),

[8] J. C. Bustillo, N. Sanchis-Gual, A. Torres-Forné, J. A. Font, A. Vajpeyi, R. Smith, C. Herdeiro, E. Radu, and S. H. W. Leong, GW190521 as a merger of Proca stars: A potential new vector boson of 8.7×10^{-13} eV, Phys. Rev. Lett. **126**, 081101 (2021).

[9] B. J. Owen and B. S. Sathyaprakash, Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement, Phys. Rev. D **60**, 022002 (1999).

[10] F. Pretorius, Binary black hole coalescence, arXiv:0710.1338.

[11] P. Ajith *et al.*, A template bank for gravitational waveforms from coalescing binary black holes. I. Nonspinning binaries, Phys. Rev. D **77**, 104017 (2008),

[12] S. A. Usman *et al.*, The PyCBC search for gravitational waves from compact binary coalescence, Classical Quantum Gravity **33**, 215004 (2016).

[13] M. Boyle *et al.*, The SXS Collaboration catalog of binary black hole simulations, Classical Quantum Gravity **36**, 195006 (2019).

[14] V. Varma, S. E. Field, M. A. Scheel, J. Blackman, D. Gerosa, L. C. Stein, L. E. Kidder, and H. P. Pfeiffer, Surrogate models for precessing binary black hole simulations with unequal masses, Phys. Rev. Res. **1**, 033015 (2019).

[15] D. J. Kaup, Klein-Gordon Geon, Phys. Rev. **172**, 1331 (1968).

[16] R. Ruffini and S. Bonazzola, System of self-gravitating particles in general relativity and the concept of an equation of state, Phys. Rev. **187**, 1767 (1969).

[17] F. E. Schunck and E. W. Mielke, General relativistic boson stars, Classical Quantum Gravity **20**, R301 (2003).

[18] R. Brito, V. Cardoso, C. A. Herdeiro, and E. Radu, Proca stars: Gravitating Bose-Einstein condensates of massive spin 1 particles, Phys. Lett. B **752**, 291 (2016).

[19] C. A. R. Herdeiro, A. M. Pombo, and E. Radu, Asymptotically flat scalar, Dirac and Proca stars: Discrete vs continuous families of solutions, Phys. Lett. B **773**, 654 (2017).

[20] C. Herdeiro, I. Perapechka, E. Radu, and Y. Shnir, Asymptotically flat spinning scalar, Dirac and Proca stars, Phys. Lett. B **797**, 134845 (2019).

[21] S. L. Liebling and C. Palenzuela, Dynamical boson stars, Living Rev. Relativity **15**, 6 (2012).

[22] N. Sanchis-Gual, F. Di Giovanni, M. Zilhão, C. Herdeiro, P. Cerdá-Durán, J. Font, and E. Radu, Nonlinear dynamics of spinning bosonic stars: Formation and stability, Phys. Rev. Lett. **123**, 221101 (2019).

[23] L. Hui, J. P. Ostriker, S. Tremaine, and E. Witten, Ultralight scalars as cosmological dark matter, Phys. Rev. D **95**, 043541 (2017).

[24] F. H. Vincent, Z. Meliani, P. Grandclement, E. Gourgoulhon, and O. Straub, Imaging a boson star at the Galactic center, Classical Quantum Gravity **33**, 105015 (2016).

[25] H. Olivares, Z. Younsi, C. M. Fromm, M. De Laurentis, O. Porth, Y. Mizuno, H. Falcke, M. Kramer, and L. Rezzolla, How to tell an accreting boson star from a black hole, Mon. Not. R. Astron. Soc. **497**, 521 (2020).

[26] C. A. R. Herdeiro, A. M. Pombo, E. Radu, P. V. P. Cunha, and N. Sanchis-Gual, The imitation game: Proca stars that can mimic the Schwarzschild shadow, J. Cosmol. Astropart. Phys. 04 (2021) 051.

[27] M. Bezares, C. Palenzuela, and C. Bona, Final fate of compact boson star mergers, Phys. Rev. D **95**, 124005 (2017).

[28] M. Bezares and C. Palenzuela, Gravitational waves from dark boson star binary mergers, Classical Quantum Gravity **35**, 234002 (2018).

[29] M. Bezares, M. Bošković, S. Liebling, C. Palenzuela, P. Pani, and E. Barausse, Gravitational waves and kicks from the merger of unequal mass, highly compact boson stars, Phys. Rev. D **105**, 064067 (2022).

[30] N. Sanchis-Gual, C. Herdeiro, J. A. Font, E. Radu, and F. Di Giovanni, Head-on collisions and orbital mergers of Proca stars, Phys. Rev. D **99**, 024017 (2019).

[31] R. Abbott *et al.* (LIGO Scientific and Virgo Collaborations), GW190521: A binary black hole merger with a total mass of $150M_{\odot}$, Phys. Rev. Lett. **125**, 101102 (2020).

[32] N. Sanchis-Gual, J. Calderón Bustillo, C. Herdeiro, E. Radu, J. A. Font, S. H. W. Leong, and A. Torres-Forné, Impact of the wavelike nature of Proca stars on their gravitational-wave emission, Phys. Rev. D, **106**, 124011 (2022).

[33] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial networks, arXiv:1406.2661.

[34] A. Hu, R. Xie, Z. Lu, A. Hu, and M. Xue, TableGAN-MCA: Evaluating membership collisions of GAN-synthesized tabular data releasing, arXiv:2107.13190.

[35] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, Photo-realistic single image super-resolution using a generative adversarial network, arXiv:1609.04802.

[36] S. E. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, Generative adversarial text to image synthesis, arXiv:1605.05396.

[37] M. Paganini, L. de Oliveira, and B. Nachman, CaloGAN: Simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks, Phys. Rev. D **97**, 014021 (2018).

[38] L. de Oliveira, M. Paganini, and B. Nachman, Learning particle physics by example: Location-aware generative adversarial networks for physics synthesis, Comput. Software Big Sci. **1**, 4 (2017).

[39] L. de Oliveira, M. Paganini, and B. Nachman, Controlling physical attributes in GAN-accelerated simulation of electromagnetic calorimeters, J. Phys. Conf. Ser. **1085**, 042017 (2018).

[40] M. Paganini, L. de Oliveira, and B. Nachman, Accelerating science with generative adversarial networks: An application to 3D particle showers in multilayer calorimeters, Phys. Rev. Lett. **120**, 042003 (2018).

[41] M. Wiese, R. Knobloch, R. Korn, and P. Kretschmer, Quant GANs: Deep generation of financial time series, Quant. Finance **20**, 1419 (2020).

[42] T. Lau, S. Xu, and X. Wang, Video content swapping using GAN, arXiv:2111.10916.

[43] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, WaveNet: A generative model for raw audio, arXiv:1609.03499.

[44] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. C. Courville, and Y. Bengio, SampleRNN: An unconditional end-to-end neural audio generation model, arXiv:1612.07837.

[45] C. Donahue, J. J. McAuley, and M. S. Puckette, Synthesizing audio with generative adversarial networks, arXiv:1802.04208.

[46] J. Blackman, S. E. Field, C. R. Galley, B. Szilágyi, M. A. Scheel, M. Tiglio, and D. A. Hemberger, Fast and accurate prediction of numerical relativity waveforms from binary black hole coalescences using surrogate models, Phys. Rev. Lett. **115**, 121102 (2015).

[47] N. Sanchis-Gual, M. Zilhão, C. Herdeiro, F. Di Giovanni, J. A. Font, and E. Radu, Synchronized gravitational atoms from mergers of bosonic stars, Phys. Rev. D **102**, 101504 (2020).

[48] Einstein Toolkit: http://www.einsteintoolkit.org.

[49] F. Loffler, J. Faber, E. Bentivegna, T. Bode, P. Diener *et al.*, The Einstein Toolkit: A community computational infrastructure for relativistic astrophysics, Classical Quantum Gravity **29**, 115001 (2012).

[50] M. Zilhão and F. Löffler, An introduction to the Einstein Toolkit, Int. J. Mod. Phys. A **28**, 1340014 (2013).

[51] E. Schnetter, S. H. Hawley, and I. Hawke, Evolutions in 3-D numerical relativity using fixed mesh refinement, Classical Quantum Gravity **21**, 1465 (2004).

[52] Cactus: http://www.cactuscode.org.

[53] H. Witek, M. Zilhao, G. Ficarra, and M. Elley *et al.*, Canuda: A public numerical relativity library to probe fundamental physics (Zenodo, 2020), 10.5281/zenodo.3565475.

[54] M. Zilhao, H. Witek, and V. Cardoso, Nonlinear interactions between black holes and Proca fields, Classical Quantum Gravity **32**, 234003 (2015).

[55] N. Sanchis-Gual, J. Calderón Bustillo, C. Herdeiro, E. Radu, J. A. Font, S. H. W. Leong, and A. Torres-Forné, Impact of the wavelike nature of Proca stars on their gravitational-wave emission, Phys. Rev. D **106**, 124011 (2022).

[56] Y. Bengio, P. Simard, and P. Frasconi, Learning long-term dependencies with gradient descent is difficult, IEEE Trans. Neural Networks **5**, 157 (1994).

[57] A. Radford, L. Metz, and S. Chintala, Unsupervised representation learning with deep convolutional generative adversarial networks, arXiv:1511.06434.

[58] C. Y. Lee, A. Toffy, G. J. Jung, and W. Han, Conditional WaveGAN, arXiv:1809.10636.

[59] A. Odena, V. Dumoulin, and C. Olah, Deconvolution and checkerboard artifacts, Distill (2016).

[60] A. Paszke *et al.*, PyTorch: An imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., Vancouver, Canada, 2019), pp. 8024–8035.

[61] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, Improved training of Wasserstein GANs, arXiv:1704.00028.

[62] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.

[63] A. Borji, Pros and cons of GAN evaluation measures: New developments, arXiv:2103.09396.

[64] A. Nitz *et al.*, GWASTRO/PyCBC: PyCBC release v1.16.11, 2020.

[65] L. N. Smith and N. Topin, Super-convergence: Very fast training of neural networks using large learning rates, arXiv:1708.07120.

[66] 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.