

PINNGraPE: Physics Informed Neural Network for Gravitational wave Parameter Estimation

Leigh Smith^{1,2}, Matteo Scialpi^{3,4}, Francesco Di Clemente⁵ and Michał Bejger^{4,6}

¹Dipartimento di Fisica, Università di Trieste, Via Valerio 2, 34127 Trieste, Italy

²INFN, Sezione di Trieste, I-34127 Trieste, Italy

³Dipartimento di Fisica e Scienze della Terra, Università di Ferrara, Via Saragat 1, 44122 Ferrara, Italy

⁴INFN Sezione di Ferrara, Via Saragat 1, 44122 Ferrara, Italy

⁵Department of Physics, University of Houston, Houston, TX 77204, USA

⁶Nicolaus Copernicus Astronomical Center, Polish Academy of Sciences, Bartycka 18, 00-716, Warsaw, Poland

E-mail: leighann.smith@units.it

Abstract.

Weakly-modelled searches for gravitational waves are essential for ensuring that all potential sources are accounted for in detection efforts, as they make minimal assumptions regarding source morphology. While these searches primarily target generic transient sources, they are also highly effective at identifying a broad range of compact binary coalescences, demonstrated by the weakly-modelled search algorithm coherent WaveBurst being the first to detect GW150914. Despite their ability to detect compact binaries with diverse properties, the accurate estimation of source parameters from their output remains to be a challenging task. To overcome this, we leverage physics-informed neural networks, which serve as a powerful tool for parameter estimation by applying physical constraints through the universal differential equation governing a compact binary system. With this approach, we rapidly infer the mass parameters of binary black hole merger systems to within 7% from only the time-frequency representation of the gravitational wave signal.

1 Introduction

coherent WaveBurst (cWB) [1, 2, 3] is a weakly-modelled gravitational wave (GW) detection algorithm which relies on excess coherent energy across a network of detectors. While it is used extensively in searches for generic GW transients which cannot rely on well-modelled waveforms, it is also highly efficient in detecting binary black hole (BBH) events without assuming a specific source waveform [4, 5, 6]. Despite this, it lacks the ability to accurately infer source parameters from the reconstructed waveform, aside from sky location and an estimate of chirp mass from the likelihood. To overcome this problem we utilise physics-informed neural networks (PINNs), which aim to learn the behaviour of physical systems by embedding governing differential equations and physical relations in the loss function of a neural network [7]. In this work, we train a PINN to learn the governing equations of a coalescing binary, with the goal of inferring mass properties of a BBH from only the GW signal. We consider the 1.5 post-Newtonian (PN) frequency evolution of the merging binary [8]:

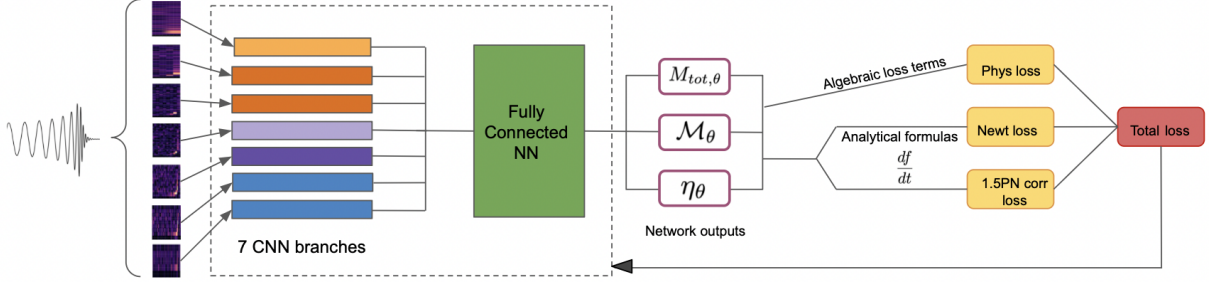


Figure 1: Diagram of the multi-branch CNN architecture. 7 TF representations are input into the network, each processed by a different CNN branch. CNN outputs are flattened and inserted into a fully-connected neural network, which directly outputs 3 mass parameters. Physical constraints are considered in the loss terms, used to update the hyper-parameters of the network.

$$\frac{df}{dt} = \frac{96}{5} \pi^{8/3} \left(\frac{GM_{\odot}}{c^3} \frac{\mathcal{M}}{M_{\odot}} \right)^{5/3} f^{11/3} \left[1 - \left(\frac{743}{336} + \frac{11}{4} \eta \right) \varepsilon^{2/3} + 4\pi\varepsilon \right], \quad \varepsilon = \frac{GM_{\odot}}{c^3} \left(\frac{M_{\text{tot}}}{M_{\odot}} \right) \pi f \quad (1)$$

for chirp mass $\mathcal{M}$, total mass M_{tot} and symmetric mass ratio η . Expanding to further PN orders increases the parameters in this relation, however for this study we consider only mass properties. These mass parameters are related to the component masses of the system through the following 3 equations:

$$\mathcal{M} = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}}, \quad M_{\text{tot}} = m_1 + m_2, \quad \eta = \frac{m_1 m_2}{(m_1 + m_2)^2}, \quad (2)$$

which can be re-arranged to form a direct relation between $\mathcal{M}$, M_{tot} and η :

$$\mathcal{M} = M_{\text{tot}} \eta^{3/5} \quad (3)$$

From the above equations we are able to construct a neural network that will learn the frequency evolution of the GW signal through physics-informed losses, thus outputting the related mass properties.

2 Data and model architecture

For a detected a signal, cWB constructs likelihood maps in the time-frequency (TF) domain, over 7 different TF resolutions. The aim is to eventually use the TF outputs directly from cWB, however for this preliminary study we construct TF data to mimic the structure of cWB outputs using the PyCBC package [9]. This data consists of BBH simulations injected into Gaussian noise coloured by the design sensitivity power spectral density of the Advanced LIGO (aLIGO) detector [10]. Each injection is transformed into the TF domain over 7 resolutions. In total we create 10,000 signals, varied over distance [500,1000] Mpc and sampled uniformly from total mass [30,100] $M_{\odot}$ and symmetric mass ratio [0.1,0.25]. The remaining mass properties are calculated via Eqs. 2 and 3. The data is divided into 70% training, 20% validation and 10% testing data.

We employ a multi-branch convolutional neural network (CNN) as the base of our model, an overview of which is seen in Fig. 1. Each of the 7 TF resolutions is input into a separate CNN branch, designed to handle the varied dimensionality of the inputs with different convolutional and maxpooling layer definitions. The outputs of the CNN branches are flattened into a fully-connected neural network, which has 3 output neurons corresponding to $\{M_{\text{tot}}, \mathcal{M}, \eta\}_{\theta}$. While only 2 output mass parameters are needed to infer all mass properties theoretically, the inclusion of all 3 parameters in Eq. 3 allows the network to learn the physical redundancies within this relation. This is enforced through the loss equation and ultimately improves the accuracy of the network. The physics-informed loss is constructed from 2 main components. The first component is a mean squared loss between output and target terms of the analytical df/dt formula, where output terms are calculated via $\{M_{\text{tot}}, \mathcal{M}, \eta\}_{\theta}$ and Eq. 1. The df/dt loss component is split by Newtonian and 1.5 PN correction terms due to the extreme magnitude difference in their values. The second component consists of auxiliary algebraic loss terms between the 3 outputs of the network.

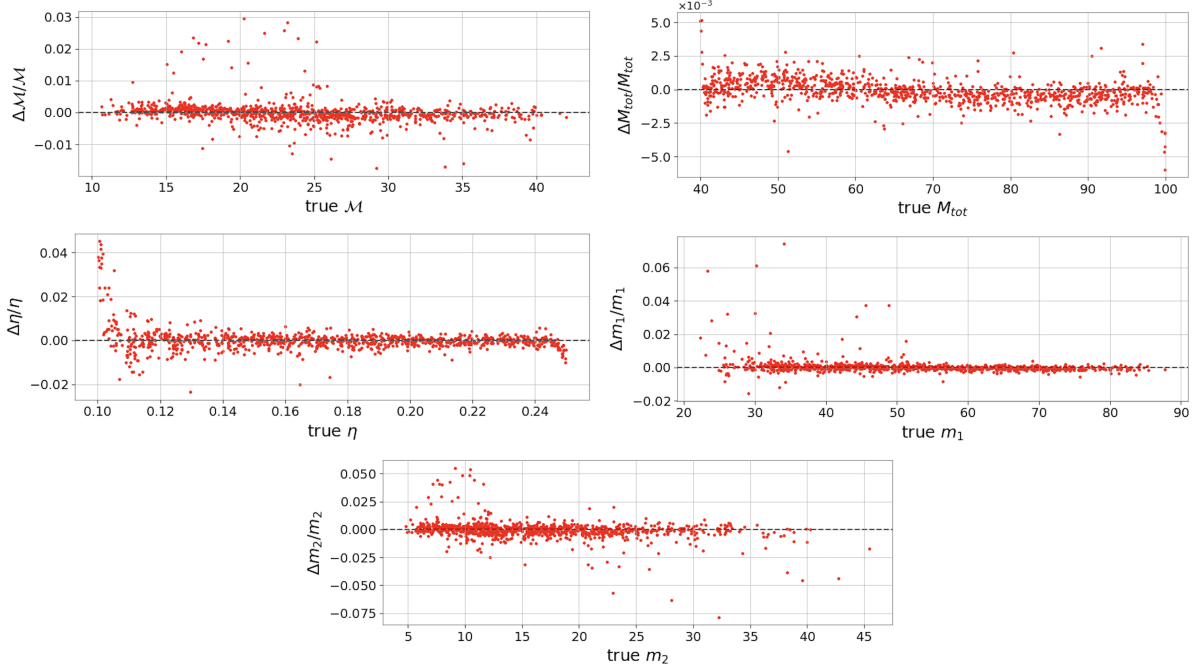


Figure 2: Relative errors of the inferred mass properties vs. true parameter. The black dashed lines represent zero relative error, while each red marker represents a signal from the test data.

These are the terms that exploit the physical redundancy within Eq. 3, which helps with the convergence and accuracy of the network. With the above considerations, the total loss is given by:

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{\text{Newt}} + \mathcal{L}_{1.5\text{PN}} + \mathcal{L}_{\text{Phys}} = & \frac{\beta_1}{B} \sum_{i=1}^B \left(\frac{df_\theta}{dt} - \frac{df_T}{dt} \right)_{\text{Newt}}^2 + \frac{\beta_2}{B} \sum_{i=1}^B \left(\frac{df_\theta}{dt} - \frac{df_T}{dt} \right)_{1.5\text{PN}}^2 \\ & + \frac{\beta_3}{B} \sum_{i=1}^B \left(M_{\text{tot},\theta} \eta_\theta^{3/5} - \mathcal{M}_\theta \right)^2 + \frac{\beta_4}{B} \sum_{i=1}^B \left(\mathcal{M}_\theta \eta_\theta^{-3/5} - M_{\text{tot},\theta} \right)^2 + \frac{\beta_5}{B} \sum_{i=1}^B \left(\left(\frac{\mathcal{M}_\theta}{M_{\text{tot},\theta}} \right)^{3/5} - \eta_\theta \right)^2 \quad (4) \end{aligned}$$

where B is the batch size and $\beta_1 = 10^{-6}, \beta_2 = 10^4, \beta_3 = \beta_4 = \beta_5 = 1$ are weighting factors. Values related to the network outputs are notated by θ , while target values are notated by T . The subscript i has been dropped from all values for simplicity. In total the network takes ~ 80 minutes to train on a GPU, with a REDUCELRONPLATEAU scheduler decreasing the learning rate to $< 10^{-6}$.

3 Discussion and future outlook

The performance of the network is tested by inputting the TF representations of signals from the test data set into the model, which infers a signal's mass properties within 2 ms. Fig. 2 presents the accuracy of the network via the relative errors for each of the inferred parameters. Looking first at the parameters directly output from the network, $\mathcal{M}$, M_{tot} , and η are inferred to within 3%, 0.5% and 5% respectively. From these parameters, the component masses m_1 and m_2 are inferred (via Eqs. 2) to within 7%. Looking more closely at the errors on the component masses, it can be seen that the majority of points lie within the 2% error region, and that only a few outliers have errors up to 7%. This is especially true for low mass values, the cause of which will be investigated in future work. In addition to this, the errors on M_{tot} and η appear to follow a sigmoid trend, with increased errors towards the boundaries of the parameter ranges. We suspect that such behaviour arises from the sigmoid final layer in the network, and alternative functions will be investigated to overcome this problem.

There are many avenues to enhance this work, with the first steps to improve the accuracy and robustness of the network. This will be accomplished through the consideration of TF signals directly from cWB and through the exploration of a more detailed data set with increased parameter ranges. In

addition to this, we plan to investigate the structure of our model further with the aim to decrease errors and biases across the parameter range. Furthermore, the natural progression of this project will be to consider further PN expansion terms in our governing equations, such as spin-orbit and spin-spin terms, which is required to generalise to spinning BBHs.

Acknowledgments

This work acknowledges support from the European Union - Next Generation EU Mission 4 Component 1 CUP J53D23001550006 with the PRIN Project No. 202275HT58, and by ICSC – Centro Nazionale di Ricerca in High Performance Computing, Big Data and Quantum Computing, funded by European Union – NextGenerationEU. This research was partially supported by the Polish National Science Center OPUS grant no. 2021/43/B/ST9/01714, and by the National Aeronautics and Space Administration (NASA) under Award 80NSSC24K0767.

References

- [1] S. Klimentenko, I. Yakushin, A. Mercer, and G. Mitselmakher, “A coherent method for detection of gravitational wave bursts,” *Classical and Quantum Gravity*, vol. 25, no. 11, p. 114029, 2008.
- [2] S. Klimentenko *et al.*, “Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors,” *Phys. Rev. D*, vol. 93, no. 4, p. 042004, 2016.
- [3] M. Drago *et al.*, “Coherent WaveBurst, a pipeline for unmodeled gravitational-wave data analysis,” *SoftwareX*, vol. 14, no. 100678, pp. 2352–7110, 2021.
- [4] B. P. Abbott *et al.*, “Observing gravitational-wave transient GW150914 with minimal assumptions,” *Phys. Rev. D*, vol. 93, no. 12, p. 122004, 2016. [Addendum: Phys.Rev.D 94, 069903 (2016)].
- [5] M. Szczepańczyk *et al.*, “Observing an intermediate-mass black hole GW190521 with minimal assumptions,” *Phys. Rev. D*, vol. 103, no. 8, p. 082002, 2021.
- [6] T. Mishra *et al.*, “Search for binary black hole mergers in the third observing run of Advanced LIGO-Virgo using coherent WaveBurst enhanced with machine learning,” *Phys. Rev. D*, vol. 105, no. 8, p. 083018, 2022.
- [7] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *J. Comput. Phys.*, vol. 378, pp. 686–707, 2019.
- [8] C. Cutler and E. E. Flanagan, “Gravitational waves from merging compact binaries: How accurately can one extract the binary’s parameters from the inspiral wave form?,” *Phys. Rev. D*, vol. 49, pp. 2658–2697, 1994.
- [9] A. Nitz *et al.*, “gwastro/pycbc: v2.3.3 release of pycbc,” January 2024.
- [10] J. Aasi *et al.*, “Advanced LIGO,” *Class. Quant. Grav.*, vol. 32, p. 074001, 2015.