Probabilistic Machine Learning for Real-Time Gravitational-Wave Inference

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Maximilian Dax

aus Bonn

Tübingen

2024

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	17.07.2025
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter/-in:	Prof. Dr. Bernhard Schölkopf
2. Berichterstatter/-in:	Prof. Dr. Philipp Hennig
3. Berichterstatter/-in:	Prof. Dr. Tilman Plehn

To You.

Abstract

Gravitational-wave (GW) astronomy has led to groundbreaking discoveries in the past decade, and with the development of next-generation detectors, its potential for future breakthroughs continues to grow. This field hinges on the ability to accurately characterize GW sources based on measured data. However, computational demands of existing inference methods impede their application to large-scale or real-time data analysis. We here present DINGO, a probabilistic machine learning framework for Bayesian GW inference that addresses these limitations with an unprecedented combination of speed and accuracy. Building on neural posterior estimation (NPE), DINGO trains deep neural networks on GW simulations to learn the mapping between measured data and GW source parameters.

We first introduce DINGO for binary black hole mergers, the most common GW source. We develop techniques to integrate symmetries (called GNPE) and to rapidly adapt to varying detector noise properties. We then augment NPE with importance sampling (NPE-IS) to correct for potential network inaccuracies. This enables asymptotically exact inference, independent verification and unbiased estimates of the Bayesian evidence, addressing important limitations of deep learning-based inference. Finally, we extend DINGO to binary neutron star mergers. We develop techniques to effectively compress long signals based on event-adaptive priors (prior conditioning) and to enable inference even before the merger. With inference times of less than a second, this provides crucial real-time information for directing searches for electromagnetic counterparts.

Our experimental evaluations encompass more than 50 real events and thousands of simulations, three different waveform models, two types of sources and two experimental setups (LIGO-Virgo-KAGRA and next-generation detectors). DINGO consistently achieves comparable accuracy to established inference methods while being orders of magnitude faster. This prepares GW data analysis for increasing detection rates, facilitates large-scale studies and can improve searches for electromagnetic counterparts. Beyond GW astronomy, DINGO contributes several broadly applicable techniques to the field of simulation-based inference, including GNPE, NPE-IS and prior-conditioning.

Zusammenfassung

Die Gravitationswellen-Astronomie hat im letzten Jahrzehnt bahnbrechende Entdeckungen ermöglicht. Mit der Entwicklung der nächsten Generation von Detektoren wächst ihr Potenzial für zukünftige Durchbrüche weiter. Ein zentraler Bestandteil dieses Forschungsfeldes ist die Charakterisierung von astrophysikalischen Gravitationswellenquellen anhand gemessener Daten. Existierende Inferenzmethoden sind allerdings so rechenintensiv, dass groß angelegte oder Echtzeitanalysen damit nur bedingt durchführbar sind. In dieser Arbeit präsentieren wir DINGO, ein probabilistisches System des maschinellen Lernens für Bayessche Inferenz von Gravitationswellen, welches diese Einschränkungen überwindet. Aufbauend auf der Methode der neuronalen Posteriorschätzung (engl. neural posterior estimation, NPE) trainiert DINGO tiefe neuronale Netzwerke mit Simulationen von Gravitationswellen, und lernt so die Zusammenhänge zwischen gemessenen Daten und Parametern, welche die Gravitationswellenquellen beschreiben.

Zunächst führen wir DINGO für Verschmelzungen von schwarzen Löchern ein, die häufigste Quellen von Gravitationswellen. Wir entwickeln Techniken, um Symmetrien zu integrieren (GNPE) und um Schwankungen im Detekorrauschen zu berücksichtigen. Anschließend kombinieren wir NPE mit Importance Sampling (NPE-IS), um potenzielle Fehler der neuronalen Netzwerke zu korrigieren. Dies ermöglicht asymptotisch exakte Inferenz und unabhängige Validierung der Ergebnisse. NPE-IS adressiert damit wichtige Einschränkungen von auf Deep Learning basierenden Inferenzmethoden. Zuletzt erweitern wir DINGO auf Verschmelzungen von Neutronensternen. Wir entwickeln Techniken, um lange Signale durch Nutzung adaptiver Prior zu komprimieren (prior conditioning) und um Inferenz bereits vor der Verschmelzung zu ermöglichen. Mit Inferenzzeiten von unter einer Sekunde liefert dies entscheidende Echtzeitinformationen für die Suche nach elektromagnetischen Signalen.

Unsere Experimente umfassen Auswertungen auf über 50 echten Gravitationswellenmessungen und tausenden Simulationen, drei verschiedene Wellenformmodelle, zwei Arten von Gravitationswellenquellen und zwei experimentelle Setups (LIGO-Virgo-KAGRA und Detektoren der nächsten Generation). DINGO erreicht eine mit etablierten Inferenzmethoden vergleichbare Genauigkeit, ist dabei jedoch um Größenordnungen schneller. Dies bereitet die Datenanalyse in der Gravitationswellen-Astronomie auf steigende Detektionsraten vor, ermöglicht groß angelegte Studien und kann die Suche nach elektromagnetischen Signalen verbessern. Mehrere für DINGO entwickelte Techniken sind allgemein anwendbar im Bereich der simulationsbasierten Inferenz, darunter GNPE, NPE-IS und prior conditioning.

Acknowledgements

I thank my advisors Bernhard, Jakob and Stephen for their supervision, support and for their crucial contributions to my research. I thank Bernhard for the opportunity to join the EI department. Stephen is also my closest collaborator, and I am grateful for everything I have learned from and with him.

I thank the rest of the PhD committee, Philipp and Antonio, for their service and interest in my research. I thank Timmy, Stephen, Simon, Jonas and Ingrid for comments on this thesis. I thank the Hector Fellow Academy for supporting me during my PhD studies, and for providing numerous networking and training opportunities. I thank all members of the HFA office for making this possible.

I thank Alessandra and Jonathan, who have been involved in my PhD projects since the start. Jonathan also came up with the name DINGO. It was great working with Jonas and Annalena at the MPI, and collaborating remotely with Michael and Nihar. I also thank the rest of the DINGO pack, including Alex, Sam, Ashwin and Vincent, as well as my other collaborators Michael D., Simon and Timmy.

I thank my colleagues and friends at the EI department. Here is an incomplete list of people that have been around over the years who made EI a great place (apologies for likely missing someone): Alex I., Alex N., Annalena, Annika, Armin, Diego, Felix, Frederik, Frederike, Gb, Gionvanni, Hamza, Heiner, Jonas K., Jonas W., Julius, Jun, Lars, Lennart, Luigi, Max, Nasim, Simon, Timmy, Vincent, Waleed, Weiyang, Wendong, Yassine, Yucen, Zeju, Zhijing, and Ann-Sohpie, Haesook, Lidia, Sabrina.

I thank all members of Jakob's lab for making me feel welcome whenever I joined their group activities (which I wish I could have done more often). I thank my colleagues Urs, Jordi, Jan and Leo for a great summer at Google in Berlin. I thank Bastian, who helped shape my approach to science during my Bachelor and Master theses, and his group at that time, in particular Stefan, Tobi and Dominik.

I am grateful for my friends, who have supported me in completing my PhD. In addition to many of the people listed above, this includes Anne, Maya, Lino, Patrick, Paul, the #YOSO crew, and my football team TV Derendingen Zwoide (I'm counting on us winning the league this season).

I am grateful for my family-in-law, Dietmar, Birgitta, Annika, Simon, Chrissy, Jonas, Miriam, Sophia, Felix and x, my grandparents, my brothers-in-law Eddy and Joschi, my sisters Katharina, Olivia and Fenja, and my parents Karin and Rainer. Thanks for your unconditional support and love.

Finally I thank my physics lab partner, roommate, best friend and wife Ingrid. Our PhD studies were demanding, with lots of deadlines, traveling and "doing this one more thing before we take it slow," but at the end of the day we always prioritize each other. You challenge my views in and outside of science and support me in everything I do. I love our random midnight card games, our table tennis matches on the dinner table and all the sports we do together, from half marathons to bouldering to tennis.

Preface

The core of this thesis consists of four chapters based on the following publications and preprints.

Chapter 2
Real-Time Gravitational Wave Science with Neural Posterior Estimation
Maximilian Dax, Stephen R. Green, Jonathan Gair, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
Phys. Rev. Lett. **127**, 241103 – Published 8 December 2021

Chapter 3
Group equivariant neural posterior estimation
Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Deistler, Bernhard Schölkopf, Jakob H. Macke
Proceedings of the Tenth International Conference on Learning Representations (ICLR 2022) – Published 28 January 2022

Chapter 4
Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference
Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
Phys. Rev. Lett. **130**, 171403 – Published 26 April 2023

Chapter 5
Real-time inference for binary neutron star mergers using machine learning
Maximilian Dax, Stephen R. Green, Jonathan Gair, Nihar Gupte, Michael Pürrer, Vivien Raymond, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
Nature **639**, 49-53 – Published 5 March 2025

During my PhD studies, I contributed to the following publications and preprints, which are not part of this thesis.

Adapting to noise distribution shifts in flow-based gravitational-wave inference
Jonas Wildberger, **Maximilian Dax**, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
Phys. Rev. D **107**, 084046 – Published 15 April 2023

Flow matching for scalable simulation-based inference
Jonas Wildberger, **Maximilian Dax**, Simon Buchholz, Stephen R. Green, Jakob H. Macke, Bernhard Schölkopf
Neural Information Processing Systems (NeurIPS) 2023 – Published 21 September 2023

Evidence for eccentricity in the population of binary black holes observed by LIGO-Virgo-KAGRA
Nihar Gupte, Antoni Ramos-Buades, Alessandra Buonanno, Jonathan Gair, M. Coleman Miller, **Maximilian Dax**, Stephen R. Green, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Isobel M. Romero-Shaw, Bernhard Schölkopf
arXiv preprint arXiv:2404.14286 – Dated 22 April 2024

Fast and Reliable Probabilistic Reflectometry Inversion with Prior-Amortized Neural Posterior Estimation
Vladimir Starostin, **Maximilian Dax**, Alexander Gerlach, Alexander Hinderhofer, Álvaro Tejero-Cantero, Frank Schreiber
Science Advances **11**, 11 – Published 14 March 2025

Flow Matching for Atmospheric Retrieval of Exoplanets: Where Reliability meets Adaptive Noise Levels
Timothy D. Gebhard, Jonas Wildberger, **Maximilian Dax**, Annalena Kofler, Daniel Angerhausen, Sascha P. Quanz, Bernhard Schölkopf
Astronomy & Astrophysics **693**, A42 – Published online 24 December 2024

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Inverse problems play a crucial role in science and engineering. They arise because theoretical models are typically developed in the causal direction of the data generating process, mapping from a physical system to observable data. Data analysis generally aims to solve the inverse direction: starting from observed data, the goal is to infer underlying properties that could have given rise to the data (Fig. 1.1). Therefore, inference often corresponds to inversion of a forward model. Solving such inverse problems enables characterization of physical systems and mechanisms that are not directly observed, as well as validation of the corresponding forward model. Therefore, the ability to efficiently and accurately solve inverse problems is one of the major drivers of scientific progress.

We here focus on inverse problems in gravitational-wave (GW) astronomy. In the past decade, this field has emerged as an important new way of probing extreme astrophysical phenomena (e.g., black hole mergers) via the GW signals emitted by these sources. A central task in GW astronomy is to estimate source properties (e.g., black hole masses and spins) based on measured GW signals. This requires inversion of general relativity-based models, which map from source properties to signals. The ability to perform accurate GW inference by solving the associated inverse problem is of great importance to GW science, forming the basis for numerous discoveries and studies.

In this thesis, we develop a simulation-based machine learning framework for GW inference. The key idea is to directly learn solutions to inverse problems by training probabilistic inference models on simulated datasets. This connects probabilistic modeling with GW science, and the goal of this thesis is to advance the state of the art in both these fields. From the perspective of GW science, we aim to address limitations of conventional inference methods, thereby enabling new scientific analyses while also preparing GW data analysis for next-generation detectors. From the machine learning perspective, we aim to contribute approaches that are inspired by but generalize beyond GW inference, advancing the capabilities of probabilistic machine learning.

In the remainder of this chapter, we provide background information on probabilistic inference (Sec. 1.1) and GW science (Sec. 1.2). We then discuss the research gaps in these fields which we aim to address here and provide a brief overview of this thesis (Sec. 1.3).

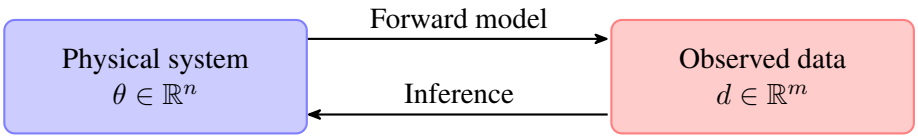


Fig. 1.1 Theoretical models in science and engineering are typically developed for the causal direction of the data generating process, and thus define a mapping from a physical system (the causes; θ) to observational data (the effects; d). Experiments and measurements often provide access only to the data d. The task of inference is then to characterize the underlying physical system in terms of parameters θ based on observed data d. Inference thus corresponds to inversion of the forward model.

1.1 Inverse problems and probabilistic inference

Given a forward model $\theta \mapsto d$, the associated inverse problem describes the task of inferring the inverse direction $d \mapsto \theta$. Here, θ denotes a set of input parameters to the forward model, which usually parameterize an underlying physical system or process (often $\theta \in \mathbb{R}^n$), and d denotes observational data (often $d \in \mathbb{R}^m$). Model inversion is a successful and broadly applicable paradigm for inference of hidden properties which are not directly captured by a measurement.

Inverse problems are ubiquitous in many disciplines. In computer vision and graphics, reconstruction of 3D structures (corresponding to θ) based on sets of 2D projections from multiple perspectives (d) can be achieved by inverting a rendering model. This is the central concept behind X-Ray computed tomography [124, 39], which infers 3D structures based on sets of 2D X-Ray scans by inverting a model for X-Ray absorption and scattering. A similar inverse problem is addressed by machine learning methods for novel view synthesis such as neural radiance fields [159] and Gaussian splatting [133]. Other inverse problems in computer vision include deblurring and inpainting, which aim to reconstruct clean images (θ) based on corrupted or partial images (d). In geophysics, rock properties (θ) can be estimated from seismic reflection or transmission data (d) via seismic inversion [199]. In astronomy, atmospheric properties of exoplanets (θ) can be estimated from observed telescope data (d) by inverting simulators for exoplanet emission spectra [155]. These are all examples of inverse problems: the quantity of interest (3D structure/clean image/rock properties/exoplanet atmosphere) is only indirectly observed via a measurement (2D projections/corrupted image/seismic data/telescope data), and there exist models to map from the former to the latter.

Forward models are often not deterministic, for example because of model uncertainties or measurement noise, and can therefore not be defined via functions $d = f(\theta)$. Instead, a stochastic forward model is described by a conditional probability distribution $p(d|\theta)$ called the likelihood. The forward model can therefore be defined explicitly via the likelihood density $p(d|\theta)$ (or a function $f(d, \theta) = z(d)p(d|\theta)$ proportional to it), or via a mechanism to sample the likelihood, $d \sim p(d|\theta)$. The density $p(d|\theta)$ quantifies how likely data d is under the model for given θ, and sampling $d \sim p(d|\theta)$ corresponds to simulating an observation for θ. Note that the density implicitly defines the corresponding simulator and vice versa, but in practice only one of those mechanisms—density evaluation or sampling—may be computationally accessible.

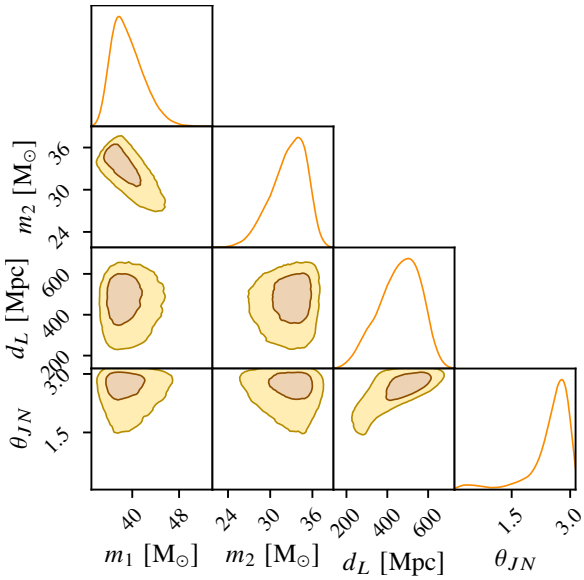


Fig. 1.2 Bayesian posterior distribution $p(\theta|d)$ over four black-hole parameters $\theta = (m_1, m_2, d_L, \theta_{JN})$ estimated from gravitational wave data d for GW150914 [4] (Fig. 1.3). One-dimensional marginals are visualized via their densities and two-dimensional marginals are indicated via 50% and 90% credible regions. The posterior captures uncertainties induced by detector noise and correlations between parameters. For example, distance d_L and inclination angle θ_{JN} are highly correlated, as both primarily effect the loudness of the gravitational wave. See Fig. A.4 for the full corner plot.

Given a forward model, the task of inference is to estimate parameters θ that could have given rise to observed data d. It is typically not possible to uniquely determine θ, for example due to incomplete data, measurement noise or degeneracies of the forward model. The inverse mapping $d \mapsto \theta$ is therefore also described by a probability distribution $p(\theta|d)$ which captures probabilistic properties such as uncertainties and correlations of the parameters (Fig. 1.2). In a Bayesian framework, the posterior is given by Bayes' theorem

$$p(\theta|d) = \frac{p(d|\theta)p(\theta)}{p(d)}, \tag{1.1}$$

where the prior $p(\theta)$ captures assumptions about θ before observing d and the evidence $p(d) = \int p(d|\theta)p(\theta)\mathrm{d}\theta$ is a normalization factor. By combining prior and model likelihood, the posterior distribution summarizes our updated knowledge about θ after observing d.

In Bayesian inference, a central task is to numerically approximate the posterior for a given observation d by generating a representative set of samples $(\theta_1, \ldots, \theta_k)$ with $\theta_i \sim p(\theta|d)$. Such samples are a useful representation, as they can capture complex posterior distributions without the need for closed-form solutions. Posterior representation via samples further enables (1) straightforward marginalization over subsets of parameters to assess uncertainties or low-dimensional correlations (Fig. 1.2), (2) cheap Monte Carlo estimation of expectation values of functions $f(\theta)$ given the data, $\mathbb{E}[f(\theta)|d] = \int f(\theta)p(\theta|d)\,\mathrm{d}\theta \approx \frac{1}{k}\sum_{i=1}^{k} f(\theta_i)$, and (3) posterior predictive checks to assess whether the forward model adequately describes the observed data. There exist a variety of methods to generate

samples from the posterior distribution. These can be grouped into approaches based on likelihood density evaluations (Sec. 1.1.1) and likelihood simulation (Sec. 1.1.2).

1.1.1 Likelihood-based inference

When the likelihood of the forward model $p(d|\theta)$ can be evaluated explicitly, likelihood-based methods such as Markov chain Monte Carlo (MCMC) [158, 117], nested sampling [206] and importance sampling [220] can be used to sample the posterior. These methods explore the parameter space by evaluating the target density for specific values of θ. For Bayesian inference, the target density is the posterior $p(\theta|d)$, which up to a normalization can be evaluated as the product of likelihood and prior. The output of these methods are sets of samples $(\theta_1, \dots, \theta_k)$, which under some assumptions are asymptotically uncorrelated and representative of the posterior. These methods are designed for unconditional distributions. In the explanations below d_o is thus considered a fixed observation of interest, and density evaluations and sampling are only performed in the space of θ.

MCMC methods generate samples via Markov chains whose equilibrium distribution is the posterior. A common MCMC method is the Metropolis-Hastings algorithm [158, 117], which adds new samples to a chain in two steps. First, a candidate θ' is proposed based on the current sample θ_t via a proposal distribution $q(\theta'|\theta_t)$. Second, the candidate is accepted as $\theta_{t+1} = \theta'$ with probability $\min\left\{\frac{p(\theta'|d_o)}{p(\theta_t|d_o)}\frac{q(\theta_t|\theta')}{q(\theta'|\theta_t)}, 1\right\}$ (see e.g. Sec. 4 in Ref. [65]); if rejected, $\theta_{t+1} = \theta_t$. This ensures that parameters with high posterior density are more likely to be sampled. Indeed, the posterior is the stationary distribution of the resulting chain if the transition q is irreducible (i.e., every θ with $p(\theta|d_o) > 0$ can be reached by the chain with finite probability) and aperiodic. While convergence is guaranteed whenever these conditions are met, the convergence rate in practice depends on the target and proposal distributions. MCMC usually applies a burn-in period, in which early samples are discarded to reduce the dependence on the starting sample. Samples are typically autocorrelated, so when (approximately) independent samples are required, the chains need to be subsampled. The necessity for burn in and subsampling can greatly reduce the sampling efficiency in practice.

Nested sampling has originally been developed for estimation of the Bayesian evidence $p(d)$, but it can also generate posterior samples. Furthermore, nested sampling is often more efficient than MCMC for multimodal inference problems, and has thus become a popular alternative [40]. The basic idea is to iteratively sample from prior regions where the likelihood exceeds a certain threshold, $p(d_o|\theta) > p^*$. The likelihood threshold p^* is increased with each iteration, progressively shrinking the prior volume and thereby focusing on higher likelihood regions.

Importance sampling represents the posterior with a set of samples $(\theta_1, \dots, \theta_k)$ with associated importance weights $(w_1, \dots, w_k)$. Samples are generated from a proposal distribution $\theta_i \sim q(\theta|d_o)$, and weights $w_i = p(\theta_i|d_o)/q(\theta_i|d_o)$ account for deviations between $p(\theta|d_o)$ and $q(\theta|d_o)$. Unweighted samples can be obtained via rejection sampling. Importance sampling requires $q(\theta|d_o)$ to cover the entire support of $p(\theta|d_o)$. The sampling efficiency further depends on how well $q(\theta|d_o)$ matches $p(\theta|d_o)$. In practice, importance sampling is not directly applicable to most problems, due to the challenges of designing a good proposal distribution $q(\theta|d_o)$ that provides sufficiently high efficiency.

In this thesis, MCMC and nested sampling are used as baselines and to provide reference results (Chapters 2,3,4,5). Importance sampling is used to augment machine learning results (Chapter 4).

1.1.2 Simulation-based inference

When the likelihood of the forward model can be sampled, $d \sim p(d|\theta)$, simulation-based inference (SBI; sometimes also called likelihood-free inference) [78] can be applied. The central idea is that the forward model enables sampling from the joint distribution

$$(\theta_i, d_i) \sim p(\theta, d) \quad \text{via} \quad \theta_i \sim p(\theta), d_i \sim p(d|\theta_i). \tag{1.2}$$

The joint distribution $p(\theta, d)$ captures all marginal and conditional distributions, including for example the posterior $p(\theta|d)$. Neural SBI leverages such samples $(\theta_i, d_i) \sim p(\theta, d)$ to train neural network-based surrogate models for the likelihood, likelihood ratios or directly for the posterior.

We here discuss SBI through the lens of Bayesian inference, although frequentist interpretations are possible in many cases. We further restrict the discussion to neural SBI, but note that there also exist traditional SBI approaches based on rejection sampling [205, 44, 45, 49, 183].

Neural posterior estimation (NPE) [175, 152, 112] directly estimates the posterior $p(\theta|d)$ with a neural density estimator $q_\psi(\theta|d)$. Here, ψ denotes learnable neural network parameters, which are optimized to achieve $q_\psi(\theta|d) \approx p(\theta|d)$. The density estimator is often parameterized with a normalizing flow [193, 178]. Normalizing flows can accurately model complex distributions, and further allow density evaluation (required for training) and sampling (required for inference). NPE training aims to minimize the marginalized Kullback–Leibler divergence between $p(\theta|d)$ and $q_\psi(\theta|d)$,

$$\begin{aligned}
\mathbb{E}_{d\sim p(d)}\left[D_{\mathrm{KL}}(p(\theta|d)||q_\psi(\theta|d))\right] &= \int p(d)p(\theta|d)\log\left(\frac{p(\theta|d)}{q_\psi(\theta|d)}\right)\,\mathrm{d}d\,\mathrm{d}\theta \\
&= \int p(\theta)p(d|\theta)\log\left(\frac{p(\theta|d)}{q_\psi(\theta|d)}\right)\,\mathrm{d}d\,\mathrm{d}\theta \\
&= \int p(\theta)p(d|\theta)\left[-\log q_\psi(\theta|d)\right]\,\mathrm{d}d\,\mathrm{d}\theta + z \\
&= \mathbb{E}_{\theta\sim p(\theta),\, d\sim p(d|\theta)}\left[-\log q_\psi(\theta|d)\right] + z.
\end{aligned} \tag{1.3}$$

Here, Bayes' theorem (1.1) is used in the second line, and z denotes the constant contribution from $\log p(\theta|d)$, which is independent of $q_\psi(\theta|d)$ and therefore irrelevant for optimization of ψ. Minimization of the Kullback-Leibler divergence can thus be achieved by minimizing the loss $L = -\log q_\psi(\theta|d)$ across samples from the joint distribution (1.2). Once trained, inference for observed data d_o can be performed by sampling from the trained density estimator, $\theta \sim q_\psi(\theta|d_o)$.

While this thesis focuses primarily on NPE, there exist a variety of other SBI methods. **Neural likelihood estimation (NLE)** [233, 91, 177, 153] trains a conditional density estimator $q(d|\theta)$ to estimate the likelihood $p(d|\theta)$. The density estimator is trained by maximizing the log probability $\log q(d|\theta)$ across samples from the joint distribution (1.2). After training, the posterior can be sampled with MCMC, using the density estimator $q(d|\theta)$ as a surrogate for the likelihood $p(d|\theta)$. **Neural**

ratio estimation (NRE) [128, 182, 76, 119, 95, 218, 160] trains a classifier to distinguish between samples from the joint $(\theta, d) \sim p(d|\theta)p(\theta)$ and marginal $(\theta, d) \sim p(\theta)p(d)$ distribution. The resulting neural network provides an estimate of the likelihood ratio $r(d|\theta_i, \theta_j) = p(d|\theta_i)/p(d|\theta_j)$, which can then be used to sample the posterior with MCMC. Finally, various lines of work explore the use of generative modeling techniques in SBI. This includes **generative adversarial training for SBI (GATSBI)** [189] based on generative adversarial networks [107], **neural posterior score estimation (NPSE)** [105, 200] based on score matching and diffusion models [207, 208, 123], **flow matching posterior estimation (FMPE)** [229] based on flow matching [148] and **consistency models for neural posterior estimation (CMPE)** [198] based on consistency models [209]. These methods directly target the posterior and could thus be regarded as NPE variants, with different density estimators and training objectives compared to the original NPE formulation.

Amortized inference

After training, the neural SBI methods described above can perform inference for any number of observations without the need for new likelihood simulations or retraining. The computational cost of simulation and training is thus *amortized* over all analyses. This makes SBI an efficient framework for performing large numbers of analyses. It also enables rapid inference of new observations, which may provide useful real-time results in time critical applications.

Sequential inference

In their generic form, SBI models are trained with samples from the joint distribution (1.2). This can be inefficient when there is only a single observation of interest d_o. The posterior $p(\theta|d_o)$ may only cover a small part of the prior $p(\theta)$ and many simulations $d \sim p(d)$ may be very different from d_o, such that the majority of training samples from the joint distribution $p(d|\theta)p(\theta)$ may not be very informative about $p(\theta|d_o)$. To better tune the training data to the observation d_o, it has been proposed to sample training parameters from a distribution $\hat{p}(\theta)$ optimized for d_o, instead of sampling from the prior $p(\theta)$ as in (1.2). An obvious choice for $\hat{p}(\theta)$ is the posterior $p(\theta|d_o)$ itself.

Sequential SBI methods for NPE [175, 152, 112, 85], NLE [177] and NRE [119, 95] train the inference model in multiple rounds, iteratively improving the proposal $\hat{p}(\theta)$. In the first round, training is performed as usual with samples from the joint distribution (1.2). After each round r, new training samples are generated based on the current inference model $(\theta_i, d_i) \sim p(d|\theta)q^r(\theta|d_o)$ (for NLE and NRE, $q^r(\theta|d_o)$ is defined implicitly and sampling requires running MCMC). This iteratively tunes the training dataset to the observation d_o. Sequential SBI methods are thus typically more simulation efficient than their generic counterparts [154] for a single observation d_o, but no longer applicable to arbitrary data $d \sim p(d)$. Posterior estimation methods such as NPE further require modifications to the training objective to account for the difference between $\hat{p}(\theta)$ and $p(\theta)$.

1.1.3 Likelihood-based versus simulation-based inference

Some forward models have a tractable likelihood $p(d|\theta)$ but cannot simulate data $d \sim p(d|\theta)$, in which case only likelihood-based inference is applicable. Conversely, other forward models can simulate

data but don't have a tractable likelihood, such that only SBI is applicable. However, many forward models provide access to both, likelihood evaluations *and* simulations. This includes in particular the gravitational wave model in Sec. 1.2.2. In such cases, likelihood-based and simulation-based methods are both applicable, and the choice of inference method involves various trade-offs. We here compare specific examples of the two inference paradigms: MCMC, which is the most common likelihood-based approach, and NPE, which is the SBI method of choice in this thesis. Note that most (but importantly not all) aspects of the discussion below transfer to the general comparison between likelihood-based and simulation-based approaches.

Computation. NPE can have great computational advantages when many observations need to be analyzed, or when rapid inference results are required for real-time applications. This is because amortized NPE performs the computation for simulation and training ahead of time (Sec. 1.1.2), enabling cheap and fast inference for any number of observations afterwards. In contrast, MCMC cannot share computation between different analyses. In some cases, these computational advantages alone may justify the choice of NPE over MCMC. On the other hand, when there are only few observations of interest and inference times are not critical, MCMC can sometimes be more efficient.

Convergence in theory. For MCMC, one can design proposal distributions which guarantee that the posterior is the equilibrium distribution. Asymptotically, for sufficiently long chains, MCMC then generates samples from the posterior. For NPE on the other hand, theoretical guarantees are weaker. Some density estimators can provably represent any well-behaved probability distribution with sufficiently large neural networks (see e.g. Sec. 2.2 in Ref. [178]), and the global optimum of the NPE training objective (Eq. (1.3)) is indeed achieved when NPE correctly predicts the posterior. However, NPE optimization is based on standard techniques for neural network training, and there exist no formal guarantees for convergence to a global optimum. From a theoretical point of view, MCMC could thus be regarded as more rigorous—although even for MCMC, convergence is not necessarily achieved in a computationally feasible number of steps.

Convergence in practice. NPE addresses two common challenges that MCMC faces in practice. First, MCMC chains are typically autocorrelated. Samples inferred with NPE density estimators on the other hand are naturally independent, removing the need for additional subsampling or thinning steps. Second, MCMC is prone to missing distributional modes. This is because MCMC typically relies on exploration of the parameter space with local proposal distributions, which impede transitions between disconnected modes. In contrast, NPE training samples are generated from the joint distribution (Eq. (1.2)), which by construction accurately captures all distributional modes. On the other hand, NPE is subject to the usual pitfalls of deep learning [120]. This includes overfitting, susceptibility to adversarial attacks and poor performance on out-of-distribution data (i.e., data very unlikely under the integrated likelihood and prior). Assessing, whether NPE training has converged may further be similarly challenging as assessing convergence of MCMC chains.

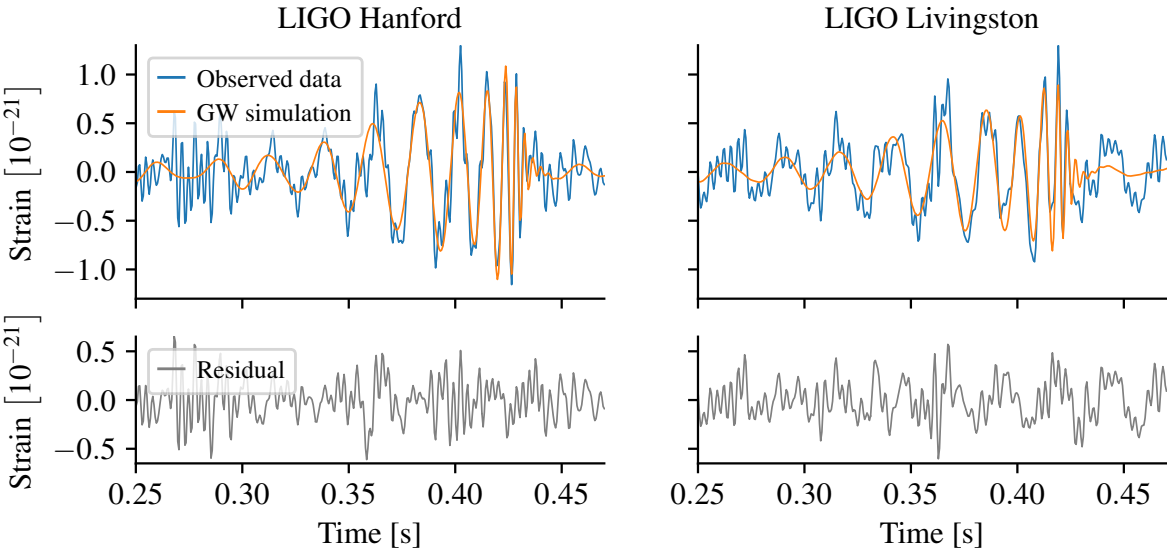


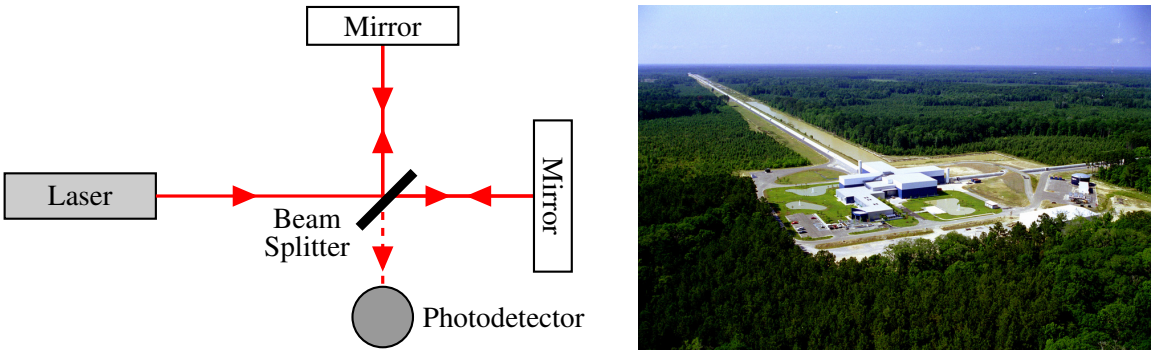
Fig. 1.3 Time series strain data for GW150914—the first-ever detected GW—comparing observed data (blue) to a corresponding GW simulation (orange) in the two LIGO detectors Hanford (left) and Livingston (right). The time is specified relative to September 14, 2015 at 09:50:45 UTC. To decrease detector noise, the frequency range is restricted to $[35, 350]$ Hz with a bandpass filter, and frequencies with high detector noise levels are removed with band-reject filters. The GW simulation is generated with the maximum likelihood parameters from a DINGO [82, 84] analysis with the waveform model IMRPhenomXPHM [185]. The bottom row shows the residual between the measured and simulated strain data. The figure layout is inspired by Fig. 1 in Ref. [4].

1.2 Gravitational waves

Gravitational waves (GWs) are deformations of spacetime that propagate at the speed of light, predicted by Einstein in 1916 [97] within his theory of general relativity. GWs are emitted by accelerated masses, just like electromagnetic waves are emitted by accelerated charges [101]. However, the gravitational interaction is so weak that extremely high masses and accelerations are required to produce detectable GWs. Realistic sources of GW measurements are thus of astrophysical origin.

Einstein's predictions of GWs have for the first time been verified in 1974 [126] with a system of two neutron stars orbiting each other. The binary system was observed through pulsed electromagnetic radiation emitted by one of the neutron stars. These measurements indicated the decay of the binary orbit over time. The corresponding energy loss was in close agreement with theoretical predictions for the energy loss from emission of GWs. This marked the first indirect observation of GWs, for which Russell A. Hulse and Joseph H. Taylor Jr. were awarded the 1993 Nobel Prize in Physics.

The first direct observation of GWs was made in 2015 [4] by the Laser Interferometer Gravitational-Wave Observatory (LIGO) [2] (Fig. 1.3). The astrophysical source was a system of two black holes, both around 30 times the mass of our Sun, around one billion light-years from Earth. The black holes spiraled around each other and merged into a single, more massive black hole. This process emitted GWs with a total energy of around 3 solar masses, which propagated through space and were detected by the two LIGO interferometers (Fig. 1.4) as tiny deformations of spacetime. Detection of these



(a) Basic Michelson interferometer

(b) LIGO Livingston detector

Fig. 1.4 GWs can be detected by measuring small distance changes with interferometry. (a) Schematic diagram of a laser-based Michelson interferometer. The interferometer splits a laser beam into two perpendicular arms, reflecting the light off mirrors at the ends of each arm. The light is then recombined to create interference patterns. Any change in the relative lengths of the arms, such as those caused by passing GWs, alters the interference pattern. (b) Photograph of the LIGO Livingston detector, showing the 4 km long interferometer arms. Courtesy Caltech/MIT/LIGO Laboratory.

GWs, referred to as GW150914, required a tremendous experimental effort. Three key scientists behind LIGO, Rainer Weiss, Kip S. Thorne and Barry C. Barish, received the 2017 Nobel Prize in Physics for this discovery.

The ability to directly observe GWs heralded a new era in astronomy, providing a novel way of probing the universe. Through GW measurements, we can now observe astrophysical events that are hard or impossible to detect in other channels (e.g., with electromagnetic radiation), such as black hole mergers. In the past decade, GW astronomy has been used to test the theory of general relativity [24] and for detailed studies of black-hole astrophysics and populations [21, 30], neutron-star physics [12] and cosmology [7]. The remainder of this section provides background information on experimental GW astronomy.

1.2.1 GW detections

GWs can be detected by directly measuring the corresponding spacetime deformations with interferometry (Fig. 1.4a). An interferometer splits a light beam into two perpendicular arms with a beam splitter. The beams are reflected back by mirrors placed at the end of each arm, recombined at the beam splitter and finally detected by a photodetector. The interference pattern of the recombined beam depends on the difference Δd of the distances traveled by the light in the two detector arms. For light with wavelength λ, the corresponding phase difference of the two beams is given by $\Delta\varphi = 2\pi\frac{\Delta d}{\lambda}$, and an additional (but constant) phase shift may arise from the beam splitter. If the total phase difference is a multiple of 2π, the beams interfere constructively and the recombined beam at the photodetector becomes stronger. Conversely, for odd multiples of π, the beams interfere destructively, leading to a vanishing (or, in practice, weaker) amplitude of the recombined beam. Therefore, the interference pattern measured by the photodetector contains information about Δd.

GWs passing through such an interferometer stretch and compress spacetime differently along the two arms. As a consequence, the distance differences between the two light paths Δd changes over time, which can be observed via the inference pattern. However, the spacetime deformations are extremely small. For example, the relative distance changes due to GW150914 were of order 10^{-21} (Fig. 1.3), and therefore changed the length of a kilometer-long interferometer arm by around 10^{-18} m, which is roughly one thousandth the size of a proton. Resolving such tiny distance changes is experimentally extremely challenging. GW observatories thus employ a variety of techniques to enhance the basic Michelson interferometer. For example, LIGO uses Fabry Perot cavities, which repeatedly reflect the beams within the individual arms and thereby increase their effective length; power and signal recycling mirrors to enhance the signal in the photodetector; and sophisticated damping mechanisms to suppress noise [2].

There exist several interferometer-based GW detectors. The two LIGO detectors Hanford (Washington, United States; 4 km long interferometer arms) and Livingston (Louisiana, United States; 4 km arms; Fig. 1.4b) made their first detection in 2015, with the aforementioned event GW150914 [4]. The Virgo detector [32] (near Pisa, Italy; 3 km arms) made its first detection in 2017 [8]. The Kamioka Gravitational Wave Detector (KAGRA) [42, 35] (Kamioka, Japan; 3 km arms; built underground) and GEO600 [90] (near Hannover, Germany; 600 m arms) are operational [26], but have not yet reported a GW detection due to their reduced sensitivity compared to LIGO and Virgo. Use of multiple detectors is crucial for GW astronomy. Coincident measurements of GW signals in spatially separated and independent detectors greatly increases the confidence in detections. Furthermore, observation in multiple detectors enhances GW source localization, as the incident direction can be constrained from differences between the detection times. Multi-detector observations also provide greater combined signal-to-noise ratios for estimation of other GW source parameters. The observatories thus collaborate closely to form a global detector network.

Since 2015, LIGO and Virgo have reported 90 candidates for GW detections [15, 22, 31, 29] from their first three observering runs, O1, O2 and O3 (Fig. 1.5). In addition, individual candidates have been reported [3] from the fourth observing run (O4, May 2023–June 2025) of the LIGO Scientific, Virgo, and KAGRA Collaboration (LVK), and frequent public alerts (e.g., Ref [146]) indicate a high detection rate in O4. All of the confident observations are believed to originate from compact binary coalescences, that is, mergers of two compact objects. Specifically, three types of events have been reported, binary black hole, neutron star–black hole, and binary neutron star mergers. Mergers involving neutron stars can emit electromagnetic signals in addition to GWs. So far, such so-called counterparts have only been observed for one event [75, 9], which has led to important scientific results [7, 10, 13, 12, 14], including an independent measurement of the Hubble constant [7] and tests of general relativity [14]. Such observations are challenging, as telescopes need to be pointed into the direction of the merger, which requires accurate source localization based on GW signals.

1.2.2 Gravitational-wave data models

The primary data product of GW detectors is a time series measurement of the spacetime strain $d(t)$ (Fig. 1.3). This is obtained from the calibrated output of the photodetector, sampled with a fixed frequency ($f_s = 16384$ Hz for LIGO), resulting in a discrete time series $d_i = d(t_i)$ with

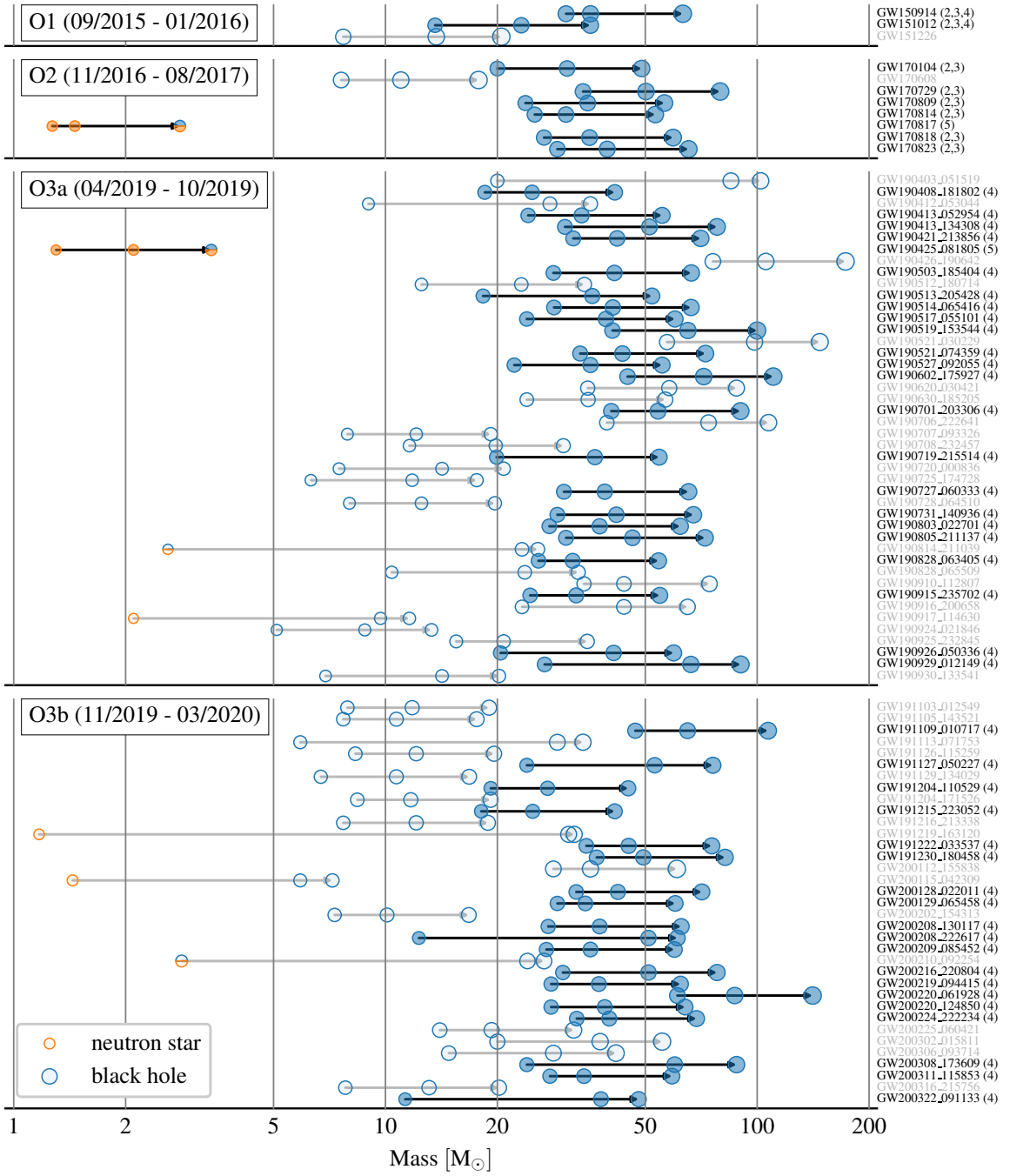


Fig. 1.5 Mergers of compact objects during the first three LIGO-Virgo observing runs O1–O3. Each line represents one event (90 in total), showing the two initial masses and the final mass connected with an arrow. Events analyzed with DINGO [82, 84] in this thesis (51 in total) are highlighted (black event names and arrows, filled circles), with the corresponding chapter numbers in parentheses.

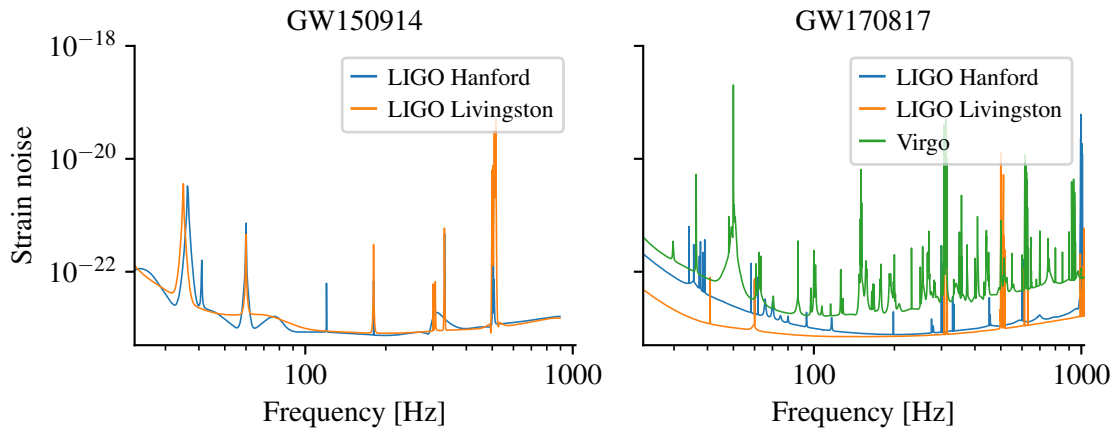


Fig. 1.6 Detector noise ASDs of the LIGO-Virgo detectors, quantifying the detector noise amplitude in each frequency bin. Detectors are upgraded between observing runs, so the noise level at the time of GW150914 [4] in O1 (left) is higher than the noise level at the time of GW170817 [13] in O2 (right). Here, the ASDs are multiplied by $\sqrt{4\Delta f}$ to make them independent of the frequency resolution Δf and dimensionless.

$\Delta t = t_{i+1} - t_i = 1/f_s$. This time series is assumed to be the superposition of GW strain $h(t)$ and additive detector noise $n(t)$, $d(t) = h(t) + n(t)$. Analysis of GW data thus requires modeling of both, detector noise as well as GW signals.

Detector noise model

The data collected by LVK detectors are affected by various types of noise. This includes fundamental sources like quantum and thermal noise, as well as technical sources, like laser frequency fluctuations and photodetector dark noise (see Sec. 3 in Ref. [1] and Sec. 3 in Ref. [18] for details).

The definition of a statistical noise model requires one to make additional assumptions. First, detector noise is commonly assumed to follow a *Gaussian* distribution. This is often a good approximation, as detector noise typically arises as the superposition of many small, (mostly) uncorrelated effects (cf. central limit theorem). Second, the noise is commonly assumed to be *stationary*, that is, statistical properties do not change under time shifts. Formally, this means that the noise covariance matrix C_{ij}—which captures correlations between noise at times t_i and t_j and is defined as $C_{ij} = \mathbb{E}[(n_i - \mu)(n_j - \mu)]$ with $\mu = \mathbb{E}(n)$—only depends on the difference $|t_i - t_j|$, or equivalently, on $|i - j|$. Note that while both these assumptions are usually applied in GW data analysis, real detector noise is neither exactly Gaussian nor stationary, and there are efforts to mitigate these deviations [18].

GW data analysis is typically performed in the Fourier transformed domain, which is particularly well suited for stationary noise. In the following, all quantities are thus represented in the Fourier domain, omitting the usual hat notation. The noise covariance matrix is diagonal in Fourier domain,

$$C_{ij} = \delta_{ij} S_n(f_i), \tag{1.4}$$

Description	Parameter	Description	Parameter
component masses	m_1, m_2	polarization	ψ
spin magnitudes	a_1, a_2	phase of coalescence	ϕ_c
spin angles	θ_1, θ_2, ϕ_{12}, ϕ_{JL}	time of coalescence	t_c
tidal parameters	λ_1, λ_2	sky position	α, β
inclination	θ_{JN}	luminosity distance	d_L

Table 1.1 Parameters describing a compact binary coalescence. Indices $1, 2$ refer to the individual compact objects (black holes or neutron stars). Parameters λ_i are only applicable to neutron stars.

with indices i, j now referring to frequency bins $f_{i,j}$. This defines the power spectral density (PSD) $S_n(f)$ and the amplitude spectral density (ASD) $\sqrt{S_n(f)}$. The ASD is a property of the detector, quantifying the noise level in each frequency bin (Fig. 1.6), which corresponds to the free parameters of the stationary and Gaussian detector noise model. The ASD can be determined empirically, for example with Welch's method [227] via the power spectrum of a long segment of signal-free data, or with tools such as BAYESWAVE [73, 149] which simultaneously fit the GW signal and noise distribution.

The ASD quantifies the sensitivity of the detector as a function of frequency—the lower the ASD, the lower the noise level, and the higher the sensitivity. The LVK detectors are most sensitive in a frequency band between roughly 10^2 Hz and 10^3 Hz, resulting in U-shaped ASDs (Fig. 1.6). The steep increase towards lower frequencies is primarily due to seismic noise, and at higher frequencies, the ASDs are dominated by quantum noise. The LVK ASDs further have several narrow peaks from electrical and mechanical sources. For example, there are peaks at 60 Hz for the US-based LIGO detectors and 50 Hz for the Europe-based Virgo detector due to the frequencies of the respective power grids. Detector noise ASDs change over time. In between observing runs, detectors are upgraded, resulting in improved sensitivity (Fig. 1.6). Even throughout an observing run, the detector noise level drifts, such that ASDs have to be estimated for each event individually.

Gravitational wave signal models

A compact binary coalescence is parameterized by its component masses (2), spins (6), tidal deformabilities (1 per neutron star involved), inclination (1), polarization (1), phase (1) and time (1) of coalescence, sky position (2), and distance (1) (Tab. 1.1). This amounts to 15 parameters for binary black hole, 16 for neutron star-black hole and 17 for binary neutron star events. These parameters θ determine the GW signal, which can be computed with waveform models.

The gold standard for waveform modeling are *numerical relativity (NR) simulations* [181, 197]. These solve Einstein's equations numerically, and are therefore accurate but also computationally expensive. Data analysis is typically performed with cheaper waveform models, such as *NR surrogate models* [47, 221], which interpolate NR simulations but are only available for restricted regions of the parameter space, *effective-one-body models* [55, 173, 190], which combine NR and perturbative calculations, and *phenomenological models* [115, 134, 50, 185]. Based on the binary parameters θ, these waveform models compute the two GW polarizations, $h_+(\theta)$, $h_\times(\theta)$, which can then be projected onto the detectors to obtain the simulated GW signal $h(\theta)$ in time or frequency domain.

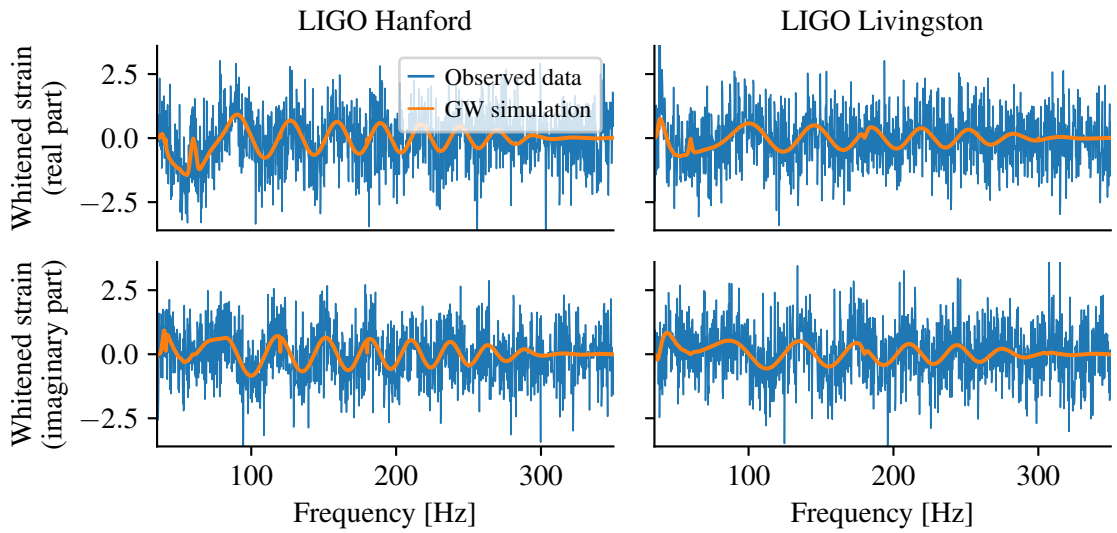


Fig. 1.7 Frequency domain data (blue) for GW150914 in the two LIGO detectors Hanford (left) and Livingston (right), obtained by Fourier transforming the time series strain (Fig. 1.3). The data are whitened by dividing by $\sqrt{S_n/(4\Delta f)}$, such that detector noise should have unit variance. The GW simulation (orange) is generated using the maximum-likelihood parameters from a DINGO analysis.

Gravitational wave likelihood

With the assumptions above, GW data $d = h(\theta) + n$ is modeled as the sum of a signal $h(\theta)$ and stationary, Gaussian noise n with PSD S_n. This implies the likelihood

$$p(d|\theta) = \prod_i p(d_i|\theta), \tag{1.5}$$

with

$$p(d_i|\theta) = \frac{1}{2\pi(S_n)_i} \exp\left(-2\Delta f \frac{|d_i - h_i(\theta)|^2}{(S_n)_i}\right), \tag{1.6}$$

where data d and signal $h(\theta)$ are both represented as complex frequency series with i indexing the frequency bins, potentially in multiple independent detectors. Intuitively, the likelihood compares the data to the waveform simulation by quantifying whether the residual $r = d - h(\theta)$ is a plausible realization of the noise model (Fig. 1.7). Specifically, evaluation of Eq. (1.6) thus corresponds to computing the probability of r_i under a Gaussian with mean 0 and standard deviation $\sigma_i = \sqrt{(S_n)_i/(4\Delta f)}$. This is done for real and imaginary part individually, resulting in a two-dimensional Whittle likelihood [228]. Similarly, the likelihood can be sampled, $d \sim p(d|\theta)$, by adding noise n with mean 0 and standard deviation σ_i to the signal $h(\theta)$, $d = h(\theta) + n$. Likelihood evaluation and sampling both have roughly the same computational cost, which is dominated by the waveform simulation $h(\theta)$.

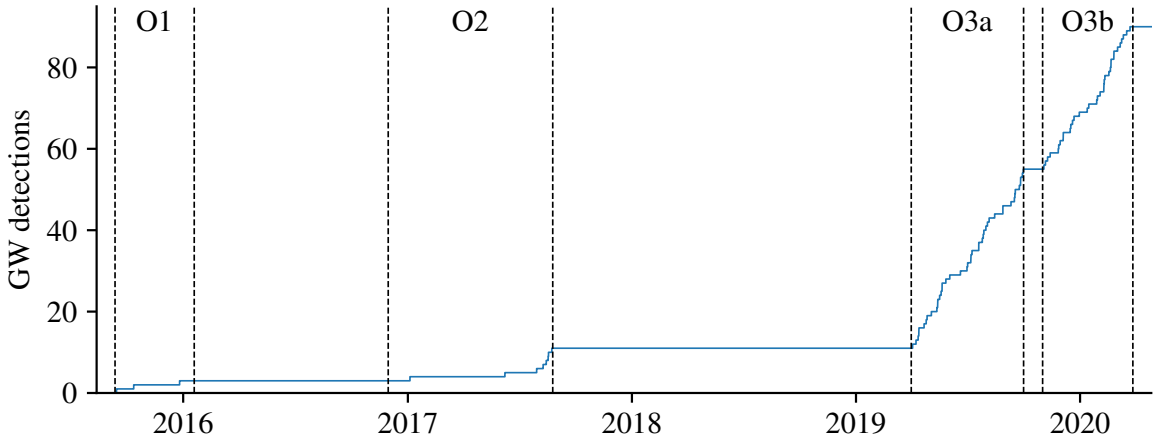


Fig. 1.8 Cumulative count of confident GW detections during the first three LIGO-Virgo observing runs O1–O3. The event rate increases between observing runs due to upgrades to detector sensitivities. It is expected that this trend continues in the current observing run O4 (May 2023–June 2025).

1.2.3 Gravitational-wave data analysis

Data analysis at the LVK can be grouped into two central tasks.

GW search [100, 166, 58, 34, 66, 196, 167, 140] aims to identify GW event candidates in the detector data and to assess their statistical significance. This is commonly achieved with a technique called matched filtering, which compares the data to a large bank of GW signal templates. These templates are generated to densely cover the parameter space of compact binary coalescences. Given measured data, matched filtering searches for the template with the highest signal-to-noise ratio in the bank. The statistical significance of a GW candidate is then estimated based on this optimal template, using the signal-to-noise ratio and other detection statistics. The corresponding template parameters further provide an initial estimate of the GW source properties. The LVK employs multiple independent search pipelines, including PyCBC [166], GstLAL[58], MBTA [34] and SPIIR [66].

GW parameter estimation aims to estimate GW source parameters, typically in a Bayesian framework (Sec. 1.1). Parameter estimation is performed for event candidates identified by search pipelines, and assumes that the data indeed contain a GW signal. Given observed data d, a likelihood $p(d|\theta)$ defined by waveform and noise models (Sec. 1.2.2) and a prior $p(\theta)$, the goal is to characterize the astrophysical source in terms of the Bayesian posterior $p(\theta|d)$ over source parameters θ. The LVK traditionally employs likelihood-based methods (Sec. 1.1.1) implemented by the LVK tools LALInference [224] or Bilby [41, 195, 210] to generate posterior samples $(\theta_1, \ldots, \theta_k)$. Accurate and reliable inference methods are of great importance in GW science, as they underlie the vast majority of downstream analyses (see for example Refs. [12, 25, 28–30]).

1.3 Overview of this thesis

Conventional GW inference (Sec. 1.2.3) often provides excellent results. However, inference times typically range from hours to months, depending on the complexity of the GW waveform model. With the growing GW detection rate (Fig. 1.8), the computational cost of inference becomes increasingly problematic. More efficient GW inference methods are thus essential to continue analyzing each GW event individually, to routinely use the most physically realistic waveform models, and to perform large-scale searches for new discoveries (e.g., eccentricity in black hole orbits, deviations from general relativity). Moreover, GW inference provides crucial information (e.g., source location, mass parameters) required to direct multi-messenger searches for potential electromagnetic counterparts. However, conventional inference is too slow to provide this information in very low latency. Finally, planned next-generation detectors will amplify the issues above, with even higher detection rates, more multi-messenger events and new data analysis problems such as overlapping signals.

This thesis introduces a paradigm for GW data analysis intended to address these limitations. Our new framework, called DINGO ("Deep INference for Gravitational-wave Observations"), augments the SBI method NPE with several new techniques to solve the challenging problem of GW inference.

Chapter 2 introduces DINGO, demonstrating for the first time that SBI can meet the high accuracy requirements in GW science while also being $\sim 10^3$ times faster than conventional methods. Therefore, this foundational chapter motivates the subsequent research efforts to develop a comprehensive framework that covers all areas of GW inference. Crucially, DINGO explicitly integrates GW symmetries into the inference algorithm to enhance its performance. Chapter 3 describes this technique in detail and generalizes it beyond the GW application.

While theoretically principled and empirically successful, DINGO lacks formal accuracy guarantees. However, GW science requires highly reliable data analysis. Chapter 4 addresses this issue with DINGO-IS, which combines DINGO with likelihood-based importance sampling. This establishes an independent mechanism to verify and potentially correct DINGO predictions, providing results that are asymptotically free from deep learning inaccuracies. Chapter 4 further extends DINGO to more physically realistic GW waveform models, one of which is so computationally expensive that conventional inference would require several months of computation per event. The ability to verify results without comparison to conventional inference is therefore crucial for such analyses.

With Chapters 2–4, DINGO can analyze most binary black holes and is tested with the most common GW waveform models. However, possibly the greatest scientific benefit of fast GW inference is its ability to improve searches for electromagnetic counterparts, which are not expected to be observed for binary black holes. In contrast, binary neutron stars are known to emit such counterparts, but are challenging to analyze with machine learning due to their long and complex signals. Chapter 5 introduces various innovations that, for the first time, enable full GW inference for binary neutron stars in less than one second. This method, DINGO-BNS, can further analyze binary neutron stars even before the merger, and scales to hour-long signals in next-generation detectors.

Chapter 2

Real-Time Gravitational Wave Science with Neural Posterior Estimation

We demonstrate unprecedented accuracy for rapid gravitational-wave parameter estimation with deep learning. Using neural networks as surrogates for Bayesian posterior distributions, we analyze eight gravitational-wave events from the first LIGO-Virgo Gravitational-Wave Transient Catalog and find very close quantitative agreement with standard inference codes, but with inference times reduced from $O(\text{day})$ to 20 seconds per event. Our networks are trained using simulated data, including an estimate of the detector-noise characteristics near the event. This encodes the signal and noise models within millions of neural-network parameters, and enables inference for any observed data consistent with the training distribution, accounting for noise nonstationarity from event to event. Our algorithm—called "DINGO"—sets a new standard in fast-and-accurate inference of physical parameters of detected gravitational-wave events, which should enable real-time data analysis without sacrificing accuracy.

Declaration

This chapter is based on the following published manuscript.

> *Real-Time Gravitational Wave Science with Neural Posterior Estimation*
> Maximilian Dax, Stephen R. Green, Jonathan Gair, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
> Phys. Rev. Lett. **127**, 241103 – Published 8 December 2021

Text and figures are adapted from the corresponding arXiv version `arXiv:2106.12594v2` with minor updates to layout and references.

Author contributions (CRediT)

Maximilian Dax: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Visualization
Stephen R. Green: Conceptualization, Methodology, Validation, Formal Analysis, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision
Jonathan Gair: Conceptualization, Formal Analysis, Writing - Review & Editing
Jakob H. Macke: Conceptualization, Writing - Review & Editing, Supervision
Alessandra Buonanno: Conceptualization, Writing - Review & Editing
Bernhard Schölkopf: Conceptualization, Writing - Review & Editing, Supervision

2.1 Introduction

Since the first detection of a signal from a pair of merging black holes [4], gravitational waves have quickly emerged as an important new probe of gravitational theory [24], neutron-star physics [12], cosmology [19], and black-hole astrophysics [21]. These scientific successes were made possible by a growing rate of detections by the LIGO [2] and Virgo [32] observatories, and their subsequent analysis and characterization as signals from merging compact binary systems. The LIGO and Virgo Collaborations (LVC) have now published results from over 50 such systems [15, 23], and this number promises to grow ever-faster as detectors are made more sensitive in the future [17].

Given a detection, Bayesian inference is used to characterize the originating source [18]. This is based on having models for the signals and the detector noise. For gravitational waves, signal models take the form of waveform predictions $h(\theta)$ depending on the source parameters θ (masses, location, etc.). Waveform models are based on solutions to Einstein's equations (and any relevant matter equations) for the two-body dynamics and gravitational radiation, using a combination of numerical-relativity and perturbative calculations [55, 51, 222] and phenomenological fitting [188, 134, 222]. Detector noise is typically modeled as stationary and Gaussian, with some spectrum which can be estimated empirically. Together, these "forward" models give rise to the likelihood $p(d|\theta)$ for the observed *strain* data d, which is assumed to consist of a signal plus noise. With the choice of a prior $p(\theta)$ over parameters, the posterior distribution is given via Bayes' theorem,

$$p(\theta|d) = \frac{p(d|\theta)p(\theta)}{p(d)}, \tag{2.1}$$

where $p(d)$ is a normalizing factor called the evidence. The posterior gives our belief about the source parameters, given the observed data.

The task of inference is to characterize the posterior by drawing *samples* from it. This can be accomplished with stochastic algorithms like Markov chain Monte Carlo (MCMC). The LVC have developed software tools such as LALInference [224] and Bilby [41, 195, 210] to carry this out. However, these algorithms are computationally expensive as they require many likelihood evaluations for each independent posterior sample $\theta \sim p(\theta|d)$, and each likelihood requires a waveform simulation. An analysis producing $\sim 10^4$ independent samples typically requires millions of waveform evaluations and a total inference time of hours to months, depending on the signal duration and waveform model. More physically-realistic waveform models [170] are also more costly, so carrying out inference for all events with the best models is an enormous computational effort. When rapid results are desired—for alerts to trigger electromagnetic follow-up of transient phenomena [6], or when processing large numbers of events—accuracy usually has to be traded off for speed, by restricting to a limited set of fast models [184, 185] or specialized inference algorithms [203, 143, 71].

In this Letter, we describe an alternative approach to gravitational-wave inference which delivers both dramatically reduced analysis time *and* high accuracy, in stark contrast to the trade-off intrinsic to standard algorithms. The basic idea is to produce a large number of simulated data sets (with associated parameters), and use these to train a type of neural network known as a *normalizing flow* to approximate the posterior. The trained network can then generate new posterior samples extremely

quickly once a detection is made. This bypasses the need to generate waveforms at inference time, thereby *amortizing* the expensive training costs over all future detections. The general approach of building such "surrogate" inverse models is called *neural posterior estimation* (NPE) [175, 152, 112], and is beginning to see application in several scientific domains [77]. When applied to gravitational waves, with all of the optimizations we describe, we call the method *Deep INference for Gravitational-wave Observations*, or DINGO.

NPE and conventional methods both involve the same inputs: a prior and a likelihood. A key difference, however, is the way in which the likelihood is used: for conventional methods, its density is *evaluated*, whereas for NPE it is used to *simulate data*, i.e., $d \sim p(d|\theta)$. This distinction is important when dealing with nonstationary or non-Gaussian detector noise, for which an analytic likelihood is either expensive or unavailable. In this case, one could nevertheless simulate data, in a noise-model-independent way, by injecting simulated signals into real noise. Our present focus is on speed and on validating DINGO on real data with the common assumption of stationary-Gaussian noise, but the ultimate aim of more accurate inference using real noise should be kept in mind.

There have been several previous studies that applied NPE or related approaches to gravitational waves [103, 67, 61, 111, 110, 86, 141, 201]; see also [80]. However, most of these are limited in some way: they either restrict the number of parameters or the distributional form of the posterior, or they do not analyze real data, or there are clear deviations from results obtained using standard algorithms. The best performance to-date was achieved in the study [110] by some of us. This was the only study to infer all 15 parameters[1] of a binary black hole (BBH) system in real data and demonstrate close agreement to standard samplers. However, even that study did not achieve full amortization, as it did not address the fact that detector noise varies from event to event. Rather, the neural network of [110] was tuned to the noise power spectral densities (PSDs) of the detectors at the time of the event analyzed, and it would require retraining for each new event.

We now present for the first time completely amortized inference for BBHs using DINGO. This is achieved by *conditioning* the neural network not only on the event strain data, but also on the detector noise PSD, which can be estimated using nearby data [224]. We also achieve unprecedented accuracy thanks to a new iterative algorithm for time-shifting the coalescence times, as well as various architecture improvements. We use our trained networks to analyze all events in the first Gravitational-Wave Transient Catalog (GWTC-1) [15] with component masses greater than $10\,\mathrm{M}_\odot$ (our prior bound) and find close (sometimes indistinguishable) quantitative agreement with standard algorithms. This Letter sets a new standard for rapid gravitational-wave inference, which should enable real-time gravitational-wave science in the near future. It shows that NPE has moved beyond toy models and is competitive with conventional algorithms. More broadly, it provides a demonstration of these new methods in a realistic use case, which we hope will inspire wider adoption in experimental science.

[1] Parameters consist of detector-frame component masses (m_1, m_2), time of coalescence at geocenter t_c, reference phase ϕ_c, sky position (α, δ), luminosity distance d_L, inclination angle θ_{JN}, spin magnitudes (a_1, a_2), spin angles $(\theta_1, \theta_2, \phi_{12}, \phi_{JL})$ [98], and polarization angle ψ.

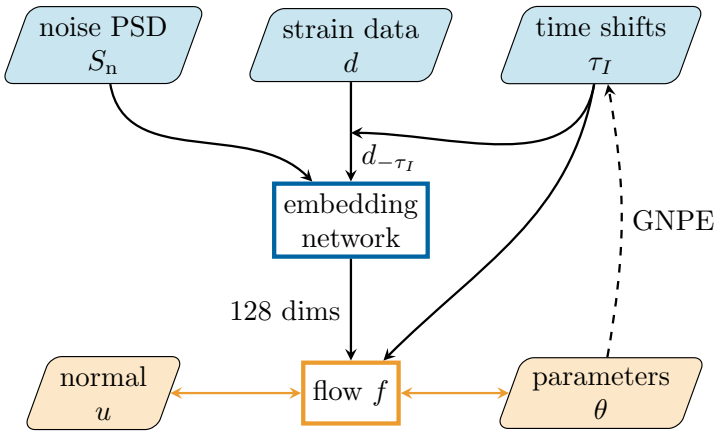


Fig. 2.1 DINGO flow chart. The posterior distribution is represented in terms of an invertible *normalizing flow* (orange), taking normally-distributed random variables u into posterior samples θ. The flow itself depends on a (compressed) representation of the noise properties S_n and the data d, as well as an estimate τ_I of the coalescence time in each detector I. The data are time-shifted by τ_I to simplify the representation. For inference, the iterative *group equivariant neural posterior estimation* (GNPE) algorithm is used to provide an estimate of τ_I, as described in the main text.

2.2 Method

The central object of DINGO is the density-estimation neural network, which defines a conditional probability distribution $q(\theta|d)$. This should be distinguished from the posterior $p(\theta|d)$, which $q(\theta|d)$ learns to approximate through training. We use so-called normalizing flows [193, 136, 176] to define a sufficiently flexible $q(\theta|d)$ via a d-dependent mapping $f_d : u \mapsto \theta$ from a simple "base" distribution $\pi(u)$,

$$q(\theta|d) = \pi\left(f_d^{-1}(\theta)\right)\left|\det J_{f_d^{-1}}\right|. \tag{2.2}$$

If $\pi(u)$ can be rapidly evaluated and sampled from, and if f_d is invertible and has simple Jacobian determinant, then $q(\theta|d)$ can also be rapidly evaluated and sampled from. Following [110], we take $\pi(u)$ to be multivariate standard normal, and f_d a composition of spline coupling flows [93], each of which is defined with a neural network.

The overall structure of DINGO is illustrated in Fig. 2.1. This contains three key enhancements compared to the study [110]. First, since the data generation process depends on the detector noise PSD S_n, we include this as additional context to the neural network, i.e., $q(\theta|d, S_\mathrm{n})$. This allows us to tune the network at inference time to the PSD estimated just prior to the event, corresponding to standard "off-source" noise estimation [224]. An alternative would be to estimate the noise "on-source" [149], but since we consider only short-duration BBH events here, the off-source approach is sufficient.

The second enhancement addresses the problem of high-dimensional observed data by using an additional neural network to first compress to a small number of features. This network (called an "embedding network") is trained alongside the flow network. Our data is in the frequency domain, between 20 Hz and 1024 Hz, with 0.125 Hz resolution, so combined with the PSDs, this gives 24,096

input dimensions for each of the two or three interferometers. The first stage of the embedding network maps this linearly to 400 components per detector. To provide an inductive bias to extract signal information, we seed this layer with the principal components of clean waveforms from our training set, and then allow these parameters to float during training. Following this, a fully-connected residual network [118] compresses to 128 features, which are provided to the flow.

Finally, we developed a new method to treat time translations of the strain data. For standard algorithms, inference of (α, δ, t_c) requires sampling over waveforms with varying coalescence times t_I in each detector I. Likewise for NPE, the network must learn to interpret strain data with different t_I. For frequency-domain data, however, time translations correspond to local phase shifts, which, although explicitly known, are challenging for neural networks to learn. Indeed, this occupied much of the network capacity in Ref. [110]. Our new approach—called *group equivariant* neural posterior estimation (GNPE)—leverages explicit knowledge of the time-translation symmetry along with *approximate* knowledge of t_I to simplify the data representation and allow the network to focus on more nontrivial parameters. For further details see [83].

For GNPE, we train the network to infer θ given perturbed coalescence times τ_I and manually-time-shifted strain data $d_{-\tau_I}$. Using maximum likelihood estimation [108], this means we minimize the loss function

$$L = \mathbb{E}_{p(\theta)}\mathbb{E}_{p(S_\mathrm{n})}\mathbb{E}_{p(d|\theta,S_\mathrm{n})}\mathbb{E}_{\kappa(\delta t_I)}\left[-\log q(\theta|d_{-t_I(\theta)-\delta t_I}, S_\mathrm{n}, t_I(\theta)+\delta t_I)\right], \tag{2.3}$$

with respect to the network parameters [135]. Here, $\mathbb{E}$ refers to the expected value over the specified distribution, which is evaluated stochastically using Monte Carlo draws. $\kappa(\delta t_I)$ is a uniform kernel used to perturb t_I. For inference, even though we do not have direct access to t_I, all parameters can be inferred using Gibbs sampling starting with an approximate t_I (obtained, e.g., using standard NPE): first, convolve t_I with $\kappa(\delta t_I)$ to obtain τ_I, then use the network to infer a new estimate for t_I; then convolve again and repeat. We find that this converges after $O(10)$ iterations.

Evaluating (2.3) requires sampling $\theta^{(i)} \sim p(\theta)$ and $S_\mathrm{n}^{(i)} \sim p(S_\mathrm{n})$, and then simulating data $d^{(i)} \sim p(d|\theta^{(i)}, S_\mathrm{n}^{(i)})$. Aside from the PSD sampling, this follows Ref. [110] very closely. In particular we use the same prior over parameters, with $m_1, m_2 \in [10, 80]\ \mathrm{M}_\odot$. We train separate networks for the noise distributions in the first (O1) and second (O2) observing runs of LIGO and Virgo, with PSD samples estimated empirically from stretches of interferometer noise data [20]. For O1, we choose the distance prior $[100, 2000]$ Mpc. For O2, we train one network for loud events with distance prior $[100, 2000]$ Mpc and another for quieter events with $[100, 6000]$ Mpc. In addition to these two-detector networks, we train a three-detector network with distance prior $[100, 1000]$ Mpc to analyze GW170814. With future enhancements of network architecture we expect to cover the entire distance range with a single network. Finally, as in Ref. [110], training data are generated from a fixed set of spin-precessing frequency-domain waveforms, described by the IMRPhenomPv2 [115, 134, 50] model, but with extrinsic parameters and noise realizations drawn randomly during training. With training sets of 5×10^6 waveforms, there is no indication of overfitting. Training takes roughly 10 days on a single NVIDIA A100. Further details on the networks and training are provided in Chapter A.

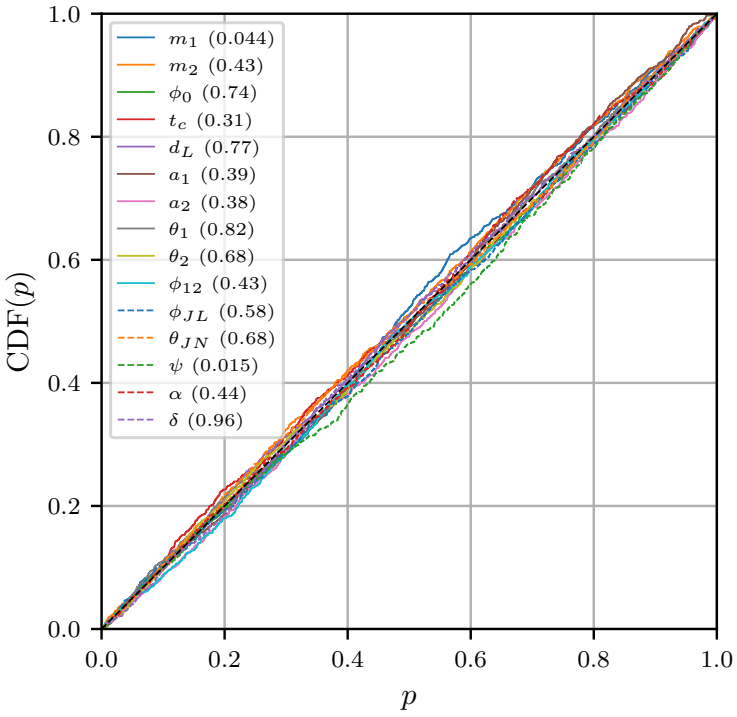
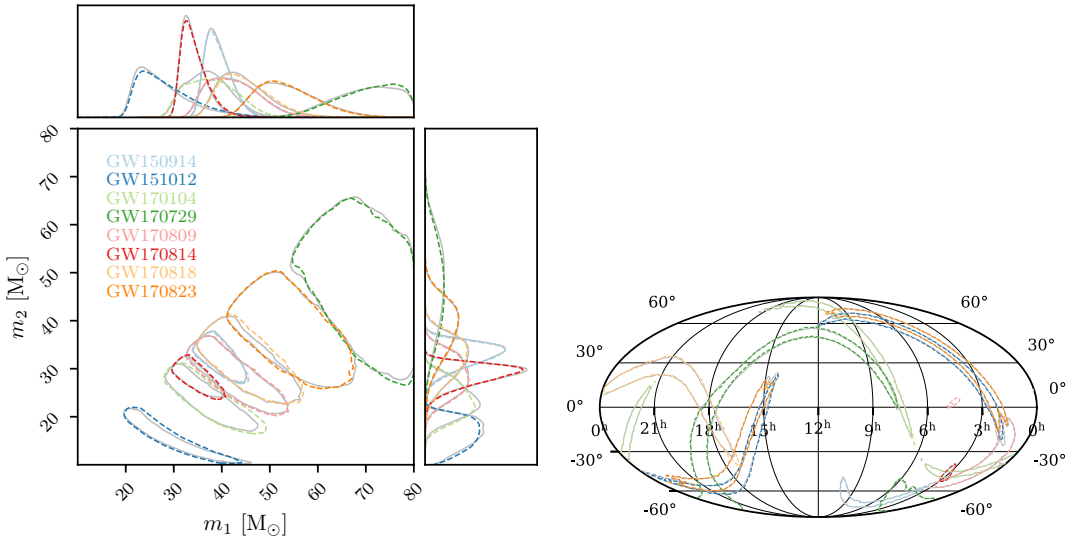


Fig. 2.2 P–P plot for 1000 injections. The legend shows the p-values of the individual parameters, with a combined p-value of 0.46.

2.3 Results

As a first test, we evaluate DINGO on data entirely consistent with the training distribution, i.e., simulated waveforms in stationary-Gaussian noise. This is an easier task than using observational data, which includes real signals in noise that is neither strictly stationary nor Gaussian, and therefore lies outside the training distribution. We sample posteriors from 1000 simulated data sets and construct a P–P plot (see Fig. 2.2). For each parameter, we compute the percentile score of the true value within its marginalized posterior, and then we plot the cumulative distribution function (CDF) of these scores. For true posteriors, the percentiles should be uniformly distributed, so the CDF should be diagonal. Kolmogorov-Smirnov test p-values are indicated in the legend, with combined p-value of 0.46. This shows that DINGO is performing properly on simulated data.

We now proceed to our main result, which is a demonstration of performance on real events. We perform inference on the eight GWTC-1 BBH events compatible with our prior, using both DINGO and LALInference MCMC. For DINGO, generation of 50,000 sample points with 30 GNPE iterations takes roughly 20 seconds. Comparisons of inferred component masses and sky position for all events show good agreement (see Fig. 2.3), including multimodality for the sky position. The one exception is GW170104, where the mass posterior is slightly flatter. Nevertheless, 90% credible intervals are in good agreement.



(a) (b)

Fig. 2.3 Comparison of (a) detector-frame component mass and (b) sky position posteriors from DINGO (colored) and LALInference (gray) for eight GWTC-1 events. 90% credible regions shown.

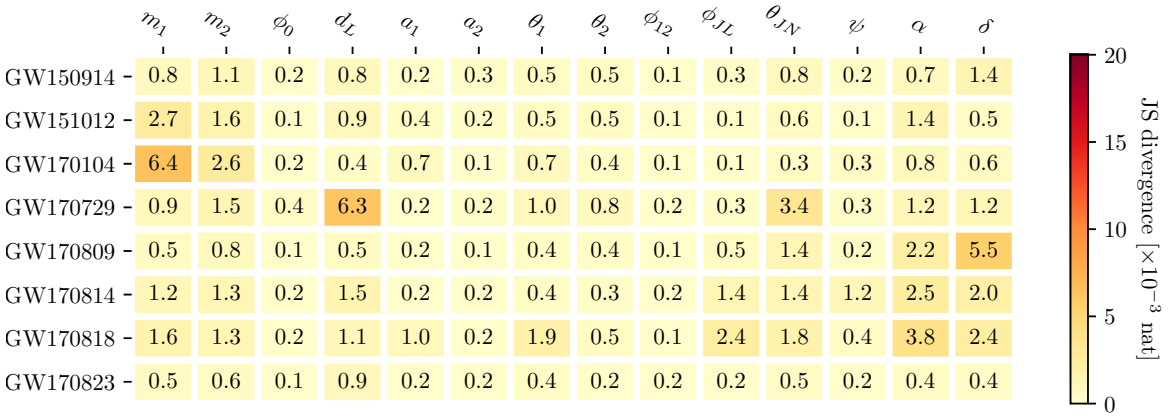


Fig. 2.4 JSDs between DINGO and LALInference marginalized posteriors, averaged over 100 realizations. The mean JSD across all events and parameters is 0.0009 nat.

For quantitative comparisons, we compute the Jensen-Shannon divergence (JSD) [147] between DINGO and LALInference one-dimensional marginalized posteriors (see Fig. 2.4). This is a symmetric divergence that measures the difference between two probability distributions, with values ranging from 0 to $\ln(2) \approx 0.69$ nat. We find a mean JSD across all events and parameters of 0.0009 nat, which is slightly higher than the variation (0.0007 nat) found between LALInference runs with identical settings but different random seeds [195]. By comparing such LALInference runs, Ref. [195] also established a maximum JSD of 0.002 nat for indistinguishability; our results are approaching this threshold, with two events below for all parameters, and the others with one to three parameters above. The slight visible disagreement between mass posteriors for GW170104 is also reflected in larger JSDs. For comparison, we note that PSD variations (see Fig. A.2) and the choice of waveform model [195] both impact the JSD at a much higher level (0.02 nat). Additional comparisons between samplers, including posteriors for all events, are provided in Chapter A.

2.4 Conclusions

In this Letter, we introduced DINGO and applied it to perform extremely fast Bayesian parameter inference for gravitational waves observed by the LIGO and Virgo detectors. We analyzed eight GWTC-1 events, and showed excellent agreement with standard algorithms, with inference times reduced by factors of 10^3–10^4. This was achieved by conditioning on the detector noise characteristics and making a number of architecture and algorithm improvements. The DINGO code is available at https://github.com/dingo-gw/dingo.

A critical component of DINGO is a new iterative algorithm—GNPE—to partially off-load the modeling of time translations from the neural network. Although convergence of GNPE may take 20 seconds, initial samples with slightly reduced accuracy can, however, be produced in just a few seconds by taking fewer iterations.

Going forward, the next steps are to extend the prior to include longer-duration binary neutron star signals [11] (for which rapid results are especially important to identify electromagnetic counterparts) and to extend to more physically-realistic waveform models, which include higher multipole modes and more accurate spin-precession effects [170]. Long and complex waveforms are much more expensive for standard algorithms, so the relative improvement in performance should be even more significant. If successful, this would also enable the routine use of the most physically-realistic waveforms, resulting in consistently reduced systematic errors. These extensions will likely require somewhat larger networks and improved data representation or compression.[2]

Another natural extension would be to study signals without making the common stationary-Gaussian idealization for the detector noise during the training stage. For DINGO, performing inference with realistic noise is simply a matter of training with simulated signals injected into real noise realizations

[2]Initial estimates based on a singular value decomposition [56] indicate that to accurately represent SEOBNRv4PHM BBH waveforms [170], the initial layers of our embedding network should be widened by a factor of roughly four. Assuming the same number of iterations and fixed hardware, the total training time would increase by about 35%. For binary neutron stars, adopting a frequency-dependent resolution [225, 57] would limit the expansion of the number of frequency bins to a factor of three.

taken from detectors. Using real noise should lead to improved accuracy that is not possible using standard likelihood-based methods, and would serve as an excellent demonstration of the advantages of NPE. For real-time analysis, it will also be necessary to develop approaches to progressively retrain networks to keep pace with changing data distributions during an observing run, e.g., as detector sensitivity is improved. All of these enhancements, particularly the treatment of nonstationary noise, will be critical for extensions to future observatories such as LISA.

Deep-learning tools are now ready to analyze the vast majority of LIGO/Virgo events. In the past, the primary challenge has been in obtaining sufficiently accurate results, but with DINGO, we have now achieved this in a realistic context. Through planned future extensions, we expect that DINGO could become one of the leading approaches to gravitational-wave inference.

Acknowledgments

We thank S. Ossokine, M. Pürrer, C. Simpson and P. Züge for helpful discussions. This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration and the Virgo Collaboration. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation (NSF) as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation. M.D. thanks the Hector Fellow Academy for support. J.H.M. and B.S. are members of the MLCoE, EXC number 2064/1 – Project number 390727645. We use `PyTorch` [179] and `nflows` [94] for the implementation of our neural networks. The plots are generated with `matplotlib` [127], `ChainConsumer` [122] and `ligo.skymap` [202].

Chapter 3

Group Equivariant Neural Posterior Estimation

Simulation-based inference with conditional neural density estimators is a powerful approach to solving inverse problems in science. However, these methods typically treat the underlying forward model as a black box, with no way to exploit geometric properties such as equivariances. Equivariances are common in scientific models, however integrating them directly into expressive inference networks (such as normalizing flows) is not straightforward. We here describe an alternative method to incorporate equivariances under joint transformations of parameters and data. Our method—called group equivariant neural posterior estimation (GNPE)—is based on self-consistently standardizing the "pose" of the data while estimating the posterior over parameters. It is architecture-independent, and applies both to exact and approximate equivariances. As a real-world application, we use GNPE for amortized inference of astrophysical binary black hole systems from gravitational-wave observations. We show that GNPE achieves state-of-the-art accuracy while reducing inference times by three orders of magnitude.

Declaration

This chapter is based on the following published manuscript.

> *Group equivariant neural posterior estimation*
> Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Deistler, Bernhard Schölkopf, Jakob H. Macke
> Proceedings of the Tenth International Conference on Learning Representations (ICLR 2022) – Published 28 January 2022

Text and figures are adapted from the corresponding arXiv version `arXiv:2111.13139v2` with minor updates to layout and references.

Author contributions (CRediT)

Maximilian Dax: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Visualization
Stephen R. Green: Conceptualization, Methodology, Validation, Formal Analysis, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision
Jonathan Gair: Conceptualization, Methodology, Formal Analysis, Writing - Review & Editing
Michael Deistler: Software, Writing - Review & Editing, Visualization
Bernhard Schölkopf: Conceptualization, Writing - Review & Editing, Supervision
Jakob H. Macke: Conceptualization, Writing - Original Draft, Writing - Review & Editing, Supervision

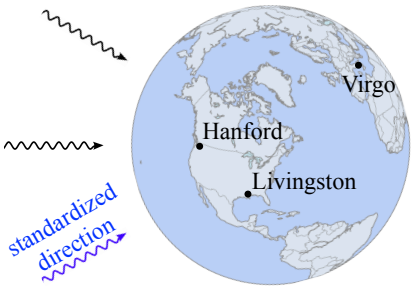


Fig. 3.1 By standardizing the source sky position, a GW signal can be made to arrive at the same time in all three LIGO/Virgo detectors. However, since this also changes the projection of the signal onto the detectors, it defines only an *approximate* equivariance. Nevertheless, our proposed GNPE algorithm simplifies inference by simultaneously inferring *and* standardizing the incident direction.

3.1 Introduction

Bayesian inference provides a means of characterizing a system by comparing models against data. Given a forward model or likelihood $p(x|\theta)$ for data x described by parameters θ, and a prior $p(\theta)$, the Bayesian posterior is proportional to the product, $p(\theta|x) \propto p(x|\theta)p(\theta)$. Sampling techniques such as Markov Chain Monte Carlo (MCMC) can be used to build up a posterior distribution provided the likelihood and prior can be evaluated.

For models with intractable or expensive likelihoods (as often arise in scientific applications) simulation-based (or likelihood-free) inference methods offer a powerful alternative [78]. In particular, neural posterior estimation (NPE) [175] uses expressive conditional density estimators such as normalizing flows [193, 178] to build surrogates for the posterior. These are trained using model simulations $x \sim p(x|\theta)$, and allow for rapid sampling for any $x \sim p(x)$, thereby amortizing training costs across future observations. NPE and other density-estimation methods for simulation-based inference [114, 177, 119] have been reported to be more simulation-efficient [154] than classical likelihood-free methods such as Approximate Bayesian Computation [204].

Training an inference network for any $x \sim p(x)$ can nevertheless present challenges due to the large number of training samples and powerful networks required. The present study is motivated by the problem of gravitational-wave (GW) data analysis. Here the task is to infer properties of astrophysical black-hole mergers based on GW signals observed at the LIGO and Virgo observatories on Earth. Due to the complexity of signal models, it has previously not been possible to train networks to estimate posteriors to the same accuracy as conventional likelihood-based methods [224, 41]. The GW posterior, however, is equivariant[1] under an overall change in the time of arrival of the data. It is also *approximately* equivariant under a joint change in the sky position and (by triangulation) individual shifts in the arrival times in each detector (Fig. 3.1). If we could constrain these parameters *a priori*, we could therefore apply time shifts to align the detector data and simplify the inference task for the remaining parameters.

More generally, we consider forward models with known equivariances under group transformations applied jointly to data and parameters. Our aim is to exploit this knowledge to *standardize the pose of the data*[2] and simplify analysis. The obvious roadblock is that the pose is contained in the set of parameters θ and is therefore unknown prior to inference. Here we describe *group equivariant* neural posterior estimation (GNPE), a method to self-consistently infer parameters *and* standardize the pose.

[1] In physics, the term "covariant" is frequently used instead of "equivariant".
[2] We adopt the language from computer vision by Jaderberg et al. [129].

The basic approach is to introduce a *proxy* for the pose—a blurred version—on which one conditions the posterior. The pose of the data is then transformed based on the proxy, placing it in a band about the standard value, and resulting in an easier inference task. Finally, the joint posterior over θ and the pose proxy can be sampled at inference time using Gibbs sampling.

The standard method to incorporating equivariances is to integrate them directly into network architectures, e.g., to use convolutional networks for translational equivariances. Although these approaches can be highly effective, they impose design constraints on network architectures. For GWs, for example, we use specialized embedding networks to extract signal waveforms from frequency-domain data, as well as expressive normalizing flows to estimate the posterior—neither of which is straightforward to make explicitly equivariant. We also have complex equivariance connections between subsets of parameters and data, including approximate equivariances. The GNPE algorithm is extremely general: it is architecture-independent, it applies whether equivariances are exact or approximate, and it allows for arbitrary equivariance relations between parameters and data.

We discuss related work in Sec. 3.2 and describe the GNPE algorithm in Sec. 3.3. In Sec. 3.4 we apply GNPE to a toy example with exact translational equivariance, showing comparable simulation efficiency to NPE combined with a convolutional network. In Sec. 3.5 we show that standard NPE does not achieve adequate accuracy for GW parameter inference, even with an essentially unlimited number of simulations. In contrast, GNPE achieves highly accurate posteriors at a computational cost three orders of magnitude lower than bespoke MCMC approaches [224]. The present paper describes the GNPE method which we developed for GW analysis [82], and extends it to general equivariance transformations which makes it applicable to a wide range of problems. A detailed description of GW results is presented in Dax et al. [82].

3.2 Related work

The most common way of integrating equivariances into machine learning algorithms is to use equivariant network architectures [142, 68]. This can be in conflict with design considerations such as data representation and flexibility of the architecture, and imposes constraints such as locality. GNPE achieves complete separation of equivariances from these considerations, requiring only the ability to efficiently transform the pose.

Normalizing flows are particularly well suited to NPE, and there has been significant progress in constructing equivariant flows [52]. However, these studies consider joint transformations of parameters of the base space and sample space—*not* joint transformation of data and parameters for *conditional* flows, as we consider here.

GNPE enables end-to-end equivariances from data to parameters. Consider, by contrast, a conditional normalizing flow with a convolutional embedding network: the equivariance persists through the embedding network but is broken by the flow. Although this may improve learning, it does not enforce an end-to-end equivariance. This contrasts with an *invariance*, for which the above would be sufficient. Finally, GNPE can also be applied if the equivariance is only *approximate*.

Several other approaches integrate domain knowledge of the forward model [43, 54] by considering a "gray-box" setting. GNPE allows us to incorporate high-level domain knowledge about approximate equivariances of forward models without requiring access to its implementation or internal states of the simulator. Rather, it can be applied to "black-box" code.

An alternative approach to incorporate geometrical knowledge into classical likelihood-free inference algorithms (e.g., Approximate Bayesian Computation, see [204]) is by constructing [99] or learning [130, 64] equivariant summary statistics $s(x)$, which are used as input to the inference algorithm instead of the raw data x. However, designing equivariant summary statistics (rather than invariant ones) can be challenging, and furthermore inference will be biased if the equivariance only holds approximately.

Past studies using machine-learning techniques for amortized GW parameter inference [103, 67, 110, 87] all consider simplified problems (e.g., only a subset of parameters, a simplified posterior, or a limited treatment of detector noise). In contrast, the GNPE-based study in Dax et al. [82] is the only one to treat the full amortized parameter inference problem with accuracy matching standard methods.

3.3 Methods

3.3.1 Neural posterior estimation

NPE [175, 112] is a simulation-based inference method that directly targets the posterior. Given a dataset of prior parameter samples $\theta^{(i)} \sim p(\theta)$ and corresponding model simulations $x^{(i)} \sim p(x|\theta^{(i)})$, it trains a neural density estimator $q(\theta|x)$ to estimate $p(\theta|x)$. This is achieved by minimizing the loss

$$\mathcal{L}_{\text{NPE}} = \mathbb{E}_{p(\theta)}\mathbb{E}_{p(x|\theta)}\left[-\log q(\theta|x)\right] \tag{3.1}$$

across the dataset of $(\theta^{(i)}, x^{(i)})$ pairs. This maximum likelihood objective leads to recovery of $p(\theta|x)$ if $q(\theta|x)$ is sufficiently flexible. Normalizing flows [193, 93] are a particularly expressive class of conditional density estimators commonly used for NPE.

NPE amortizes inference: once $q(\theta|x)$ is trained, inference is very fast for any observed data x_{o}, so training costs are shared across observations. The approach is also extremely flexible, as it treats the forward model as a black box, relying only on prior samples and model simulations. In many situations, however, these data have known structure that one wants to exploit to improve learning.

3.3.2 Equivariances under transformation groups

In this work we describe a generic method to incorporate equivariances under joint transformations of θ and x into NPE. A typical example arises when inferring the position of an object from image data. In this case, if we spatially translate an image x by some offset $\vec{d}$—effected by *relabeling the pixels*—then the inferred position θ should also transform by $\vec{d}$—by *addition* to the position coordinates θ. Translations are composable and invertible, and there exists a trivial identity translation, so the set of translations has a natural group structure. Our method works for any continuous transformation group, including rotations, dilations, etc., and in this section we keep the discussion general.

For a transformation group G, we denote the action of $g \in G$ on parameters and data as

$$\theta \to g\theta, \tag{3.2}$$

$$x \to T_g x. \tag{3.3}$$

Here, T_g refers to the group representation under which the data transform (e.g., for image translations, the pixel relabeling). We adopt the natural convention that G is defined by its action on θ, so we do not introduce an explicit representation on parameters. The posterior distribution $p(\theta|x)$ is said to be *equivariant* under G if, when the parameter and data spaces are jointly G-transformed, the posterior is unchanged, i.e.,

$$p(\theta|x) = p(g\theta|T_g x)|\det J_g|, \qquad \forall g \in G. \tag{3.4}$$

The right-hand side comes from the change-of-variables rule. For translations the Jacobian J_g has unit determinant, but we include it for generality. For NPE, we are concerned with equivariant posteriors, however it is often more natural to think of equivariant forward models (or likelihoods). An equivariant likelihood and an *invariant* prior together yield an equivariant posterior (App. B.1.1).

Our goal is to use equivariances to simplify the data—to G-transform x such that θ is taken to a fiducial value. For the image example, this could mean translating the object of interest to the center. In general, θ can also include parameters unchanged under G (e.g., the color of the object), so we denote the corresponding standardized parameters by θ_0. These are related to θ by a group transformation denoted g^θ, such that $g^\theta\theta_0 = \theta$. We refer to g^θ as the "pose" of θ, and standardizing the pose means to take it to the group identity element $e \in G$. Applying $T_{(g^\theta)^{-1}}$ to the data space effectively reduces its dimensionality, making it easier to interpret for a neural network.

Although the preceding discussion applies to equivariances that hold exactly, our method in fact generalizes to *approximate* equivariances. We say that a posterior is approximately equivariant under G if (3.4) does *not* hold, but standardizing the pose nevertheless reduces the effective dimensionality of the dataset. An approximately equivariant posterior can arise if an exact equivariance of the forward model is broken by a non-invariant prior, or if the forward model is itself non-equivariant.

3.3.3 Group equivariant neural posterior estimation

We are now presented with the basic problem that we resolve in this work: how to simultaneously infer the pose of a signal and use that inferred pose to standardize (or align) the data so as to simplify the analysis. This is a circular problem because one cannot standardize the pose (contained in model parameters θ) without first inferring the pose from the data; and conversely one cannot easily infer the pose without first simplifying the data by standardizing the pose.

Our resolution is to start with a rough estimate of the pose, and iteratively (1) transform the data based on a pose estimate, and (2) estimate the pose based on the transformed data. To do so, we expand the parameter space to include *approximate* pose parameters $\hat{g} \in G$. These "pose proxies" are defined using a kernel to blur the true pose, i.e., $\hat{g} = g^\theta\epsilon$ for $\epsilon \sim \kappa(\epsilon)$; then $p(\hat{g}|\theta) = \kappa\left((g^\theta)^{-1}\hat{g}\right)$. The kernel $\kappa(\epsilon)$ is a distribution over group elements, which should be chosen to be concentrated around

e; we furthermore choose it to be symmetric. Natural choices for $\kappa(\epsilon)$ include Gaussian and uniform distributions. For translations, the pose proxy is simply the true position with additive noise.

Consider now the posterior distribution $p(\theta, \hat{g}|x)$ over the expanded parameter space. Our iterative algorithm comes from Gibbs sampling this distribution [194] (Fig. 3.2), i.e., alternately sampling θ and $\hat{g}$, conditional on the other parameter and x,

$$\theta \sim p(\theta|x, \hat{g}), \tag{3.5}$$

$$\hat{g} \sim p(\hat{g}|x, \theta). \tag{3.6}$$

The second step just amounts to blurring the pose, since $p(\hat{g}|x, \theta) = p(\hat{g}|\theta) = \kappa\left((g^\theta)^{-1}\hat{g}\right)$. The key first step uses a neural density estimator q that is trained taking advantage of a standardized pose.

For an **equivariant** posterior, the distribution (3.5) can be rewritten as (App. B.1.2)

$$p(\theta|x, \hat{g}) = p\left(\hat{g}^{-1}\theta|T_{\hat{g}^{-1}}x, \hat{g}^{-1}\hat{g}\right)\left|\det J_{\hat{g}}^{-1}\right| \equiv p(\theta'|x')\left|\det J_{\hat{g}}^{-1}\right|. \tag{3.7}$$

For the last equality we defined $\theta' \equiv \hat{g}^{-1}\theta$ and $x' \equiv T_{\hat{g}^{-1}}x$, and we dropped the constant argument $\hat{g}^{-1}\hat{g} = e$. This expresses $p(\theta|x, \hat{g})$ in terms of the $\hat{g}$-standardized data x'—which is much easier to estimate. We train a neural density estimator $q(\theta'|x')$ to approximate this, by minimizing the loss,

$$\mathcal{L}_{\text{GNPE}} = \mathbb{E}_{p(\theta)}\mathbb{E}_{p(x|\theta)}\mathbb{E}_{p(\hat{g}|\theta)}\left[-\log q\left(\hat{g}^{-1}\theta|T_{\hat{g}^{-1}}x\right)\right]. \tag{3.8}$$

With a trained $q(\theta'|x')$,

$$\theta \sim p(\theta|x, \hat{g}) \qquad \Longleftrightarrow \qquad \theta = \hat{g}\theta', \quad \theta' \sim q(\theta'|T_{\hat{g}^{-1}}x). \tag{3.9}$$

The estimated posterior is equivariant by construction (App. B.1.3).

For an **approximately-equivariant** posterior, (3.5) cannot be transformed to be independent of $\hat{g}$. We are nevertheless able to use the conditioning on $\hat{g}$ to approximately align x. We therefore train a neural density estimator $q(\theta|x', \hat{g})$, by minimizing the loss

$$\mathcal{L}_{\text{GNPE}} = \mathbb{E}_{p(\theta)}\mathbb{E}_{p(x|\theta)}\mathbb{E}_{p(\hat{g}|\theta)}\left[-\log q\left(\theta|T_{\hat{g}^{-1}}x, \hat{g}\right)\right]. \tag{3.10}$$

In general, one may have a combination of exact and approximate equivariances (see, e.g., Sec. 3.5).

3.3.4 Gibbs convergence

The Gibbs-sampling procedure constructs a Markov chain with equilibrium distribution $p(\theta, \hat{g}|x)$. For convergence, the chain must be transient, aperiodic and irreducible [194, 106]. For sensible choices of $\kappa(\epsilon)$ the chain is transient and aperiodic by construction. Further, irreducibility means that the entire posterior can be reached starting from any point, which should be possible even for disconnected posteriors provided the kernel is sufficiently broad. In general, burn-in truncation and thinning of

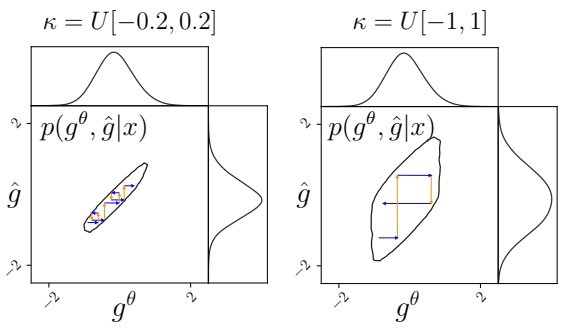


Fig. 3.2 We infer $p(g^\theta, \hat{g}|x)$ with Gibbs sampling by alternately sampling (1) $g^\theta \sim p(g^\theta|x, \hat{g})$ (blue) and (2) $\hat{g} \sim p(\hat{g}|x, g^\theta)$ (orange). For (1) we use a density estimator $q(g^\theta|T_{\hat{g}^{-1}}(x), \hat{g})$, for (2) the definition $\hat{g} = g^\theta + \epsilon$, $\epsilon \sim \kappa(\epsilon)$. Pose standardization with $T_{\hat{g}^{-1}}$ is only allowed due to conditioning on $\hat{g}$. Increasing the width of κ accelerates convergence (due to larger steps in parameter space), at the cost of $\hat{g}$ being a worse approximation for g^θ, and therefore pose alignment being less effective.

the chain is required to ensure (approximately) independent samples. By marginalizing over $\hat{g}$ (i.e., ignoring it) we obtain samples from the posterior $p(\theta|x)$, as desired.[3]

Convergence of the chain also informs our choice of $\kappa(\epsilon)$. For wide $\kappa(\epsilon)$, only a few Gibbs iterations are needed to traverse the joint posterior $p(\theta, \hat{g}|x)$, whereas for narrow $\kappa(\epsilon)$ many steps are required (Fig. 3.2). In the limiting case of $\kappa(\epsilon)$ a delta distribution (i.e., no blurring) the chain does not deviate from its initial position and therefore fails to converge.[4] Conversely, a narrower $\kappa(\epsilon)$ better constrains the pose, which improves the accuracy of the density estimator. The width of κ should be chosen based on this practical trade-off between speed and accuracy; the standard deviation of a typical pose posterior is usually a good starting point.

In practice, we obtain N samples in parallel by constructing an ensemble of N Markov chains. We initialize these using samples from a second neural density estimator $q_{\text{init}}(g^\theta|x)$, trained using standard NPE. Gibbs sampling yields a sequence of sample sets $\{\theta_j^{(i)}\}_{i=1}^N$, $j = 0, 1, 2, \ldots$, each of which represents a distribution $Q_j(\theta|x)$ over parameters. Assuming a perfectly trained network, one iteration applied to sample set j yields an updated distribution,

$$Q_{j+1}(\theta|x) = p(\theta|x) \left[\frac{Q_j(\cdot|x) \,\bar{*}\, \kappa}{p(\cdot|x) \,\bar{*}\, \kappa} * \kappa \right] (g^\theta). \tag{3.11}$$

The "$*$" symbol denotes group convolution and "$\bar{*}$" the combination of marginalization and group convolution (see App. B.1.4 for details). The true posterior $p(\theta|x)$ is clearly a fixed point of this sequence, with the number of iterations to convergence determined by κ and the accuracy of the initialization network q_{init}.

3.4 Toy example: damped harmonic oscillator

We now apply GNPE to invert a simple model of a damped harmonic oscillator. The forward model gives the time-dependent position x of the oscillator, conditional on its real frequency ω_0, damping ratio β, and time of excitation τ. The time series x is therefore a damped sinusoid starting at τ

[3] In practice, this results only in *approximate* samples due to the asymptotic behaviour of Gibbs sampling, and a potential mismatch between the trained q and the targeted true posterior.

[4] This also explains why introducing the pose proxy is needed at all: GNPE would not work without it!

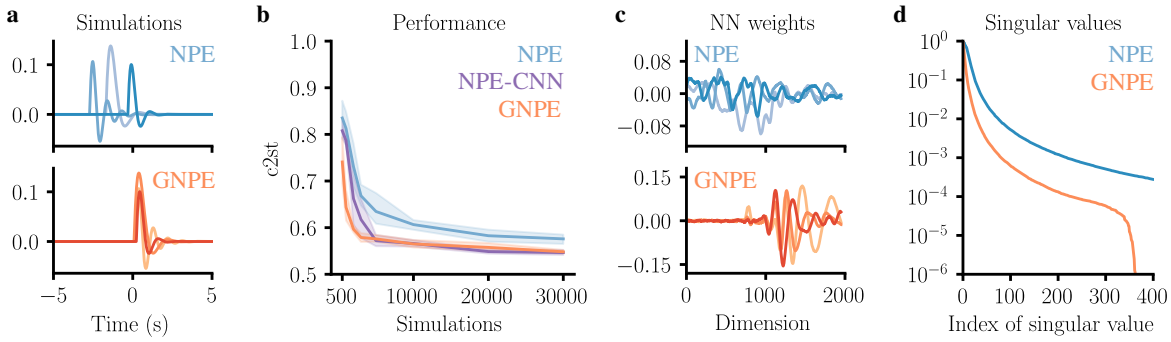


Fig. 3.3 Comparison of standard NPE (blue) and GNPE (orange) for the damped harmonic oscillator. **a)** Three sample inputs to the neural density estimators showing GNPE data are pose-standardized. **b)** c2st score performance (best: 0.5, worst: 1.0): GNPE significantly outperforms equivariance-agnostic NPE, and is on par with NPE with a convolutional embedding network (purple). **c)** Example filters from the first linear layer of fully trained networks. GNPE filters more clearly capture oscillatory modes. **d)** Singular values of the training data. Inputs to GNPE x' have smaller effective dimension than raw inputs x to NPE.

(and zero before). Noise is introduced via a normally-distributed perturbation of the parameters $\theta = (\omega_0, \beta, \tau)$, resulting in a Gaussian posterior $p(\theta|x)$ (further details in App. B.3.1). The model is constructed such that the posterior is equivariant under translations in τ,

$$p(\omega_0, \beta, \tau + \Delta\tau | T_{\Delta\tau} x) = p(\omega_0, \beta, \tau | x), \tag{3.12}$$

so we take τ to be the pose. The equivariance of this model is exact, but it could easily be made approximate by, e.g., introducing τ-dependent noise. The prior $p(\tau)$ extends from -5 s to 0 s, so for NPE, the density estimator must learn to interpret data from oscillators excited throughout this range.

For GNPE, we shift the data to align the pose near $\tau = 0$ using a Gaussian kernel $\kappa = \mathcal{N}[0, (0.1\text{ s})^2]$ (Fig. 3.3a). We then train a neural density estimator $q(\theta'|x')$ to approximate $p(\theta'|x')$, where $\theta' \equiv (\omega_0, \beta, -\epsilon)$ and $x' \equiv T_{-(\tau+\epsilon)} x$ are pose-standardized. We take $q(\theta'|x')$ to be diagonal Gaussian, matching the known form of the posterior. For each experiment, we train until the validation loss stops decreasing. We also train a neural density estimator $q_{\text{init}}(\tau|x)$ with standard NPE on the same dataset to generate initial GNPE seeds. To generate N posterior samples we proceed as follows:

1. Sample $\tau^{(i)} \sim q_{\text{init}}(\tau|x)$, $i = 1, \ldots, N$;
2. Sample $\epsilon^{(i)} \sim \kappa(\epsilon)$, set $\hat{\tau}^{(i)} = \tau^{(i)} + \epsilon^{(i)}$, and time-translate the data, $x'^{(i)} = T_{-\hat{\tau}^{(i)}} x$;
3. Sample $\theta'^{(i)} \sim q(\theta'|x'^{(i)})$, and undo the time translation $\hat{\tau}^{(i)}$ to obtain $\theta^{(i)}$.

We repeat steps 2 and 3 until the distribution over τ converges. For this toy example and our choice of κ only one iteration is required. For further details of the implementation see App. B.3.2.

We evaluate GNPE on five simulations by comparing inferred samples to ground-truth posteriors using the c2st score [102, 151]. This corresponds to the test accuracy of a classifier trained to discriminate samples from the target and inferred distributions, and ranges from 0.5 (best) to 1.0 (worst). As

baselines we evaluate standard NPE (i) with a network architecture identical to GNPE and (ii) with a convolutional embedding network (NPE-CNN; see App. B.3.2). Both approaches that leverage the equivariance, GNPE (by standardizing the pose) and NPE-CNN (by using a translation-equivariant embedding network), perform similarly well and far outperform standard NPE (Fig. 3.3b). This underscores the importance of equivariance awareness. The fact that the NPE network is trained to interpret signals from oscillators excited at arbitrary τ, whereas GNPE focuses on signals starting around $\tau = 0$ (up to a small ϵ perturbation) is also reflected in in simplified filters in the first layer of the network (Fig. 3.3c) and a reduced effective dimension of the input data to GNPE (Fig. 3.3d).

3.5 Gravitational-wave parameter inference

Gravitational waves—propagating ripples of space and time—were first detected in 2015, from the inspiral, merger, and ringdown of a pair of black holes [4]. Since that time, the two LIGO detectors (Hanford and Livingston) [2] as well as the Virgo detector [32] have observed signals from over 50 coalescences of compact binaries involving either black holes or neutron stars [15, 23, 31]. Key scientific results from these observations have included measurements of the properties of stellar-origin black holes that have provided new insights into their origin and evolution [21]; an independent measurement of the local expansion rate of the Universe, the Hubble constant [7]; and new constraints on the properties of gravity and matter under extreme conditions [12, 24].

Quasicircular binary black hole (BBH) systems are characterized by 15 parameters θ, including the component masses and spins, as well as the space-time position and orientation of the system (Tab. B.1). Given these parameters, Einstein's theory of general relativity predicts the motion and emitted gravitational radiation of the binary. The GWs propagate across billions of light-years to Earth, where they produce a time-series signal $h_I(\theta)$ in each of the LIGO/Virgo interferometers $I = \mathrm{H}, \mathrm{L}, \mathrm{V}$. The signals on Earth are very weak and embedded in detector noise n_I. In part to have a tractable likelihood, the noise is approximated as additive and stationary Gaussian. The signal and noise models give rise to a likelihood $p(x|\theta)$ for observed data $x = \{h_I(\theta) + n_I\}_{I=\mathrm{H,L,V}}$.

Once the LIGO/Virgo detection pipelines are triggered, classical stochastic samplers are typically employed to determine the parameters of the progenitor system using Bayesian inference [224, 41]. However, these methods require millions of likelihood evaluations (and hence expensive waveform simulations) for each event analyzed. Even using fast waveform models, it can take $O(\mathrm{day})$ to analyze a single BBH. Faster inference methods are therefore highly desirable to cope with growing event rates, more realistic (and expensive) waveform models, and to make rapid localization predictions for possible multimessenger counterparts. Rapid amortized methods such as NPE have the potential to transform GW data analysis. However, due to the complexity and high dimensionality[5] of GW data, it has been a challenge [103, 67, 110, 87] to obtain results of comparable accuracy and completeness to classical samplers. We now show how GNPE can be used to exploit equivariances to greatly simplify the inference problem and achieve for the first time performance indistinguishable from "ground truth" stochastic samplers—at drastically reduced inference times.

[5] In our work, we analyze 8 s data segments between 20 Hz and 1024 Hz. Including also noise information, this results in 24,099 input dimensions per detector.

3.5.1 Equivariances of sky position and coalescence time

We consider the analysis of BBH systems. Included among the parameters θ are the time of coalescence t_{c} (as measured at geocenter) and the sky position (right ascension α, declination δ). Since GWs propagate at the speed of light, these are related to the times of arrival t_I of the signal in each of the interferometers.[6] Our priors (based on the precision of detection pipelines) constrain t_I to a range of ≈ 20 ms, which is much wider than typical posteriors. Standard NPE inference networks must therefore be trained on simulations with substantial time shifts.

The detector coalescence times t_I—equivalently, (t_c, α, δ)—can alternatively be interpreted as the pose of the data, and standardized using GNPE. The group G transforming the pose factorizes into a direct product of absolute and relative time shifts,

$$G = G_{\mathrm{abs}} \times G_{\mathrm{rel}}. \tag{3.13}$$

Group elements $g_{\mathrm{abs}} \in G_{\mathrm{abs}}$ act by uniform translation of all t_I, whereas $g_{\mathrm{rel}} \in G_{\mathrm{rel}}$ act by individual translation of t_{L} and t_{V}. We work with data in frequency domain, where time translations act by multiplication, $T_g x_I = e^{-2\pi i f \Delta t_I} x_I$. Absolute time shifts correspond to a shift in t_c, and are an *exact* equivariance of the posterior, $p(g_{\mathrm{abs}}\theta | T_{g_{\mathrm{abs}}} x) = p(\theta|x)$. Relative time shifts correspond to a change in (α, δ) (as well as t_c). This is only an *approximate* equivariance, since a change in sky position changes the projection of the incident signal onto the detector arms, leading to a subdominant change to the signal morphology in each detector.

3.5.2 Application of GNPE

We use GNPE to standardize the pose within a band around $t_I = 0$. We consider two modes defined by different uniform blurring kernels. The "accurate" mode uses a narrow kernel $\kappa_{\mathrm{narrow}} = U[-1\ \mathrm{ms}, 1\ \mathrm{ms}]^{n_I}$, whereas the "fast" mode uses a wide kernel $\kappa_{\mathrm{wide}} = U[-3\ \mathrm{ms}, 3\ \mathrm{ms}]^{n_I}$. The latter is intended to converge in just one GNPE iteration, at the cost of having to interpret a wider range of data.

We define the blurred pose proxy $\hat{g}_I \equiv t_I + \epsilon_I$, where $\epsilon_I \sim \kappa(\epsilon_I)$. We then train a conditional density estimator $q(\theta'|x', \hat{g}_{\mathrm{rel}})$, where $\theta' = \hat{g}_{\mathrm{abs}}^{-1}\theta$ and $x' = T_{\hat{g}^{-1}} x$. That is, we condition q on the relative time shift (since this is an approximate equivariance) and we translate parameters by the absolute time shift (since this is an exact equivariance). We always transform the data by the full time shift. We train a density estimator $q_{\mathrm{init}}(\{t_I\}_{I=\mathrm{H,L,V}}|x)$ using standard NPE to infer initial pose estimates.

The difficulty of the inference problem (high data dimensionality, significant noise levels, complex forward model) combined with high accuracy requirements to be scientifically useful requires careful design decisions. In particular, we initialize the first layer of the embedding network with principal components of clean waveforms to provide an inductive bias to extract useful information. We further use an expressive neural-spline normalizing flow [93] to model the complicated GW posterior structure. See App. B.4.2 for details of network architecture and training.

[6] We consider observations made in either $n_I = 2$ or 3 interferometers.

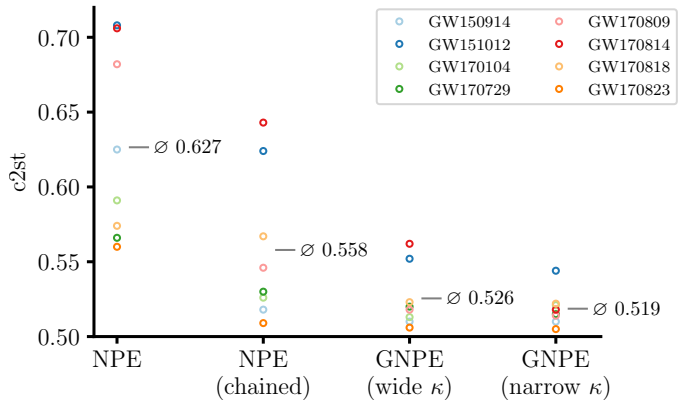


Fig. 3.4 Comparison of estimated posteriors against LALINFERENCE MCMC for eight GW events, as quantified by c2st (best: 0.50, worst: 1.00). GNPE with a wide kernel outperforms both NPE baselines, while being only marginally slower (1 iteration $\sim$ 2 s). With a narrow kernel and 30 iterations ($\sim$ 60 s), we achieve c2st < 0.55 across all events. $\varnothing$ indicates the average across all eight events. For an alternative metric (MSE) see Fig. B.3.

3.5.3 Results

We evaluate performance on all eight BBH events from the first Gravitational-Wave Transient Catalog [15] consistent with our prior (component masses greater than 10 $M_\odot$). We generate reference posteriors with the LIGO/Virgo MCMC code LALINFERENCE [224]. We quantify the deviation between NPE samples and the reference samples using c2st.

We compare performance against two baselines, standard NPE and a modified approach that partially standardizes the pose ("chained NPE"). For the latter, we use the chain rule to decompose the posterior,

$$p(\theta|x) = p(\phi, \lambda|x) = p(\phi|x, \lambda) \cdot p(\lambda|x), \tag{3.14}$$

where $\lambda = (t_c, \alpha, \delta)$ are the pose parameters and $\phi \subset \theta$ collects the remaining 12 parameters. We use standard NPE to train a flow $q(\lambda|x)$ to estimate $p(\lambda|x)$, and a flow $q(\phi|x', \lambda)$ to estimate $p(\phi|x, \lambda)$. The latter flow is conditioned on λ, which we use to standardize the pose of x. In contrast to GNPE, this approach is sensitive to the initial pose estimate $q(\lambda|x)$, which limits accuracy (Figs. 3.4 and B.8). We note that all hyperparameters of the flow and training protocol (see App. B.4.2) were extensively optimized on NPE, and then transferred to GNPE without modification, resulting in conservative estimates of the performance advantage of GNPE. Fast-mode GNPE converges in one iteration, whereas accurate-mode requires 30 (convergence is assessed by the JS divergence between the inferred pose posteriors from two successive iterations).

Standard NPE performs well on some GW events but lacks the required accuracy for most of them, with c2st scores up to 0.71 (Fig. 3.4). Chained NPE performs better across the dataset, but performs poorly on events such as GW170814, for which the initial pose estimate is inaccurate. Indeed, we find that inaccuracies of that baseline can be almost entirely attributed to the initial pose estimate (Fig. B.7). Fast-mode GNPE with only one iteration is already more robust to this effect due to the blurring operation of the pose proxy (Fig. B.8). Both GNPE models significantly outperform the baselines, with accurate-mode obtaining c2st scores < 0.55 across all eight events. We emphasize that the c2st score is sensitive to any deviation between reference samples and samples from the inferred

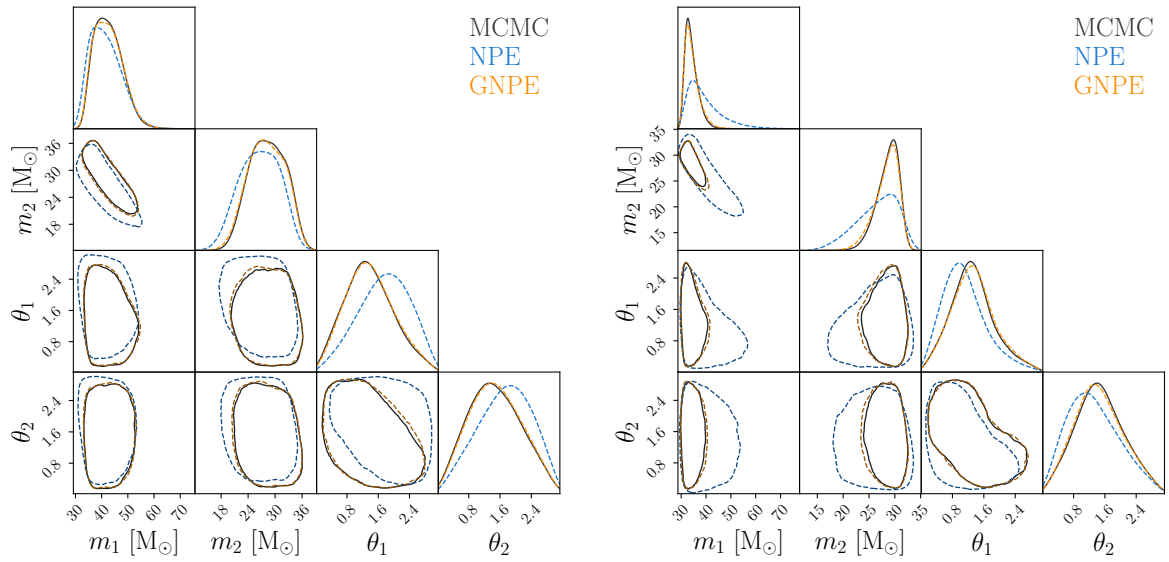


Fig. 3.5 Corner plots for the GW events GW170809 (left) and GW170814 (right), plotting 1D marginals on the diagonal and 90% credible regions for the 2D correlations. We display the two black hole masses m_1 and m_2 and two spin parameters θ_1 and θ_2 (note that the full posterior is 15-dimensional). NPE does not accurately reproduce the MCMC posterior, while accurate-mode GNPE matches the MCMC results well. For a plot with all baselines see Fig. B.5.

posterior. On a recent benchmark by Lueckmann et al. [154] on examples with much lower parameter *and* data dimensions, even state-of-the-art SBI algorithms rarely reached c2st scores below 0.6. The fact that GNPE achieves scores around 0.52—i.e., posteriors which are nearly indistinguishable from the reference—on this challenging, high-dimensional, real-world example underscores the power of exploiting equivariances with GNPE.

Finally, we visualize posteriors for two events, GW170809 and GW170814, in Fig. 3.5. The quantitative agreement between GNPE and MCMC (Fig. 3.4) is visible from the overlapping marginals for all parameters displayed. NPE, by contrast, deviates significantly from MCMC in terms of shape and position. Note that we show a failure case of NPE here; for other events, such as GW170823, deviations of NPE from the reference posterior are less clearly visible.

3.6 Conclusions

We described GNPE, an approach to incorporate exact—and even *approximate*—equivariances under joint transformations of data and parameters into simulation-based inference. GNPE can be applied to black-box scientific forward models and any inference network architecture. It requires similar training times compared to NPE, while the added complexity at inference time depends on the number of GNPE iterations (adjustable, but typically $O(10)$). We show with two examples that exploiting equivariances with GNPE can yield large gains in simulation efficiency and accuracy.

For the motivating problem of GW parameter estimation, GNPE achieves for the first time rapid amortized inference with results virtually indistinguishable from MCMC [82]. This is an extremely challenging "real-world" scientific problem, with high-dimensional input data, complex signals, and significant noise levels. It combines exact and approximate equivariances, and there is no clear path to success without their inclusion along with GW-specialized architectures and expressive density estimators.

Acknowledgments

We thank A. Buonanno, T. Gebhard, J.M. Lückmann, S. Ossokine, M. Pürrer and C. Simpson for helpful discussions. We thank the anonymous reviewer for coming up with the illustration of GNPE in App. B.2. This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration and the Virgo Collaboration. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation (NSF) as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation. M. Dax thanks the Hector Fellow Academy for support. M. Deistler thanks the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for support. B.S. and J.H.M. are members of the MLCoE, EXC number 2064/1 – Project number 390727645. This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A. We use `PyTorch` [179], `nflows` [94] and `sbi` [217] for the implementation of our neural networks. The plots are generated with `matplotlib` [127] and `ChainConsumer` [122].

Chapter 4

Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference

We combine amortized neural posterior estimation with importance sampling for fast and accurate gravitational-wave inference. We first generate a rapid proposal for the Bayesian posterior using neural networks, and then attach importance weights based on the underlying likelihood and prior. This provides (1) a corrected posterior free from network inaccuracies, (2) a performance diagnostic (the sample efficiency) for assessing the proposal and identifying failure cases, and (3) an unbiased estimate of the Bayesian evidence. By establishing this independent verification and correction mechanism we address some of the most frequent criticisms against deep learning for scientific inference. We carry out a large study analyzing 42 binary black hole mergers observed by LIGO and Virgo with the SEOBNRv4PHM and IMRPhenomXPHM waveform models. This shows a median sample efficiency of $\approx 10\%$ (two orders-of-magnitude better than standard samplers) as well as a ten-fold reduction in the statistical uncertainty in the log evidence. Given these advantages, we expect a significant impact on gravitational-wave inference, and for this approach to serve as a paradigm for harnessing deep learning methods in scientific applications.

Declaration

This chapter is based on the following published manuscript.

> *Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference*
> Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
> Phys. Rev. Lett. **130**, 171403 – Published 26 April 2023

Text and figures are adapted from the corresponding arXiv version `arXiv:2210.05686v2` with minor updates to layout and references.

Author contributions (CRediT)

Maximilian Dax: Conceptualization, Methodology, Software, Validation, Formal Analysis, Investigation, Writing - Original Draft, Writing - Review & Editing, Visualization
Stephen R. Green: Conceptualization, Methodology, Software, Validation, Formal Analysis, Writing - Original Draft, Writing - Review & Editing, Visualization, Supervision
Jonathan Gair: Methodology, Formal Analysis, Writing - Review & Editing
Michael Pürrer: Software, Writing - Review & Editing
Jonas Wildberger: Software, Writing - Review & Editing
Jakob H. Macke: Writing - Review & Editing, Supervision
Alessandra Buonanno: Writing - Review & Editing
Bernhard Schölkopf: Writing - Review & Editing, Supervision

4.1 Introduction

Bayesian inference is a key paradigm for scientific discovery. In the context of gravitational waves (GWs), it underlies analyses including individual-event parameter estimation [29], tests of gravity [25], neutron-star physics [12], populations [30], and cosmology [28]. Given a prior $p(\theta)$ and a model likelihood $p(d|\theta)$, the Bayesian posterior

$$p(\theta|d) = \frac{p(d|\theta)p(\theta)}{p(d)} \tag{4.1}$$

summarises, as a probability distribution, our knowledge of the model parameters θ after observing data d. When $p(d|\theta)$ is tractable (as in the case of GWs) likelihood-based samplers such as Markov chain Monte Carlo (MCMC) [158, 117] or nested sampling [206] are typically used to draw samples from the posterior. If it is possible to *sample* $d \sim p(d|\theta)$ (i.e., simulate data) one can alternatively use amortized simulation-based (or likelihood-free) inference methods [78]. These approaches are based on deep neural networks and can be several orders-of-magnitude faster at inference time. For GW inference, they have also been shown to achieve similar accuracy to MCMC [82]. In general, however, it is not clear how well such networks generalize to out-of-distribution data and they lack diagnostics to be confident in results [59]. These powerful approaches are therefore rarely used in applications where accuracy is important and likelihoods are tractable.

In this Letter, we achieve the best of both worlds by combining likelihood-free and likelihood-based methods for GW parameter estimation. We take samples from DINGO[1] [82]—a fast and accurate likelihood-free method using normalizing flows [193, 136, 93, 178]—and treat these as a proposal for importance sampling [220]. The combined method ("DINGO-IS") generates samples from the exact posterior and now provides an estimate of the Bayesian evidence $p(d)$. Moreover, the importance sampling efficiency arises as a powerful and objective performance metric, which flags potential failure cases. Importance sampling is fully parallelizable.

After describing the method more fully in the following section, we verify on two real events that DINGO-IS produces results consistent with standard inference codes [224, 41, 195, 210]. Our main result is an analysis of 42 events from the Second and Third Gravitational-Wave Transient Catalogs (GWTC-2 and GWTC-3) [23, 29], using two waveform models, IMRPhenomXPHM [185] and SEOBNRv4PHM [170]. Due to the long waveform simulation times, SEOBNRv4PHM inference would take several months per event with stochastic samplers. However DINGO-IS with 64 CPU cores takes just 10 hours for these waveforms. (Initial DINGO samples are available typically in under a minute.) Our results indicate that DINGO(-IS) performs well for the majority of events, and that failure cases are indeed flagged by low sample efficiency. We also find that the log evidence is recovered with statistical uncertainty reduced by a factor of 10 compared to standard samplers.

Machine learning methods have seen numerous applications in GW astronomy, including to detection and parameter estimation [80]. For parameter estimation, these methods have included variational inference [103, 111], likelihood ratio estimation [86], and posterior estimation with normalizing

[1] Deep INference for Gravitational-wave Observations.

flows [111, 110, 82, 62]. Aside from directly estimating parameters, normalizing flows have also been used to accelerate classical samplers, with significant efficiency improvements [231].

Neural density estimation and importance sampling have previously been combined [172] under the guise of "neural importance sampling" [165], and similar approaches have been applied in several contexts [168, 36, 132, 213]. Our contributions are to (1) extend this to amortized simulation-based inference, (2) use it to improve results generated with classical inference methods such as MCMC, and (3) to highlight how the use of a forward Kullback-Leibler (KL) loss improves reliability. We also apply it to the challenging real-world problem of GW inference.[2] We demonstrate results that far outperform classical methods in terms of sample efficiency and parallelizability, while maintaining accuracy and including simple diagnostics. We therefore expect this work to accelerate the development and verification of probabilistic deep learning approaches across science.

4.2 Method

DINGO trains a conditional density-estimation neural network $q(\theta|d)$ to approximate $p(\theta|d)$ based on simulated data sets (θ, d) with $\theta \sim p(\theta)$, $d \sim p(d|\theta)$—an approach called neural posterior estimation (NPE) [175]. Once trained, DINGO can rapidly produce (approximate) posterior samples for any measured data d. In practice, results may deviate from the true posterior due to insufficient training, lack of network expressivity, or out-of-distribution (OOD) data (i.e., data inconsistent with the training distribution). Although it was shown in [82] that these deviations are often negligible, verification of results requires comparing against expensive standard samplers.

Here, we describe an efficient method to *verify* and *correct* DINGO results using importance sampling (IS) [220]. Starting from a collection of n samples $\theta_i \sim q(\theta|d)$ (the "proposal") we assign to each one an importance weight $w_i = p(d|\theta_i)p(\theta_i)/q(\theta_i|d)$. For a perfect proposal, $w_i =$ constant, but more generally the number of *effective samples* is related to the variance, $n_{\text{eff}} = (\sum_i w_i)^2 / \sum_i (w_i^2)$ [139]. The *sample efficiency* $\epsilon = n_{\text{eff}}/n \in (0, 1]$ arises naturally as a quality measure of the proposal.

Importance sampling requires evaluation of $p(d|\theta)p(\theta)$ rather than the normalized posterior. The Bayesian evidence can then be estimated from the normalization of the weights as $p(d) = 1/n \sum_i w_i$. The standard deviation of the log evidence, $\sigma_{\log p(d)} = \sqrt{(1-\epsilon)/(n \cdot \epsilon)}$ (see Sec. C.1), scales with $1/\sqrt{n}$, enabling very precise estimates. The evidence is furthermore unbiased if the support of the posterior is fully covered by the proposal distribution [171]. The *log* evidence does have a bias, but this scales as $1/n$, and in all cases considered here is completely negligible (see Sec. C.1). If $q(\theta|d)$ fails to cover the entire posterior, the evidence itself would also be biased, toward lower values.

NPE is particularly well-suited for IS because of two key properties. First, by construction the proposal has tractable density, such that we can not only sample from $q(\theta|d)$, but also evaluate it. Second, the NPE proposal is expected to always cover the entire posterior support. This is because, during training, NPE minimizes the *forward* KL divergence $D_{\text{KL}}(p(\theta|d)||q(\theta|d))$. This diverges

[2] A similar approach using convolutional networks to parametrize Gaussian and von Mises proposals was used to estimate the sky position alone [137] Using the normalizing flow proposal (as we do here) significantly improves the flexiblity of the conditional density estimator and enables inference of all parameters.

unless $\text{supp}(p(\theta|d)) \subseteq \text{supp}(q(\theta|d))$, making the loss "probability-mass covering". Probability mass coverage is not guaranteed for finite sets of samples generated with stochastic samplers like MCMC (which can miss distributional modes), or machine learning methods with other training objectives like variational inference [131, 226, 193].

Neural importance sampling can in fact be used to improve posterior samples from *any* inference method provided the likelihood is tractable. If the method provides only samples (without density) then one must first train an (unconditional) density estimator $q(\theta)$ (e.g., a normalizing flow [193, 136, 176]) to use as proposal. This is generally fast for an unconditional flow, and using the forward KL loss guarantees that the proposal will cover the samples. Success, however, relies on the quality of the initial samples: if they are light-tailed, sample efficiency will be poor, and if they are not mass-covering, the evidence will be biased. Nevertheless, for initial samples that well represent the posterior, this technique can provide quick verification and improvement.

In the context of GWs, we refer to neural importance sampling with DINGO as DINGO-IS. Although this technique requires likelihood evaluations at inference time, in practice it is much faster than other likelihood-based methods because of its high sample efficiency and parallelizability. Indeed, DINGO samples are independent and identically distributed, trivially enabling full parallelization of likelihood evaluations. This is a crucial advantage compared to inherently sequential methods such as MCMC.

4.3 Results

For our experiments, we prepare DINGO networks as described in [82], with several modifications. First, we extend the priors over component masses to $m_1, m_2 \in [10, 120]\ \mathrm{M}_\odot$ and dimensionless spin magnitudes to $a_1, a_2 \in [0, 0.99]$. We also use the waveform models IMRPhenomXPHM [185] and SEOBNRv4PHM [170], which include higher radiative multipoles and more realistic precession. Finally, in addition to networks for the first observing run of LIGO and Virgo (O1), we also train networks based on O3 noise. For the O3 analyses, we found performance improved by training separate DINGO models with distance priors $[0.1, 3]$ Gpc, $[0.1, 6]$ Gpc and $[0.1, 12]$ Gpc. We continue to use frequency-domain strain data in the range $[20, 1024]$ Hz with $\Delta f = 0.125$ Hz and identical data conditioning as in [82]. The network architecture, hyperparameters, and training algorithm are also unchanged. We consider the two LIGO [1] detectors for all analyses, and leave inclusion of Virgo [33] data to a future publication of a complete catalog.

In our experiments, we found that DINGO often has difficulty resolving the phase parameter ϕ_c. Although ϕ_c itself is of little physical interest, it is nevertheless needed to evaluate the likelihood for importance sampling. We therefore sample ϕ_c synthetically, by first evaluating the likelihood across a ϕ_c grid and caching the waveform modes for efficiency (see Sec. C.2). This approach is similar to standard phase marginalization [223, 224, 219], but it is valid even with higher modes; it can therefore be adapted also to stochastic samplers.

For DINGO-IS, with 10^5 proposal samples per event, the total time for inference using one NVIDIA A100 GPU and 64 CPU cores is typically less than 1 hour for IMRPhenomXPHM and ≈ 10 hours for

	Mean JSD	Max JSD	$\log p(d)$
DINGO	2.2	7.2 (α)	-
DINGO-IS	0.5	1.4 (d_{L})	-15831.87 ± 0.01
BILBY	1.8	4.0 (d_{L})	-15831.78 ± 0.10
DINGO	9.0	53.4 (M_{c})	-
DINGO-IS	0.7	2.2 (α)	-16412.88 ± 0.01
BILBY	1.1	4.1 (α)	-16412.73 ± 0.09

Table 4.1 Performance for GW150914 (upper block) and GW151012 (lower) with waveform model IMRPhenomXPHM. The Jensen-Shannon divergence (JSD) quantifies the deviation from LALINFERENCE-MCMC for one-dimensional marginal posteriors (all values in 10^{-3} nat). The mean is taken across all parameters. Posteriors with a maximum JSD $\leq 2 \times 10^{-3}$ nat are considered indistinguishable [195]; here, maxima occur for right ascension α, luminosity distance d_{L}, and chirp mass M_{c}. We also report BILBY-DYNESTY results.

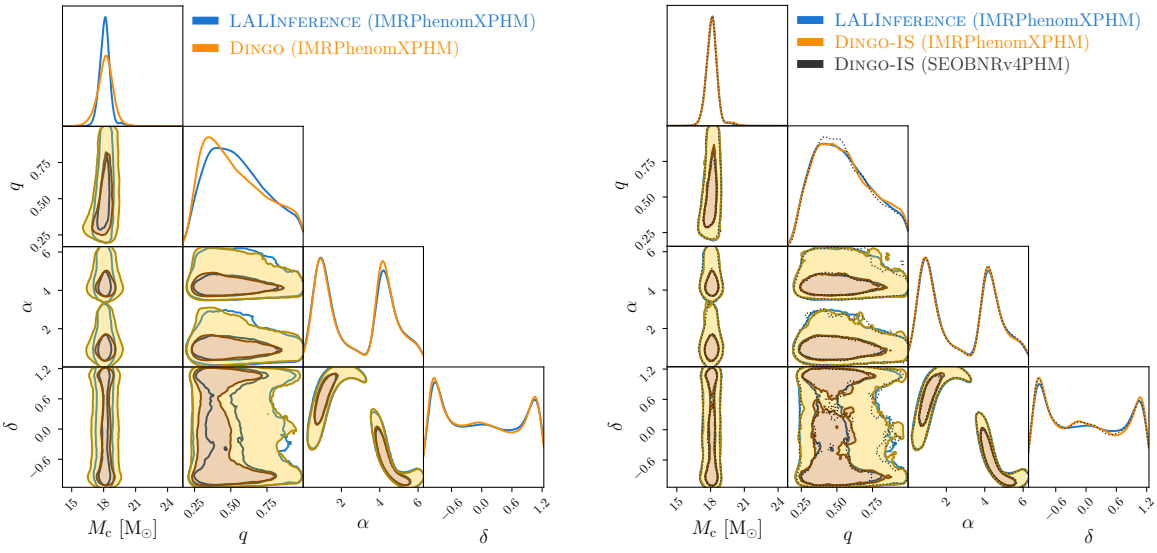


Fig. 4.1 Chirp mass (M_{c}), mass ratio (q) and sky position (α, δ) parameters for GW151012, comparing inference with DINGO and LALINFERENCE-MCMC. Even when initial DINGO results deviate from LALINFERENCE posteriors (left panel), IS leads to almost perfect agreement (right). For comparison, the right panel also shows results for SEOBNRv4PHM.

Event	$\log p(d)$	ϵ	Event	$\log p(d)$	ϵ
GW190408	-16178.332 ± 0.012	6.9%	GW190926	-16015.813 ± 0.019	2.8%
_181802	-16178.172 ± 0.010	9.3%	_050336	-16015.861 ± 0.009	12.1%
GW190413	-15571.413 ± 0.006	22.5%	GW190929	-16146.666 ± 0.018	3.2%
_052954	-15571.391 ± 0.005	26.3%	_012149	-16146.591 ± 0.021	2.4%
GW190413	-16399.331 ± 0.009	12.4%	GW191109	-17925.064 ± 0.025	1.7%
_134308	-16399.139 ± 0.014	4.7%	_010717	-17922.762 ± 0.041	0.6%
GW190421	-15983.248 ± 0.008	15.3%	GW191127	-16759.328 ± 0.019	2.7%
_213856	-15983.131 ± 0.010	9.4%	_050227	-16758.102 ± 0.029	1.2%
GW190503	-16582.865 ± 0.022	2.0%	‡GW191204	-15984.455 ± 0.015	4.2%
_185404	-16583.352 ± 0.027	1.4%	_110529	-15983.618 ± 0.063	0.3%
GW190513	-15946.462 ± 0.043	0.6%	GW191215	-16001.286 ± 0.013	5.8%
_205428	-15946.581 ± 0.017	3.4%	_223052	-16000.846 ± 0.052	0.4%
GW190514	-16556.466 ± 0.009	11.6%	GW191222	-15871.521 ± 0.007	16.5%
_065416	-16556.314 ± 0.017	3.5%	_033537	-15871.450 ± 0.005	25.8%
GW190517	-16271.048 ± 0.027	1.3%	GW191230	-15913.798 ± 0.009	12.2%
_055101	-16272.428 ± 0.034	0.9%	_180458	-15913.918 ± 0.010	8.8%
GW190519	-15991.171 ± 0.008	15.2%	GW200128	-16305.128 ± 0.013	6.1%
_153544	-15991.287 ± 0.068	0.2%	_022011	-16304.510 ± 0.007	18.3%
GW190521	-16008.876 ± 0.008	13.4%	‡GW200129	-16226.851 ± 0.109	0.1%
_074359	-16008.037 ± 0.015	4.2%	_065458	-16231.203 ± 0.051	0.4%
GW190527	-16119.012 ± 0.008	13.8%	GW200208	-16136.381 ± 0.007	16.6%
_092055	-16118.781 ± 0.013	6.1%	_130117	-16136.531 ± 0.009	11.2%
GW190602	-16036.993 ± 0.006	25.0%	GW200208	-16775.200 ± 0.011	7.4%
_175927	-16037.529 ± 0.006	23.5%	_222617	-16774.582 ± 0.021	2.2%
GW190701	-16521.381 ± 0.040	0.6%	GW200209	-16383.847 ± 0.009	12.5%
_203306	-16521.609 ± 0.010	10.1%	_085452	-16384.157 ± 0.025	1.6%
GW190719	-15850.492 ± 0.008	13.4%	GW200216	-16215.703 ± 0.017	3.4%
_215514	-15850.339 ± 0.011	8.0%	_220804	-16215.540 ± 0.018	3.1%
GW190727	-15992.017 ± 0.009	10.3%	GW200219	-16133.457 ± 0.011	9.6%
_060333	-15992.428 ± 0.005	30.8%	_094415	-16133.157 ± 0.017	4.0%
GW190731	-16376.777 ± 0.005	32.6%	GW200220	-16303.782 ± 0.007	17.3%
_140936	-16376.763 ± 0.005	31.0%	_061928	-16303.087 ± 0.026	1.5%
GW190803	-16132.409 ± 0.006	21.4%	GW200220	-16136.600 ± 0.008	13.2%
_022701	-16132.408 ± 0.005	27.8%	_124850	-16136.519 ± 0.037	0.7%
GW190805	-16073.261 ± 0.006	20.0%	GW200224	-16138.613 ± 0.006	22.5%
_211137	-16073.656 ± 0.007	16.6%	_222234	-16139.101 ± 0.006	21.4%
GW190828	-16137.220 ± 0.009	12.2%	‡GW200308	-16173.938 ± 0.013	6.0%
_063405	-16136.799 ± 0.010	9.1%	_173609	-16173.692 ± 0.025	1.7%
GW190909	-16061.634 ± 0.011	7.4%	GW200311	-16117.505 ± 0.011	7.4%
_114149	-16061.275 ± 0.016	3.8%	_115853	-16117.583 ± 0.009	11.9%
GW190915	-16083.960 ± 0.015	20.8%	‡GW200322	-16313.568 ± 0.307	0.0%
_235702	-16083.937 ± 0.027	4.8%	_091133	-16313.110 ± 0.105	0.1%

Table 4.2 42 BBH events from GWTC-3 analyzed with DINGO-IS. We report the log evidence $\log p(d)$ and the sample efficiency ϵ for the two waveform models IMRPhenomXPHM (upper rows) and SEOBNRv4PHM (lower rows). Highlighting colors indicate the sample efficiency (green: high; yellow: medium; orange/red: low); DINGO-IS results can be trusted for medium and high ϵ (see Chapter C). Events in gray suffer from data quality issues [23, 29]. ‡See remarks on these events in text.

SEOBNRv4PHM. In both cases, the computation time is dominated by waveform simulations, which could be further reduced using more CPUs. The rest of the time is taken up to generate the initial DINGO proposal samples.[3]

We first validate DINGO-IS against standard inference codes for two real events, GW150914 and GW151012, using IMRPhenomXPHM. (For SEOBNRv4PHM it is not feasible to run classical samplers, and one would instead need to use faster methods such as RIFT [174, 143].) We generate reference posteriors using LALINFERENCE-MCMC [224], and compare one-dimensional marginalized posteriors for each parameter using the Jensen-Shannon divergence (Tab. 4.1). For both events, the initial small deviations of DINGO samples from the reference are made negligible[4] using DINGO-IS (see Fig. 4.1 for a qualitative demonstration). We find sample efficiencies of $\epsilon = 28.8\%$ and $\epsilon = 12.5\%$ for GW150914 and GW151012, respectively.

For the evidence, we compare against BILBY-DYNESTY [41, 195, 210], since nested sampling generally provides a more accurate estimate than MCMC. In Tab. 4.1 we see that DINGO-IS is more precise by a factor of ≈ 10, but the BILBY evidence is larger for both events by roughly one standard deviation. This deviation could be statistical, but it could also indicate a bias in one of the methods. (Recall that IS requires the proposal to be mass-covering for an unbiased evidence.) To further investigate for GW151012, we perform neural importance sampling starting from 10^6 BILBY samples (see Sec. C.3.2). This achieves a slightly lower $\epsilon = 8.3\%$ than DINGO-IS, but $\log p(d) = -16412.89 \pm 0.01$ in close agreement. While this does not fully rule out a bias in DINGO-IS samples (since the test is not fully independent) we take this as an indication that DINGO-IS indeed infers an unbiased evidence. More generally, it showcases how our method can be extended to improve the output of stochastic samplers.

We now perform a large study analyzing all 42 events in GWTC-2 [23] and GWTC-3 [29] that are consistent with our mass prior.[5] We stress that a study of this scope would be infeasible with standard codes, since SEOBNRv4PHM inference for a single event would take several months. Across all events we achieve a median sampling efficiency of $\epsilon = 10.9\%$ for IMRPhenomXPHM and $\epsilon = 4.4\%$ for SEOBNRv4PHM (Tab. 4.2). For most events, the initial DINGO results are already accurate and only deviate slightly from DINGO-IS; furthermore, DINGO-IS shows excellent agreement between the two waveform models (see Sec. C.6 for more detailed comparisons). Note that these results are based on highly complex precessing higher-mode waveform models, and do not include any mitigation of noise transients (see below). With the simpler IMRPhenomPv2 [115, 134, 50] model and a smaller mass prior (in a study on drifting detector noise distributions [230]) DINGO-IS achieves an even larger median sample efficiency of $\epsilon = 36.8\%$ on 37 events.

[3] It takes longer to generate the proposal than to produce low-latency DINGO samples (≈ 20 s) because of the group-equivariant NPE (GNPE) algorithm [82, 83] (which breaks access to the density) and the synthetic phase recovery. See Chapter C for details.

[4] Initial deviations are larger than those reported in [82] since we use a more complicated waveform model and a larger prior, while keeping the size of the neural network and training time the same. Any remaining deviations after importance sampling can in principle also be due to sampling inaccuracies of LALINFERENCE-MCMC. Note that a direct comparison to published LIGO-Virgo-KAGRA results is impeded by different data settings.

[5] Lower mass events produce longer signals, so extending DINGO to these may require improved methods for data compression [225, 57]. This will be particularly relevant for binary neutron stars.

Importance sampling guarantees robust results by marking failure cases with a low sample efficiency. By this metric, DINGO struggles slightly with chirp masses near the lower prior boundary (GW191204_110529 and GW200322_091133). For such systems, efficiency may be improved by increasing the prior range used for training. Events with known data quality issues also often have low sample efficiency (see Tab. 4.2): several low-ϵ events are contaminated by glitch artifacts (which would be mitigated in a more complete analysis [23, 29]); GW200129_065458, in addition to having a glitch [180], may not be well modeled by either of our waveform models due to having strong precession [116]; and GW200322_091133 may be simply a Gaussian noise fluctuation [163]. In these cases, DINGO-IS marks events for additional investigation.

Data quality issues such as non-Gaussian noise or observed signals that do not match models correspond to OOD data, i.e., data not consistent with the training distribution. Since OOD data are not seen during training, DINGO cannot be expected to return their true posterior, which results in a low sample efficiency. As an additional test, running DINGO-IS on signal-free data with a blip glitch [74] in the LIGO Hanford detector (GPS time 1238613687.5) results in $\epsilon \approx 0.001\%$. Likewise, we find that DINGO-IS successfully flags adversarial examples [215, 109] that are intentionally corrupted to mislead the inference network ($\epsilon \approx 0.01\%$; see Sec. C.5)—addressing a common failure mode of neural networks. Our general view, therefore, is that although there can be various reasons for low-ϵ results, it often serves as a useful heuristic to identify OOD events.

4.4 Conclusions

We have described the use of importance sampling to improve the results of NPE in amortized inference problems, and we applied it to the case of GWs. Neural importance sampling provides rapid verification of results and corrects any inaccuracies in deep learning output; it provides an evidence estimate with precision far exceeding that of classical samplers; and it marks potentially OOD data for further investigation. With high sample efficiency and rapid initial results, DINGO-IS becomes a comprehensive inference tool for accurately analyzing the large numbers of BBH events expected soon.

High sample efficiencies are predicated on a high quality proposal, which DINGO thankfully provides. A key element is the probability-mass covering property, which is guaranteed by the forward KL training loss. This tends to produce broad tails, which are downweighted in importance sampling. *Overly* broad proposals would nevertheless result in low sample efficiency, so highly expressive density estimators such as normalizing flows are essential, along with DINGO innovations such as GNPE and GW training data augmentation. DINGO posteriors are rarely light tailed, but this does occasionally lead to underestimated evidence for small n.

With the inclusion of importance sampling, the DINGO pipeline can now be used in several different ways. When low latency is desired, complete posteriors are still available without importance sampling in a matter of seconds. Results include sky position and mass parameters and could therefore play an important role in directing electromagnetic followup observations once we extend DINGO to mergers involving neutron stars (see footnote 5). By comparing against DINGO-IS, we have shown that in the majority of cases, initial results are already very reliable, with only minor deviations in marginal

distributions. Indeed, validation of DINGO results was a major motivation in exploring importance sampling.

When high accuracy is desired, DINGO-IS reweights results to the true posterior and includes an estimate of the evidence. Results are verified and include probability mass-covering guarantees that ensure secondary modes are not missed. Sample efficiencies are often two orders-of-magnitude higher than MCMC or nested sampling, and importance sampling is fully parallelizable. As a consequence, results are typically available within an hour for IMRPhenomXPHM, or 10 hours for SEOBNRv4PHM. This represents a significant advantage when considering the event rates likely to be reached with advanced detectors (three per week or higher in the upcoming LIGO-Virgo-KAGRA observing run O4).

DINGO-IS opens several new possibilities for GW analysis: (1) rapid inference means that the most accurate waveform models, which include all physical effects, could be used for all events; (2) high-precision evidences enable detailed model comparison; and (3) low sample efficiencies can identify data that do not fit the noise or waveform model. We believe that these results have highlighted clear benefits of combining likelihood-free and likelihood-based methods in Bayesian inference. Going forward, as DINGO-IS validates and builds trust in DINGO, it will help to set the stage for noise-model free inference, which is truly likelihood-free.

The code for DINGO and DINGO-IS is available at https://github.com/dingo-gw/dingo.

Acknowledgments

We thank V. Raymond for encouraging us to pursue importance sampling in the early stages of the project, and C. García Quirós, N. Gupte, S. Ossokine, A. Ramos-Buades and R. Smith for useful discussions. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation. This research has made use of data or software obtained from the Gravitational Wave Open Science Center (gw-openscience.org), a service of LIGO Laboratory, the LIGO Scientific Collaboration, the Virgo Collaboration, and KAGRA. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation (NSF) as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. The construction and operation of KAGRA are funded by Ministry of Education, Culture, Sports, Science and Technology (MEXT), and Japan Society for the Promotion of Science (JSPS), National Research Foundation (NRF) and Ministry of Science and ICT (MSIT) in Korea, Academia Sinica (AS) and the Ministry of Science and Technology (MoST) in Taiwan. M.D. thanks the Hector Fellow Academy for support. J.H.M. and B.S. are members of the MLCoE, EXC number 2064/1 – Project number 390727645 and the Tübingen AI Center funded by the German Ministry for Science and Education (FKZ 01IS18039A). For the implementation of DINGO we use

`PyTorch` [179], `nflows` [94], `LALSimulation` [145] and the adam optimizer [135]. The plots are generated with `matplotlib` [127] and `ChainConsumer` [122].

Chapter 5

Real-Time Inference for Binary Neutron Star Mergers using Machine Learning

Mergers of binary neutron stars (BNSs) emit signals in both the gravitational-wave (GW) and electromagnetic (EM) spectra. Famously, the 2017 multi-messenger observation of GW170817 [11, 9] led to scientific discoveries across cosmology [7], nuclear physics [10, 13, 12], and gravity [14]. Central to these results were the sky localization and distance obtained from GW data, which, in the case of GW170817, helped to identify the associated EM transient, AT 2017gfo [75], 11 hours after the GW signal. Fast analysis of GW data is critical for directing time-sensitive EM observations; however, due to challenges arising from the length and complexity of signals, it is often necessary to make approximations that sacrifice accuracy. Here, we present a machine learning framework that performs complete BNS inference in just one second without making any such approximations. Our approach enhances multi-messenger observations by providing (i) accurate localization even before the merger; (ii) improved localization precision by $\sim 30\%$ compared to approximate low-latency methods; and (iii) detailed information on luminosity distance, inclination, and masses, which can be used to prioritize expensive telescope time. Additionally, the flexibility and reduced cost of our method open new opportunities for equation-of-state studies. Finally, we demonstrate that our method scales to extremely long signals, up to an hour in length, thus serving as a blueprint for data analysis for next-generation ground- and space-based detectors.

Declaration

This chapter is based on the following published manuscript.

> *Real-time inference for binary neutron star mergers using machine learning*
> Maximilian Dax, Stephen R. Green, Jonathan Gair, Nihar Gupte, Michael Pürrer, Vivien Raymond, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, Bernhard Schölkopf
> Nature **639**, 49-53 – Published 5 March 2025

Text and figures are adapted from the arXiv version `arXiv:2407.09602v2` with minor updates to layout and references.

Author contributions

Maximilian Dax led the research with input from **Stephen R. Green** and **Bernhard Schölkopf**. **Maximilian Dax** and **Stephen R. Green** conceived the main methodology. **Maximilian Dax** devised and carried out the implementation and experimental analysis. **Maximilian Dax** and **Stephen R. Green** developed the DINGO code, with contributions from **Nihar Gupte**, **Michael Pürrer**, and **Jonas Wildberger**. **Nihar Gupte** implemented the JAX waveform model, and **Vivien Raymond** proposed to explore equation-of-state inference. **Maximilian Dax** and **Stephen R. Green** wrote the paper. **All authors** contributed to scientific discussions and paper editing.

5.1 Introduction

Fast and accurate inference of binary neutron stars (BNSs) from gravitational-wave (GW) data is a critical challenge facing multi-messenger astronomy. For a BNS, the GW signal is visible by the LIGO-Virgo-KAGRA (LVK) [1, 33, 42] observatories minutes before any electromagnetic counterpart, and encodes information on source characterization, distance, sky location, and orientation necessary for pointing and prioritizing optical telescopes. However, the length of BNS signals makes conventional Bayesian inference techniques [224, 41] too slow to be useful in low-latency applications. Instead, once a GW signal is identified by detection pipelines [100, 166, 58, 34, 66, 196, 167, 140], approximate algorithms are used for providing initial alerts (e.g., Bayestar [203], which uses the signal-to-noise [SNR] time series rather than the complete strain data and gives localization in seconds). Other methods focus on accelerating likelihood evaluations without incurring loss of precision (e.g., using reduced-order quadratures), with the state-of-the-art delivering localization in six minutes, and full inference in two hours [162].

Simulation-based machine learning offers a powerful alternative for GW inference (see Sec. D.1.6 for related work). With simulation-based inference (SBI) [78], neural networks are trained to encode probabilistic estimates of astrophysical source parameters conditional on data. Trained networks then enable extremely fast analysis for new data sets, amortizing upfront training costs across observations. In past work, we developed the DINGO framework for binary black holes (BBHs) [111, 110, 82, 84], which performs accurate inference in seconds, including strong accuracy guarantees when coupled with importance sampling. However, when applied to BNS, machine learning approaches such as DINGO are beset by the same challenges facing traditional methods due to long signal durations. Indeed, DINGO becomes unreliable even for low-mass BBHs (chirp masses $\lesssim 15\ \mathrm{M}_\odot$), with signals longer than roughly 16 s. A BNS lasts for hundreds of seconds for the LVK and will reach hours for next-generation detectors (XG, e.g., Cosmic Explorer [192] and Einstein Telescope [187]). From the neural network perspective, this corresponds to time or frequency series input with up to tens of millions of dimensions, a thousand-fold increase over BBH.

In this study, we overcome these challenges by leveraging perturbative BNS physics information to simplify and compress the data. However, this simplification requires approximate knowledge of the source itself and is hence valid only over a small portion of the parameter space. We solve this problem using a new algorithm called "prior-conditioning," which enables us to construct networks that can be adapted at inference time to subsets of the prior volume. Our new framework, called DINGO-BNS, makes no (practically relevant) approximation, and takes just *one second* for accurate inference of all 17 BNS parameters (Fig 5.1). DINGO-BNS can also infer all of these parameters *minutes before the merger* based on partial inspiral-only information, estimates which can be continuously updated as more data become available (Fig. 5.2a). Near-real-time or pre-merger alerts can then be provided to astronomers, facilitating potential discoveries of precursor and prompt electromagnetic counterparts [63, 211, 164].

Our results are faster and more complete than any existing low latency algorithm, with the accuracy of offline parameter estimation codes. Compared to Bayestar, we achieve median reductions in the size of the 90% credible sky region of 30% (Fig. 5.2b). Finally, DINGO-BNS exhibits excellent scaling to

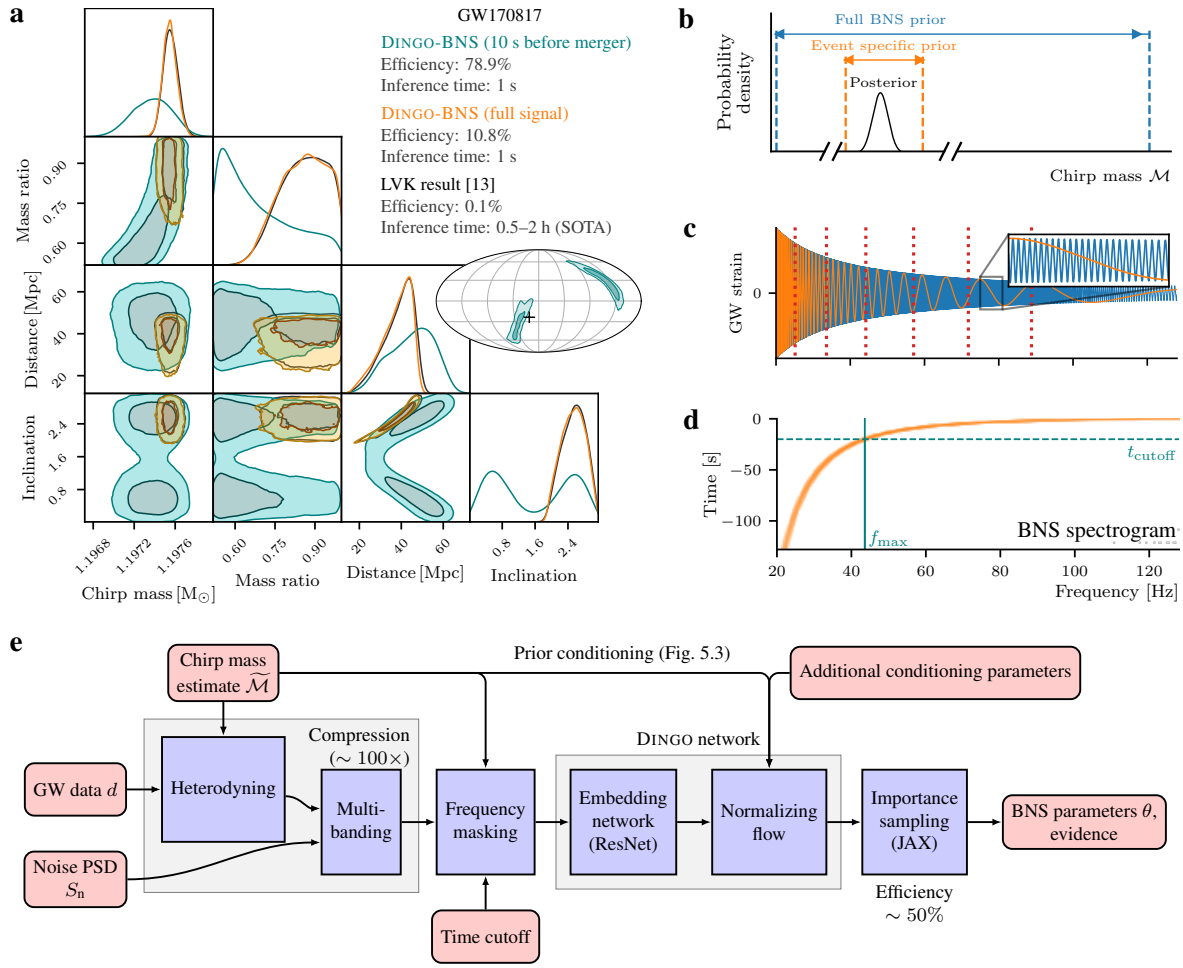


Fig. 5.1 **Real-time GW inference for BNS is enabled by several innovations.** (a) DINGO-BNS estimates all BNS parameters in just one second (orange), reproducing LVK results [13] (black) three orders of magnitude faster than existing methods [162, 232]. DINGO-BNS can also analyze partial data before the merger occurs (teal). Fast analysis results are crucial for directing electromagnetic searches for prompt or even precursor signals. Note that GW170817 overlapped with a loud glitch, which could explain why the true sky position lies in the tail of the pre-merger distribution. (b) For a given event, the chirp mass posterior (black) is tightly constrained compared to the prior (blue), so a restricted chirp mass prior (orange) is sufficient, and moreover simplifies analysis. With our prior-conditioning technique, we train a single neural network that can be instantly tuned to an event-specific prior lying anywhere within the full volume. (c) We compress data by a factor of ~ 100 by first factoring out ("heterodyning") the predominant phase evolution of the signal (blue), based on a chirp mass estimate $\widetilde{\mathcal{M}}$ associated to the event-specific prior. The resulting simplified signal (orange) is down-sampled in resolution, reducing data dimensionality (coarser resolution at high frequencies; bands indicated by dotted red lines). (d) To enable pre-merger inference, we mask out the strain frequency series according to the cut-off time. (e) All of these components are integrated into a single neural network that can be trained end-to-end and produce 10^5 weighted samples per second, with typical sampling efficiencies of 50%.

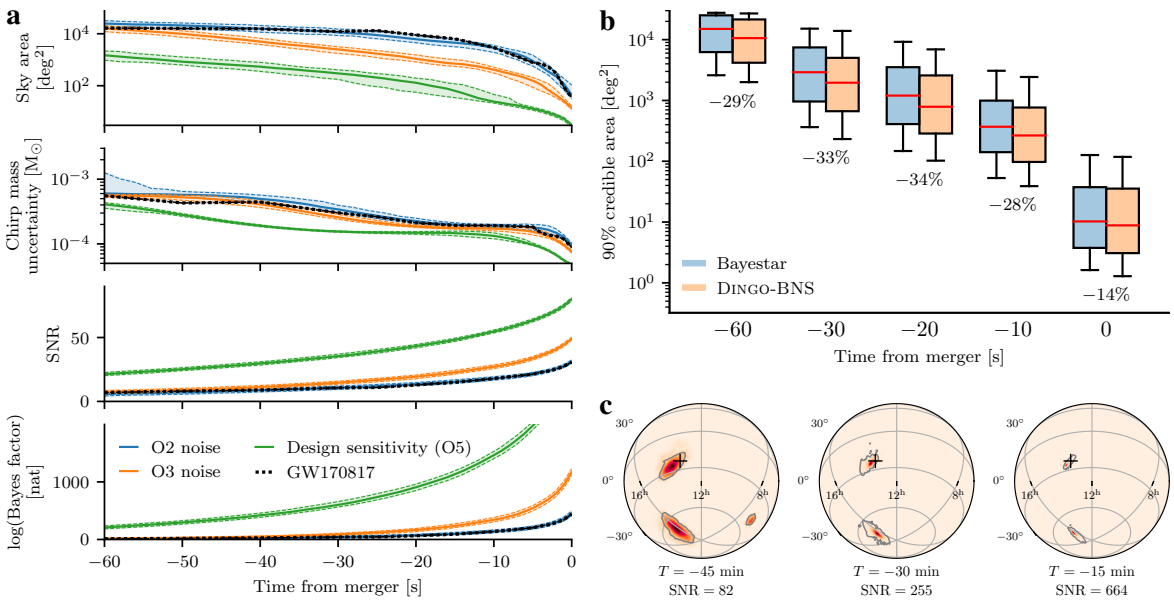


Fig. 5.2 **Pre-merger inference with DINGO-BNS.** (a) Evolution of pre-merger estimates for GW170817 (black) and GW170817-like simulations injected into different noise levels (colors). We display the 90% credible sky area, the standard deviation of the chirp mass, the accumulated signal-to-noise and the log Bayes factor comparing the signal and noise models. All of these quantities are inferred with a latency of ~ 1 second. Dotted lines represent 10th/90th percentiles. (b) Sky localization area at 90% credible level for various premerger times, comparing against Bayestar. The boxplots display the median (red line), quartiles (colored box) and 10th/90th percentiles (whiskers). DINGO-BNS localization is consistently more precise. (c) Premerger sky localization for a GW170817-like event injected into Cosmic Explorer noise, using a minimum frequency of 6 Hz. The black marker indicates the injection coordinates, and gray outline the 90% credible area.

longer signals (see Sec. D.1.3), and we demonstrate XG pre-merger inference for signals up to an hour in length (Fig. 5.2c).

5.2 DINGO-BNS

For given GW data d, we characterize the source in terms of the posterior probability distribution $p(\theta|d)$ over BNS parameters θ. Parameters include component masses (2), spins (6), orientation, sky position (2), luminosity distance, polarization, time and phase of coalescence, and (in contrast to black holes) tidal deformabilities (2). Following [82], we use simulated GW datasets to train a density estimation neural network $q(\theta|d)$ (a normalizing flow) to approximate $p(\theta|d)$. Once trained, inference for new d simply requires sampling $\theta \sim q(\theta|d)$. We obtain asymptotically exact results by augmenting samples with importance weights using the GW likelihood function [18]. This framework, called DINGO-IS [84], has been successfully applied to black hole mergers, however the length of BNS signals renders the naïve transfer of machine learning methods impossible.

DINGO-BNS makes several innovations to tackle this challenge (Fig. 5.1e), including using knowledge of specific BNS signal morphology to compress data in a non-lossy way; conditioning the network on the compressor using prior-conditioning; frequency masking based on the pre-merger time and chirp mass; and conditioning on parameter subsets for incorporating multi-messenger information or expectations from nuclear models. The philosophy underlying our approach is that the full BNS problem is too hard for existing neural architectures, so we divide the parameter and data spaces into manageable portions based on known physical information. We then combine all of these variable design choices into a single network that we can instantly tune to the context at hand.

5.2.1 Data compression and prior conditioning

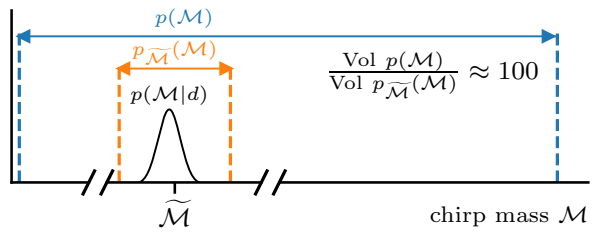
We adapt two GW analysis techniques to the SBI context, heterodyning [69, 70, 235] to simplify the data, and multibanding [225, 161] to reduce the data dimension without loss of information. During the long inspiral period, a BNS signal exhibits a "chirp," with phase evolution (to leading order in the post-Newtonian expansion [48]),

$$\varphi(f;\mathcal{M}) = \frac{3}{128}\left(\frac{\pi G\mathcal{M}f}{c^3}\right)^{-5/3}, \tag{5.1}$$

where $\mathcal{M} = (m_1 m_2)^{3/5}/(m_1+m_2)^{1/5}$ is the chirp mass of the system, with m_1, m_2 the component masses. Given an approximation $\widetilde{\mathcal{M}}$ to the chirp mass, we heterodyne the (frequency-domain) data by multiplying by $e^{i\varphi(f;\widetilde{\mathcal{M}})}$, reducing the number of oscillations in the signal by several orders of magnitude (Fig. 5.1c). Given heterodyned data, we apply multibanding by partitioning the domain into (empirically-determined) frequency bands, and coarsening the resolution in higher bands such that the (heterodyned) signal is preserved.

Since the compression described requires $\widetilde{\mathcal{M}}$ to approximate the chirp mass, it cannot be done across the entire BNS prior volume using a single $\widetilde{\mathcal{M}}$ value. DINGO-BNS therefore uses prior-conditioning to restrict to an event-specific prior over which data is compressed. The restricted volume additionally

a



b *Training step*

1: $\widetilde{\mathcal{M}} \sim \hat{p}(\widetilde{\mathcal{M}})$ ▷ Sample $\widetilde{\mathcal{M}}$ from hyperprior
2: $\mathcal{M} \sim p_{\widetilde{\mathcal{M}}}(\mathcal{M})$ ▷ Sample $\mathcal{M}$ from restricted prior
3: $d \leftarrow d(\mathcal{M})$ ▷ Simulate data
4: $d_{\widetilde{\mathcal{M}}} \xleftarrow{\widetilde{\mathcal{M}}} d$ ▷ Compress data, see (5.1c)
5: Optimize $-\log q(\mathcal{M} - \widetilde{\mathcal{M}} | d_{\widetilde{\mathcal{M}}}, \widetilde{\mathcal{M}})$

c *Inference*

Require: $\widetilde{\mathcal{M}} \approx \mathcal{M}_{\text{true}}$ ▷ Choose approximate $\widetilde{\mathcal{M}}$
1: $d_{\widetilde{\mathcal{M}}} \xleftarrow{\widetilde{\mathcal{M}}} d$ ▷ Compress data
2: $\delta\mathcal{M} \sim q(\delta\mathcal{M} | d_{\widetilde{\mathcal{M}}}, \widetilde{\mathcal{M}})$ ▷ Sample $\delta\mathcal{M}$ from network
3: $\mathcal{M} \leftarrow \widetilde{\mathcal{M}} + \delta\mathcal{M}$

Fig. 5.3 **Prior conditioning enables event-specific compression.** We train an SBI model simultaneously across a range of priors, each parametrized by a reference chirp mass $\widetilde{\mathcal{M}}$. For each (narrow) prior $p_{\widetilde{\mathcal{M}}}(\mathcal{M})$, we apply heterodyning and multibanding compression. This compression simplifies the data distribution that the model must learn and reduces its dimensionality. For simplicity in this presentation, we omit parameters other than the chirp mass.

simplifies the density estimation task. By conditioning on the choice of restriction, prior-conditioning trains a network that is *tunable* to this choice but otherwise applicable over the whole volume (Fig. 5.3). Inference requires an estimate $\widetilde{\mathcal{M}}$ of the chirp mass $\mathcal{M}$, which can be determined quickly by sweeping across the prior (see Sec. D.1.2).

5.2.2 Frequency masking

In contrast to past work, DINGO-BNS also allows for strain frequency series with varying f_{min} and f_{max}. For a given analysis, f_{min} is chosen based on $\widetilde{\mathcal{M}}$ and the segment duration, as the minimum frequency present in the signal in a given GW detector network. This masking is necessary for consistency with frequency-domain waveform models, which assume infinite duration. Choosing f_{max}, by contrast, determines the end time of the data stream analyzed to enable pre-merger inference (see Sec. D.1.4).

5.2.3 Conditioning on parameter subsets

The DINGO-BNS framework (and SBI in general) allows for considerable flexibility in terms of quickly marginalizing over and conditioning on parameters. Conditioning on a parameter allows

us to set it to a fixed value, e.g., to incorporate knowledge of that parameter from other sources. In our study, we trained DINGO-BNS networks conditioned on the sky position, i.e., we learned $p(\theta \setminus \{\alpha, \delta\}|d, \alpha, \delta)$, where α, δ denote the right ascension and declination, respectively. Such a network allows us to incorporate precise multi-messenger localization to obtain tighter constraints on the remaining parameters, potentially enabling real-time feedback on whether optical candidates should be prioritized for detailed spectroscopy [16]. In this way, DINGO-BNS can enable new modes of interaction between GW and electromagnetic observers, potentially transforming how we prioritize and respond to multi-messenger events. We have also explored parameter-conditioning to accelerate offline nuclear equation-of-state analyses (see Sec. D.1.5).

5.3 Experiments

We generate training data using simulated BNS waveforms (including spin-precession and tidal contributions, but without higher angular multipoles [89]) with additive stationary Gaussian detector noise. When relevant, networks are also trained with power spectral density (PSD)-conditioning to enable instant tuning to noise levels at the time of an event. At inference time, we validate and correct results using importance sampling, thus guaranteeing their accuracy provided a sufficient effective sample size is obtained [84]. We accelerate the importance sampling step using JAX waveform and likelihood implementations [96, 232, 234].

We performed four studies using DINGO-BNS: (a) pre-merger analysis of the first BNS detected, GW170817, as well as equivalent injections (simulated data sets) at varying noise levels; (b) pre-merger analysis of a range of injections in LVK design sensitivity noise; (c) after-merger analysis of the two detected BNS events, GW170817 and GW190425, reproducing published LVK results; and (d) pre-merger analysis of injections in Cosmic Explorer noise (with a minimum frequency of 6 Hz, corresponding to an hour long signal). We use the importance sampling efficiency as a primary performance metric, finding average values of 45.8%, 48.5%, 31.0%, and 35.6% in experiments (a), (b), (c), (d), respectively. With these high efficiencies, inference for 10^4 effective samples takes roughly one second on an H100 GPU (see Sec. D.2.2). Efficiencies are generally higher for pre-merger, likely because the waveform morphology is simplest in the early inspiral.

5.4 Discussion

Prior conditioning works well for BNS inference, and it could be extended to address further challenges in GW astronomy (e.g., isolation of events from overlapping backgound signals in XG) and other scientific domains. In the future we would like to explore our prior-conditioning approach to data compression for black hole-neutron star systems and low-mass BBHs. This is nontrivial because such systems can emit GWs in higher angular radiation multipoles (i.e., beyond the $(l, m) = (2, 2)$ mode that we assume here), which evolve according to integer multiples of (5.1), and so would require an improved heterodyning algorithm to factor out the chirp. Higher modes are not present in BNS signals since the stars are very nearly equal mass.

Another exciting prospect for SBI is a more realistic treatment of detector noise. Indeed, since BNS inspirals have long duration, noise non-stationarities and non-Gaussianities are more likely to manifest. DINGO-BNS currently assumes stationary Gaussian noise and is supplied with an off-source estimate of the PSD. However, by training on realistic detector noise, our approach can in principle learn to fully characterize the noise jointly with the signal, including any deviations from stationarity and Gaussianity. This approach is akin to on-source PSD and glitch modeling [73], but allows for more general noise and automatically marginalizes over uncertainties. Initial steps in this direction have already been taken for intermediate-mass binary black holes [191]. Improved noise treatments such as those afforded by SBI will become crucial for reducing systematic error as detectors become more sensitive [81].

Finally, although DINGO-BNS is intended to be used for parameter estimation following a trigger by dedicated search pipelines, its speed opens the possibility to run continuously on all data as they are taken. Either the signal-to-noise ratio or Bayesian evidence time series generated by DINGO-BNS could then be used as a detection statistic, forming an end-to-end detection and parameter estimation pipeline. To implement this would require calibrating these statistics to determine false alarm rates, as well as careful comparisons against existing algorithms to establish efficacy.

Acknowledgments

We thank S. Buchholz, T. Dent, A. Kofler and S. Morisaki for useful discussions. This research has made use of data or software obtained from the Gravitational Wave Open Science Center (gwosc.org), a service of the LIGO Scientific Collaboration, the Virgo Collaboration, and KAGRA. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation, as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. KAGRA is supported by Ministry of Education, Culture, Sports, Science and Technology (MEXT), Japan Society for the Promotion of Science (JSPS) in Japan; National Research Foundation (NRF) and Ministry of Science and ICT (MSIT) in Korea; Academia Sinica (AS) and National Science and Technology Council (NSTC) in Taiwan. M.D. thanks the Hector Fellow Academy for support. This work was supported by the German Research Foundation (DFG) through Germany's Excellence Strategy – EXC- Number 2064/1 – Project number 390727645). The computational work for this manuscript was carried out on the Atlas cluster at the Max Planck Institute for Intelligent Systems in Tübingen, Germany, and the Lakshmi and Hypatia clusters at the Max Planck Institute for Gravitational Physics in Potsdam, Germany. V.R. is supported by the UK's Science and Technology Facilities Council grant ST/V005618/1.

Chapter 6

Conclusion

The GW perspective

This thesis introduces DINGO, a probabilistic machine learning approach to characterize astrophysical GW sources (mergers of black holes or neutron stars) based on corresponding GW measurements. DINGO achieves comparable accuracy to conventional inference methods while being multiple orders of magnitude faster. This efficiency prepares GW data analysis for increasing detection rates and further enables routine use of complex waveform models and large-scale studies. For example, an analysis with DINGO found the strongest evidence for eccentric black hole orbits to date [113], which could reveal important information about the formation of binary black holes. For binary neutron stars, DINGO-BNS performs full inference in less than one second, providing real-time source localization. This could help revolutionize searches for electromagnetic counterparts to GW signals, one of the most promising fields in GW astronomy. DINGO is reviewed for production use by the LIGO-Virgo-KAGRA collaboration, where it is currently operating in parallel with traditional codes on GW events in the fourth observing run.

Going forward, there are various possible extensions to this work. At present, it is still challenging to apply DINGO to sources with low chirp masses $\mathcal{M} < 15\ \mathrm{M}_\odot$ (Figure 6.1), because these emit very long signals. For binary neutron stars (the source type with the lowest chirp mass range), DINGO-BNS solves this problem with a specialized compression technique. However, other low-mass sources, such as neutron star-black hole systems and low-mass binary black holes, emit in higher angular radiation multipoles. This makes the DINGO-BNS compression technique less effective and further innovations are likely required to handle such sources. Extension to these systems would close the final gap in the parameter space, making DINGO applicable to all LIGO-Virgo-KAGRA detections.

Continued efforts to further improve DINGO's reliability and efficiency will be crucial to further establish it in the GW community. A present limitation is that changes to the inference configuration often require training new DINGO models from scratch. This could be addressed by building a foundation model that can adjust such settings at inference time. Another practical issue is that the symmetry-enhanced GNPE framework (Chapters 2,3) leads to substantially slower inference when combined with importance sampling (Chapter 4; see Sec. C.3.1 for the technical reason). Replacing

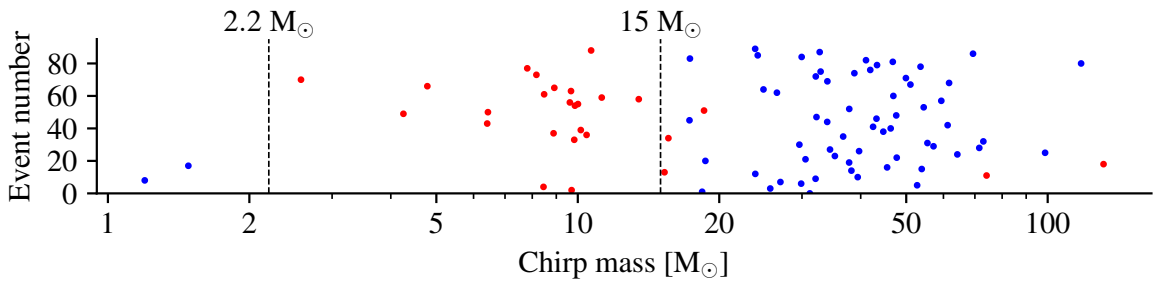


Fig. 6.1 Out of the 90 confident GW detections in the first three LIGO-Virgo observing runs, 63 events (blue) have been analyzed with DINGO in this thesis or in Ref. [113]. At present, DINGO is applicable to binary neutron stars with $\mathcal{M} \leq 2.2\,\mathrm{M}_\odot$ or to binary black holes with $\mathcal{M} \geq 15\,\mathrm{M}_\odot$. Most events that have not yet been analyzed with DINGO (red) fall within this mass gap. Analysis of such events is challenging, as low masses lead to long GW signals and thus high dimensional data.

this with an alternative method to integrate the corresponding symmetries (e.g., via an improved neural network architecture) could thus further reduce inference times.

Our research also opens up more exploratory directions. For example, one could make full use of the simulation-based paradigm by injecting GW signals into real detector noise during DINGO training, resulting in noise model-free inference. Our framework could further be modified to directly infer black hole population parameters from multiple GW events [27, 144], instead of analyzing each one individually. While the focus of this thesis is on parameter estimation, pretrained DINGO networks could also be finetuned for GW search, as they encode GW waveforms and detector noise properties.

Finally, next-generation detectors will revolutionize GW astronomy in the coming decades with vast opportunities for scientific discoveries. These detectors will be able to resolve new astrophysical sources, such as massive black-hole binaries and extreme mass ratio inspirals as well as thousands of simultaneous galactic binaries. At the same time, GW data analysis will be faced with new challenges. These include further increased detection rates (e.g., $10^5 - 10^6$ black hole mergers per year detected by Einstein Telescope [186, 156, 53] and Cosmic Explorer [192]) as well as measurements of louder, longer, more complex, and potentially overlapping GW signals. In the space-based Laser Interferometer Space Antenna [38], GW signals will be so abundant and strongly overlapping that inference will have to be performed simultaneously for all signals in terms of a global fit [72, 150]. While so far we have only briefly explored DINGO in the context of next-generation detectors (Chapter 5), we believe that it has great potential to address some of the open problems in this new era.

The machine learning perspective

This thesis builds on Bayesian simulation-based inference [78], specifically neural posterior estimation (NPE) [175, 152, 112] with normalizing flows [178, 93], and crucially on early applications of these techniques to GW inference by some of my close collaborators [111, 110]. This thesis further introduces a range of new methods, some of which are applicable beyond GW science. *Group-equivariant NPE* (GNPE, Chapter 3) describes a general way to integrate domain knowledge, such as physical symmetries, into conditional density estimation. *Importance-sampled NPE* (NPE-IS, Chapter 4) estab-

lishes an inference framework that verifies and potentially corrects its own results. It further provides a precise estimate of the Bayesian evidence, which is often hard to compute otherwise. In his talk at the 09/2024 PHYSTAT workshop, Gilles Louppe dubbed this a "cheat code for asymptotically exact inference." Indeed, NPE-IS addresses important problems in the field of simulation-based inference, where it is often difficult to assess the validity of inference results. NPE-IS is applicable whenever the likelihood can be simulated and evaluated, and has already been adopted in other fields including exoplanet astronomy [104] and material science [212]. With *prior conditioning* (Chapter 5), a single NPE network supports a range of priors and also prior-adaptive data transformations and compression. These techniques were developed to address challenges in GW inference (involving noisy high-dimensional data and complicated GW models) and the extraordinarily high accuracy requirements, but are nevertheless broadly applicable. This underscores the potential of application-driven machine learning research to contribute general methods that are useful beyond the specific scientific domain.

Reflecting on the techniques developed in this thesis, careful decomposition of the inference problem stands out as a central concept. Rather than estimating the target distribution (the posterior) directly, it is often more effective to break it into components aligned with the underlying physical structure, and to estimate these individually. This approach simplifies distributional properties and thereby facilitates the density estimation task. For example, GNPE and prior conditioning both leverage this idea to integrate symmetries or information from perturbative physics. We found that such decompositions often have a greater effect on the performance for fixed computational budget than other design choices, such as neural architecture or hyperparameters.

In recent years, progress in deep learning has been driven by an ever-increasing scale of computation and data, resulting in extremely large models [169, 216, 92]. This trend aligns with Richard Sutton's famous blogpost "The Bitter Lesson" from 2019 [214], which states that "general methods that leverage computation are ultimately the most effective, and by a large margin." For the research in this thesis, computational scale is indeed an important factor. Our neural networks have been trained with billions of samples, for days to weeks on state-of-the-art GPUs, fully utilizing the computational resources available to us. On the other hand, we also needed to develop new domain-inspired techniques to obtain sufficiently accurate results. Extensively leveraging both, computation and domain knowledge, is thus crucial for our framework, and we could not have accomplished comparable results if we neglected one of the two. In general, augmentation of scalable computation-driven techniques with knowledge-based components is an exciting research direction to enhance the efficiency and applicability of machine learning, in particular in the context of scientific applications. We hope that the ideas presented in this thesis inspire further adoption of such hybrid approaches in GW science, in other scientific domains, and in related machine learning fields.

References

[1] J. Aasi et al. Advanced LIGO. *Class. Quant. Grav.*, 32:074001, 2015. doi: 10.1088/0264-9381/32/7/074001.

[2] J. Aasi et al. Advanced LIGO. *Class. Quant. Grav.*, 32:074001, 2015. doi: 10.1088/0264-9381/32/7/074001.

[3] A. G. Abac et al. Observation of Gravitational Waves from the Coalescence of a 2.5–4.5 $M_\odot$ Compact Object and a Neutron Star. *Astrophys. J. Lett.*, 970(2):L34, 2024. doi: 10.3847/2041-8213/ad5beb.

[4] B. Abbott et al. Observation of Gravitational Waves from a Binary Black Hole Merger. *Phys. Rev. Lett.*, 116(6):061102, 2016. doi: 10.1103/PhysRevLett.116.061102.

[5] B. Abbott et al. GW190425: Observation of a Compact Binary Coalescence with Total Mass $\sim 3.4 M_\odot$. *Astrophys. J. Lett.*, 892(1):L3, 2020. doi: 10.3847/2041-8213/ab75f5.

[6] B. P. Abbott et al. Multi-messenger Observations of a Binary Neutron Star Merger. *Astrophys. J. Lett.*, 848(2):L12, 2017. doi: 10.3847/2041-8213/aa91c9.

[7] B. P. Abbott et al. A gravitational-wave standard siren measurement of the Hubble constant. *Nature*, 551(7678):85–88, 2017. doi: 10.1038/nature24471.

[8] B. P. Abbott et al. GW170814: A Three-Detector Observation of Gravitational Waves from a Binary Black Hole Coalescence. *Phys. Rev. Lett.*, 119(14):141101, 2017. doi: 10.1103/PhysRevLett.119.141101.

[9] B. P. Abbott et al. Multi-messenger Observations of a Binary Neutron Star Merger. *Astrophys. J. Lett.*, 848(2):L12, 2017. doi: 10.3847/2041-8213/aa91c9.

[10] B. P. Abbott et al. Gravitational Waves and Gamma-rays from a Binary Neutron Star Merger: GW170817 and GRB 170817A. *Astrophys. J. Lett.*, 848(2):L13, 2017. doi: 10.3847/2041-8213/aa920c.

[11] B. P. Abbott et al. GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral. *Phys. Rev. Lett.*, 119(16):161101, 2017. doi: 10.1103/PhysRevLett.119.161101.

[12] B. P. Abbott et al. GW170817: Measurements of neutron star radii and equation of state. *Phys. Rev. Lett.*, 121(16):161101, 2018. doi: 10.1103/PhysRevLett.121.161101.

[13] B. P. Abbott et al. Properties of the binary neutron star merger GW170817. *Phys. Rev.*, X9(1): 011001, 2019. doi: 10.1103/PhysRevX.9.011001.

[14] B. P. Abbott et al. Tests of General Relativity with GW170817. *Phys. Rev. Lett.*, 123(1):011102, 2019. doi: 10.1103/PhysRevLett.123.011102.

[15] B. P. Abbott et al. 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(3):031040, 2019. doi: 10.1103/PhysRevX.9.031040.

[16] B. P. Abbott et al. Low-latency Gravitational-wave Alerts for Multimessenger Astronomy during the Second Advanced LIGO and Virgo Observing Run. *Astrophys. J.*, 875(2):161, 2019. doi: 10.3847/1538-4357/ab0e8f.

[17] B. P. Abbott et al. Prospects for observing and localizing gravitational-wave transients with Advanced LIGO, Advanced Virgo and KAGRA. *Living Rev. Rel.*, 23(1):3, 2020. doi: 10.1007/s41114-020-00026-9.

[18] B. P. Abbott et al. A guide to LIGO–Virgo detector noise and extraction of transient gravitational-wave signals. *Class. Quant. Grav.*, 37(5):055002, 2020. doi: 10.1088/1361-6382/ab685e.

[19] B. P. Abbott et al. A Gravitational-wave Measurement of the Hubble Constant Following the Second Observing Run of Advanced LIGO and Virgo. *Astrophys. J.*, 909(2):218, 2021. doi: 10.3847/1538-4357/abdcb7.

[20] R. Abbott et al. Open data from the first and second observing runs of Advanced LIGO and Advanced Virgo. *SoftwareX*, 13:100658, 2021. doi: 10.1016/j.softx.2021.100658.

[21] R. Abbott et al. Population Properties of Compact Objects from the Second LIGO-Virgo Gravitational-Wave Transient Catalog. *Astrophys. J. Lett.*, 913(1):L7, 2021. doi: 10.3847/2041-8213/abe949.

[22] R. Abbott et al. 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. doi: 10.1103/PhysRevX.11.021053.

[23] R. Abbott et al. 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. doi: 10.1103/PhysRevX.11.021053.

[24] R. Abbott et al. Tests of general relativity with binary black holes from the second LIGO-Virgo gravitational-wave transient catalog. *Phys. Rev. D*, 103(12):122002, 2021. doi: 10.1103/PhysRevD.103.122002.

[25] R. Abbott et al. Tests of General Relativity with GWTC-3. *arXiv preprint arXiv:2112.06861*, 12 2021.

[26] R. Abbott et al. First joint observation by the underground gravitational-wave detector KAGRA with GEO 600. *PTEP*, 2022(6):063F01, 2022. doi: 10.1093/ptep/ptac073.

[27] R. Abbott et al. Population of Merging Compact Binaries Inferred Using Gravitational Waves through GWTC-3. *Phys. Rev. X*, 13(1):011048, 2023. doi: 10.1103/PhysRevX.13.011048.

[28] R. Abbott et al. Constraints on the Cosmic Expansion History from GWTC–3. *Astrophys. J.*, 949(2):76, 2023. doi: 10.3847/1538-4357/ac74bb.

[29] R. Abbott et al. GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo during the Second Part of the Third Observing Run. *Phys. Rev. X*, 13(4):041039, 2023. doi: 10.1103/PhysRevX.13.041039.

[30] R. Abbott et al. Population of Merging Compact Binaries Inferred Using Gravitational Waves through GWTC-3. *Phys. Rev. X*, 13(1):011048, 2023. doi: 10.1103/PhysRevX.13.011048.

[31] R. Abbott et al. GWTC-2.1: Deep extended catalog of compact binary coalescences observed by LIGO and Virgo during the first half of the third observing run. *Phys. Rev. D*, 109(2):022001, 2024. doi: 10.1103/PhysRevD.109.022001.

[32] F. Acernese et al. Advanced Virgo: a second-generation interferometric gravitational wave detector. *Class. Quant. Grav.*, 32(2):024001, 2015. doi: 10.1088/0264-9381/32/2/024001.

[33] F. Acernese et al. Advanced Virgo: a second-generation interferometric gravitational wave detector. *Class. Quant. Grav.*, 32(2):024001, 2015. doi: 10.1088/0264-9381/32/2/024001.

[34] T. Adams, D. Buskulic, V. Germain, G. M. Guidi, F. Marion, M. Montani, B. Mours, F. Piergiovanni, and G. Wang. Low-latency analysis pipeline for compact binary coalescences in the advanced gravitational wave detector era. *Class. Quant. Grav.*, 33(17):175012, 2016. doi: 10.1088/0264-9381/33/17/175012.

[35] T. Akutsu et al. Overview of KAGRA: Detector design and construction history. *PTEP*, 2021 (5):05A101, 2021. doi: 10.1093/ptep/ptaa125.

[36] M. S. Albergo, G. Kanwar, and P. E. Shanahan. Flow-based generative models for Markov chain Monte Carlo in lattice field theory. *Phys. Rev. D*, 100(3):034515, 2019. doi: 10.1103/PhysRevD.100.034515.

[37] J. Alvey, U. Bhardwaj, S. Nissanke, and C. Weniger. What to do when things get crowded? Scalable joint analysis of overlapping gravitational wave signals. *arXiv preprint arXiv:2308.06318*, 8 2023.

[38] P. Amaro-Seoane, H. Audley, S. Babak, J. Baker, E. Barausse, P. Bender, E. Berti, P. Binetruy, M. Born, D. Bortoluzzi, et al. Laser interferometer space antenna. *arXiv preprint arXiv:1702.00786*, 2017.

[39] J. Ambrose. Computerized transverse axial scanning (tomography): Part 2. clinical application. *The British journal of radiology*, 46(552):1023–1047, 1973.

[40] G. Ashton, N. Bernstein, J. Buchner, X. Chen, G. Csányi, A. Fowlie, F. Feroz, M. Griffiths, W. Handley, M. Habeck, et al. Nested sampling for physical scientists. *Nature Reviews Methods Primers*, 2(1):39, 2022.

[41] G. Ashton et al. BILBY: A user-friendly Bayesian inference library for gravitational-wave astronomy. *Astrophys. J. Suppl.*, 241(2):27, 2019. doi: 10.3847/1538-4365/ab06fc.

[42] Y. Aso, Y. Michimura, K. Somiya, M. Ando, O. Miyakawa, T. Sekiguchi, D. Tatsumi, and H. Yamamoto. Interferometer design of the KAGRA gravitational wave detector. *Phys. Rev. D*, 88(4):043007, 2013. doi: 10.1103/PhysRevD.88.043007.

[43] A. G. Baydin, L. Shao, W. Bhimji, L. Heinrich, L. Meadows, J. Liu, A. Munk, S. Naderiparizi, B. Gram-Hansen, G. Louppe, et al. Etalumis: Bringing probabilistic programming to scientific simulators at scale. In *Proceedings of the international conference for high performance computing, networking, storage and analysis*, pages 1–24, 2019.

[44] M. A. Beaumont, W. Zhang, and D. J. Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

[45] M. A. Beaumont, J.-M. Cornuet, J.-M. Marin, and C. P. Robert. Adaptive approximate bayesian computation. *Biometrika*, 96(4):983–990, 2009.

[46] U. Bhardwaj, J. Alvey, B. K. Miller, S. Nissanke, and C. Weniger. Sequential simulation-based inference for gravitational wave signals. *Phys. Rev. D*, 108(4):042004, 2023. doi: 10.1103/PhysRevD.108.042004.

[47] 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(12):121102, 2015. doi: 10.1103/PhysRevLett.115.121102.

[48] L. Blanchet. Gravitational Radiation from Post-Newtonian Sources and Inspiralling Compact Binaries. *Living Rev. Rel.*, 17:2, 2014. doi: 10.12942/lrr-2014-2.

[49] M. G. B. Blum and O. François. Non-linear regression models for approximate bayesian computation. *Stat. Comput.*, 20(1):63–73, 2010. doi: 10.1007/s11222-009-9116-0. URL https://doi.org/10.1007/s11222-009-9116-0.

[50] A. Bohé, M. Hannam, S. Husa, F. Ohme, M. Pürrer, and P. Schmidt. PhenomPv2 – technical notes for the LAL implementation. *LIGO Technical Document, LIGO-T1500602-v4*, 2016. URL https://dcc.ligo.org/LIGO-T1500602/public.

[51] A. Bohé et al. Improved effective-one-body model of spinning, nonprecessing binary black holes for the era of gravitational-wave astrophysics with advanced detectors. *Phys. Rev. D*, 95 (4):044028, 2017. doi: 10.1103/PhysRevD.95.044028.

[52] D. Boyda, G. Kanwar, S. Racanière, D. J. Rezende, M. S. Albergo, K. Cranmer, D. C. Hackett, and P. E. Shanahan. Sampling using su (n) gauge equivariant flows. *Physical Review D*, 103 (7):074504, 2021.

[53] M. Branchesi et al. Science with the Einstein Telescope: a comparison of different designs. *JCAP*, 07:068, 2023. doi: 10.1088/1475-7516/2023/07/068.

[54] J. Brehmer, G. Louppe, J. Pavez, and K. Cranmer. Mining gold from implicit models to improve likelihood-free inference. *Proceedings of the National Academy of Sciences*, 117(10): 5242–5249, 2020.

[55] A. Buonanno and T. Damour. Effective one-body approach to general relativistic two-body dynamics. *Phys. Rev. D*, 59:084006, 1999. doi: 10.1103/PhysRevD.59.084006.

[56] K. Cannon, A. Chapman, C. Hanna, D. Keppel, A. C. Searle, and A. J. Weinstein. Singular value decomposition applied to compact binary coalescence gravitational-wave signals. *Phys. Rev. D*, 82:044025, 2010. doi: 10.1103/PhysRevD.82.044025.

[57] K. Cannon et al. Toward Early-Warning Detection of Gravitational Waves from Compact Binary Coalescence. *Astrophys. J.*, 748:136, 2012. doi: 10.1088/0004-637X/748/2/136.

[58] K. Cannon et al. GstLAL: A software framework for gravitational wave discovery. *arXiv preprint arXiv:2010.05082*, 10 2020.

[59] P. Cannon, D. Ward, and S. M. Schmon. Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*, 2022.

[60] C. Chatterjee and L. Wen. Premerger Sky Localization of Gravitational Waves from Binary Neutron Star Mergers Using Deep Learning. *Astrophys. J.*, 959(2):76, 2023. doi: 10.3847/1538-4357/accffb.

[61] C. Chatterjee, L. Wen, K. Vinsen, M. Kovalam, and A. Datta. Using Deep Learning to Localize Gravitational Wave Sources. *Phys. Rev. D*, 100(10):103025, 2019. doi: 10.1103/PhysRevD.100.103025.

[62] C. Chatterjee, M. Kovalam, L. Wen, D. Beveridge, F. Diakogiannis, and K. Vinsen. Rapid Localization of Gravitational Wave Sources from Compact Binary Coalescences Using Deep Learning. *Astrophys. J.*, 959(1):42, 2023. doi: 10.3847/1538-4357/ad08b7.

[63] S. S. Chaudhary et al. Low-latency gravitational wave alert products and their performance at the time of the fourth LIGO-Virgo-KAGRA observing run. *Proc. Nat. Acad. Sci.*, 121(18): e2316474121, 2024. doi: 10.1073/pnas.2316474121.

[64] Y. Chen, D. Zhang, M. U. Gutmann, A. Courville, and Z. Zhu. Neural approximate sufficient statistics for implicit models. In *Ninth International Conference on Learning Representations 2021*, 2021.

[65] S. Chib and E. Greenberg. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4):327–335, 1995.

[66] Q. Chu et al. SPIIR online coherent pipeline to search for gravitational waves from compact binary coalescences. *Phys. Rev. D*, 105(2):024023, 2022. doi: 10.1103/PhysRevD.105.024023.

[67] A. J. K. Chua and M. Vallisneri. Learning Bayesian posteriors with neural networks for gravitational-wave inference. *Phys. Rev. Lett.*, 124(4):041102, 2020. doi: 10.1103/PhysRevLett.124.041102.

[68] T. Cohen and M. Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016.

[69] N. J. Cornish. Fast Fisher Matrices and Lazy Likelihoods. *arXiv preprint arXiv:1007.4820*, 7 2010.

[70] N. J. Cornish. Heterodyned likelihood for rapid gravitational wave parameter inference. *Phys. Rev. D*, 104(10):104054, 2021. doi: 10.1103/PhysRevD.104.104054.

[71] N. J. Cornish. Rapid and Robust Parameter Inference for Binary Mergers. *Phys. Rev. D*, 103 (10):104057, 2021. doi: 10.1103/PhysRevD.103.104057.

[72] N. J. Cornish and J. Crowder. LISA data analysis using MCMC methods. *Phys. Rev. D*, 72: 043005, 2005. doi: 10.1103/PhysRevD.72.043005.

[73] N. J. Cornish and T. B. Littenberg. BayesWave: Bayesian Inference for Gravitational Wave Bursts and Instrument Glitches. *Class. Quant. Grav.*, 32(13):135012, 2015. doi: 10.1088/0264-9381/32/13/135012.

[74] S. Coughlin, M. Zevin, S. Bahaadini, N. Rohani, S. Allen, C. Berry, K. Crowston, M. Harandi, C. Jackson, V. Kalogera, A. Katsaggelos, V. Noroozi, C. Osterlund, O. Patane, J. Smith, S. Soni, and L. Trouille. Gravity Spy Machine Learning Classifications of LIGO Glitches from Observing Runs O1, O2, O3a, and O3b, Nov. 2021. URL https://doi.org/10.5281/zenodo.5649212.

[75] D. A. Coulter et al. Swope Supernova Survey 2017a (SSS17a), the Optical Counterpart to a Gravitational Wave Source. *Science*, 358:1556, 2017. doi: 10.1126/science.aap9811.

[76] K. Cranmer, J. Pavez, and G. Louppe. Approximating likelihood ratios with calibrated discriminative classifiers. *arXiv preprint arXiv:1506.02169*, 2015.

[77] K. Cranmer, J. Brehmer, and G. Louppe. The frontier of simulation-based inference. *Proc. Nat. Acad. Sci.*, 117(48):30055–30062, 2020. doi: 10.1073/pnas.1912789117.

[78] K. Cranmer, J. Brehmer, and G. Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[79] M. Crisostomi, K. Dey, E. Barausse, and R. Trotta. Neural posterior estimation with guaranteed exact coverage: The ringdown of GW150914. *Phys. Rev. D*, 108(4):044029, 2023. doi: 10.1103/PhysRevD.108.044029.

[80] E. Cuoco, J. Powell, M. Cavaglià, K. Ackley, M. Bejger, C. Chatterjee, M. Coughlin, S. Coughlin, P. Easter, R. Essick, et al. Enhancing gravitational-wave science with machine learning. *Machine Learning: Science and Technology*, 2(1):011002, 5 2020. doi: 10.1088/2632-2153/abb93a.

[81] D. Davis et al. LIGO detector characterization in the second and third observing runs. *Class. Quant. Grav.*, 38(13):135014, 2021. doi: 10.1088/1361-6382/abfd85.

[82] 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(24): 241103, 2021. doi: 10.1103/PhysRevLett.127.241103.

[83] M. Dax, S. R. Green, J. Gair, M. Deistler, B. Schölkopf, and J. H. Macke. Group equivariant neural posterior estimation. In *International Conference on Learning Representations*, 11 2022.

[84] M. Dax, S. R. Green, J. Gair, M. Pürrer, J. Wildberger, J. H. Macke, A. Buonanno, and B. Schölkopf. Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference. *Phys. Rev. Lett.*, 130(17):171403, 2023. doi: 10.1103/PhysRevLett.130.171403.

[85] M. Deistler, P. J. Goncalves, and J. H. Macke. Truncated proposals for scalable and hassle-free simulation-based inference. *Advances in Neural Information Processing Systems*, 35: 23135–23149, 2022.

[86] A. Delaunoy, A. Wehenkel, T. Hinderer, S. Nissanke, C. Weniger, A. R. Williamson, and G. Louppe. Lightning-Fast Gravitational Wave Parameter Inference through Neural Amortization. In *Third Workshop on Machine Learning and the Physical Sciences*, 10 2020.

[87] A. Delaunoy, A. Wehenkel, T. Hinderer, S. Nissanke, C. Weniger, A. R. Williamson, and G. Louppe. Lightning-fast gravitational wave parameter inference through neural amortization. *Third Workshop on Machine Learning and the Physical Sciences (NeurIPS 2020)*, 2020.

[88] T. Dietrich, S. Bernuzzi, and W. Tichy. Closed-form tidal approximants for binary neutron star gravitational waveforms constructed from high-resolution numerical relativity simulations. *Phys. Rev. D*, 96(12):121501, 2017. doi: 10.1103/PhysRevD.96.121501.

[89] T. Dietrich et al. Matter imprints in waveform models for neutron star binaries: Tidal and self-spin effects. *Phys. Rev. D*, 99(2):024029, 2019. doi: 10.1103/PhysRevD.99.024029.

[90] K. L. Dooley et al. GEO 600 and the GEO-HF upgrade program: successes and challenges. *Class. Quant. Grav.*, 33:075009, 2016. doi: 10.1088/0264-9381/33/7/075009.

[91] C. C. Drovandi, C. Grazian, K. Mengersen, and C. Robert. Approximating the likelihood in approximate bayesian computation, 2018.

[92] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[93] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios. Neural spline flows. In *Advances in Neural Information Processing Systems*, pages 7509–7520, 2019.

[94] C. Durkan, A. Bekasov, I. Murray, and G. Papamakarios. nflows: normalizing flows in PyTorch, Nov. 2020. URL https://doi.org/10.5281/zenodo.4296287.

[95] C. Durkan, I. Murray, and G. Papamakarios. On contrastive learning for likelihood-free inference. In *International conference on machine learning*, pages 2771–2781. PMLR, 2020.

[96] T. D. P. Edwards, K. W. K. Wong, K. K. H. Lam, A. Coogan, D. Foreman-Mackey, M. Isi, and A. Zimmerman. ripple: Differentiable and Hardware-Accelerated Waveforms for Gravitational Wave Data Analysis. *arXiv preprint arXiv:2302.05329*, 2 2023.

[97] A. Einstein. Approximative integration of the field equations of gravitation. *Sitzungsber. Preuss. Akad. Wiss. Berlin (Math. Phys.)*, 1916(688-696):1, 1916.

[98] B. Farr, E. Ochsner, W. M. Farr, and R. O'Shaughnessy. A more effective coordinate system for parameter estimation of precessing compact binaries from gravitational waves. *Phys. Rev. D*, 90(2):024018, 2014. doi: 10.1103/PhysRevD.90.024018.

[99] P. Fearnhead and D. Prangle. Constructing summary statistics for approximate bayesian computation: semi-automatic approximate bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474, 2012.

[100] L. S. Finn. Detection, measurement and gravitational radiation. *Phys. Rev. D*, 46:5236–5249, 1992. doi: 10.1103/PhysRevD.46.5236.

[101] E. E. Flanagan and S. A. Hughes. The Basics of gravitational wave theory. *New J. Phys.*, 7: 204, 2005. doi: 10.1088/1367-2630/7/1/204.

[102] J. Friedman. On multivariate goodness-of-fit and two-sample testing. In *Conference on Statistical Problems in Particle Physics, Astrophysics and Cosmology*, 2004.

[103] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith. Bayesian parameter estimation using conditional variational autoencoders for gravitational-wave astronomy. *Nature Phys.*, 18(1):112–117, 2022. doi: 10.1038/s41567-021-01425-7.

[104] T. D. Gebhard, J. Wildberger, M. Dax, A. Kofler, D. Angerhausen, S. P. Quanz, and B. Schölkopf. Flow matching for atmospheric retrieval of exoplanets: Where reliability meets adaptive noise levels. *arXiv preprint arXiv:2410.21477*, 2024.

[105] T. Geffner, G. Papamakarios, and A. Mnih. Compositional score modeling for simulation-based inference. *Proceedings of the 40th International Conference on Machine Learning*, 202: 11098–11116, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/geffner23a.html.

[106] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian data analysis*. CRC press, 2013.

[107] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[108] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[109] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[110] S. R. Green and J. Gair. Complete parameter inference for GW150914 using deep learning. *Mach. Learn. Sci. Tech.*, 2(3):03LT01, 2021. doi: 10.1088/2632-2153/abfaed.

[111] S. R. Green, C. Simpson, and J. Gair. Gravitational-wave parameter estimation with autoregressive neural network flows. *Phys. Rev. D*, 102(10):104057, 2020. doi: 10.1103/PhysRevD.102.104057.

[112] D. Greenberg, M. Nonnenmacher, and J. Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pages 2404–2414. PMLR, 2019.

[113] N. Gupte et al. Evidence for eccentricity in the population of binary black holes observed by LIGO-Virgo-KAGRA. *arXiv preprint arXiv:2404.14286*, 4 2024.

[114] M. U. Gutmann and J. Corander. Bayesian optimization for likelihood-free inference of simulator-based statistical models. *The Journal of Machine Learning Research*, 17(1):4256–4302, 2016.

[115] M. Hannam, P. Schmidt, A. Bohé, L. Haegel, S. Husa, F. Ohme, G. Pratten, and M. Pürrer. Simple model of complete precessing black-hole-binary gravitational waveforms. *Phys. Rev. Lett.*, 113:151101, Oct 2014. doi: 10.1103/PhysRevLett.113.151101. URL https://link.aps.org/doi/10.1103/PhysRevLett.113.151101.

[116] M. Hannam et al. General-relativistic precession in a black-hole binary. *Nature*, 610(7933):652–655, 2022. doi: 10.1038/s41586-022-05212-z.

[117] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 04 1970. ISSN 0006-3444. doi: 10.1093/biomet/57.1.97. URL https://doi.org/10.1093/biomet/57.1.97.

[118] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[119] J. Hermans, V. Begy, and G. Louppe. Likelihood-free mcmc with approximate likelihood ratios. In *Proceedings of the 37th International Conference on Machine Learning*, volume 98 of *Proceedings of Machine Learning Research*. PMLR, 2020.

[120] J. Hermans, A. Delaunoy, F. Rozet, A. Wehenkel, V. Begy, and G. Louppe. A trust crisis in simulation-based inference? your posterior approximations can be unfaithful. *arXiv preprint arXiv:2110.06581*, 2021.

[121] F. Hernandez Vivanco, R. Smith, E. Thrane, P. D. Lasky, C. Talbot, and V. Raymond. Measuring the neutron star equation of state with gravitational waves: The first forty binary neutron star merger observations. *Phys. Rev. D*, 100:103009, Nov 2019. doi: 10.1103/PhysRevD.100.103009. URL https://link.aps.org/doi/10.1103/PhysRevD.100.103009.

[122] S. R. Hinton. ChainConsumer. *The Journal of Open Source Software*, 1:00045, Aug. 2016. doi: 10.21105/joss.00045.

[123] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[124] G. N. Hounsfield. Computerized transverse axial scanning (tomography): Part 1. description of system. *The British journal of radiology*, 46(552):1016–1022, 1973.

[125] Q. Hu and J. Veitch. Rapid Premerger Localization of Binary Neutron Stars in Third-generation Gravitational-wave Detectors. *Astrophys. J. Lett.*, 958(2):L43, 2023. doi: 10.3847/2041-8213/ad0ed4.

[126] R. A. Hulse and J. H. Taylor. Discovery of a pulsar in a binary system. *Astrophysical Journal, vol. 195, Jan. 15, 1975, pt. 2, p. L51-L53.*, 195:L51–L53, 1975.

[127] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9 (3):90–95, 2007. doi: 10.1109/MCSE.2007.55.

[128] R. Izbicki, A. Lee, and C. Schafer. High-dimensional density ratio estimation with extensions to approximate likelihood computation. In *Artificial intelligence and statistics*, pages 420–429. PMLR, 2014.

[129] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28:2017–2025, 2015.

[130] B. Jiang, T.-y. Wu, C. Zheng, and W. H. Wong. Learning summary statistic for approximate bayesian computation via deep neural network. *Statistica Sinica*, pages 1595–1618, 2017.

[131] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[132] G. Kanwar, M. S. Albergo, D. Boyda, K. Cranmer, D. C. Hackett, S. Racanière, D. J. Rezende, and P. E. Shanahan. Equivariant flow-based sampling for lattice gauge theory. *Phys. Rev. Lett.*, 125(12):121601, 2020. doi: 10.1103/PhysRevLett.125.121601.

[133] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

[134] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Pürrer, X. J. Forteza, and A. Bohé. Frequency-domain gravitational waves from nonprecessing black-hole binaries. II. A phenomenological model for the advanced detector era. *Phys. Rev.*, D93(4):044007, 2016. doi: 10.1103/PhysRevD.93.044007.

[135] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*, 2015.

[136] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling. Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems*, pages 4743–4751, 2016.

[137] A. Kolmus, G. Baltus, J. Janquart, T. van Laarhoven, S. Caudill, and T. Heskes. Fast sky localization of gravitational waves using deep learning seeded importance sampling. *Phys. Rev. D*, 106(2):023032, 2022. doi: 10.1103/PhysRevD.106.023032.

[138] A. Kolmus, J. Janquart, T. Baka, T. van Laarhoven, C. Van Den Broeck, and T. Heskes. Tuning neural posterior estimation for gravitational wave inference. *arXiv preprint arXiv:2403.02443*, 3 2024.

[139] A. Kong. A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Rep*, 348, 1992.

[140] M. Kovalam, M. A. K. Patwary, A. K. Sreekumar, L. Wen, F. H. Panther, and Q. Chu. Early Warnings of Binary Neutron Star Coalescence Using the SPIIR Search. *Astrophys. J. Lett.*, 927 (1):L9, 2022. doi: 10.3847/2041-8213/ac5687.

[141] P. G. Krastev, K. Gill, V. A. Villar, and E. Berger. Detection and Parameter Estimation of Gravitational Waves from Binary Neutron-Star Mergers in Real LIGO Data using Deep Learning. *Phys. Lett. B*, 815:136161, 2021. doi: 10.1016/j.physletb.2021.136161.

[142] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[143] J. Lange, R. O'Shaughnessy, and M. Rizzo. Rapid and accurate parameter inference for coalescing, precessing compact binaries. *arXiv preprint arXiv:1805.10457*, 5 2018.

[144] K. Leyde, S. R. Green, A. Toubiana, and J. Gair. Gravitational wave populations and cosmology with neural posterior estimation. *Phys. Rev. D*, 109(6):064056, 2024. doi: 10.1103/PhysRevD.109.064056.

[145] LIGO Scientific Collaboration. LIGO Algorithm Library - LALSuite. free software (GPL), 2018.

[146] Ligo Scientific Collaboration, VIRGO Collaboration, and Kagra Collaboration. LIGO/Virgo/KAGRA S241114bi: Identification of a GW compact binary merger candidate. *GRB Coordinates Network*, 38228:1, Nov. 2024.

[147] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991. doi: 10.1109/18.61115.

[148] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

[149] T. B. Littenberg and N. J. Cornish. Bayesian inference for spectral estimation of gravitational wave detector noise. *Phys. Rev. D*, 91(8):084034, 2015. doi: 10.1103/PhysRevD.91.084034.

[150] T. B. Littenberg and N. J. Cornish. Prototype global analysis of LISA data with multiple source types. *Phys. Rev. D*, 107(6):063004, 2023. doi: 10.1103/PhysRevD.107.063004.

[151] D. Lopez-Paz and M. Oquab. Revisiting classifier two-sample tests. In *5th International Conference on Learning Representations, ICLR*, 2017.

[152] J.-M. Lueckmann, P. J. Gonçalves, G. Bassetto, K. Öcal, M. Nonnenmacher, and J. H. Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1289–1299, 2017.

[153] J.-M. Lueckmann, G. Bassetto, T. Karaletsos, and J. H. Macke. Likelihood-free inference with emulator networks. In *Symposium on Advances in Approximate Bayesian Inference*, pages 32–53. PMLR, 2019.

[154] J.-M. Lueckmann, J. Boelts, D. Greenberg, P. Goncalves, and J. Macke. Benchmarking simulation-based inference. In *International Conference on Artificial Intelligence and Statistics*, pages 343–351. PMLR, 2021.

[155] N. Madhusudhan. Atmospheric Retrieval of Exoplanets. In H. Deeg and J. Belmonte, editors, *Handbook of Exoplanets*, pages 1–30. Springer International Publishing, Cham, 2018. ISBN 978-3-319-30648-3. doi: 10.1007/978-3-319-30648-3_104-1.

[156] M. Maggiore et al. Science Case for the Einstein Telescope. *JCAP*, 03:050, 2020. doi: 10.1088/1475-7516/2020/03/050.

[157] J. McGinn, A. Mukherjee, J. Irwin, C. Messenger, M. J. Williams, and I. S. Heng. Rapid neutron star equation of state inference with Normalising Flows. *arXiv preprint arXiv:2403.17462*, 3 2024.

[158] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6): 1087–1092, 1953.

[159] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.

[160] B. K. Miller, C. Weniger, and P. Forré. Contrastive neural ratio estimation. *Advances in Neural Information Processing Systems*, 35:3262–3278, 2022.

[161] S. Morisaki. Accelerating parameter estimation of gravitational waves from compact binary coalescence using adaptive frequency resolutions. *Phys. Rev. D*, 104(4):044062, 2021. doi: 10.1103/PhysRevD.104.044062.

[162] S. Morisaki, R. Smith, L. Tsukada, S. Sachdev, S. Stevenson, C. Talbot, and A. Zimmerman. Rapid localization and inference on compact binary coalescences with the Advanced LIGO-Virgo-KAGRA gravitational-wave detector network. *Phys. Rev. D*, 108(12):123040, 2023. doi: 10.1103/PhysRevD.108.123040.

[163] G. Morras, J. F. N. n. Siles, J. Garcia-Bellido, and E. R. Morales. False alarms induced by Gaussian noise in gravitational wave detectors. *Phys. Rev. D*, 107(2):023027, 2023. doi: 10.1103/PhysRevD.107.023027.

[164] E. R. Most and A. A. Philippov. Electromagnetic precursor flares from the late inspiral of neutron star binaries. *Mon. Not. Roy. Astron. Soc.*, 515(2):2710–2724, 2022. doi: 10.1093/mnras/stac1909.

[165] T. Müller, B. McWilliams, F. Rousselle, M. Gross, and J. Novák. Neural importance sampling. *ACM Transactions on Graphics (TOG)*, 38(5):1–19, 2019.

[166] A. H. Nitz, T. Dal Canton, D. Davis, and S. Reyes. Rapid detection of gravitational waves from compact binary mergers with PyCBC Live. *Phys. Rev. D*, 98(2):024050, 2018. doi: 10.1103/PhysRevD.98.024050.

[167] A. H. Nitz, M. Schäfer, and T. Dal Canton. Gravitational-wave Merger Forecasting: Scenarios for the Early Detection and Localization of Compact-binary Mergers with Ground-based Observatories. *Astrophys. J. Lett.*, 902:L29, 2020. doi: 10.3847/2041-8213/abbc10.

[168] F. Noé, S. Olsson, J. Köhler, and H. Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.

[169] OpenAI. GPT-4 Technical Report, 2023.

[170] S. Ossokine et al. Multipolar Effective-One-Body Waveforms for Precessing Binary Black Holes: Construction and Validation. *Phys. Rev. D*, 102(4):044055, 2020. doi: 10.1103/PhysRevD.102.044055.

[171] A. B. Owen. *Monte Carlo theory, methods and examples*. https://artowen.su.domains/mc/, 2013.

[172] B. Paige and F. Wood. Inference networks for sequential monte carlo in graphical models. In *International Conference on Machine Learning*, pages 3040–3049. PMLR, 2016.

[173] Y. Pan, A. Buonanno, A. Taracchini, L. E. Kidder, A. H. Mroué, H. P. Pfeiffer, M. A. Scheel, and B. Szilágyi. Inspiral-merger-ringdown waveforms of spinning, precessing black-hole binaries in the effective-one-body formalism. *Phys. Rev. D*, 89(8):084006, 2014. doi: 10.1103/PhysRevD.89.084006.

[174] C. Pankow, P. Brady, E. Ochsner, and R. O'Shaughnessy. Novel scheme for rapid parallel parameter estimation of gravitational waves from compact binary coalescences. *Phys. Rev. D*, 92(2):023002, 2015. doi: 10.1103/PhysRevD.92.023002.

[175] G. Papamakarios and I. Murray. Fast ε-free inference of simulation models with bayesian conditional density estimation. In *Advances in neural information processing systems*, 2016.

[176] G. Papamakarios, T. Pavlakou, and I. Murray. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, pages 2338–2347, 2017.

[177] G. Papamakarios, D. Sterratt, and I. Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 89 of *Proceedings of Machine Learning Research*, pages 837–848. PMLR, 2019.

[178] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *The Journal of Machine Learning Research*, 22(1):2617–2680, 2021.

[179] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[180] E. Payne, S. Hourihane, J. Golomb, R. Udall, D. Davis, and K. Chatziioannou. Curious case of GW200129: Interplay between spin-precession inference and data-quality issues. *Phys. Rev. D*, 106(10):104017, 2022. doi: 10.1103/PhysRevD.106.104017.

[181] H. P. Pfeiffer, L. E. Kidder, M. A. Scheel, and S. A. Teukolsky. A Multidomain spectral method for solving elliptic equations. *Comput. Phys. Commun.*, 152:253–273, 2003. doi: 10.1016/S0010-4655(02)00847-0.

[182] K. Pham, D. Nott, and S. Chaudhuri. A note on approximating abc-mcmc using flexible classifiers. *Stat*, 3, 03 2014. doi: 10.1002/sta4.56.

[183] D. Prangle, P. Fearnhead, M. P. Cox, P. J. Biggs, and N. P. French. Semi-automatic selection of summary statistics for abc model choice, 2014.

[184] G. Pratten, S. Husa, C. Garcia-Quiros, M. Colleoni, A. Ramos-Buades, H. Estelles, and R. Jaume. Setting the cornerstone for a family of models for gravitational waves from compact binaries: The dominant harmonic for nonprecessing quasicircular black holes. *Phys. Rev. D*, 102(6):064001, 2020. doi: 10.1103/PhysRevD.102.064001.

[185] G. Pratten et al. Computationally efficient models for the dominant and subdominant harmonic modes of precessing binary black holes. *Phys. Rev. D*, 103(10):104056, 2021. doi: 10.1103/PhysRevD.103.104056.

[186] M. Punturo et al. The Einstein Telescope: A third-generation gravitational wave observatory. *Class. Quant. Grav.*, 27:194002, 2010. doi: 10.1088/0264-9381/27/19/194002.

[187] M. Punturo et al. The third generation of gravitational wave observatories and their science reach. *Class. Quant. Grav.*, 27:084007, 2010. doi: 10.1088/0264-9381/27/8/084007.

[188] M. Pürrer. Frequency domain reduced order model of aligned-spin effective-one-body waveforms with generic mass-ratios and spins. *Phys. Rev. D*, 93(6):064041, 2016. doi: 10.1103/PhysRevD.93.064041.

[189] P. Ramesh, J.-M. Lueckmann, J. Boelts, Á. Tejero-Cantero, D. S. Greenberg, P. J. Goncalves, and J. H. Macke. GATSBI: Generative adversarial training for simulation-based inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=kR1hC6j48Tp.

[190] A. Ramos-Buades, A. Buonanno, H. Estellés, M. Khalil, D. P. Mihaylov, S. Ossokine, L. Pompili, and M. Shiferaw. Next generation of accurate and efficient multipolar precessing-spin effective-one-body waveforms for binary black holes. *Phys. Rev. D*, 108(12):124037, 2023. doi: 10.1103/PhysRevD.108.124037.

[191] V. Raymond, S. Al-Shammari, and A. Göttel. Simulation-based Inference for Gravitational-waves from Intermediate-Mass Binary Black Holes in Real Noise. *arXiv preprint arXiv:2406.03935*, 6 2024.

[192] D. Reitze et al. Cosmic Explorer: The U.S. Contribution to Gravitational-Wave Astronomy beyond LIGO. *Bull. Am. Astron. Soc.*, 51(7):035, 2019.

[193] D. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538, 2015.

[194] G. O. Roberts and A. F. Smith. Simple conditions for the convergence of the gibbs sampler and metropolis-hastings algorithms. *Stochastic processes and their applications*, 49(2):207–216, 1994.

[195] I. M. Romero-Shaw et al. Bayesian inference for compact binary coalescences with bilby: validation and application to the first LIGO–Virgo gravitational-wave transient catalogue. *Mon. Not. Roy. Astron. Soc.*, 499(3):3295–3319, 2020. doi: 10.1093/mnras/staa2850.

[196] S. Sachdev et al. An Early-warning System for Electromagnetic Follow-up of Gravitational-wave Events. *Astrophys. J. Lett.*, 905(2):L25, 2020. doi: 10.3847/2041-8213/abc753.

[197] M. A. Scheel, M. Boyle, T. Chu, L. E. Kidder, K. D. Matthews, and H. P. Pfeiffer. High-accuracy waveforms for binary black hole inspiral, merger, and ringdown. *Phys. Rev. D*, 79: 024003, 2009. doi: 10.1103/PhysRevD.79.024003.

[198] M. Schmitt, V. Pratz, U. Köthe, P.-C. Bürkner, and S. T. Radev. Consistency models for scalable and fast simulation-based inference. *arXiv preprint arXiv:2312.05440*, 2023.

[199] G. T. Schuster. *Seismic inversion*. Society of Exploration Geophysicists, 2017.

[200] L. Sharrock, J. Simons, S. Liu, and M. Beaumont. Sequential neural score estimation: Likelihood-free inference with conditional score based diffusion models. *Proceedings of the 41st International Conference on Machine Learning*, 235:44565–44602, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/sharrock24a.html.

[201] 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(1):015007, 2022. doi: 10.1088/2632-2153/ac3843.

[202] L. Singer. ligo.skymap. https://lscsoft.docs.ligo.org/ligo.skymap/, 2020.

[203] L. P. Singer and L. R. Price. Rapid Bayesian position reconstruction for gravitational-wave transients. *Phys. Rev. D*, 93(2):024013, 2016. doi: 10.1103/PhysRevD.93.024013.

[204] S. A. Sisson, Y. Fan, and M. Beaumont. *Handbook of approximate Bayesian computation*. CRC Press, 2018.

[205] S. A. Sisson, Y. Fan, and M. A. Beaumont. Overview of abc. In *Handbook of approximate Bayesian computation*, pages 3–54. Chapman and Hall/CRC, 2018.

[206] J. Skilling. Nested sampling for general Bayesian computation. *Bayesian Analysis*, 1(4):833 – 859, 2006. doi: 10.1214/06-BA127. URL https://doi.org/10.1214/06-BA127.

[207] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[208] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[209] Y. Song, P. Dhariwal, M. Chen, and I. Sutskever. Consistency models. In *International Conference on Machine Learning*, pages 32211–32252. PMLR, 2023.

[210] J. S. Speagle. dynesty: a dynamic nested sampling package for estimating bayesian posteriors and evidences. *Monthly Notices of the Royal Astronomical Society*, 493(3):3132–3158, Feb 2020. ISSN 1365-2966. doi: 10.1093/mnras/staa278. URL http://dx.doi.org/10.1093/mnras/staa278.

[211] N. Sridhar, J. Zrake, B. D. Metzger, L. Sironi, and D. Giannios. Shock-powered radio precursors of neutron star mergers from accelerating relativistic binary winds. *Mon. Not. Roy. Astron. Soc.*, 501(3):3184–3202, 2021. doi: 10.1093/mnras/staa3794.

[212] V. Starostin, M. Dax, A. Gerlach, A. Hinderhofer, Á. Tejero-Cantero, and F. Schreiber. Fast and reliable probabilistic reflectometry inversion with prior-amortized neural posterior estimation. *arXiv preprint arXiv:2407.18648*, 7 2024.

[213] H. Sun, K. L. Bouman, P. Tiede, J. J. Wang, S. Blunt, and D. Mawet. α-deep probabilistic inference (α-dpi): Efficient uncertainty quantification from exoplanet astrometry to black hole feature extraction. *The Astrophysical Journal*, 932(2):99, jun 2022. doi: 10.3847/1538-4357/ac6be9. URL https://dx.doi.org/10.3847/1538-4357/ac6be9.

[214] R. Sutton. The bitter lesson. http://www.incompleteideas.net/IncIdeas/BitterLesson.html, 2019. Accessed: 2024-11-25.

[215] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

[216] G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[217] A. Tejero-Cantero, J. Boelts, M. Deistler, J.-M. Lueckmann, C. Durkan, P. J. Gonçalves, D. S. Greenberg, and J. H. Macke. sbi: A toolkit for simulation-based inference. *Journal of Open Source Software*, 5(52):2505, 2020. doi: 10.21105/joss.02505.

[218] O. Thomas, R. Dutta, J. Corander, S. Kaski, and M. U. Gutmann. Likelihood-free inference by ratio estimation, 2020.

[219] E. Thrane and C. Talbot. An introduction to Bayesian inference in gravitational-wave astronomy: parameter estimation, model selection, and hierarchical models. *Publ. Astron. Soc. Austral.*, 36:e010, 2019. doi: 10.1017/pasa.2019.2. [Erratum: Publ.Astron.Soc.Austral. 37, e036 (2020)].

[220] S. T. Tokdar and R. E. Kass. Importance sampling: a review. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(1):54–60, 2010.

[221] V. Varma, S. E. Field, M. A. Scheel, J. Blackman, D. Gerosa, L. C. Stein, L. E. Kidder, and H. P. Pfeiffer. Surrogate models for precessing binary black hole simulations with unequal masses. *Phys. Rev. Research.*, 1:033015, 2019. doi: 10.1103/PhysRevResearch.1.033015.

[222] 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(6): 064045, 2019. doi: 10.1103/PhysRevD.99.064045.

[223] J. Veitch and W. Del Pozzo. Analytic marginalisation of phase parameter. *URL: https://dcc. ligo. org/LIGO-T1300326/public*, 2013.

[224] J. Veitch, V. Raymond, B. Farr, W. Farr, P. Graff, S. Vitale, et al. Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library. *Phys. Rev.*, D91(4):042003, 2015. doi: 10.1103/PhysRevD.91.042003.

[225] S. Vinciguerra, J. Veitch, and I. Mandel. Accelerating gravitational wave parameter estimation with multi-band template interpolation. *Class. Quant. Grav.*, 34(11):115006, 2017. doi: 10.1088/1361-6382/aa6d44.

[226] M. J. Wainwright, M. I. Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[227] P. Welch. The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Transactions on audio and electroacoustics*, 15(2):70–73, 1967.

[228] P. Whittle. *Hypothesis testing in time series analysis*. PhD thesis, Uppsala, 1951.

[229] J. Wildberger, M. Dax, S. Buchholz, S. R. Green, J. H. Macke, and B. Schölkopf. Flow matching for scalable simulation-based inference. *NeurIPS 2023*, 12 2023.

[230] J. Wildberger, M. Dax, S. R. Green, J. Gair, M. Pürrer, J. H. Macke, A. Buonanno, and B. Schölkopf. Adapting to noise distribution shifts in flow-based gravitational-wave inference. *Phys. Rev. D*, 107(8):084046, 2023. doi: 10.1103/PhysRevD.107.084046.

[231] M. J. Williams, J. Veitch, and C. Messenger. Nested sampling with normalizing flows for gravitational-wave inference. *Phys. Rev. D*, 103(10):103006, 2021. doi: 10.1103/PhysRevD.103.103006.

[232] K. W. K. Wong, M. Isi, and T. D. P. Edwards. Fast Gravitational-wave Parameter Estimation without Compromises. *Astrophys. J.*, 958(2):129, 2023. doi: 10.3847/1538-4357/acf5cd.

[233] S. Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466: 1102–4, 08 2010. doi: 10.1038/nature09319.

[234] T. Wouters, P. T. H. Pang, T. Dietrich, and C. Van Den Broeck. Robust parameter estimation within minutes on gravitational wave signals from binary neutron star inspirals. *arXiv preprint arXiv:2404.11397*, 4 2024.

[235] B. Zackay, L. Dai, and T. Venumadhav. Relative Binning and Fast Likelihood Evaluation for Gravitational Wave Parameter Estimation. *arXiv preprint arXiv:1806.08792*, 6 2018.

Appendix A

Real-Time Gravitational Wave Science with Neural Posterior Estimation

A.1 Training data

We perform inference over the full 15D parameter space for quasicircular binary black holes, which includes detector-frame component masses m_1, m_2, time of coalescence at geocenter t_c, reference phase ϕ_c, sky position (right ascension α and declination δ), luminosity distance d_L, inclination angle θ_{JN}, spin magnitudes a_1, a_2, tilt angles θ_1, θ_2, other spin angles ϕ_{12}, ϕ_{JL} [98], and polarization angle ψ. Priors are taken to be the same as in Ref. [110]: standard over all angles, and uniform in all other parameters, with $m_1 \geq m_2$, $m_1, m_2 \in [10, 80]$ $\mathrm{M}_\odot$, $a_1, a_2 \in [0, 0.88]$, and $t_c \in [-0.1, 0.1]$ s. We found that it was difficult to train a neural network for accurate inference over the entire relevant range of luminosity distance, so we partition the prior as shown in Tab. A.1. To perturb the signal coalescence times t_I for GNPE ,we use a uniform kernel $\kappa(\delta t_I)$ in the range $[-1, 1]$ ms.

Training data consist of labeled strain data sets (θ, d) and associated noise power spectral densities S_n. To construct the data sets, we first draw samples from the prior, $\theta \sim p(\theta)$. This is done in two stages as in Ref. [110]: first, intrinsic parameters are sampled ahead of training, and waveforms are generated and saved based on these; second, extrinsic parameters are sampled during training and applied to the waveforms, since this involves simple transformations. For our purposes, intrinsic

Observing run	Detectors	Distance range [Mpc]
O1	HL	[100, 2000]
O2	HL	[100, 2000]
		[100, 6000]
	HLV	[100, 1000]

Table A.1 Neural networks are trained based on noise from a particular observing run, number of detectors, and distance range.

Observing run	Detector	Number of PSDs
O1	H	2444
	L	2414
O2	H	4670
	L	3873
	V	864

Table A.2 Number of PSDs estimated for each observing run and detector.

parameters consist of $\theta_{\text{intrinsic}} = (m_1, m_2, \phi_c, \theta_{JN}, a_1, a_2, \theta_1, \theta_2, \phi_{12}, \phi_{JL})$ and extrinsic parameters are $\theta_{\text{extrinsic}} = (t_c, \alpha, \delta, d_L, \psi)$.

Each strain data set i consists of a waveform with additive stationary Gaussian noise, $d^{(i)} = h(\theta^{(i)}) + n^{(i)}$. This is represented in frequency domain, with $f_{\text{min}} = 20$ Hz, $f_{\text{max}} = 1024$ Hz, and $\Delta f = 0.125$ Hz, corresponding to a duration of 8 s. Waveforms are generated using the IMRPhenomPv2 frequency-domain model [115, 134, 50], which is fast (so that comparisons against standard samplers are feasible) and also includes spin-precession effects. We save intrinsic waveforms to disk in an SVD representation, which is accurate to a mismatch of $2 \cdot 10^{-5}$ for the 99.9th percentile of the data. Noise realizations are generated during training, after first sampling an associated PSD, i.e., $S_{\text{n}}^{(i)} \sim p(S_{\text{n}})$, $n^{(i)} \sim p(S_{\text{n}}^{(i)})$. We construct training sets based on 5×10^6 sets of intrinsic parameters, but by sampling extrinsic parameters and noise realizations during training, the effective size of the training set is infinite in these dimensions.

To construct the empirical PSD distributions $p(S_{\text{n}})$, we estimate PSDs using noise data from each observing run. Using `BURST_CAT2` data from GWOSC [20], we identify stretches of at least 1024 s in duration that do not overlap with events. Each PSD is estimated by taking a 1024 s stretch of data, dividing this into non-overlapping 8 s subintervals, and using the Welch "median-average" method to average PSDs estimated on each of these [224]. We use a Tukey window with a roll-off of 0.4 s. For inference, we use the same construction to estimate the PSD from detector data just prior to an event. The number of PSDs obtained for each observing run is given in Tab. A.2.

A.2 Neural network

There are two main components to our conditional density-estimation neural network, the embedding network for compressing data to a sufficiently small number of features, and the normalizing flow, which produces the Bayesian posterior from these features.

A.2.1 Embedding network

For each of the two or three detectors, the embedding network takes as input the real and imaginary parts of the whitened frequency-domain strain data, as well as the inverse amplitude spectral density (ASD). This results in 24,096 inputs per detector. We provide the inverse ASD rather than the PSD because of the more numerically stable behavior of spectral lines. We scale the ASD with a constant factor of 10^{23}.

The first embedding layer is a linear mapping that serves to drastically reduce the number of dimensions. This is initialized based on a singular value decomposition (SVD) to provide an inductive bias to facilitate training. The strain data are initially projected onto the first $n_{\text{SVD}} = 200$ (complex) singular vectors, defined based on a set of 50,000 signal waveforms drawn from the prior. The inverse ASDs are likewise mapped to n_{SVD} complex numbers, which are added to the projected strain; this projection is initialized to 0. After this initial layer, the data has therefore been projected onto $2n_{\text{detectors}}n_{\text{SVD}}$ (real) features (i.e., either 800 or 1200 components, depending on the number of detectors). There is no nonlinear activation following this layer.

Following the SVD layer is a fully-connected residual network. This consists of a sequence of two-layer residual blocks; prior to each linear mapping are batch normalization layers and ELU activation functions. We take six blocks each of 1024, 512, 256, and 128 hidden dimensions, resulting in a total of 48 hidden layers in the residual network.

A.2.2 Normalizing flow

Our normalizing flow is very similar to that of [110]. This consists of a sequence of coupling transforms, each of which transforms half of the components of a sample element-wise based on the context information and the values of the untransformed components. We use a rational-quadratic spline coupling transform [93], with the same parameters as in [110], except for the number of residual blocks, which is reduced from 10 to 5. Between the coupling transforms, the ordering of the sample components is randomized to ensure all components are sufficiently transformed by the sequence of transforms. We also increase the total number of coupling transforms to 30 from 15 in [110]. The flow therefore consists of 300 hidden layers. In total, the embedding network and the flow combined have $1.31 \cdot 10^8$ learnable parameters for $n_{\text{detectors}} = 2$ and $1.42 \cdot 10^8$ for $n_{\text{detectors}} = 3$.

The context information for the flow consists of the 128 features output from the embedding network, as well as the two or three perturbed detector coalescence times τ_I.

We also train a neural network to provide an initial estimate for t_I, which is used as a starting point for the iterative GNPE algorithm. This does not require τ_I as context, but otherwise has the same form as the main network.

A.2.3 Training

Training consists of two stages, an initial pretraining stage and a fine-tuning stage. During the pretraining stage, the SVD layer is frozen, and all noise realizations are drawn from an average PSD for the observing run and detector. This is designed to simplify early-stage training. During the fine-tuning stage, the SVD layer is unfrozen, and PSDs are randomly drawn from the empirical distribution, and noise realizations are drawn from these.

Our networks are trained for 300 epochs of pretraining and 150 epochs of fine tuning with a batch size of 4096. We use the Adam optimizer [135]. We begin training with learning rates of $3 \cdot 10^{-4}$ and $3 \cdot 10^{-5}$ in the pretraining and finetuning stages, respectively, and decrease the learning rate by a factor of 2 when the validation loss has not improved in the previous 10 epochs. For three

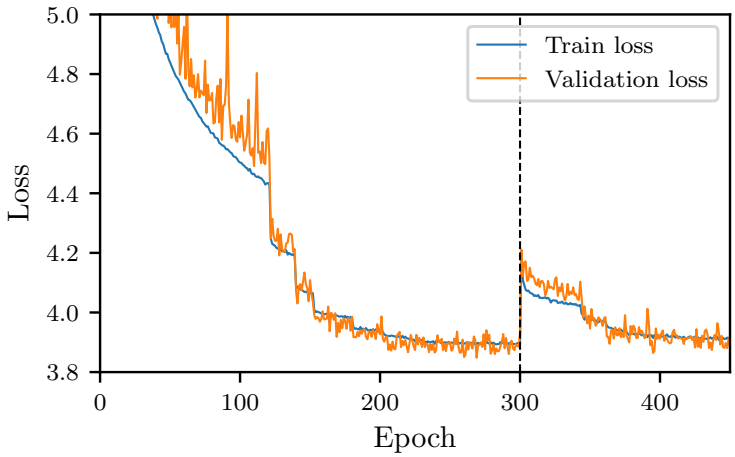


Fig. A.1 Loss as a function of training epoch for the O1 neural network. The vertical line denotes the beginning of the fine-tuning period.

detectors, we reduce the batch size to 2048 and the initial learning rates to $2 \cdot 10^{-4}$ and $2 \cdot 10^{-5}$ due to memory limitations. As shown in Fig. A.1, the loss jumps at the beginning of the fine-tuning stage. This occurs because the distribution of training data becomes much broader with the inclusion of varying noise PSDs. The final loss at the end of fine tuning is just above the pretraining loss, which indicates that the network has learned to process the varying PSDs to the same performance level of the fixed pretraining PSD. During training, we reserve 2% of the training set for validation to check for overfitting. Since the training and validation loss are in close agreement in Fig. A.1 we conclude that overfitting is minimal. Training 450 epochs with a batch size of 4096 takes roughly 10 days on a single NVIDIA A100 GPU.[1]

A.3 Effect of PSD

To give a sense of the size of the effect of the PSD on the posterior, we perform inference on GW150914, where we whiten the strain data with the correct PSD, but provide instead the PSD for GW151012 as context to the neural network. This gives a mean JSD across all parameters of 0.005 nat and a maximum JSD of 0.030 nat when compared against the DINGO result using the correct PSD. Both of these numbers significantly exceed the JSDs between DINGO and LALInference runs for all events [largest mean JSD: 0.001 nat (GW170729); largest maximum JSD: 0.006 nat (GW170104, m_1)]. This demonstrates that conditioning the neural network on the PSD is required at the level of accuracy we achieve in this study.

The worst performing parameters with the incorrect PSD are t_c, d_L, α, and δ, which have a mean JSD of 0.020 nat (see Fig. A.2). These results are consistent with the expectation that the main effect due to using the wrong PSD is to cause a change in the inferred amplitude of the signal in each detector:

[1] With an NVIDIA V100 GPU (16GB) training takes 16-18 days, and inference roughly a minute per event (rather than 20 seconds).

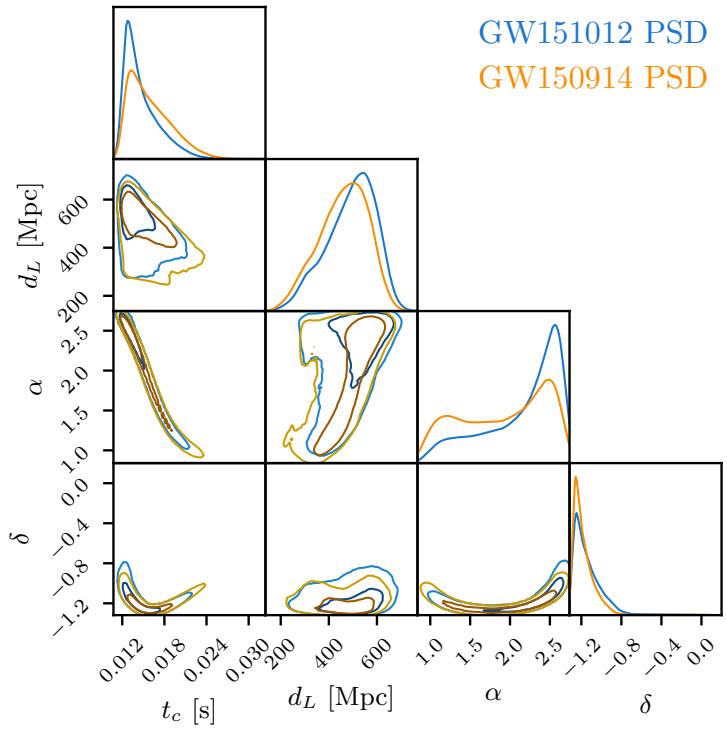


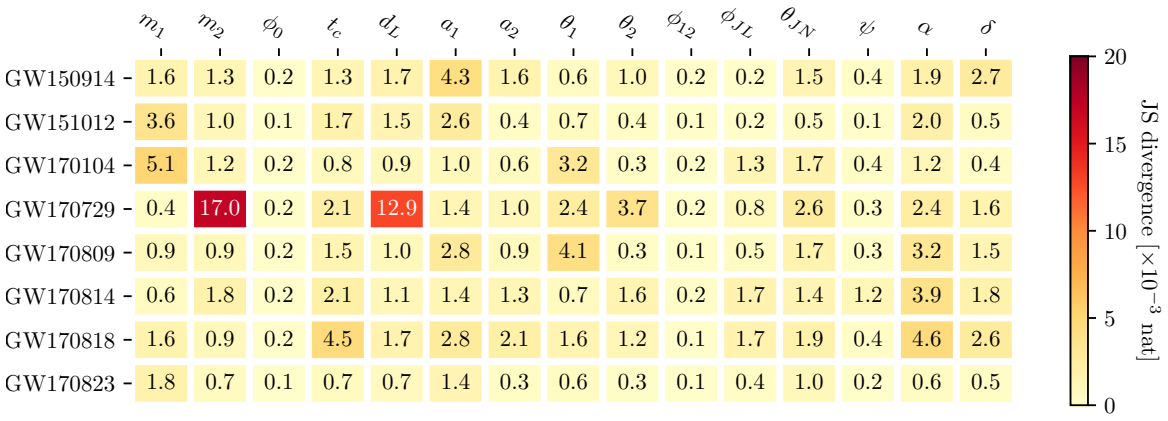
Fig. A.2 Comparison between DINGO evaluated on GW150914 using correct PSD as context, and using GW151012 PSD. These four parameters have a mean JSD of 0.020 nat.

d_L becomes biased due to the overall amplitude being off, whereas sky position and t_c become biased from relative incorrect amplitudes in the two detectors.

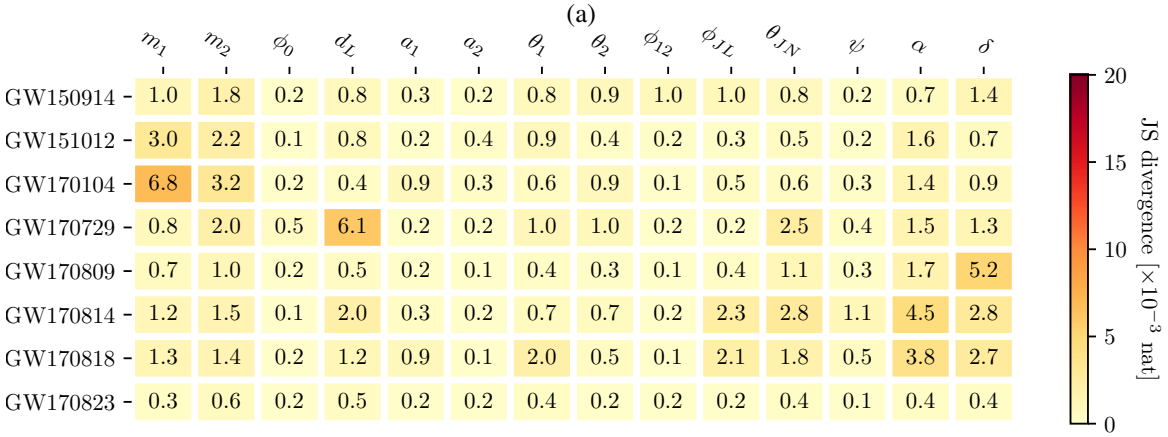
A.4 Comparisons against standard samplers

We compare our results against LALInference [224] with MCMC and nested sampling algorithms, and with Bilby [41, 195] with the dynesty [210] nested sampling algorithm. To compare as closely as possible with DINGO, we use the same data conditioning for strain data and PSDs. However, we sample in chirp mass $\mathcal{M} = (m_1 m_2)^{3/5}/(m_1 + m_2)^{1/5}$ and mass ratio $q = m_2/m_1$ instead of component masses, since this simplifies the form of the posterior and improves convergence. We also sample sky position in the detector-based azimuth/zenith reference frame rather than the (α, δ) sky frame for Bilby. This also simplifies the form of the posterior to improve sampling [195].

With standard samplers it is possible to analytically marginalize over some parameters to reduce the dimensionality of the space. This improves sampling performance, with the marginalized parameters reconstructed in post-processing. For LALInference, we marginalize over time of coalescence, and with Bilby we marginalize over time, distance, and phase. Phase marginalization is only valid in the absence of precession, but we had difficulty obtaining converged results without it. For IMRPhenomPv2 waveforms, this should not lead to a significant difference, but it should be kept in mind. For Bilby, we use `nlive=4000` and `nact=50`.



(a)



(b)

Fig. A.3 JSDs between (a) DINGO and Bilby with time-distance-phase marginalization, and (b) DINGO and LALInference nested sampling with time marginalization. (Note that LALInference does not provide time posteriors when marginalization is used.) JSDs are calculated from 10,000 samples of each distribution, using Gaussian kernel density estimation. The mean values over 100 different sample realizations are reported. The mean JSDs over all events and parameters are 0.0015 nat and 0.0010 nat for (a) and (b), respectively. As mentioned in the main text, the average JSD between LALInference runs with identical settings but different random seeds is 0.0007 nat, which sets a *practical* lower bound to the achievable values. Moreover, the DINGO framework allows us to obtain an arbitrary number of independent samples from the same distribution. This enables us to compute the JSDs between two sets of samples, that are, by construction, sampled from the same distribution. This value, 0.0002 nat, provides a *fundamental* lower bound for perfectly optimized samplers.

Event	$\mathcal{M}$ [$M_\odot$]	q	χ_{eff}	χ_p
GW150914	$31.0^{+1.5}_{-1.5}$	$0.85^{+0.13}_{-0.21}$	$-0.03^{+0.11}_{-0.12}$	$0.33^{+0.40}_{-0.27}$
	$31.1^{+1.5}_{-1.5}$	$0.85^{+0.13}_{-0.20}$	$-0.02^{+0.11}_{-0.12}$	$0.33^{+0.41}_{-0.27}$
GW151012	$18.2^{+1.1}_{-1.0}$	$0.60^{+0.35}_{-0.32}$	$0.01^{+0.20}_{-0.18}$	$0.30^{+0.40}_{-0.23}$
	$18.1^{+0.8}_{-0.7}$	$0.63^{+0.33}_{-0.34}$	$-0.00^{+0.21}_{-0.15}$	$0.30^{+0.40}_{-0.23}$
GW170104	$25.5^{+1.7}_{-1.8}$	$0.62^{+0.33}_{-0.23}$	$-0.06^{+0.16}_{-0.19}$	$0.39^{+0.34}_{-0.28}$
	$25.4^{+1.6}_{-1.6}$	$0.63^{+0.31}_{-0.22}$	$-0.07^{+0.15}_{-0.17}$	$0.38^{+0.34}_{-0.27}$
GW170729	$49.2^{+7.7}_{-8.1}$	$0.65^{+0.30}_{-0.24}$	$0.25^{+0.22}_{-0.26}$	$0.38^{+0.33}_{-0.26}$
	$49.6^{+7.6}_{-8.2}$	$0.68^{+0.28}_{-0.25}$	$0.27^{+0.22}_{-0.27}$	$0.38^{+0.32}_{-0.26}$
GW170809	$29.8^{+2.2}_{-1.9}$	$0.67^{+0.29}_{-0.24}$	$0.06^{+0.18}_{-0.16}$	$0.35^{+0.38}_{-0.27}$
	$29.9^{+2.1}_{-1.8}$	$0.68^{+0.28}_{-0.24}$	$0.07^{+0.17}_{-0.15}$	$0.35^{+0.38}_{-0.27}$
GW170814	$27.2^{+1.2}_{-1.2}$	$0.86^{+0.13}_{-0.24}$	$0.08^{+0.13}_{-0.12}$	$0.50^{+0.31}_{-0.38}$
	$27.1^{+1.1}_{-1.1}$	$0.86^{+0.13}_{-0.23}$	$0.08^{+0.12}_{-0.11}$	$0.52^{+0.29}_{-0.39}$
GW170818	$32.7^{+2.9}_{-2.8}$	$0.73^{+0.24}_{-0.27}$	$-0.02^{+0.21}_{-0.23}$	$0.51^{+0.30}_{-0.35}$
	$32.5^{+2.7}_{-2.6}$	$0.74^{+0.23}_{-0.27}$	$-0.05^{+0.20}_{-0.22}$	$0.53^{+0.28}_{-0.36}$
GW170823	$38.9^{+4.3}_{-4.1}$	$0.74^{+0.23}_{-0.28}$	$0.06^{+0.20}_{-0.20}$	$0.41^{+0.36}_{-0.31}$
	$38.9^{+4.3}_{-3.9}$	$0.73^{+0.24}_{-0.28}$	$0.06^{+0.20}_{-0.20}$	$0.41^{+0.36}_{-0.31}$

Table A.3 Comparison between DINGO (first line) and LALInference MCMC (second line) credible intervals. Median values and 90% credible intervals are quoted.

We find closest agreement with LALInference MCMC, which is what we report in the main text, and is displayed on all posterior plots. For completeness we include in Fig. A.3 comparisons against LALInference nested sampling and Bilby with phase marginalization. In general, when using phase marginalization, spin and sky position are less well recovered, as were some events. To give a sense of how JSD values translate into parameter estimates, we provide 90% credible intervals in Tab. A.3, which are all in extremely close agreement.

Deviations between posteriors obtained from DINGO and standard samplers are associated with multiple sources of error. Firstly, imperfect training of DINGO can lead to inaccurate results. Indeed, training the neural networks is challenging due to the high dimensionality of the input data to this inference problem, which inspired us to adopt the GNPE method [83] to improve convergence. While the P–P plot in Fig. 2 of the main text suggests that our networks are well converged, small deviations between posteriors for real events could arise due to the networks not being fully converged. Secondly, even the well-established standard samplers do not produce perfect posterior samples. The mean JSD between LALInference runs with identical settings but different random seeds is a factor 3.5 higher than that expected for samples from identical distributions (see Fig. A.3 for details). Thirdly, perfect agreement between (ideal implementations of) DINGO and standard samplers is only expected for data consistent with the training distribution. In reality, however, where the noise is neither perfectly stationary nor Gaussian, the measured data is slightly out-of-distribution. In those cases, there is no theoretical guarantee that DINGO extrapolates to this data in the same way as standard samplers. However, as stated in the conclusion, DINGO is not limited to stationary Gaussian noise, and we plan to lift this assumption in future work.

In the remainder of this section, we include 1D and 2D marginalized posteriors for all nontrivial parameters for all events. All DINGO posteriors are produced with 50,000 samples, except for the skymaps, which use 10,000. LALInference posteriors use all samples produced with the given sampler settings, typically of order 30,000–50,000.

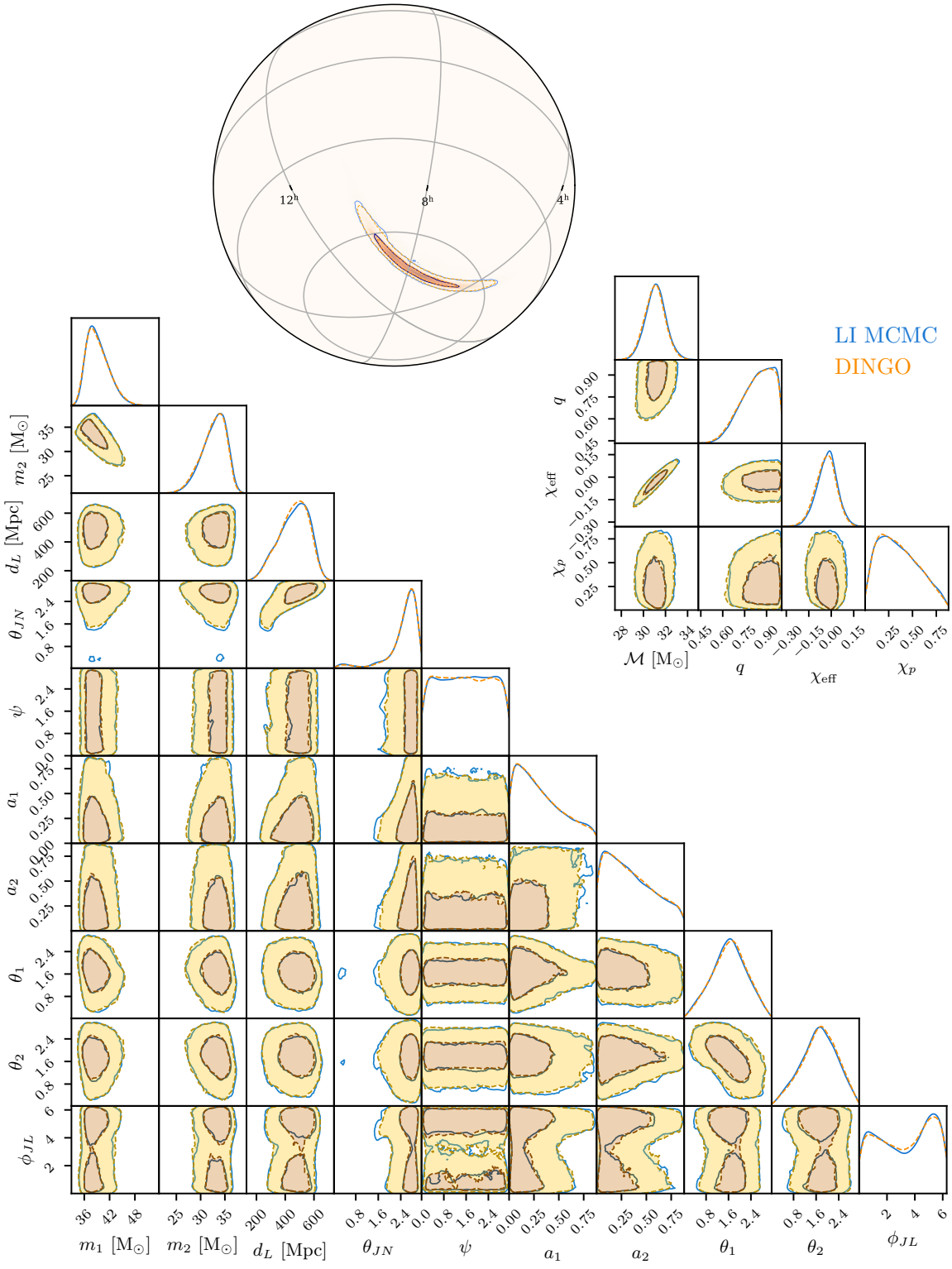


Fig. A.4 Marginalized one- and two- dimensional posterior distributions for GW150914 over a subset of parameters, comparing DINGO (orange) and LALInference MCMC (blue). Relevant derived parameters plotted on right. Contours represent 50% and 90% credible regions. Posteriors reweighted to uniform source frame distance prior.

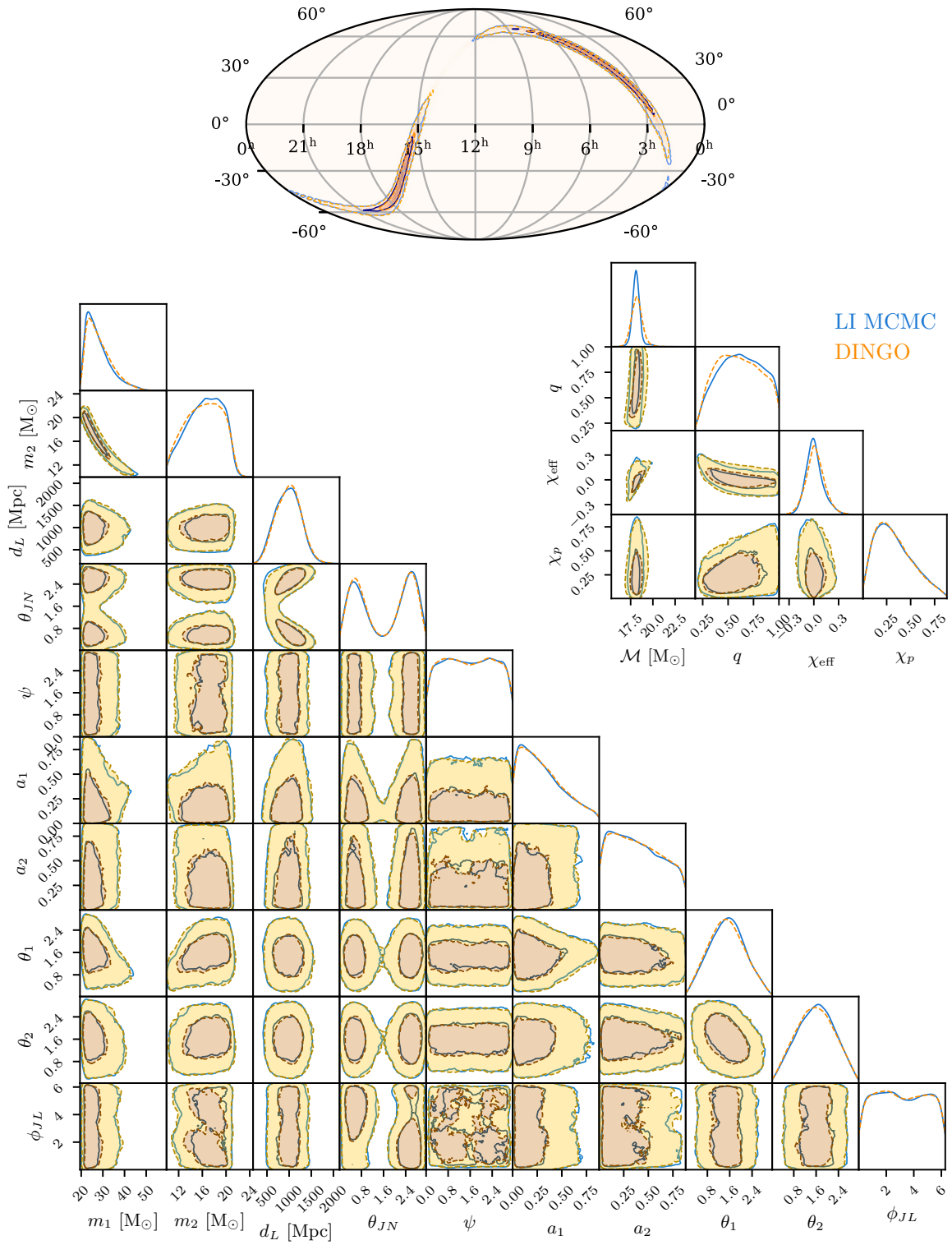


Fig. A.5 GW151012.

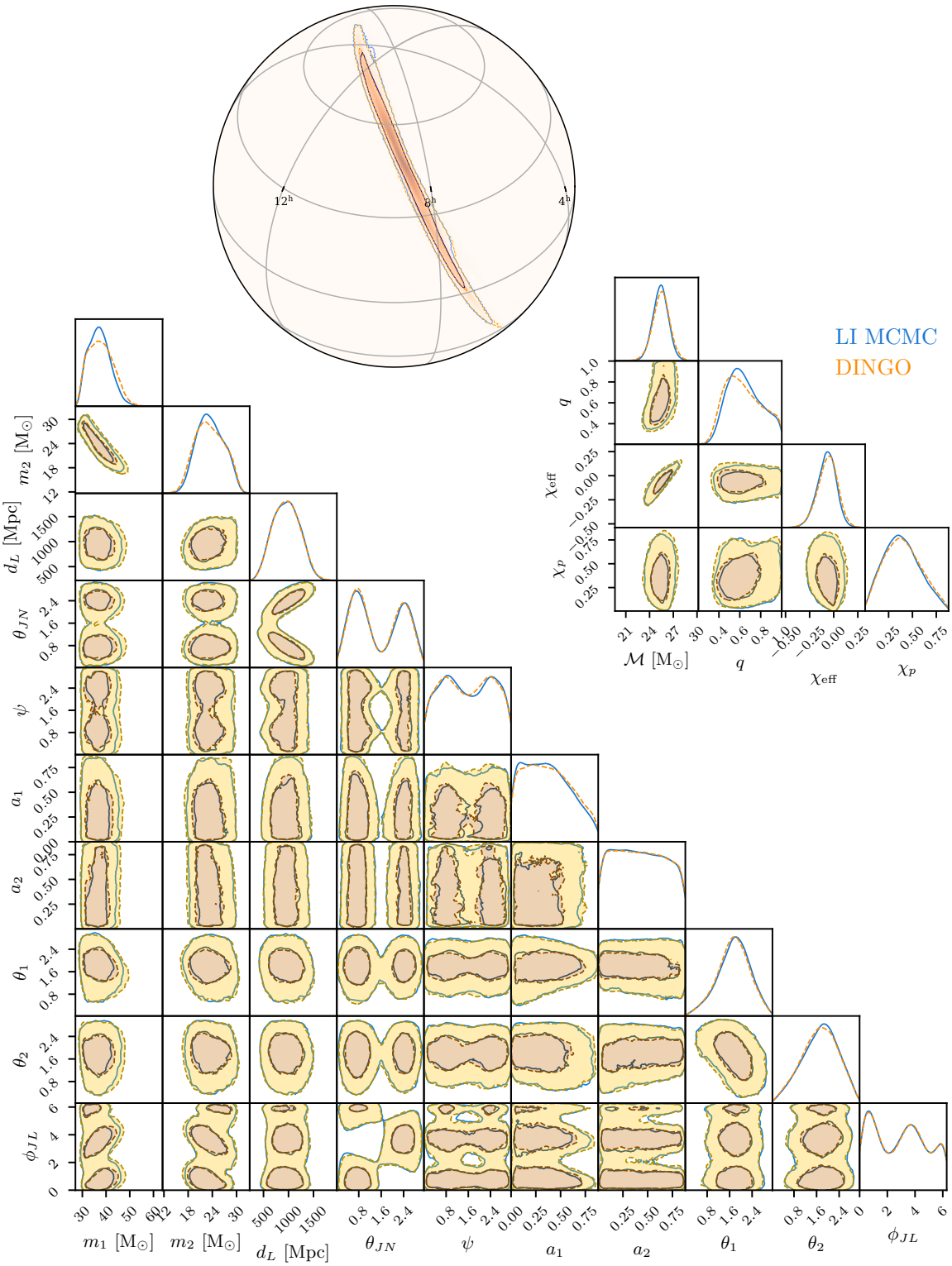


Fig. A.6 GW170104.

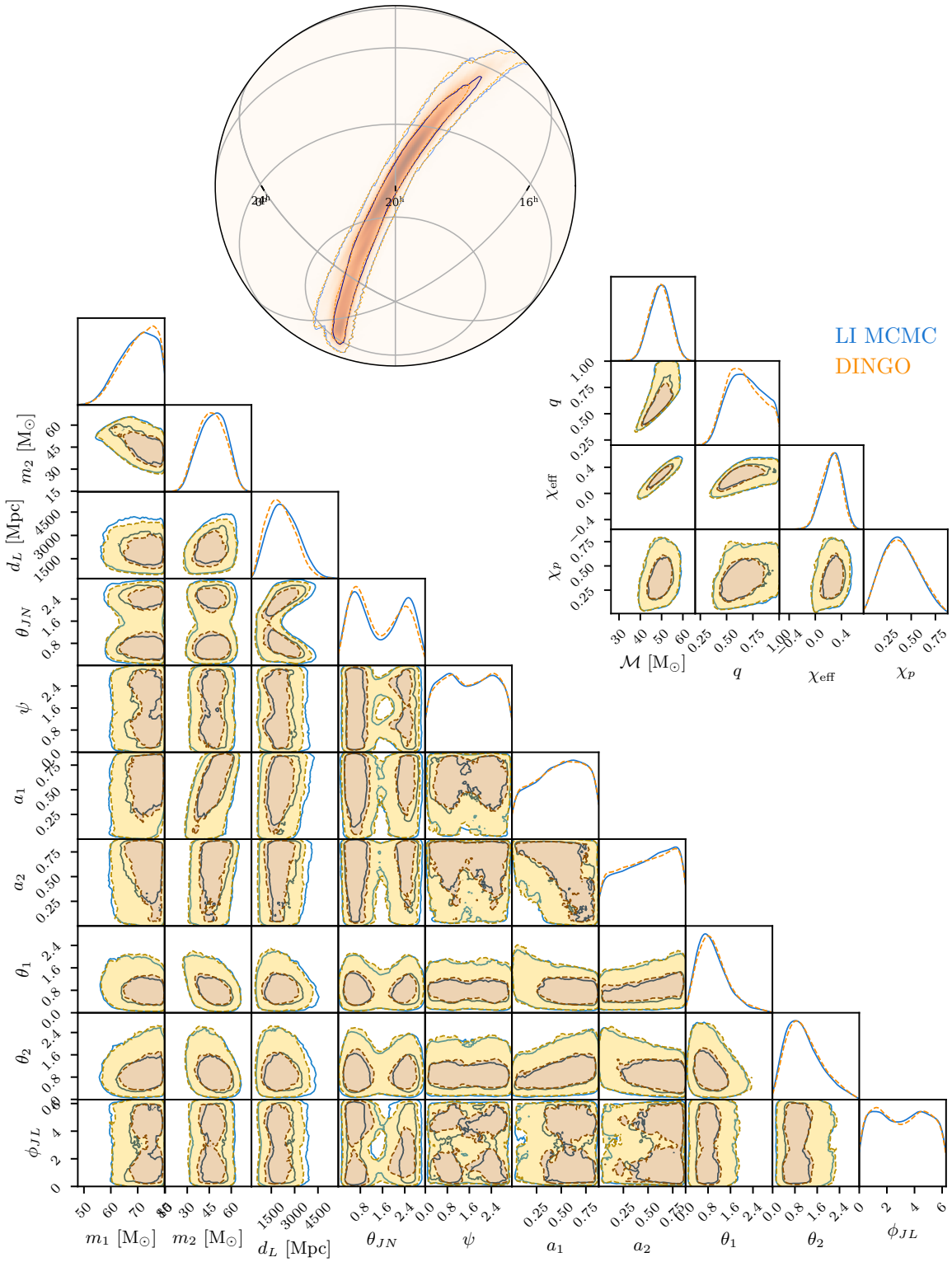


Fig. A.7 GW170729.

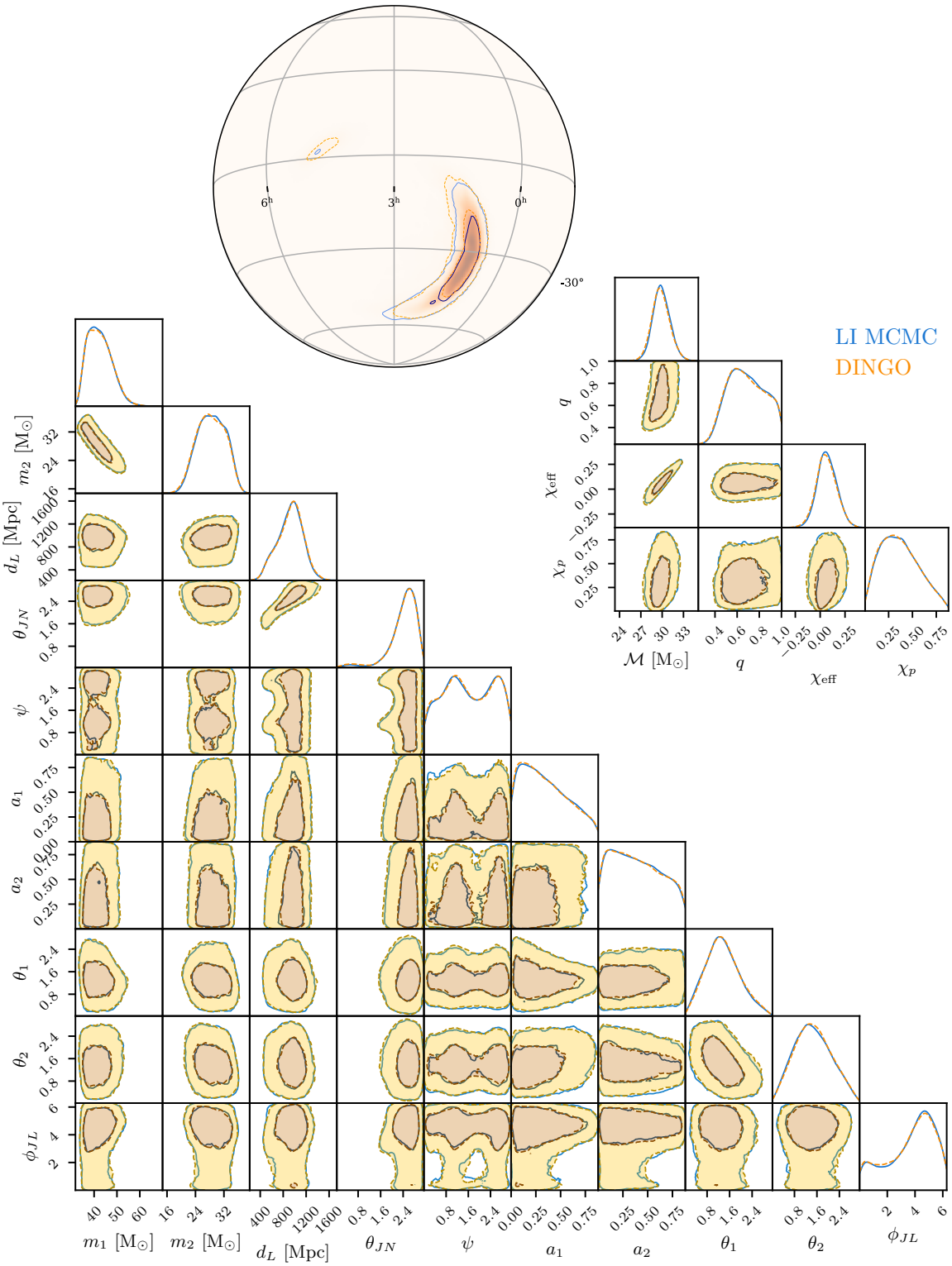


Fig. A.8 GW170809.

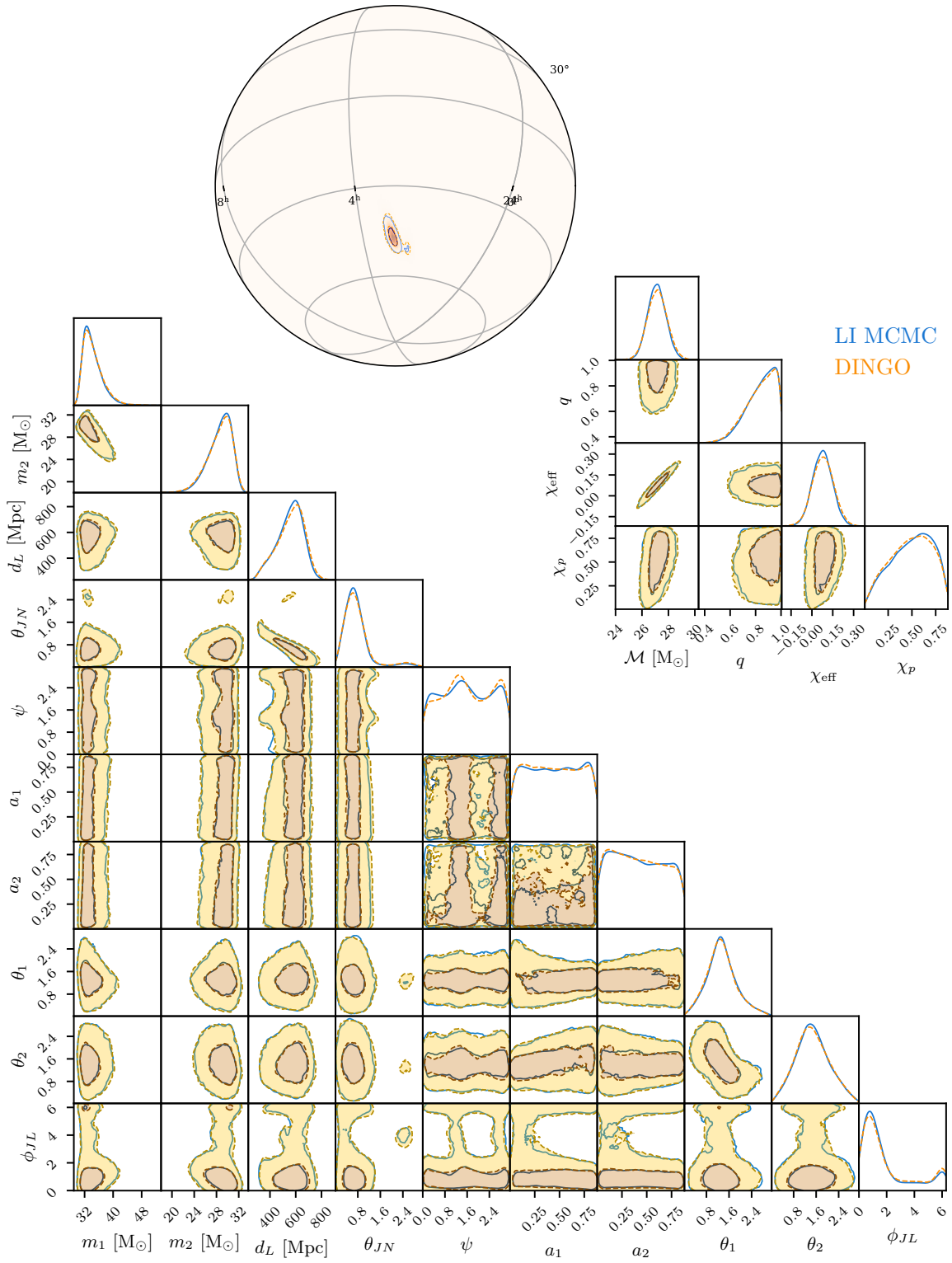


Fig. A.9 GW170814. This is the only event analyzed as a three-detector event.

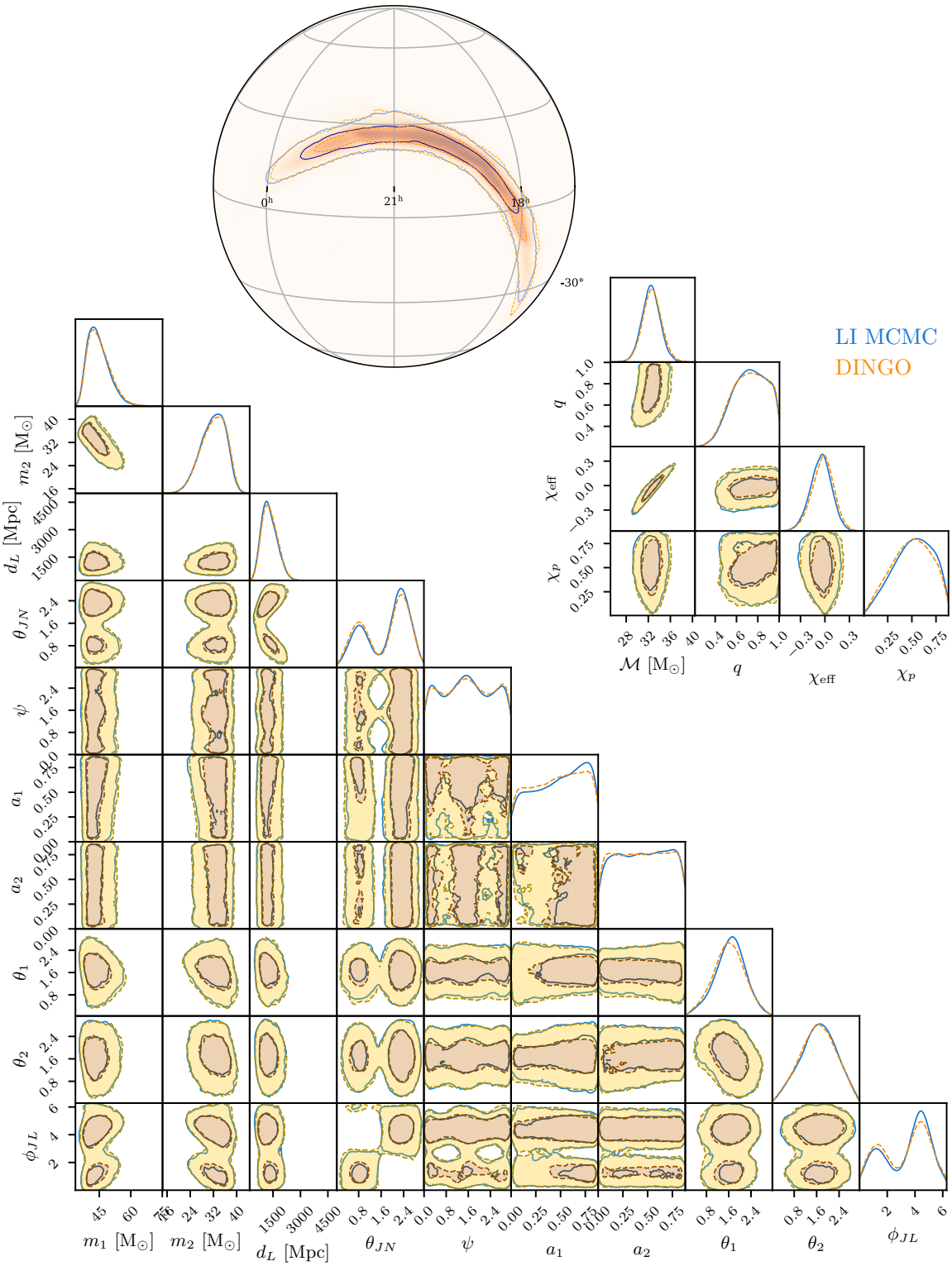


Fig. A.10 GW170818.

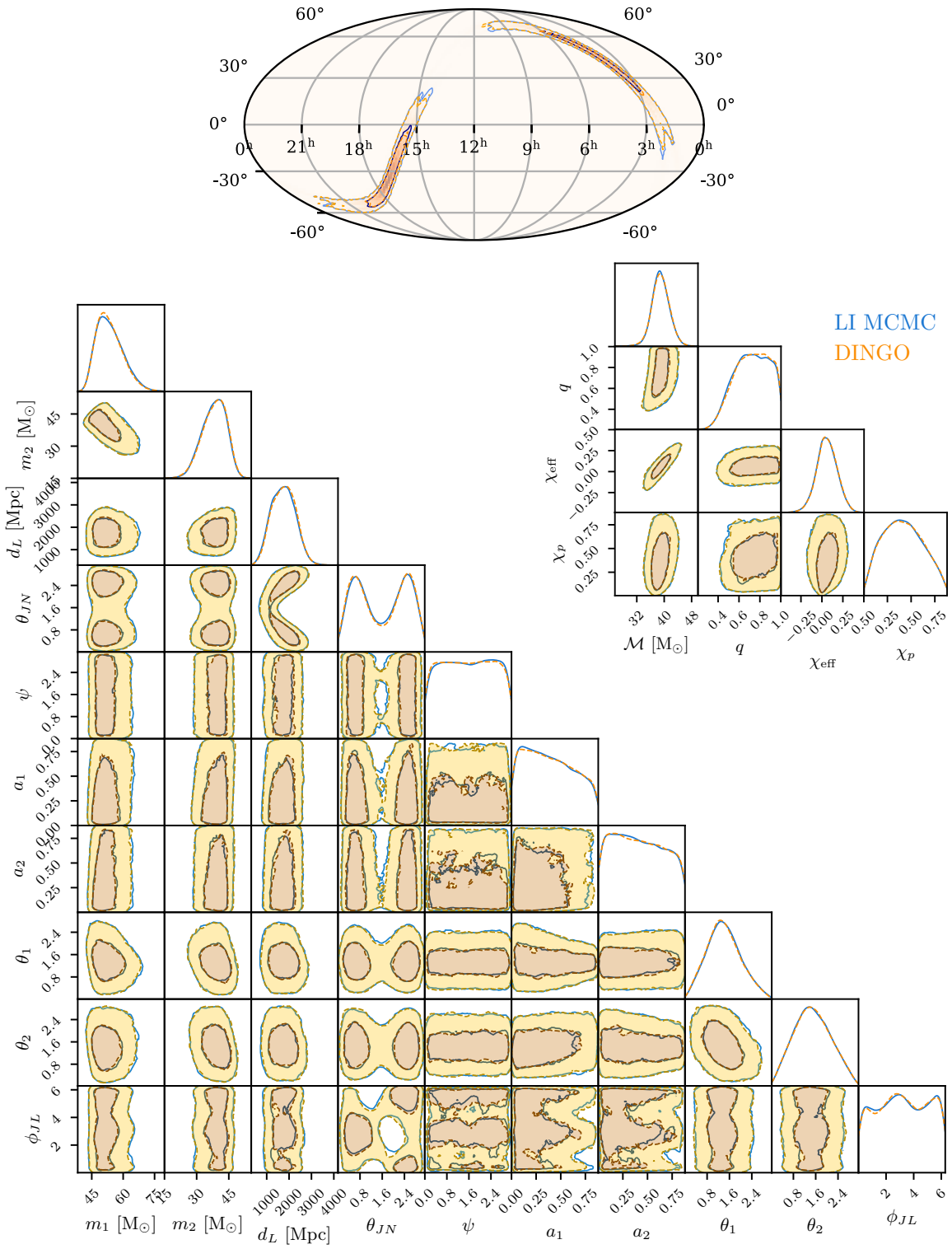


Fig. A.11 GW170823.

Appendix B

Group Equivariant Neural Posterior Estimation

B.1 Derivations

B.1.1 Equivariance relations

Consider a system with an exact equivariance under a joint transformation of parameters θ and observations x,

$$\theta \to \theta' = g\theta, \tag{B.1}$$

$$x \to x' = T_g x. \tag{B.2}$$

An invariant prior fulfills the relation

$$p(\theta) = p(\theta') \left|\det J_g\right|, \tag{B.3}$$

where the Jacobian J_g arises from the change of variables rule for probability distributions. A similar relation holds for an equivariant likelihood,

$$p(x|\theta) = p(x'|\theta') \left|\det J_T\right|. \tag{B.4}$$

An invariant prior and an equivariant likelihood further imply for the evidence $p(x)$

$$p(x) = \int p(x|\theta)p(\theta)d\theta = \int p(x'|\theta') \left|\det J_T\right| p(\theta') \left|\det J_g\right| \, d\theta = p(x') \left|\det J_T\right|. \tag{B.5}$$

Combining an invariant prior with an equivariant likelihood thus leads to the equivariance relation

$$\begin{aligned} p(\theta|x) &= \frac{p(x|\theta)p(\theta)}{p(x)} = \frac{p(x'|\theta')\,|\det J_T|\,p(\theta')\,|\det J_g|}{p(x')\,|\det J_T|} \\ &= p(\theta'|x')\,|\det J_g| \end{aligned} \tag{B.6}$$

for the posterior, where we used Bayes' theorem and equations (B.3), (B.4), and (B.5).

B.1.2 Equivariance of $p(\theta|x, \hat{g})$

Here we derive that an equivariant posterior $p(\theta|x)$ remains equivariant if the distribution is also conditioned on the proxy $\hat{g}$, as used in equation (3.7). With the definition $p(\hat{g}|\theta) = \kappa\left((g^\theta)^{-1}\hat{g}\right)$ from section 3.3.3, $p(\hat{g}|\theta)$ is equivariant under joint application of $h \in G$ to $\hat{g}$ and θ,

$$p(\hat{g}|\theta) = \kappa\left((g^\theta)^{-1}\hat{g}\right) = \kappa\left((g^\theta)^{-1}h^{-1}h\hat{g}\right) = \kappa\left((hg^\theta)^{-1}h\hat{g}\right) = p(h\hat{g}|h\theta). \tag{B.7}$$

where for the last equality we used $g^{h\theta} = hg^\theta$. This implies, that $p(\hat{g}|x)$ is equivariant under joint application of h and T_h,

$$\begin{aligned} p(\hat{g}|x) &= \int p(\hat{g}|\theta, x)p(\theta|x)\,d\theta \overset{\text{(B.7),(3.4)}}{=} \int p(h\hat{g}|h\theta)p(h\theta|T_h x)\,|\det J_h|\;d\theta \\ &= p(h\hat{g}|T_h x). \end{aligned} \tag{B.8}$$

in the second step we used $p(\hat{g}|\theta, x) = p(\hat{g}|\theta)$. From these relations, the equivariance relation used in equation (3.7) follows,

$$\begin{aligned} p(\theta|x, \hat{g}) &= \frac{p(\theta, \hat{g}|x)}{p(\hat{g}|x)} = \frac{p(\hat{g}|\theta, x)p(\theta|x)}{p(\hat{g}|x)} \overset{\text{(B.7),(3.4),(B.8)}}{=} \frac{p(h\hat{g}|h\theta)p(h\theta|T_h x)}{p(h\hat{g}|T_h x)}\,|\det J_h| \\ &= p(h\theta|T_h x, h\hat{g})\,|\det J_h|\,. \end{aligned} \tag{B.9}$$

B.1.3 Exact equivariance of inferred posterior

Consider a posterior that is exactly equivariant under G,

$$p(\theta|x) = p(g\theta|T_g x)|\det J_g|, \qquad \forall g \in G. \tag{B.10}$$

We here show that the posterior estimated using GNPE is equivariant under G by construction. This holds regardless of whether $q(\theta'|x')$ has fully converged to $p(\theta'|x')$.

With GNPE, the equivariant posterior $p(\theta|x_o)$ for an observation x_o is inferred by alternately sampling

$$\theta^{(i)} \sim p(\theta|x_o, \hat{g}^{(i-1)}) \qquad \Longleftrightarrow \qquad \theta^{(i)} = \hat{g}^{(i-1)}\theta'^{(i)}, \quad \theta'^{(i)} \sim q(\theta'|T_{(\hat{g}^{(i-1)})^{-1}}x_o), \tag{B.11}$$

$$\hat{g}^{(i)} \sim p(\hat{g}|x_o, \theta^{(i)}) \qquad \Longleftrightarrow \qquad \hat{g}^{(i)} = g^{\theta^{(i)}}\epsilon, \quad \epsilon \sim \kappa(\epsilon), \tag{B.12}$$

see also equation (3.9). Now consider a different observation $\tilde{x}_o = T_h x_o$ that is obtained by altering the pose of x_o with T_h, where h is an arbitrary element of the equivariance group G. Applying the joint transformation

$$\hat{g} \to h\hat{g}, \tag{B.13}$$

$$x_o \to T_h x_o, \tag{B.14}$$

in (B.11) leaves θ' invariant,

$$q(\theta'|T_{(h\hat{g})^{-1}} T_h x_o) = q(\theta'|T_{\hat{g}^{-1}}(T_h)^{-1} T_h x_o) = q(\theta'|T_{\hat{g}^{-1}} x_o). \tag{B.15}$$

We thus find that θ in (B.11) transforms equivariantly under joint application of (B.13) and (B.14),

$$\theta = \hat{g}\theta' \to (h\hat{g})\theta' = h(\hat{g}\theta') = h\theta. \tag{B.16}$$

Conversely, applying

$$\theta \to h\theta \tag{B.17}$$

in (B.12) transforms $\hat{g}$ by

$$\hat{g} = g^\theta \epsilon \to g^{(h\theta)} \epsilon = h g^\theta \epsilon = h\hat{g}. \tag{B.18}$$

The θ samples (obtained by marginalizing over $\hat{g}$) thus transform $\theta \to h\theta$ under $x_o \to T_h x_0$, which is consistent with the desired equivariance (B.10).

Another intuitive way to see this is to consider running an implementation of the Gibbs sampling steps (B.11) and (B.12) with fixed random seed for two observations x_o (initialized with $\hat{g}^{(0)} = \hat{g}_{x_o}$) and $T_h x_o$ (initialized with $\hat{g}^{(0)} = h\hat{g}_{x_o}$). The Gibbs sampler will yield parameters samples $(\theta_i)_{i=1}^N$ for x_o, and the *exact same* samples $(h\theta_i)_{i=1}^N$ for $T_h x_o$, up to the global transformation by h. The reason is that the density estimator $q(\theta'|x')$ is queried with the same x' for both observations x_o and $T_h x_o$ in each iteration i. Since the truncated, thinned samples are asymptotically independent of the initialization, this shows that (B.10) is fulfilled by construction.

B.1.4 Iterative inference and convergence

GNPE leverages a neural density estimator of the form $q(\theta|x', \hat{g})$ to obtain samples from the joint distribution $p(\theta, \hat{g}|x)$. This is done by iterative sampling as described in section 3.3. Here we derive equation (3.11), which states how a distribution $Q_j(\theta|x)$ is updated by a single GNPE iteration.

Given a distribution $Q_j^\theta(\theta|x)$ of θ samples in iteration j, we infer samples for the pose proxy $\hat{g}$ for the next iteration by (i) extracting the pose g^θ from θ (this essentially involves marginalizing over all non pose related parameters) and (ii) blurring the pose g^θ with the kernel κ, corresponding to a

group convolution.[1] We denote this combination of marginalization and group convolution with the "$\bar{*}$" symbol,

$$Q_{j+1}^{\hat{g}}(\hat{g}|x) = \int d\theta\, Q_j^\theta(\theta|x)\kappa((g^\theta)^{-1}\hat{g}) = \left(Q_j^\theta(\cdot|x) \,\bar{*}\, \kappa\right)(\hat{g}). \tag{B.19}$$

For a given proxy sample $\hat{g}$, a (perfectly trained) neural density estimator infers θ with

$$p(\theta|x,\hat{g}) = \frac{p(\theta,\hat{g}|x)}{p(\hat{g}|x)} = \frac{p(\hat{g}|x,\theta)p(\theta|x)}{p(\hat{g}|x)} = p(\theta|x)\frac{\kappa((g^\theta)^{-1}\hat{g})}{(p^\theta(\cdot|x)\,\bar{*}\,\kappa)\,(\hat{g})}, \tag{B.20}$$

where we used $p(\hat{g}|\theta) = \kappa((g^\theta)^{-1}\hat{g})$. Combining (B.19) and (B.20), the updated distribution over θ samples reads

$$\begin{aligned}
Q_{j+1}^\theta(\theta|x) &= \int d\hat{g}\, p(\theta|x,\hat{g})\, Q_{j+1}^{\hat{g}}(\hat{g}|x) \\
&= \int d\hat{g}\, \left(Q_j^\theta(\cdot|x)\,\bar{*}\,\kappa\right)(\hat{g})\, p(\theta|x)\frac{\kappa((g^\theta)^{-1}\hat{g})}{((p^\theta(\cdot|x)\,\bar{*}\,\kappa))(\hat{g})} \\
&= p(\theta|x)\int d\hat{g}\, \frac{\left(Q_j^\theta(\cdot|x)\,\bar{*}\,\kappa\right)(\hat{g})}{((p^\theta(\cdot|x)\,\bar{*}\,\kappa))(\hat{g})}\,\kappa((g^\theta)^{-1}\hat{g}) \\
&= p(\theta|x)\left(\frac{Q_j^\theta(\cdot|x)\,\bar{*}\,\kappa}{p^\theta(\cdot|x)\,\bar{*}\,\kappa} \;*\; \kappa^{(-)}\right)(\hat{g}).
\end{aligned} \tag{B.21}$$

Here, $\kappa^{(-)}$ denotes the reflected kernel, $\kappa^{(-)}(g) = \kappa(g^{-1})\,\forall g$. Since we choose a symmetric kernel in practice, we use $\kappa = \kappa^{(-)}$ in (3.11).

In this notation, the initialization of the pose g^θ in iteration 0 with q_{init} simply means setting $Q_0(\cdot|x)\,\bar{*}\,\kappa = q_{\text{init}}(\cdot|x) * \kappa$.

B.2 GNPE for simple Gaussian likelihood and prior

Consider a simple forward model $\tau \to x$ with a given prior

$$p(\tau) = \mathcal{N}(-5,1)[\tau] \tag{B.22}$$

and a likelihood

$$p(x|\tau) = \mathcal{N}(\tau,1)[x], \tag{B.23}$$

where the normal distribution is defined by

$$\mathcal{N}(\mu,\sigma^2)[x] = \frac{\exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)}{\sqrt{2\pi}\sigma}. \tag{B.24}$$

[1] We define a group convolution as $(A * B)(\hat{g}) = \int dg\, A(g)B(g^{-1}\hat{g})$, which is the natural extension of a standard convolution.

The evidence can be computed from the prior (B.22) and likelihood (B.23), and reads

$$p(x) = \int d\tau p(\tau) p(x|\tau) = \int d\tau \mathcal{N}(-5, 1)[\tau] \mathcal{N}(\tau, 1)[x] = \mathcal{N}(-5, 2)[x]. \tag{B.25}$$

The posterior is then given via Bayes' theorem and reads

$$p(\tau|x) = \frac{p(x|\tau)p(\tau)}{p(x)} = \frac{\mathcal{N}(\tau, 1)[x]\mathcal{N}(-5, 1)[\tau]}{\mathcal{N}(-5, \sqrt{2})[x]} = \mathcal{N}\left(\frac{x-5}{2}, 1/2\right)[\tau]. \tag{B.26}$$

B.2.1 Equivariances

The likelihood (B.23) is equivariant under G, i.e., the joint transformation

$$\begin{aligned} \tau &\to g\tau = \tau + \Delta\tau, \\ x &\to T_g^l x = x + \Delta\tau. \end{aligned} \tag{B.27}$$

This follows directly from (B.23) and (B.24). If the prior was invariant, then this equivariance would be inherited by the posterior, see App. B.1.1. However, the prior is not invariant. It turns out that the posterior is still equivariant, but x transforms under a different representation than it does for the equivariance of the likelihood. Specifically, the posterior (B.26) is equivariant under joint transformation

$$\begin{aligned} \tau &\to g\tau = \tau + \Delta\tau, \\ x &\to T_g^p x = x + 2 \cdot \Delta\tau, \end{aligned} \tag{B.28}$$

which again directly follows from (B.26) and (B.24). Importantly, $T_g^p \neq T_g^l$, i.e., the representation under which x transforms is different for the equivariance of the likelihood and the posterior. For GNPE, the relevant equivariance is that of the posterior, i.e. the set of transformations (B.28), see also equation (3.4). The equivariance relation of the posterior thus reads

$$p(\tau|x) = p(g\tau|T_g^p x)|\det J_g|, \qquad \forall g \in G. \tag{B.29}$$

B.2.2 GNPE

We choose τ as the pose, which we aim to standardize with GNPE. We define the corresponding proxy as

$$\hat{\tau} = \tau + \epsilon, \quad \epsilon \sim \kappa(\epsilon) = \mathcal{N}(0, 1)[\epsilon]. \tag{B.30}$$

We can use GNPE to incorporate the exact equivariance of the posterior by construction. To that end we define

$$\begin{aligned} \tau' &= g^{(-\hat{\tau})}\tau = \tau - \hat{\tau}, \\ x' &= T^p_{g^{(-\hat{\tau})}} x = x - 2 \cdot \hat{\tau}. \end{aligned} \tag{B.31}$$

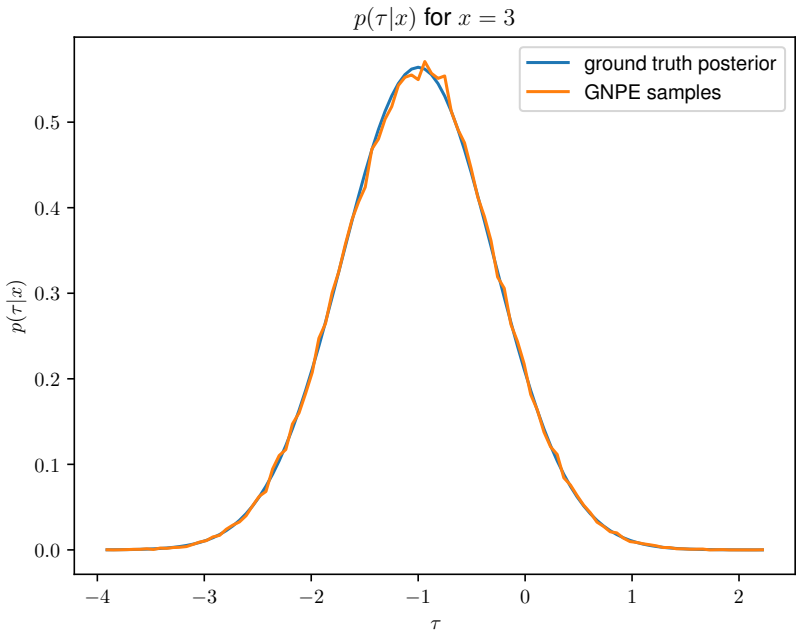


Fig. B.1 Posterior $p(\tau|x = 3)$ (blue) and the corresponding inferred GNPE samples (orange).

We then train a neural density estimator to estimate $p(\tau'|x')$. This distribution is of the same form as $p(\tau|x)$ and simply given by

$$p(\tau'|x') \overset{\text{(B.29),(B.26)}}{=} \mathcal{N}\left(\frac{x'-5}{2}, 1/2\right)[\tau'] \tag{B.32}$$

due to the equivariance (B.29). We here assume a neural density estimator that estimates (B.32) perfectly. For GNPE, we

1. Initialize $\tau^{(1)} = 0$;
2. Sample $\hat{\tau}^{(1)}$ by $\hat{\tau}^{(1)} = \tau^{(1)} + \epsilon,\ \epsilon \sim \mathcal{N}(0, 1)[\epsilon]$, and compute τ' and x' via (B.31);
3. Sample $\tau^{(2)}$ by $\tau^{(2)} = \tau'^{(2)} + \hat{\tau}^{(1)}$, with $\tau'^{(2)} \sim p(\tau'|x') = \mathcal{N}\left(\frac{x'-5}{2}, 1/2\right)[\tau']$;

and repeat (2) and (3) multiple times. This constructs a Markov chain. To obtain (approximately independent) posterior samples $\tau \sim p(\tau|x)$, we truncate to account for burn-in, thin the chain and marginalize over $\hat{\tau}$. We find that the chain indeed converges to the correct posterior (B.26), see Fig. B.1.

B.3 Toy Example

B.3.1 Forward model

The toy model in section 3.4 describes the motion of a damped harmonic oscillator that is initially at rest and excited at time τ with an infinitely short pulse. The time evolution of that system is governed by the differential equation

$$\frac{d^2}{dt^2}x(t) + 2\beta\omega_0\frac{d}{dt}x(t) + \omega_0^2 x(t) = \delta(t-\tau), \tag{B.33}$$

where ω_0 denotes the undamped angular frequency and β the damping ratio. The solution for the time series $x(t)$ is given by the Green's function for the corresponding differential operator and reads

$$x(t) = \begin{cases} 0, & t \leq \tau \\ e^{-\beta\omega_0(t-\tau)} \cdot \frac{\sin\left(\sqrt{1-\beta^2}\omega_0(t-\tau)\right)}{\sqrt{1-\beta^2}\omega_0}, & t > \tau. \end{cases} \tag{B.34}$$

This equation describes a deterministic, injective mapping between parameters $\theta = (\omega_0, \beta, \tau)$ and a time series observation x,

$$x = f(\theta). \tag{B.35}$$

This implies a likelihood $p(x|\theta) = \delta(x - f(\theta))$, and thus a point-like posterior. To showcase (G)NPE on this toy problem we introduce stochasticity by setting

$$x = f(\theta + \delta\theta) \tag{B.36}$$

instead. We sample $\delta\theta$ from an uncorrelated Gaussian distribution

$$\delta\theta \sim \mathcal{N}(0, \Sigma), \qquad \Sigma = \begin{pmatrix} \sigma_{\omega_0}^2 & 0 & 0 \\ 0 & \sigma_\beta^2 & 0 \\ 0 & 0 & \sigma_\tau^2 \end{pmatrix}, \tag{B.37}$$

with $\sigma_{\omega_0} = 0.3$ Hz, $\sigma_\beta = 0.03$ and $\sigma_\tau = 0.3$ s. Due to the injectivity of f, the posterior $p(\theta|x)$ reduces to the probability $p(\delta\theta = f^{-1}(x) - \theta)$. With a uniform prior, and neglecting boundary effects, this implies an uncorrelated Gaussian posterior $p(\theta|x)$ centered around $f^{-1}(x)$ with standard deviations as specified above. We choose this approach over, e.g., adding noise straight to observations to keep the problem as simple as possible, such that the focus remains on the comparison of GNPE and NPE. In particular, knowing that the ground truth posteriors are Gaussian, we can use a simple Gaussian density estimator.

We choose uniform priors

$$p(\omega_0) = \mathcal{U}[3, 10]\ \text{Hz}, \qquad p(\beta) = \mathcal{U}[0.2, 0.5], \qquad p(\tau) = \mathcal{U}[-5, 0]\ \text{s}. \tag{B.38}$$

The observational data x is the discretized time series in the interval $[-5, +5]$ s with 2000 evenly sampled bins. When applying time shifts with GNPE, we impose cyclic boundary conditions.

B.3.2 Implementation

We use a Gaussian density estimator for all methods (since we know that the true posterior is Gaussian). For NPE, we use a feedforward neural network with [128, 32, 16] hidden units and with ReLU activation functions as an embedding network. For NPE-CNN, we use a three-layer convolutional embedding network with kernel sizes [5,5,5], stride 1, [6,12,12] channels, average pooling with kernel size 7 and stride 7, and ReLU activation functions. For GNPE, we use the same architecture as for NPE for both, $q(\theta'|x')$ and $q_{\text{init}}(\tau|x)$. For further hyperparameters, we use the defaults of the sbi package [217].

B.3.3 Results

For all methods, we compute the average classifier two-sample test score (c2st) based on 10,000 samples from the estimated and the ground truth posterior for five different simulations. We then average the accuracy across 10 different seeds.

B.4 Gravitational wave parameter inference

B.4.1 Forward model and amortization

The forward model mapping binary black hole parameters θ (Tab. B.1) to simulated measurements x in the detectors consists of two stages. Firstly, the waveform polarizations $h(\theta)$ for given parameters θ are computed with the waveform model IMRPhenomPv2 [115, 134, 50]. Secondly, the signals are projected onto the detectors, and noise is added to obtain a realistic signal x. To a good approximation, we assume the noise to be Gaussian and stationary over the duration of a single GW signal. However, the noise spectrum, determined by the power spectral density (PSD) S_{n}, drifts over the duration of an observing run. To fully amortize the computational cost, we use a variety of different PSDs S_{n} in training, and additionally condition the inference network on S_{n}. At inference time, this enables instant tuning of the inference network to the PSD estimated at the time of the event, see Dax et al. [82] for details. Since this conditioning on S_{n} has no effect on the GNPE algorithm outlined in this work, we keep it implicit in all equations.

B.4.2 Network architecture and training

The inference network consists of an embedding network, that reduces the high dimensional input data to a 128 dimensional feature vector, and the normalizing flow, that takes this feature vector as input. For each detector, the input to the embedding network consists of the complex-valued frequency domain strain in the range $[20\text{ Hz}, 1024\text{ Hz}]$ with a resolution of 0.125 Hz, and PSD information $(10^{46} \cdot S_{\text{n}})^{-1/2}$ with the same binning. This results to a total of $(3 \cdot 8{,}033) = 24{,}099$ real input bins per detector. The first module of the embedding network consists of a linear layer per detector, that maps this 24,096 dimensional input to 400 components. We initialize this compression layer with

Table B.1 Priors for the astrophysical binary black hole parameters used to train the inference network. Priors are uniform over the specified range unless indicated otherwise. We train networks with different distance ranges for the two observing runs O1 and O2 due to the different detector sensitivities. At inference time, a cosmological distance prior is imposed by reweighting samples according to their distance.

Description	Parameter	Prior
component masses	m_1, m_2	$[10, 80]\ \mathrm{M}_\odot$, $m_1 \geq m_2$
spin magnitudes	a_1, a_2	$[0, 0.88]$
spin angles	θ_1, θ_2, ϕ_{12}, ϕ_{JL}	standard as in Farr et al. [98]
time of coalescence	t_c	$[-0.1, 0.1]$ s
luminosity distance	d_L	O1, 2 detectors: $[100, 2000]$ Mpc
		O2, 2 detectors: $[100, 2000]$ Mpc and $[100, 6000]$ Mpc
		O2, 3 detectors: $[100, 1000]$ Mpc
reference phase	ϕ_c	$[0, 2\pi]$
inclination	θ_{JN}	$[0, \pi]$ uniform in sine
polarization	ψ	$[0, \pi]$
sky position	α, β	uniform over sky

PCA components of raw waveforms. This provides a strong inductive bias to the network to filter out GW signals from extremely noisy data. Note that this important step is only possible since GNPE is architecture independent—it is for instance not compatible with a convolutional neural network. Following this compression layer, we use a series of 24 fully-connected residual blocks with two layers each to compress the output to the desired 128 dimensional feature vector. We use batch normalization and ELU activation functions. Importantly, the conditioning of the flow on the proxy $\hat{g}_{\text{rel.}}$ is done *after* the embedding network, by concatenating $\hat{g}_{\text{rel.}}$ to the embedded feature vector.

Following this, we use a neural spline flow [93] with rational-quadratic spline coupling transforms as density estimator. We use 30 such transforms, each of which is associated with 5 two-layer residual blocks with hidden dimension 512. In total, the inference network has 348 hidden layers and $1.31 \cdot 10^8$ (for two detectors) or $1.42 \cdot 10^8$ (for three detectors) learnable parameters.

We train the inference network with a data set of $5 \cdot 10^6$ waveforms with parameters θ sampled from the priors specified in table B.1, and reserve 2% of the data for validation. We pretrain the network with learning rate of $3 \cdot 10^{-4}$ for 300 epochs with fixed PSD, and finetune for another 150 epochs with learning rate of $3 \cdot 10^{-5}$ with varying PSDs. With batch size 4,096, training takes 16-18 days on a NVIDIA Tesla V100 GPU.

B.4.3 Results

The c2st scores between inferred posterior and the MCMC reference shown in Fig. 3.4 are computed using the code and default hyperparameters of Lueckmann et al. [154]. For each event, we compute the c2st score of 10,000 samples for inferred and target posterior. Fig. 3.4 displays the mean of the score across 5 different sample realizations, Fig. B.2 additionally shows the corresponding standard deviation. For technical reasons we use only 12 of the 15 inferred parameters; specifically we omit

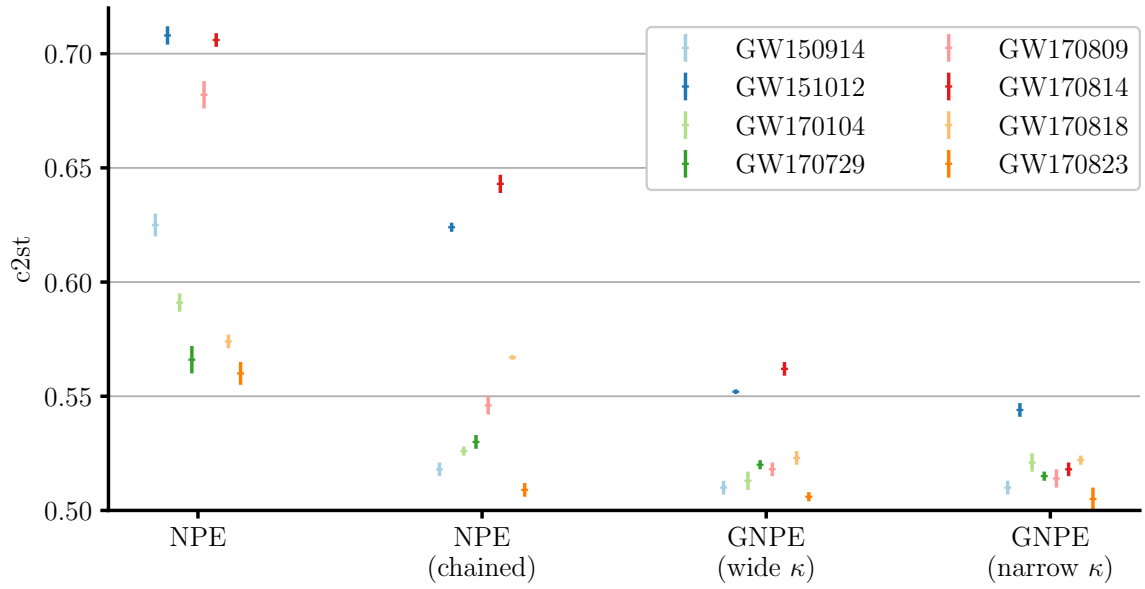


Fig. B.2 c2st scores quantifying the deviation between the inferred posteriors and the MCMC reference. This is an extended version of Fig. 3.4.

the geocentric time of coalescence t_c (since the reference posteriors generated with LALInference do not contain that variable) and the sky position parameters α and δ (since the NPE baseline with chain rule decomposition infers these in another basis).

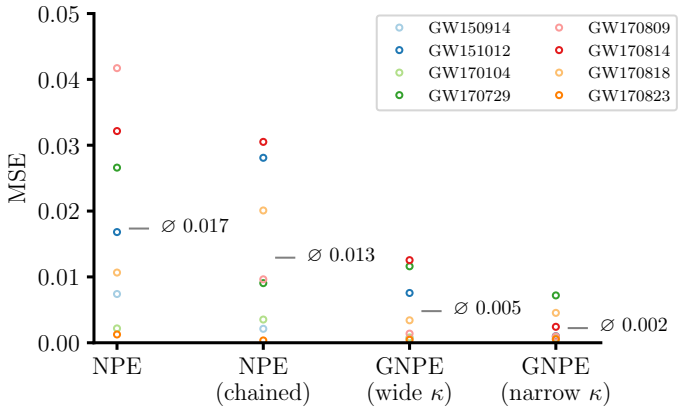


Fig. B.3 Comparison of estimated posteriors against LALINFERENCE MCMC for eight GW events, as quantified by the mean squared error (MSE) of the sample means. Before computing the means, we normalize each dimension such that the prior has a standard deviation of 1. ∅ indicates the average across all eight events. GNPE with a narrow kernel consistently outperforms the baselines, which is in accordance with Fig. 3.4.

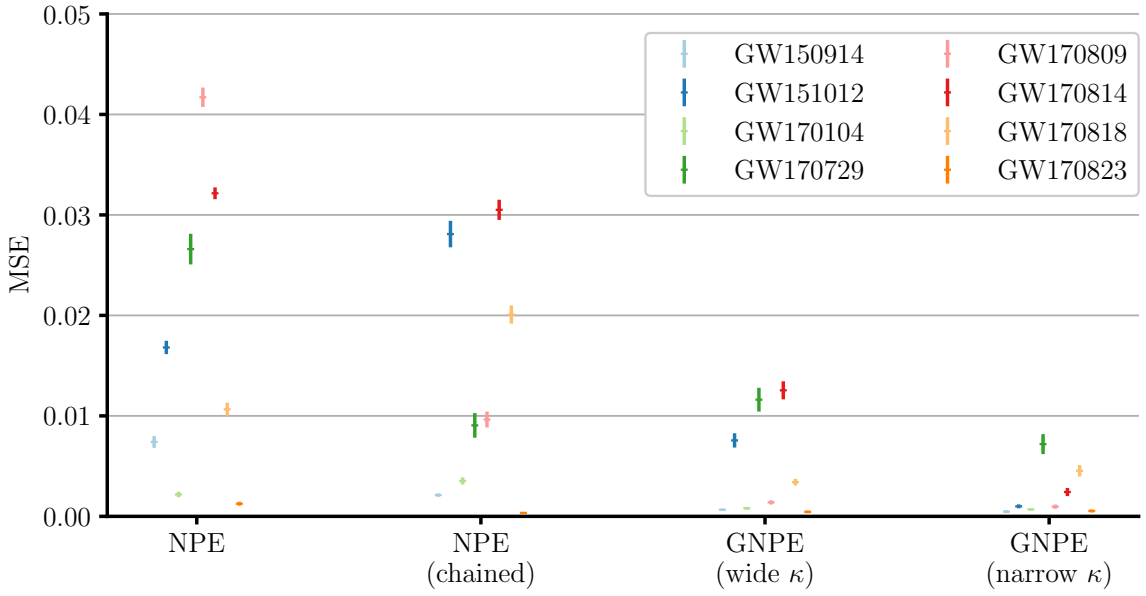


Fig. B.4 MSE between inferred posteriors and MCMC reference. Extended version of Fig. B.3.

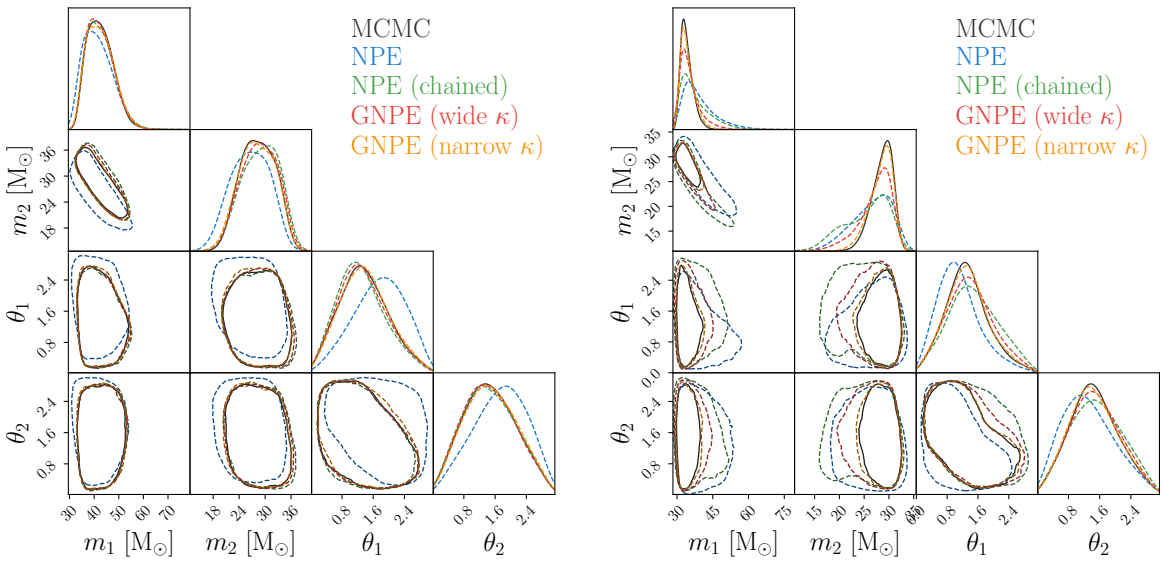


Fig. B.5 Corner plots for the GW events GW170809 (left) and GW170814 (right), plotting 1D marginals on the diagonal and 90% credible regions for the 2D correlations. We display the two black hole masses m_1 and m_2 and two spin parameters θ_1 and θ_2 (note that the full posterior is 15-dimensional). This extends Fig. 3.5 by also displaying the results from chained NPE and GNPE with wide κ.

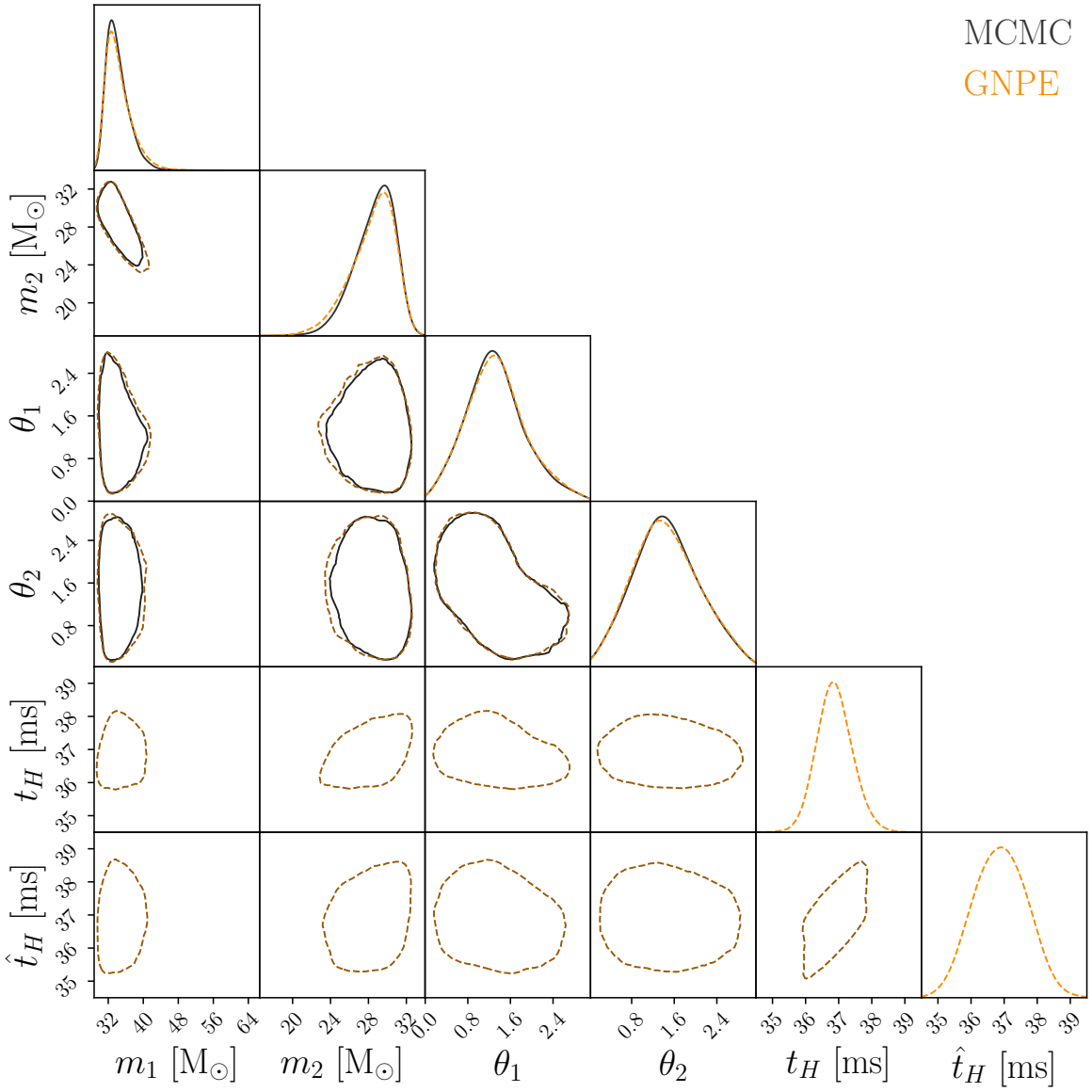


Fig. B.6 Corner plot for the GW event GW170814 plotting 1D marginals on the diagonal and 90% credible regions for the 2D correlations. We display the two black hole masses m_1 and m_2 and two spin parameters θ_1 and θ_2 that are also shown in Fig. 3.5. We additionally display one of the pose parameters t_H and the corresponding proxy $\hat{t}_H$ from the last GNPE iteration. In training, the neural density estimator learned that the true pose t_H differs by at most 1 ms from the proxy $\hat{t}_H$ that it is conditioned on (since we chose a kernel $\kappa_{\text{narrow}} = U[-1\text{ ms}, 1\text{ ms}]^{n_I}$, see section 3.5.2). This explains the strong correlation between t_H and $\hat{t}_H$ we observe. For the same reason, the observed correlations between the $\hat{t}_H$ and the non-pose parameters $(m_1, m_2, \theta_1, \theta_2)$ are similar to those between the true pose t_H and $(m_1, m_2, \theta_1, \theta_2)$.

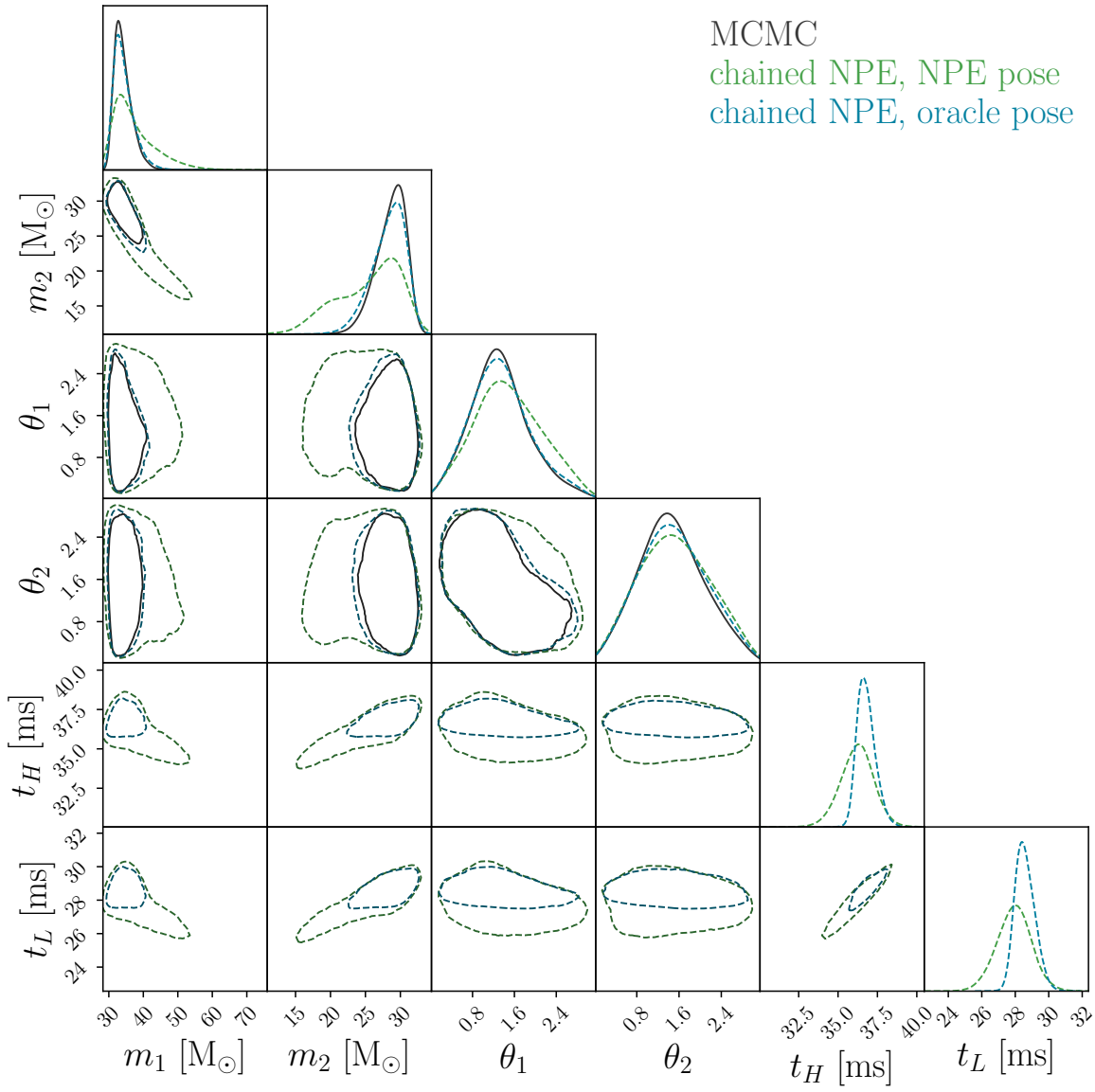


Fig. B.7 Corner plot for GW170814 with the four parameters $(m_1, m_2, \theta_1, \theta_2)$ that are also displayed in Fig. 3.5, as well as the pose (t_H, t_L). We compare chained NPE as described in section 3.5.3 to an oracle version: for the earlier the pose is inferred using standard NPE (green) while for the latter we take an oracle pose provided by a (slow) nested sampling algorithm (teal). We observe, that the result using the oracle pose matches the MCMC reference posterior well, while the other one shows clear deviations. Both versions use the same density estimator for the non-pose parameters $\phi \subset \theta$. This demonstrates that inaccuracies of the chained NPE baselines can be almost entirely attributed to inaccurate inital estimates of the pose. Poor pose estimates can occur since the density estimator trained to extract the pose operates on non pose-standardized data.
Note: The MCMC reference algorithm LALInference does not provide full pose information since it automatically marginalizes over t_c. For the oracle pose we thus employ the nested sampling algorithm bilby [41].

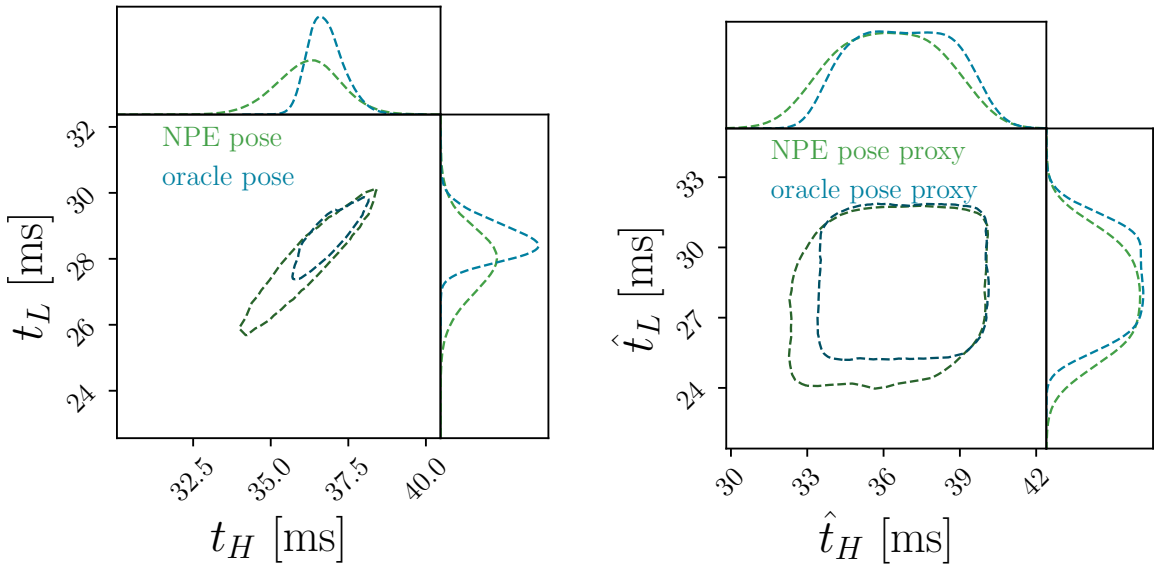


Fig. B.8 Left: Pose parameters t_H and t_L for the GW event GW170814, estimated with the neural density estimator q_{init} with standard NPE (green), as well as the "true" pose inferred with bilby (teal). Right: Pose proxies $\hat{t}_H$ and $\hat{t}_L$ for the wide kernel $\kappa_{\text{wide}} = U[-3\text{ ms}, 3\text{ ms}]^{n_I}$. These are obtained from the pose estimates in the left panel via a convolution with κ_{wide}. We observe that the deviation between the oracle and the NPE estimate is substantially smaller for the pose proxy than for the pose itself due to the blurring operation. This leads to a better performance of fast-mode GNPE (with κ_{wide} and only one iteration) compared to chained NPE in section 3.5.3.

Appendix C

Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference

C.1 Importance-sampled Bayesian evidence

The Bayesian evidence is given by

$$p(d) = \int d\theta p(d|\theta)p(\theta) = \int d\theta \frac{p(d|\theta)p(\theta)}{q(\theta|d)} q(\theta|d), \tag{C.1}$$

which can be estimated using n samples $\theta_i \sim q(\theta|d)$ in the Monte Carlo approximation as $p(d) = \hat{\mu}_w$ with

$$\hat{\mu}_w = \frac{1}{n} \sum_i \frac{p(d|\theta_i)p(\theta_i)}{q(\theta_i|d)} = \frac{1}{n} \sum_i w_i \tag{C.2}$$

where $w_i = p(d|\theta_i)p(\theta_i)/q(\theta_i|d)$ are the weights used for importance sampling. The variance for this Monte Carlo estimate is given by

$$\begin{aligned} \sigma_w^2 &= \text{Var}\left[\frac{p(d|\theta)p(\theta)}{q(\theta|d)}\right] \approx \frac{1}{n} \sum_i (w_i - \hat{\mu}_w)^2 \\ &= \hat{\mu}_w^2 \cdot \frac{1}{n} \sum_i [\bar{w}_i - 1]^2 = \hat{\mu}_w^2 \cdot \left(\frac{1}{n} \sum_i \bar{w}_i^2 - 1\right) \\ &= \hat{\mu}_w^2 \cdot \left(\frac{n - n_{\text{eff}}}{n_{\text{eff}}}\right) = \hat{\mu}_w^2 \cdot \left(\frac{1-\epsilon}{\epsilon}\right), \end{aligned} \tag{C.3}$$

where we denote normalized weights with $\bar{w}_i = w_i/\hat{\mu}_w$ and the sample efficiency with $\epsilon = n_{\text{eff}}/n$. Since we use n samples to estimate $p(d) = \hat{\mu}_w$, the standard deviation of the evidence is given

by

$$\sigma_{p(d)} = \frac{\sigma_w}{\sqrt{n}} = p(d)\sqrt{\frac{1-\epsilon}{n\cdot\epsilon}}. \tag{C.4}$$

In practice, we are interested in the log evidence, for which the uncertainty is

$$\sigma_{\log p(d)} = \frac{\sigma_{p(d)}}{p(d)} = \sqrt{\frac{1-\epsilon}{n\cdot\epsilon}}. \tag{C.5}$$

C.1.1 Bias

Since $p(\theta)$ and $q(\theta|d)$ are normalized, Eq. (C.1) provides an unbiased estimate for $p(d)$ [171],

$$\mathbb{E}\left[\frac{1}{n}\sum_i w_i\right] = \mathbb{E}[\hat{\mu}_w] = p(d). \tag{C.6}$$

The logarithm of the evidence however has a bias. Defining $Y = \hat{\mu}_w - p(d)$, we find

$$\begin{aligned}
\mathbb{E}[\log\hat{\mu}_w] &= \mathbb{E}\left[\log\left(p(d) + p(d)\cdot\frac{\hat{\mu}_w - p(d)}{p(d)}\right)\right] \\
&= \log p(d) + \mathbb{E}\left[\log\left(1+\frac{Y}{p(d)}\right)\right] \\
&= \log p(d) + \mathbb{E}\left[\frac{Y}{p(d)} - \frac{1}{2}\left(\frac{Y}{p(d)}\right)^2\right] \\
&= \log p(d) - \frac{\sigma_w^2}{2p(d)^2 n} = \log p(d) - \frac{1-\epsilon}{2n\epsilon}
\end{aligned} \tag{C.7}$$

where we used $\mathbb{E}[Y] = 0$ and $\mathrm{Var}[Y] = \sigma_w^2/n$ and neglected terms of order $\mathcal{O}((Y/p(d))^3)$. The bias of the log evidence thus depends on the sample efficiency $\epsilon = n_{\text{eff}}/n$ and scales with $1/n$. Given that the uncertainty of $\log\hat{\mu}_w$ scales with $1/\sqrt{n}$, this bias is completely negligible in practice.

C.2 Analytic estimate of the phase parameter

The parameter ϕ_c describes the phase of the gravitational wave at a fixed reference frequency. It provides no physical insight, but it is necessary to define a complete likelihood [223]. While the marginal $p(\phi_c|d)$ usually has a simple structure, the *conditional* distribution $p(\phi_c|d,\tilde{\theta})$, where $\tilde{\theta}$ denotes the 14 remaining parameters, is typically very tightly constrained. Furthermore, ϕ_c is strongly correlated with $\tilde{\theta}$. We observed that DINGO has difficulties learning the phase parameter, and often infers the prior instead, $q(\phi_c|d,\tilde{\theta}) = p(\phi_c)$. While we did not find this to have a negative impact on the remaining parameters, it leads to a substantially reduced sample efficiency.

Inspired by phase marginalization [223, 219], a technique commonly used to increase the efficiency of stochastic samplers, we analytically estimate ϕ_c. The approach outlined below differs in two ways from typical phase marginalization—(1) we retrieve ϕ_c instead of marginalizing over it, and

(2) this technique is exact even in the presence of higher modes, where phase marginalization is an approximation.

We decompose our posterior estimate into

$$q(\theta|d) = p(\phi_{\rm c}|d, \tilde{\theta})q(\tilde{\theta}|d), \tag{C.8}$$

where $q(\tilde{\theta}|d)$ is estimated with DINGO. For each DINGO sample $\tilde{\theta} \sim q(\tilde{\theta}|d)$, we then *synthetically* sample $\phi_{\rm c}$ using the analytic likelihood. This is done by evaluating $p(\phi_{\rm c}|d, \tilde{\theta})$ on a uniform grid over $\phi_{\rm c}$ with 5001 points in the range $[0, 2\pi]$ and interpolating in between.

Each likelihood evaluation requires a waveform simulation, which accounts for the bulk of the computational cost. As we outline below, by caching suitable combinations of the waveform modes, we can cheaply evaluate waveform polarizations for arbitrary ϕ_c. Hence sampling the synthetic $\phi_{\rm c}$ is barely more expensive than a single likelihood evaluation.

C.2.1 Phase transformations

We work in the L_0 frame, which aligns the z axis with the orbital angular momentum of the binary at the reference frequency, and takes ϕ_c as the azimuthal angle of the observer relative to the axis connecting the two bodies. In these coordinates, the observer is located at $(\theta, \phi) = (\iota, \pi/2 - \phi_c)$, where ι is the inclination of the binary. This is convenient for caching the modes, since ϕ_c enters the waveform entirely via the spin-weighted spherical harmonics (as opposed to the modes themselves).

Waveform modes $h_{\ell m}$ combine into polarizations $h_{+,\times}$ as

$$h_+ - ih_\times = h = \sum_{\ell,m} h_{\ell m} \, {}_{-2}Y_{\ell m}(\theta, \phi), \tag{C.9}$$

In frequency domain,

$$\tilde{h}_+(f) = \frac{1}{2}\left[\tilde{h}(f) + \tilde{h}^*(-f)\right], \tag{C.10}$$

$$\tilde{h}_\times(f) = \frac{i}{2}\left[\tilde{h}(f) - \tilde{h}^*(-f)\right]. \tag{C.11}$$

Considering just the plus polarization and substituting for the mode expansion,

$$\begin{aligned}\tilde{h}_+(f) = \frac{1}{2}\sum_{\ell,m}\Big[&\tilde{h}_{\ell m}(f) \, {}_{-2}Y_{\ell m}(\theta, \phi) \\ &+ \tilde{h}^*_{\ell m}(-f) \, {}_{-2}Y^*_{\ell m}(\theta, \phi)\Big].\end{aligned} \tag{C.12}$$

Now we use the fact that the ϕ-dependence enters the spin-weighted spherical harmonics as ${}_{-2}Y_{\ell m}(\theta, \phi) = {}_{-2}Y_{\ell m}(\theta, 0)e^{im\phi}$. Since h_+ is real, we only need to consider $f > 0$. In the L_0 frame, we can then write

$$\tilde{h}_+(f > 0) = \sum_m \tilde{h}_{+,m}(f)e^{-im\phi_c}, \tag{C.13}$$

where we have grouped the terms according to their m-dependence,

$$\begin{aligned}\tilde{h}_{+,m}(f) &= \frac{1}{2}\sum_{\ell}\Big[\tilde{h}_{\ell m}(f)\,{}_{-2}Y_{\ell m}\left(\iota,\frac{\pi}{2}\right) \\ &\quad + \tilde{h}^{*}_{\ell,-m}(-f)\,{}_{-2}Y^{*}_{\ell,-m}\left(\iota,\frac{\pi}{2}\right)\Big]. \end{aligned} \tag{C.14}$$

Notice that we combined the positive frequency parts of modes with azimuthal number m together with negative frequency modes of azimuthal number $-m$. With this decomposition, we only need to cache the $\tilde{h}_{+,m}$. Likewise for the cross polarization, we have

$$\tilde{h}_{\times}(f>0) = \sum_{m}\tilde{h}_{\times,m}(f)e^{-im\phi_c}, \tag{C.15}$$

where

$$\begin{aligned}h_{\times,m}(f) &= \frac{i}{2}\sum_{\ell}\Big[\tilde{h}_{\ell m}(f)\,{}_{-2}Y_{\ell m}\left(\iota,\frac{\pi}{2}\right) \\ &\quad - \tilde{h}^{*}_{\ell,-m}(-f)\,{}_{-2}Y^{*}_{\ell,-m}\left(\iota,\frac{\pi}{2}\right)\Big]. \end{aligned} \tag{C.16}$$

One additional complication arises because waveform models are usually given in terms of Cartesian spin components, and ϕ_c also enters into their definition in terms of the spin parameters used for parameter estimation. Consequently the modes retain a dependence on ϕ_c. We overcome this by fixing the phase parameter used in effecting this transformation. This results in a slightly different definition of the spin parameters θ_{JN} and ϕ_{JL}, which we undo in post-processing. Since the standard priors are invariant under this transformation, other parameters are not affected.

This approach enables likelihood evaluations on a $\phi_{\rm c}$ grid at the computational cost of a single likelihood evaluation, plus a small additional cost for the inner products.[1] The implementation is fully contained in the DINGO package, which uses low-level LALSimulation [145] functions to compute frequency domain modes in the L_0 frame, and combines them into the $\tilde{h}_{+/\times,m}$. For SEOBNRv4PHM, this requires Fourier transforming the time domain modes provided by LALSIMULATION in L_0 frame. For IMRPhenomXPHM it requires transforming from J to L_0 frame, such that the $\phi_{\rm c}$ dependence enters via the spherical harmonics, not via the modes themselves.

C.3 Density recovery

IS requires access to the density of the inferred samples. While for NPE, this density is tractable, this is not necessarily the case for other inference methods. Below, we describe how we use neural density estimation to recover the density in these cases.

[1] For IMRPhenomXPHM, computing the individual modes with the LALSIMULATION function `SimInspiralChooseFDModes` is substantially more expensive than computing the combined polarizations with `SimInspiralFD`. This is because `SimInspiralFD` caches information when internally computing the modes, whereas `SimInspiralChooseFDModes` does not.

	2 dimensions	14 dimensions
flow steps	5	20
hidden dimension	256	256
transform blocks	4	4
bins	8	8
training samples	$4 \cdot 10^5$	10^6
batch size	4096	8192
epochs	20	60
optimizer	adam [135]	adam [135]
learning rate	0.002	0.001
training time on A100 GPU	7 minutes	1 hour

Table C.1 Settings for the neural spline flow [93] architecture (upper part) and training (lower) used for density recovery. For DINGO-IS with GNPE, we need to estimate a two dimensional distribution over the proxy parameters, which requires a smaller network than the distribution over the 14 dimensional parameter space used for BILBY-IS.

C.3.1 Group equivariant neural posterior estimation

DINGO uses an iterative algorithm called *group equivariant* NPE (GNPE) [82, 83] to integrate physical symmetries and thereby improve the accuracy of inference. With GNPE, we train a density estimation network $q(\theta|d, \hat{t}_I(\theta))$ that is also conditional on a set of GNPE *proxy parameters* $\hat{t}_I$. These parameters are defined as blurred versions of the coalescence times t_I in the individual interferometers (which can be computed as a function of θ) as

$$\hat{t}_I = t_I + \epsilon_I, \quad \epsilon_I \sim \kappa(\epsilon), \tag{C.17}$$

with $\kappa = U[-1\text{ ms}, 1\text{ms}]$. With GNPE, we iteratively infer the posterior $p(\theta, \hat{t}_I|d)$ in the joint parameter space with Gibbs sampling, and obtain the posterior over θ by marginalizing over $\hat{t}_I$. We use a paralllelized Gibbs sampler that typically converges after 30 iterations, but some events require up to 500 iterations. Each iteration corresponds to a forward pass through the density estimator $q(\theta|d, \hat{t}_I(\theta))$. 500 GNPE iterations for a batch of $5 \cdot 10^4$ samples take about 6 minutes on an A100 GPU.

In contrast to NPE, GNPE does not have a tractable density. To recover the density, we first generate $4 \cdot 10^5$ GNPE samples (48 minutes on one GPU or 6 minutes on eight GPUs for 500 iterations). We then train an unconditional normalizing flow $q(\hat{t}_I)$ to estimate the distribution over the inferred proxy parameters with a maximum likelihood objective. We use a neural spline flow with rational-quadratic spline coupling transforms [93] with the hyperparameters from Tab. C.1. Once trained, we can sample without the need for additional GNPE iterations via

$$\theta \sim q(\theta|d, \hat{t}_I), \ \hat{t}_I \sim q(\hat{t}_I). \tag{C.18}$$

The proposal density is now tractable,

$$\log q(\theta, \hat{t}_I|d) = \log q(\theta|d, \hat{t}_I) + \log q(\hat{t}_I). \tag{C.19}$$

We then perform IS in the *joint* parameter space $(\theta, \hat{t}_I)$, where the target density is given by

$$\begin{aligned}\log p(\theta, \hat{t}_I|d) = &- \log p(d) + \log p(d|\theta) + \log p(\theta) \\ &+ \sum_I \log \kappa(\hat{t}_I - t_I).\end{aligned} \tag{C.20}$$

The last term accounts for $p(\hat{t}_I|\theta)$. As described in the main part, we omit $\log p(d)$ and estimate this from the normalization of the weights.

Alternatively, we could also train an unconditional density estimator for the converged θ samples, but this is less sample efficient and more costly to train.

C.3.2 Stochastic samplers

We apply IS to BILBY-DYNESTY [41, 195, 210], which is based on nested sampling. To recover the density, we first generate $\approx 10^6$ posterior samples with 50 BILBY runs with identical settings. With `nlive=1000` and `nact=5`, this takes about one day per run on 10 CPUs, when using the IMRPhenomXPHM model. One typically uses larger `nact` for production results, but this substantially increases the computational cost. For reference, the runs for GW150914 and GW151012 reported in the main paper with `nlive=4000` and `nact=50` took about a week. We then estimate the distribution over the BILBY samples by training an unconditional normalizing flow $q(\theta)$, see Tab. C.1. To ensure a fair comparison with DINGO, we also use the analytic estimate for the phase parameter, such that we only need to estimate the distribution over the remaining 14 parameters. Due to the higher dimensional parameter space compared to DINGO (for which we only need to recover the two dimensional density over $\hat{t}_I$), we need more samples and a larger normalizing flow for the density estimate.

For GW151012, BILBY-IS achieves a sample efficiency $\epsilon = 8.3\%$, compared to $\epsilon = 12.5\%$ for DINGO-IS, and estimates an evidence of $\log p(d) = -16412.89 \pm 0.01$. Since BILBY-IS is computationally very expensive, we do not expect it to be routinely used, but rather view it as an insightful diagnostic.

C.4 Importance sampling convergence

Due to the probability mass covering training objective, DINGO inaccuracies tend to show up as overly broad posteriors (Fig. C.1). When the tails of the posterior are overestimated by DINGO, a low sample efficiency may be encountered due to many low-weight samples. These cases are straightforward to handle with DINGO-IS. The sample efficiency is approximately constant, and the statistical uncertainty of the evidence fully captures the error, even for low n_{eff}. To get smooth marginals one simply needs to generate more samples, which is cheap with DINGO.

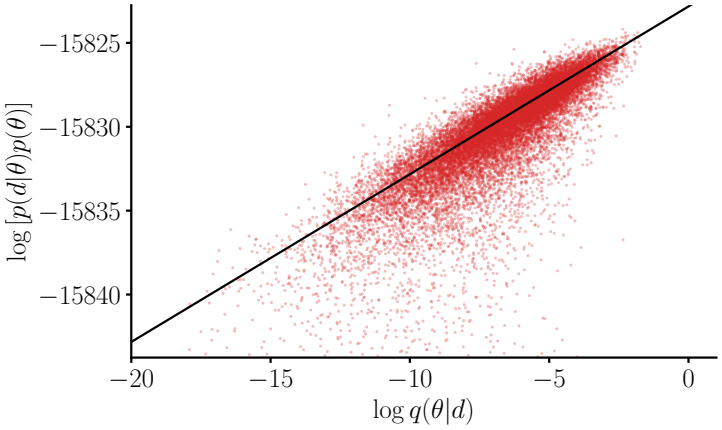


Fig. C.1 DINGO samples $\theta \sim q(\theta|d)$ for GW150914, comparing the inferred density $q(\theta|d)$ to the unnormalized posterior $p(d|\theta)p(\theta)$. The density ratios correspond to the importance weights, the Bayesian evidence $p(d)$ is estimated via their normalization. Samples of a perfect DINGO model would lie on the black line with offset $\log p(d) = -15831.87$. Deviations between DINGO and the true posterior are primarily found below that line, but rarely above. This is a manifestation of the probability-mass covering behavior, making DINGO particularly well-suited for importance sampling.

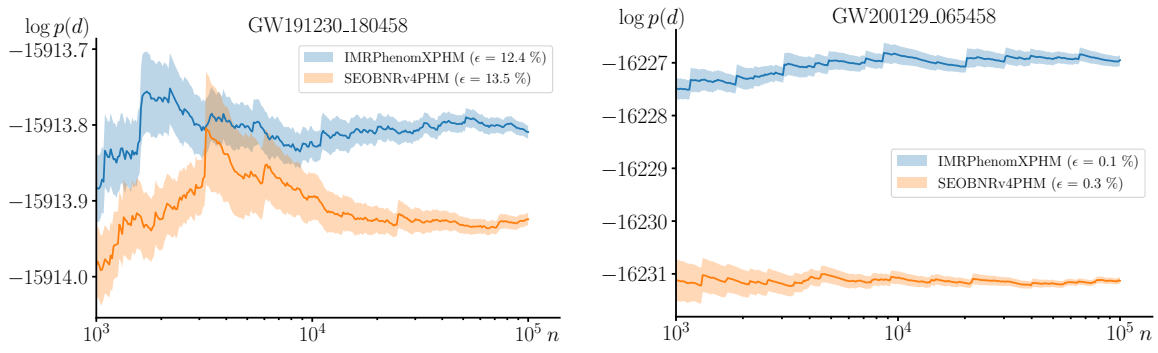


Fig. C.2 Evidence $\log p(d)$ as a function of the number of importance samples n. For constant sample efficiency ϵ, the statistical uncertainty scales with $1/\sqrt{n}$, leading to precise estimates when DINGO-IS works well (left). When the DINGO posterior is too light tailed (right), samples from the tails of the distribution are assigned very large IS weights, leading to bumps in the evidence whenever a high-weight sample is encountered.

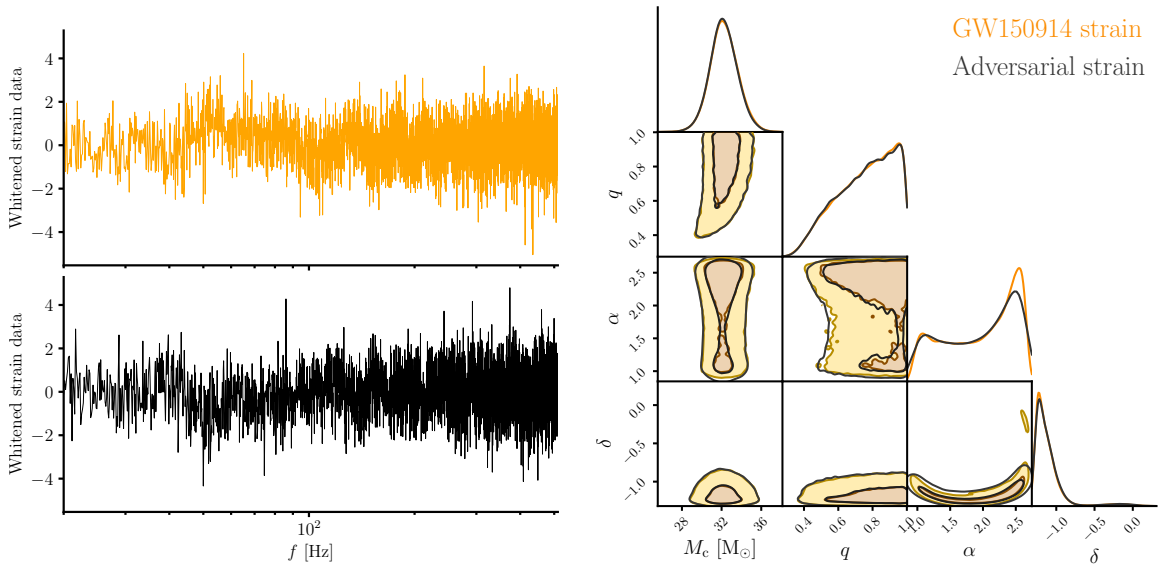


Fig. C.3 Left: Strain data (real part) in the LIGO Hanford detector. The upper row shows the measured data for GW150914, the lower row shows an adversarial example that is synthetically generated to mislead the inference network. Right: The inference network infers almost identical posteriors for both strain datasets.

In contrast, DINGO posteriors should rarely be light tailed. For real data, however, parts of the parameter space are occasionally strongly undersampled, which is problematic for IS. Indeed, for small n, the light tails may not be sampled at all, resulting in an underestimate of the evidence and the magnitude of its statistical error. Moreover, when a sample from the tail is encountered it has very large importance weight, which greatly decreases the sample efficiency. In order to assess the validity of IS results with low sample efficiency it is therefore useful to check whether $\log p(d)$ has converged as a function of n (Fig. C.2). If DINGO is not truly mass covering, the IS weights are not upper-bounded, and the sample efficiency approaches zero with increasing n. This happens for the OOD event GW200129_065458.

Fortunately non-convergence is rare, and for the majority of events, DINGO posteriors are indeed mass covering and heavy tailed. Even when this is not the case and the sample efficiency is very low, the DINGO marginals are often still accurate. This is because the light tailed parts of the parameter space are often negligibly small and randomly distributed throughout the parameter space. In such cases one can apply *batched* self-normalized IS: instead of normalizing the weights of all n samples simultaneously, one normalizes batches of size $k < n$. This regularizes IS by decreasing the largest possible weight from n to k. This should be done with caution, as it introduces a bias which is only small if the undersampled regions carry an overall low probability mass, or are distributed unsystematically throughout the parameter space.

C.5 Robustness to adversarial examples

An adversarial example [215, 109] refers to data $d_{\text{adv.}}$ that is specifically designed to mislead a neural network. Such examples can be generated by following gradients of the network output (or some function thereof) starting from some real data d_{true} and sequentially adding small perturbations to maximally change the output. Although the resulting adversarial example $d_{\text{adv.}}$ is often barely distinguishable from d_{true}, the neural network output can change dramatically.

In the context of posterior estimation, the output is a high-dimensional distribution which one can alter in multiple ways. We tried to shift or truncate the predicted DINGO distribution $q(\theta|d_{\text{adv.}})$ by applying only minimal modifications to the data. We found that DINGO is remarkably robust to such attacks, its output could barely be changed without significantly changing the input data d. This unusual robustness is attributed to two factors. First, the training data itself is very noisy, which regularizes DINGO models. Second, the first layer of DINGO networks is seeded with principal components of clean GW signals [82], so adversarial perturbations are projected onto the manifold of GW signals.

We thus explore a slightly different notion of adversarial attacks. Starting from strain data d initialized with random Gaussian noise, we aim to modify d such that DINGO estimates identical posteriors for $d_{\text{adv.}}$ and the real strain data d_{true} for GW150914. Specifically, we minimize the KL divergence $D_{\text{KL}}(q(\theta|d_{\text{true}})||q(\theta|d_{\text{adv.}}))$ via

$$d_{\text{adv.}} = \arg\max_d \mathbb{E}_{\theta\sim q(\theta|d_{\text{true}})} \log q(\theta|d). \tag{C.21}$$

In contrast to the technique mentioned above, we here do not constrain the difference between $d_{\text{adv.}}$ and d_{true} to be small. To optimize (C.21) we need to take gradients of the DINGO density with respect to d, which is intractable with the iterative GNPE [82, 83] method. Instead, we use a DINGO network trained with standard NPE. We use the adam [135] optimizer with a learning rate of 0.03 to optimize Eq. (C.21) with 400 gradient steps (batch size 1024). The resulting strain d_{adv} is visibly different from the true GW150914 strain d_{true}, but the estimated DINGO posteriors are almost identical (Fig. C.3).

With DINGO-IS, we find a sample efficiency of $\epsilon = 1.48\%$ for the real GW150914 strain d_{true}. This is substantially smaller than the sample efficiency achieved with GNPE ($\epsilon = 28.8\%$), since standard NPE does not use the physical symmetries and is hence less accurate. However, the DINGO-IS posterior is still accurate and the evidence estimate ($\log p(d) = -15831.88 \pm 0.03$) is in good agreement with the result reported in the main paper. For $d_{\text{adv.}}$ on the other hand, DINGO-IS achieves a sample efficiency of $\epsilon = 0.006\%$, clearly identifying the adversarial example as a DINGO failure case.

C.6 Additional Results

Fig. C.4 shows one-dimensional marginal posteriors for a subset of GW events analyzed in the main paper, comparing the two waveform models IMRPhenomXPHM and SEOBNRv4PHM. We see that the models give results that appear to be in good agreement.

Fig. C.5 shows posterior marginals for several GW events from O3. A large sample efficiency often corresponds to good agreement of the DINGO and DINGO-IS marginals. The sample efficiency is sensitive to deviations in the full 15 dimensional parameter space, so small sample efficiencies do not necessarily imply inaccurate *marginal* distributions.

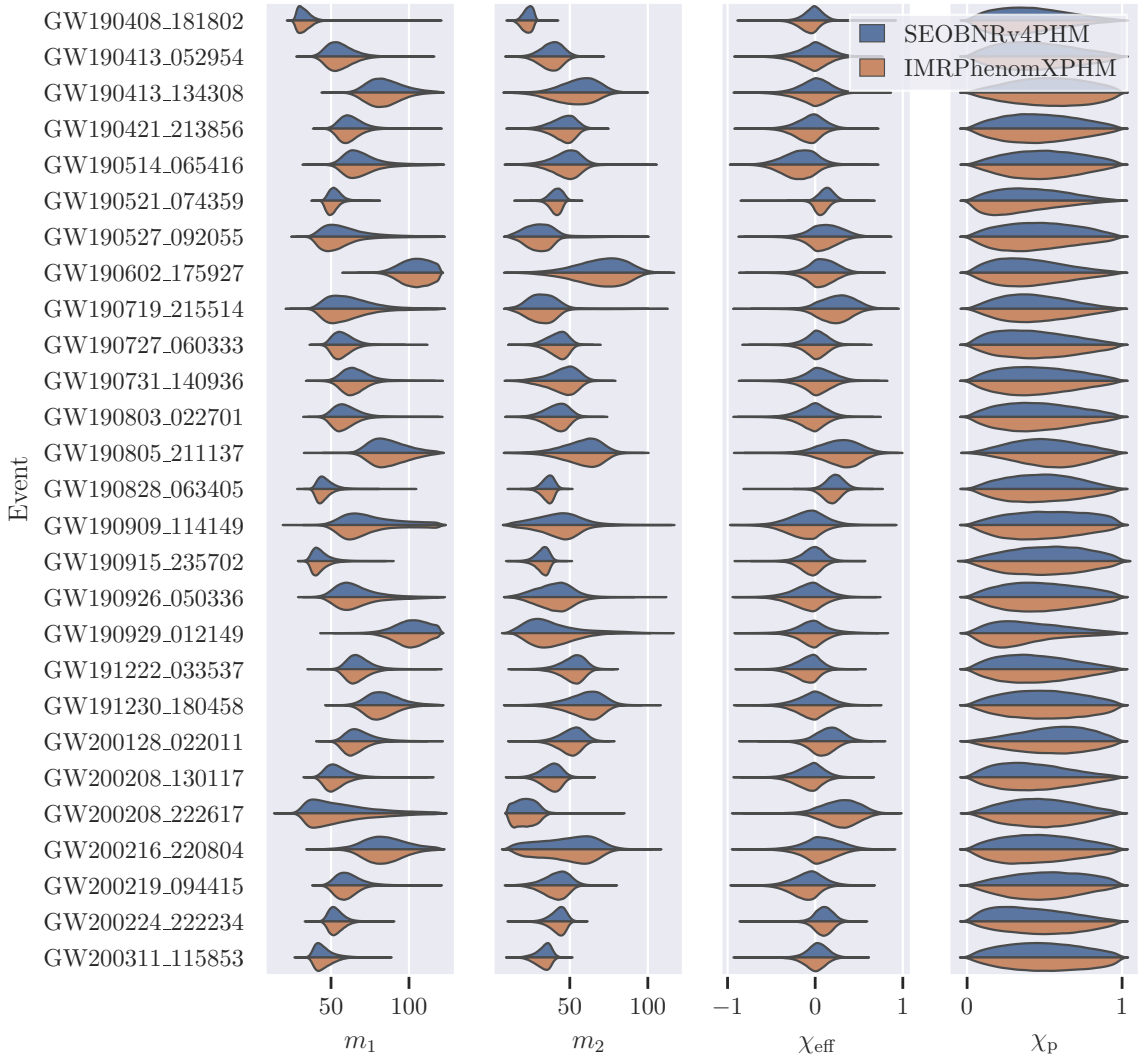


Fig. C.4 Posterior distributions for component masses and effective spin parameters. We show all events from the main paper with $\epsilon > 2\%$ for both waveform models to ensure smooth posteriors. We observe good agreement between the two waveform models. In a future publication we will include a more complete catalog, which incorporates Virgo data, and includes a more careful treatment of noise artifacts and data conditioning.

Fig. C.5 Marginalized one- and two- dimensional posterior distributions for selected O3 events, comparing DINGO (solid lines) and DINGO-IS (dashed) inference results with waveform models IMRPhenomXPHM and SEOBNRv4PHM. Contours represent 90% credible regions. For events with high (upper row) or medium (middle row) sample efficiency, the initial DINGO results are often accurate and only deviate slightly from DINGO-IS results. For events with low effective sample size (lower row), the DINGO-IS contours are often not smooth. Yet, the initial DINGO results may capture the marginals well, see GW191109_010717.

Appendix D

Real-Time Gravitational-Wave Inference for Binary Neutron Stars using Machine Learning

D.1 Machine learning framework

The Bayesian posterior $p(\theta|d) = p(d|\theta)p(\theta)/p(d)$ is defined in terms of a prior $p(\theta)$ and a likelihood $p(d|\theta)$. For GW inference, the likelihood is constructed by combining models for waveforms and detector noise. The Bayesian evidence $p(d)$ corresponds to the normalization of the posterior, and it can be used for model comparison.

Our framework is based on neural posterior estimation (NPE) [175, 152, 112], which trains a density estimation neural network $q(\theta|d)$ to estimate $p(\theta|d)$. We parameterize $q(\theta|d)$ with a conditional normalizing flow [193, 93]. Training minimizes the loss $L = -\log q(\theta|d)$ across a dataset (θ_i, d_i) of parameters $\theta_i \sim p(\theta)$ paired with corresponding likelihood simulations $d_i \sim p(d|\theta_i)$. After training, $q(\theta|d)$ serves as a surrogate for $p(\theta|d)$, and inference for any observed data d_o can be performed by sampling $\theta \sim q(\theta|d_\mathrm{o})$. DINGO [82, 84] uses a group-equivariant formulation of NPE (GNPE [83, 82]), which simplifies GW data by aligning coalescence times in the different detectors. However, this comes at the cost of longer inference times, so we do not use GNPE for DINGO-BNS.

At inference, we correct for potential inaccuracies of $q(\theta|d)$ with importance sampling [84], by assigning weight $w_i = p(d|\theta_i)p(\theta_i)/q(\theta_i|d)$ to each sample $\theta_i \sim q(\theta_i|d)$. A set of n weighted samples (w_i, θ_i) corresponds to $n_\mathrm{eff} = \left(\sum_i w_i\right)^2 / \left(\sum_i w_i^2\right)$ *effective* samples from the posterior $p(\theta|d)$. This reweighting enables asymptotically exact results, and the sample efficiency $\epsilon = n_\mathrm{eff}/n$ serves as a performance metric. The normalization of the weights further provides an unbiased estimate of the Bayesian evidence $p(d) = \left(\sum_i w_i\right)/n$.

Below, we describe in more detail the technical innovations of DINGO-BNS that enable scaling of this framework to BNS signals.

D.1.1 Prior conditioning

An NPE model $q(\theta|d)$ estimates the posterior $p(\theta|d)$ for a fixed prior $p(\theta)$. Choosing a broad prior enhances the general applicability of the NPE model, but it also implies worse tuning to specific events (for which smaller priors may be sufficient). This is a general trade-off in NPE, but it is particularly dramatic for BNS inference, where typical events constrain the chirp mass to $\sim 10^{-3}$ of the prior volume. Thus, for an individual BNS event, a tight chirp mass prior would have been sufficient (Fig. D.1b) and moreover would have enabled effective heterodyning [69, 70, 235]; but in order to cover generic BNS events, we need to train the NPE network with a large prior (see Tab. D.1). We resolve this trade-off with a new technique called prior conditioning. The key idea is to train an NPE model with *multiple different* (restricted) priors simultaneously. On each of these priors, we are allowed to apply an independent transformation to the data, which we use to heterodyne the GW strain with respect to the approximate chirp mass. Training a prior-conditioned model requires hierarchical sampling

$$\theta \sim p_\rho(\theta),\ \rho \sim \hat{p}(\rho), \tag{D.1}$$

where $p_\rho(\theta)$ is a prior family parameterized by ρ and $\hat{p}(\rho)$ is a corresponding hyperprior. We additionally condition the NPE model $q(\theta|d,\rho)$ on ρ. This model can then perform inference for any desired prior $p_\rho(\theta)$, by simply providing the corresponding ρ. This effectively amortizes the training cost over different choices of the prior.

We apply prior conditioning for the chirp mass $\mathcal{M}$, using a set of priors $p_{\widetilde{\mathcal{M}}}(\mathcal{M}) = U_{m_1,m_2}(\widetilde{\mathcal{M}} - \Delta\mathcal{M}, \widetilde{\mathcal{M}}+\Delta\mathcal{M})$. Here, $U_{m_1,m_2}(\mathcal{M}_{\text{min}}, \mathcal{M}_{\text{max}})$ denotes a distribution over $\mathcal{M}$ with support $[\mathcal{M}_{\text{min}}, \mathcal{M}_{\text{max}}]$, within which component masses m_1, m_2 are uniformly distributed. We use fixed $\Delta\mathcal{M} = 0.005\ \text{M}_\odot$ and choose a hyperprior $\hat{p}(\widetilde{\mathcal{M}})$ covering the expected range of $\mathcal{M}$ for LVK detections of BNS (see Tab. D.1). As $\Delta\mathcal{M}$ is small, $\widetilde{\mathcal{M}}$ is a good approximation for any $\mathcal{M}$ within the restricted prior $p_{\widetilde{\mathcal{M}}}(\mathcal{M})$ and we can thus use $\widetilde{\mathcal{M}}$ for heterodyning. The resulting model $q(\theta|d_{\widetilde{\mathcal{M}}}, \widetilde{\mathcal{M}})$ can then perform inference with event-optimized heterodyning and prior (via choice of appropriate $\widetilde{\mathcal{M}}$), but is nevertheless applicable to the entire range of the hyperprior.

Inference results are independent of $\widetilde{\mathcal{M}}$ as long as the posterior $p(\mathcal{M}|d)$ is fully covered by $[\widetilde{\mathcal{M}} - \Delta\mathcal{M}, \widetilde{\mathcal{M}} + \Delta\mathcal{M}]$. For BNS, $p(\mathcal{M}|d)$ is typically tightly constrained and we can use a coarse estimate of $\mathcal{M}$ for $\widetilde{\mathcal{M}}$. This can either be taken from a GW search pipeline or rapidly computed from $q(\theta|d_{\widetilde{\mathcal{M}}}, \widetilde{\mathcal{M}})$ itself by sweeping the hyperprior (see below). Note that for shorter GW signals from black hole mergers, $p(\mathcal{M}|d)$ is generally less well constrained. Transfer of prior conditioning would thus require larger (and potentially flexible) values of $\Delta\mathcal{M}$. Alternatively, the prior range can be extended at inference time by iterative Gibbs sampling of $\mathcal{M}$ and $\widetilde{\mathcal{M}}$, similar to the GNPE algorithm [82, 83].

Prior conditioning is a general SBI technique that enables choice of prior at inference time. This can also be achieved with sequential NPE [175, 152, 112, 85]. However, in contrast to prior condditioning, these techniqes require simulations and retraining for each observation, resulting in more expensive and slower inference. We here use prior conditioning with priors of fixed width for the chirp mass,

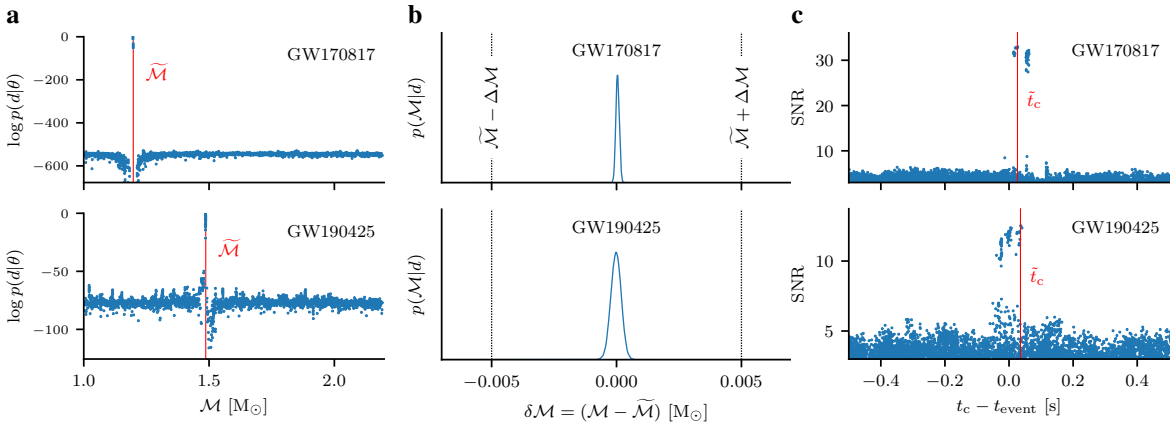


Fig. D.1 (a) Log likelihoods generated from a scan over different values of $\widetilde{\mathcal{M}}$ with a DINGO-BNS network. The final $\widetilde{\mathcal{M}}$ is chosen as the maximum likelihood $\mathcal{M}$ (red line; $\widetilde{\mathcal{M}} = 1.1975\ \mathrm{M}_\odot$ for GW170817, $\widetilde{\mathcal{M}} = 1.4868\ \mathrm{M}_\odot$ for GW190425). (b) Posterior marginal $p(\mathcal{M}|d)$. The prior (dashed lines) determined by the scan from (a) fully covers the marginal. (c) A combined scan over $\mathcal{M}$ and t_c successfully identifies GW170817 (with $\hat{t}_c = 1187008882.43$) and GW190425 (with $\hat{t}_c = 1240215503.04$).

and optional additional conditioning on fixed values for other parameters (corresponding to Dirac delta priors). Extension to more complicated priors and hyperpriors is straightforward.

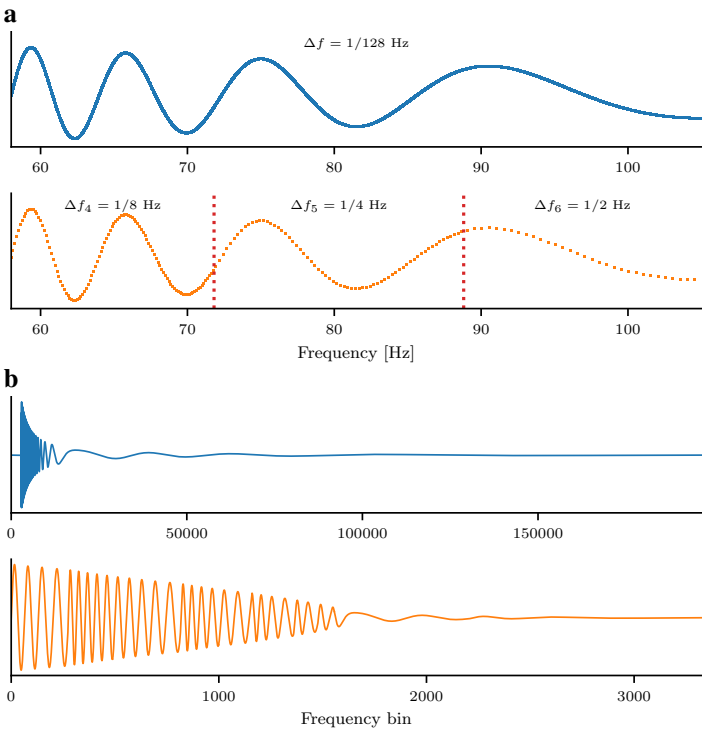
D.1.2 Independent estimation of chirp mass and merger times

Running DINGO-BNS requires an initial estimate of the chirp mass $\mathcal{M}$ (to determine $\widetilde{\mathcal{M}}$ for the network) and the merger time t_c (to trigger the analysis). Matched filter searches can identify the presence of a compact binary signal signal and its chirp mass and merger time in low-latency [100, 166, 58, 34, 66]. Specialized "early warning" searches designed to produce output before the coalescence can further provide a rough indication of sky position and distance [196, 167, 140]. When available, output of such pipelines can be used to trigger a DINGO analysis and provide estimates for $\mathcal{M}$ and t_c.

We here describe an alternative independent approach of obtaining these parameters, using only the trained DINGO-BNS model. We compute $\widetilde{\mathcal{M}}$ by sweeping the entire hyperprior $\hat{p}(\widetilde{\mathcal{M}}) = U_{m_1,m_2}(\widetilde{\mathcal{M}}_{\mathrm{min}}, \widetilde{\mathcal{M}}_{\mathrm{max}})$. Specifically, we run DINGO-BNS with a set of prior centers

$$\widetilde{\mathcal{M}}_i = \widetilde{\mathcal{M}}_{\mathrm{min}} + i \cdot \Delta\mathcal{M},\ i \in [0, (\widetilde{\mathcal{M}}_{\mathrm{max}} - \widetilde{\mathcal{M}}_{\mathrm{min}})/\Delta\mathcal{M}]. \tag{D.2}$$

The inference models in this study are trained with hyperprior ranges of up to $[1.0, 2.2]\,\mathrm{M}_\odot$. For $\Delta\mathcal{M} = 0.005\ \mathrm{M}_\odot$, we can thus cover the entire global chirp mass range using 241 (overlapping) local priors. We run DINGO-BNS for all local priors $\widetilde{\mathcal{M}}_i$ in parallel, with 10 samples per $\widetilde{\mathcal{M}}_i$. This requires DINGO-BNS inference of only a few thousand samples, which takes less than one second. We use the chirp mass $\mathcal{M}$ of the maximum likelihood sample as the prior center $\widetilde{\mathcal{M}}$ for the analysis



c

Δf_i [Hz]	LVK $\hat{f}_i$ [Hz]	CE $\hat{f}_i$ [Hz]
1/8192		5
1/4096		5
1/2048		5
1/1024		7
1/512		10
1/256		13
1/128	19	18
1/64	25	24
1/32	34	32
1/16	44	42
1/8	57	54
1/4	72	68
1/2	89	84
1	420	364
2		765

Fig. D.2 Frequency multibanding. (a) The period of (heterodyned) GW signals decreases with increasing frequency. The native frequency resolution (blue) thus oversamples the signal at high frequencies. Frequency multibanding (band boundaries indicated by dotted red lines) adapts to the signal variation, decreasing the resolution at higher frequencies (orange). (b) The multibanded domain therefore requires fewer frequency bins, and the signal variation is more homogeneous across bins. (c) Multibanded frequency domain partitions for LVK ($f_{\text{min}} = 19.4$ Hz, compression factor ~ 60) and CE ($f_{\text{min}} = 5$ Hz, compression factor ~ 650) experiments. We use a smaller chirp mass prior for the CE experiments (Tab. D.1), which allows a slightly coarser resolution compared to LVK (corresponding to lower $\hat{f}_i$). The first two bands for CE are skipped entirely, which is a consequence of the reduced signal variation with heterodyning.

(Fig. D.1a). Note that the exact choice of $\widetilde{\mathcal{M}}$ does not matter, as long as the inferred posterior is fully covered by $[\widetilde{\mathcal{M}} - \Delta\mathcal{M}, \widetilde{\mathcal{M}} + \Delta\mathcal{M}]$ (Fig. D.1b).

The merger time t_c can be inferred by continuously running this $\widetilde{\mathcal{M}}$ scan on the input data stream, sliding the t_c prior in real time over the incoming data. With inference times of one second, continuous analysis could be achieved on just a few parallel computational nodes, constantly running on the input data stream. Event candidates can then be identified by analyzing the SNR, triggering upon exceeding some defined threshold (Fig. D.1c). This scan could be performed at an arbitrary (but fixed) time prior to the merger.

This scan successfully estimates $\mathcal{M}$ and t_c for both real BNS events (Fig D.1). However, we have not tested this at a large scale on detector noise to compute false alarm rates, as DINGO-BNS is primarily intended for parameter estimation. Existing search and early warning pipelines are likely more robust for event identification, in particular in the presence of non-stationary detector noise.

D.1.3 Frequency multibanding

Although the native resolution of a frequency series is determined by the duration T of the corresponding time series ($\Delta f = 1/T$), we can average adjacent frequency bins wherever the signal is roughly constant. This enables data compression with only negligible loss of information. We here employ frequency multibanding, which divides the frequency range $[f_{\text{min}}, f_{\text{max}}]$ into N bands of decreasing resolution. Frequency band i covers the range $[\hat{f}_i, \hat{f}_{i+1})$ with $\Delta f_i = 2^i \Delta f_0$, where $\hat{f}_0 = f_{\text{min}}$, $\hat{f}_N = f_{\text{max}}$ and Δf_0 is the native resolution of the frequency series. Within a band i, the multibanded domain thus compresses the data by a factor of 2^i (Fig. D.2), which is achieved by averaging 2^i sequential bins from the original frequency series ("decimation"). To achieve optimal compression, we empirically choose the smallest possible nodes $\hat{f}_i$ for which GW signals are still fully resolved. Specifically, we simulate a set of 10^3 heterodyned GW signals and demand that every period of these signals is covered by at least 32 bins in the resulting multibanded frequency domain. This is done before generating the training dataset, and the multibanded domain then remains fixed during dataset generation and training. The optimized resolution achieves compression factors between 60 and 650 (Fig. D.2c). Care needs to be taken that our approximations are valid in the presence of detector noise. We now investigate how multibanding affects *data simulation* (for training) and the *likelihood* (for importance sampling).

Data simulation

GW data is simulated as the sum of a signal and detector noise, $d = h(\theta) + n$. The detector noise in frequency bin j is given by

$$n_j \sim \mathcal{N}(0, \sigma\sqrt{S_j}), \quad \sigma = \sqrt{\frac{w}{4\Delta f}}, \tag{D.3}$$

where S denotes the detector noise PSD and σ takes into account the frequency resolution and the Tukey window factor w. Note that n is a complex frequency series, which we ignore in our notation, as the considerations here hold for real and imaginary part individually. It is conventional to work with whitened data

$$d_j^{\text{w}} = h_j^{\text{w}}(\theta) + n_j^{\text{w}} = \frac{h_j(\theta) + n_j}{\sqrt{S_j}}, \tag{D.4}$$

in which case $n_j^{\text{w}} \sim \mathcal{N}(0, \sigma)$.

We convert to multibanded frequency domain by averaging sets of $N_i = 2^i$ bins,

$$\overline{d_j^{\text{w}}} = \frac{1}{N_i} \sum_{k=m_j}^{m_j+N_i-1} (h_k^{\text{w}} + n_k^{\text{w}}) = \overline{h_j^{\text{w}}} + \overline{n_j^{\text{w}}}, \tag{D.5}$$

where j denotes the bin in the multibanded domain, m_j denotes the starting index of the decimation window for j in the native domain, and i indexes the frequency band associated with j. Since $\overline{n_j^{\text{w}}}$ is an average of N_i Gaussian random variables with standard deviation σ, it follows that $\overline{n_j^{\text{w}}}$ is also

Gaussian with standard deviation

$$\sigma_i = \sigma/\sqrt{N_i} = \sqrt{\frac{w}{4\Delta f N_i}} = \sqrt{\frac{w}{4\Delta f_i}}. \tag{D.6}$$

We can thus simulate the detector noise directly in the multibanded domain, by updating $\sigma \to \sigma_i$, corresponding to $\Delta f \to \Delta f_i$. For the whitened signal we find

$$\overline{h_j^{\mathrm{w}}} = \frac{1}{N_i} \sum_{k=m_j}^{m_j+N_i-1} \frac{h_k}{\sqrt{S_k}} \approx \overline{h_j} \sum_{k=m_j}^{m_j+N_i-1} \frac{1}{\sqrt{S_k}}, \tag{D.7}$$

assuming an approximately constant signal h within the decimation window, $\overline{h_j} \approx h_k, \forall k \in [m_j, m_j + N_i - 1]$. For frequency-domain waveform models, We can thus directly compute the signal $\overline{h}_j$ in the multibanded domain by simply evaluating the model at frequencies $\overline{f_j}$. For whitening, we replace $1/\sqrt{S} \to \overline{1/\sqrt{S}}$.

In summary, we can directly generate BNS data in the multibanded frequency domain, by (1) updating the noise standard deviation according to the multibanded resolution, (2) appropriately decimating noise PSDs and (3) computating signals and noise realizations in the compressed domain. These operations are carefully designed to be consistent with data processing of real BNS observations, which for DINGO-BNS are first whitened in the native domain and then decimated to the multibanded domain. This process relies on the assumption that signals are constant within decimation windows, and we ensure that this is (approximately) fulfilled when determining the multibanded resolution. Indeed, for signals generated directly in the multibanded domain we find mismatches of at most $\sim 10^{-7}$ when comparing to signals that are properly decimated from the native domain.

Likelihood evaluations

We also use frequency multibanding to evaluate the likelihood for importance sampling. The standard Whittle likelihood used in GW astronomy [18] reads

$$\log p(d|\theta) = -\frac{1}{2} \sum_k \frac{|d_k^{\mathrm{w}} - h_k^{\mathrm{w}}(\theta)|^2}{\sigma^2}, \tag{D.8}$$

up to a normaliztion constant. The sum extends over all bins k in the native frequency domain. Assuming a constant signal (as above) and PSD within each decimation window, we can directly compute the likelihood in the multibanded domain

$$\log p(d|\theta) \approx -\frac{1}{2} \sum_j \frac{|\overline{d_j^{\mathrm{w}}} - \overline{h_j^{\mathrm{w}}}(\theta)|^2}{\sigma_{i(j)}^2}. \tag{D.9}$$

The assumptions are not exactly fulfilled in practice; for additional corrections see [161]. For importance sampling, we can always evaluate the exact likelihood in the native frequency domain instead. In this case, the result is no longer subject to any approximations, even if the DINGO-BNS proposal is generated with a network using multibanded data. With the full likelihood for GW170817,

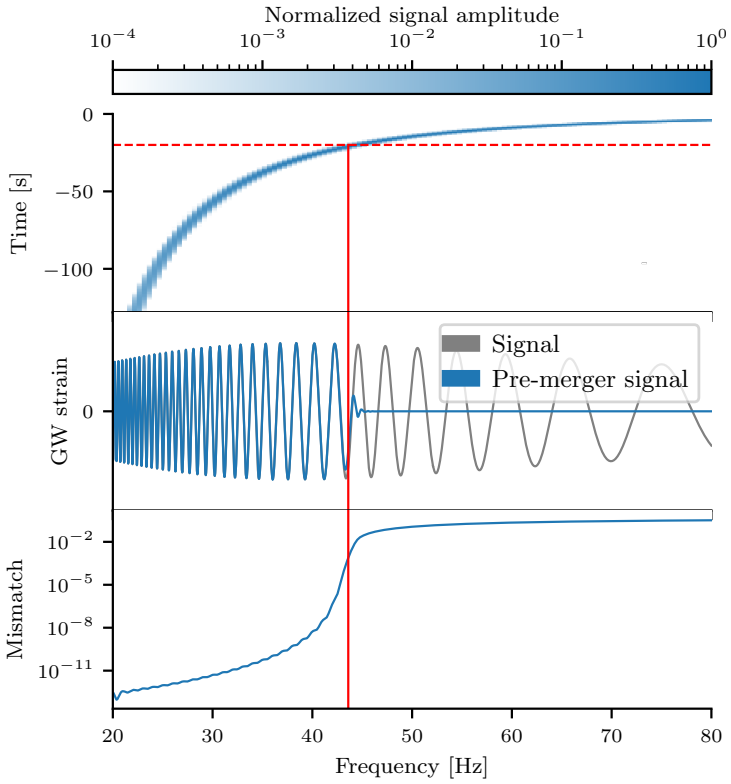


Fig. D.3 Time-domain truncation of BNS signals at time t_{max} (red dashed line) before the merger can be approximated by truncation at a corresponding maximum frequency (red solid line) in frequency domain. Below frequency $f_{\text{max}}(t, \mathcal{M})$, the truncated signal (blue in center panel) matches the original signal (gray). Above $f_{\text{max}}(t, \mathcal{M})$, the amplitude of the truncated signal quickly approaches zero. We determine $f_{\text{max}}(t, \mathcal{M})$ empirically, by allowing mismatches between truncated and original signals of at most 10^{-3} (lower panel). Analogously, truncation for $t < t_{\text{min}}$ can be achieved by imposing a minimum frequency cutoff $f_{\text{min}}(t, \mathcal{M})$.

we find a sample efficiency of 11.0% with an inference time of 13 seconds for 50,000 samples. The deviation from the result obtained with the multibanded likelihood is negligible (Jensen-Shannon divergence less than $5 \cdot 10^{-4}$ nat for all parameters). This demonstrates that use of the multibanded resolution has no practically relevant impact on the results.

D.1.4 Frequency masking

Since the GW likelihood (and our framework) use frequency domain, but data are taken in time domain, it is necessary to convert data by windowing and Fourier transforming. However, frequency domain waveform models assume infinite time duration, leading to inconsistencies with finite time segments $[t_{\text{min}}, t_{\text{max}}]$. As the frequency evolution of the inspiral is tightly constrained by the chirp mass $\mathcal{M}$, we can compute boundaries $f_{\text{min}}(t_{\text{min}}, \mathcal{M})$ and $f_{\text{max}}(t_{\text{max}}, \mathcal{M})$, such that signals are not corrupted by finite-duration effects within $[f_{\text{min}}, f_{\text{max}}]$, and are negligibly small outside of that range (Fig. D.3).

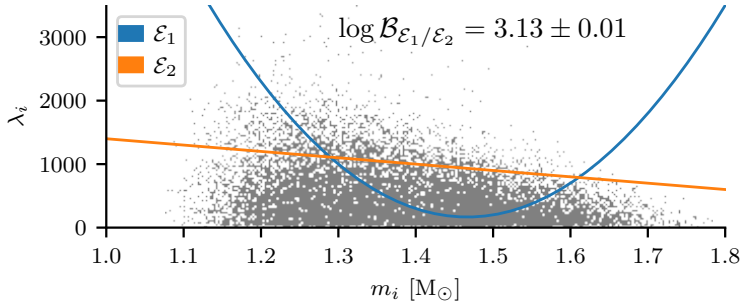


Fig. D.4 Neutron-star EOS imply a functional relation $\lambda_i = \lambda_i(m_i)$ between tidal parameters λ_i and component masses m_i. The likelihood $p(d|\mathcal{E})$ for an EOS $\mathcal{E}$ given the GW data d requires integrating the posterior $p(\theta|d)$ along the corresponding hyperplane. No posterior samples (gray) will be exactly on the corresponding hyperplane (exemplary coloured lines), hence the standard Bayesian inference techniques are not directly applicable [121]. DINGO-BNS provides various possibilities to directly compute this quantity, enabling comparison of different EOS in terms of the Bayes factors $\mathcal{B}$.

We approximate the lower bound $f_{\text{min}}(t_{\text{min}}, \mathcal{M})$ using the leading order in the post-Newtonian relationship between time and frequency,

$$f_{\text{0PN}}(t, \mathcal{M}) = \frac{1}{8\pi} \left(\frac{-t}{5}\right)^{-3/8} \left(\frac{G\mathcal{M}}{c^3}\right)^{-5/8}. \tag{D.10}$$

For a network designed for fixed data duration T, we set $f_{\text{min}}(T, \mathcal{M}) = f_{\text{0PN}}(-T, \mathcal{M}) + f_{\text{buffer}}$ (we use $f_{\text{buffer}} = 1\,\text{Hz}$ for LVK and $f_{\text{buffer}} = 0.5\,\text{Hz}$ for XG setups).

For the upper bound, we found that $f_{\text{0PN}}(t, \mathcal{M})$ is not sufficiently accurate. Instead, we determine $f_{\text{max}}(t, \mathcal{M})$ empirically by simulating a set of signals (with parameters $\theta \sim p(\theta)$), and computing mismatches between signals with and without truncation at $t > t_{\text{max}}$. For a given set of simulations, we choose $f_{\text{max}}(t, \mathcal{M})$ as the highest frequency for which all mismatches are at most 10^{-3}. To avoid additional computation at inference time, we cache the results in a lookup table for $f_{\text{max}}(t, \mathcal{M})$.

Both bounds depend on the chirp mass $\mathcal{M}$, and the upper bound additionally depends on the pre-merger time. To enable inference for arbitrary configurations, we train a single network with variable frequency bounds. During training, we compute $f_{\text{min}}(T, \widetilde{\mathcal{M}})$ with the center $\widetilde{\mathcal{M}}$ of the local chirp mass prior. The upper frequency bound f_{max} is sampled randomly (uniform in frequency bins of the multibanded frequency domain) to allow for arbitrary pre-merger times. Data outside of $[f_{\text{min}}, f_{\text{max}}]$ is zero-masked.

D.1.5 Equation-of-state likelihood

A nuclear equation of state (EOS) implies a functional relationship between neutron star masses m_i and tidal deformabilities λ_i. The likelihood $p(d|\mathcal{E})$ for a given EOS $\mathcal{E}$ and data d can be computed by inte-

grating the GW likelihood along the hyperplane defined by the EOS constraint $\lambda_i = \lambda_i^{\mathcal{E}}(m_i)$,

$$\begin{aligned} p(d|\mathcal{E}) &= \int p(d|\theta)p(\theta)\delta(\lambda_i - \lambda_i^{\mathcal{E}}(m_i))\,\mathrm{d}\theta \\ &= \int p(d|m_1, m_2, \lambda_1^{\mathcal{E}}(m_1), \lambda_2^{\mathcal{E}}(m_2))\,\mathrm{d}m_1\mathrm{d}m_2. \end{aligned} \tag{D.11}$$

Here, $p(d|m_1, m_2, \lambda_1, \lambda_2)$ is the Bayesian evidence of d conditional on $(m_1, m_2, \lambda_1, \lambda_2)$. To calculate (D.11) using Monte Carlo integration, it is necessary to repeatedly evaluate the integrand, which is extremely expensive using traditional methods (e.g., nested sampling).

With DINGO-BNS, there are two fast ways to evaluate the integrand, using either a conditional or a marginal network: (1) a marginal network $q(m_1, m_2, \lambda_1, \lambda_2|d)$ directly provides an unnormalized estimate of the conditional evidence $p(d|m_1, m_2, \lambda_1, \lambda_2)$ (sufficient for model comparison, but not subject to our usual accuracy guarantees); or (2) a conditional network $p(\theta|d; m_1, m_2, \lambda_1, \lambda_2)$ provides the normalized conditional evidence via importance sampling (including accuracy guarantees). Option (1) allows for 10^5 evaluations per second, whereas option (2) only allows for 10^3 assuming 10^2 weighted samples per evaluation. By combining (1) and (2), we can achieve speed and accuracy, by using the marginal network (1) to define a histogram proposal for Monte Carlo integration with the integrand from (2). We test this on GW170817 data with two polynomial EOS constraints $\lambda = \lambda^{\mathcal{E}}(m)$ (Fig. D.4), finding good sample efficiencies of $\sim 50\%$, small uncertainties $\sigma_{\log p(d|\mathcal{E})} \sim 0.01$ and computation times of $1-3$ s for the integral (D.11). Alternatively, the proposal could also be generated with a network $q(m_1, m_2|d)$, which additionally marginalizes over λ_i. Finally, for a parametric EOS, a DINGO-BNS network could be conditioned on EOS parameters, allowing for direct EOS inference. This variety of approaches emphasizes the flexibility of SBI for EOS inference.

D.1.6 Related work

Machine learning for GW astronomy is an active area of research [80]. Several studies explore machine learning inference for black hole mergers [103, 111, 86, 111, 110, 82, 231, 84, 37, 79, 46, 229, 138]. There have also been applications specifically to BNS inference, notably the GW-SkyLocator algorithm [60], which estimates the sky position using the SNR time series (similar to Bayestar), and JIM [232, 234], which uses hardware acceleration and machine learning to speed up conventional samplers and achieve full inference in 25 minutes. The ASTREOS framework uses machine learning for BNS equation-of-state inference [157]. Pre-merger localization with conventional techniques has also been explored in [125].

D.2 Experimental details

For our experiments, we train DINGO-BNS networks using the hyperparameters and neural architecture [118, 93] from Ref. [82], with a slightly larger embedding network. For the LVK experiments, we use a dataset with $3 \cdot 10^7$ training samples and train for 200 epochs, for CE we use $6 \cdot 10^7$ training samples and train for 100 epochs. We use three detectors for LVK (LIGO-Hanford, LIGO-Livingston, and Virgo) and two detectors for CE (primary detector at location of LIGO-Hanford, secondary

	LVK	CE
$\mathcal{M}$ [$M_\odot$]	$[1.0, 2.2]$	$[1.15, 1.25]$
m_1	[1.0, 3.2]	[0.95, 2.4]
m_2	[1.0, 2.0]	[0.95, 2.4]
$a_{1,2}$	[0, 0.05]	[0, 0.05]
λ_1	[0, 5000]	[0, 5000]
λ_2	[0, 10000]	[0, 10000]
d_L [Mpc]	[10, 100]	[20, 50] / [1000, 2000]
t_c [s]	[-0.1, 0.1] / [-0.03, 0.03]	[-1,1] / [-0.03, 0.03]

Table D.1 Training priors for chirp mass $\mathcal{M}$, component masses $m_{1,2}$, spin magnitudes $a_{1,2}$, tidal deformabilities $\lambda_{1,2}$, luminosity distance d_L and merger time t_c. All priors are uniform, except for chirp mass, which is sampled uniform in component masses. At inference, d_L can be reweighted to the standard prior (uniform in comoving volume). For t_c we use a broader prior for pre-merger inference than for full inference (separated by "/" symbol) to account for higher uncertainties. LVK priors are chosen to cover expected LVK BNS detections. CE priors for $\mathcal{M}$ and d_L are reduced compared to LVK to decrease the computational cost of training. Priors for parameters not displayed here are standard.

detector at location of LIGO-Livingston). The networks are trained with the priors displayed in Tab. D.1.

In the first experiment, we evaluate DINGO-BNS models on 200 simulated GW datasets, generated using a fixed GW signal with GW170817-like parameters and simulated LVK detector noise. We use noise PSDs from the second (O2) and third (O3) LVK observing runs as well as LVK design sensitivity. For each noise level, we train one pre-merger network ($f \in [23, 200]$ Hz) and one network for inference with the full signal, including the merger ($f \in [23, 1024]$). The latter network is only used for after-merger inference, as we found that separation into two networks improves the performance. The pre-merger network is trained with frequency masking with the masking bound f_max sampled in range $[28, 200]$ Hz, enabling inference up to 60 seconds before the merger.

In the second experiment, we analyze 1000 simulated GW datasets, with GW signal parameters randomly sampled from the prior (Tab. D.1; $\mathcal{M}$ prior reduced to range $[1.0, 1.5]$ $M_\odot$ and d_L prior reweighted to a uniform distribution in comoving volume) and with design sensitivity noise PSDs. We again train one pre-merger network ($f \in [19.4, 200]$ Hz) and one after-merger network ($f \in [19.4, 1024]$ Hz). The pre-merger network is trained with frequency masking with the masking bound f_max sampled in range $[25, 200]$ Hz, enabling inference up to 60 seconds before the merger for $\mathcal{M} \leq 1.5$ $M_\odot$. Both networks are additionally trained with lower frequency masking, with $f_\mathrm{min}(\widetilde{\mathcal{M}})$ determined as explained above, ensuring an optimal frequency range for any chirp mass. For each DINGO-BNS result, we generate a skymap using a kernel density estimator implemented by `ligo.skymap` [202]. For the sky localization comparison between DINGO-BNS and Bayestar, we run Bayestar based on the GW signal template generated with the maximum likelihood parameters from the DINGO-BNS analysis. We note that Bayestar is designed as a low-latency pipeline and typically run with (coarser) parameter estimates from search templates. Therefore, the reported Bayestar

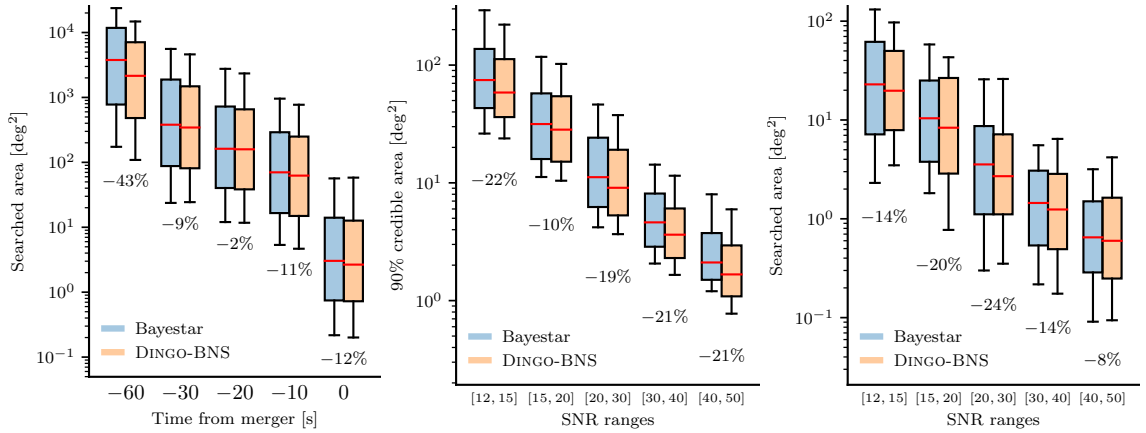


Fig. D.5 Localization comparison between Bayestar and DINGO-BNS, in terms of the 90% credible area and the searched area. The comparisons according to SNR are based only on results after the merger.

runs may deviate slightly from the realistic LVK setup. However, our results are consistent with Ref. [162], which also found a $\sim 30\%$ precision improvement over Bayestar localization (using LVK search triggers). Both, DINGO-BNS and Ref. [162], perform full Bayesian BNS inference and should therefore have identical localization improvements over Bayestar (assuming ideal accuracy, which for DINGO-BNS is validated with consistently high importance sampling efficiency). Differences to the localization comparison in Ref. [162] are thus primarily attributed to different configurations for Bayestar and slightly different injection priors. Additional results for the localization comparison are shown in Fig. D.5.

In the third experiment, we reproduce the public LVK results for GW170817 [11, 13] and GW190425 [5] with DINGO-BNS. We use the same priors and data settings as the LVK, but we do not marginalize over calibration uncertainty. We find good sample efficiencies for both events (10.8% for GW170817 and 51.3% for GW190425) and good agreement with the LVK results (Fig. D.6). The LVK results use detector noise PSDs generated with BayesWave [73], which are not available prior to the merger. For our pre-merger analysis of GW170817 in the main part we thus use a PSD generated with the Welch method. The GW170817 signal overlapped with a loud glitch in the LIGO-Livingston detector [11], and we use the glitch subtracted data provided by the LVK in our analyses. Since such data would not be available prior to the merger, pre-merger inference of BNS events overlapping with glitches would in practice also require fast glitch mitigation methods.

In the fourth experiment, we analyze simulated CE data using the anticipated noise PSDs for the primary and secondary detectors. We train a DINGO-BNS network for pre-merger inference with $f \in [6, 11]$ Hz, with the upper frequency masking bound f_{max} sampled in range $[7, 11]$ Hz. This supports a signal length of 4096 seconds, with pre-merger inference between 45 and 15 minutes prior to the merger. We inject signals with GW170817-like parameters for distance, masses and inclination, to investigate how well GW170817-like event can be localized in the CE detector. We also train a network on the full frequency range $[6, 1024]$ Hz for after-merger inference, with a reduced distance prior to control the SNR (Tab. D.1).

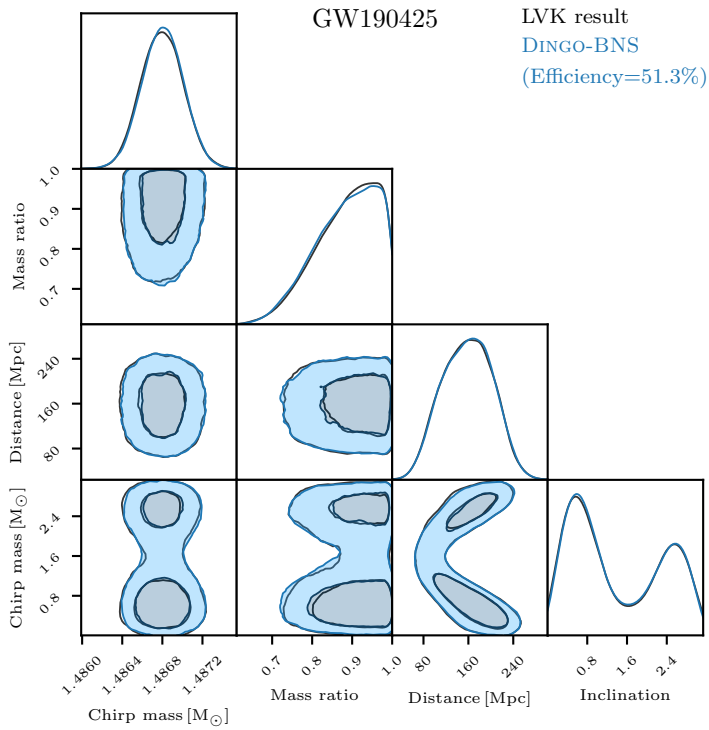


Fig. D.6 Inference results for GW190425, displaying the 50% and 90% credible regions for the 2D marginals. DINGO-BNS shows good agreement with the public LVK result.

D.2.1 Sample efficiencies

We report sample efficiencies for all injections studies in Fig. D.7. Importance-sampled DINGO-BNS results are accurate even with low efficiency, provided that a sufficient *absolute* number of effective samples can be generated. The efficiency nevertheless is a valuable diagnostic to asses the performance of the trained inference networks.

In LVK experiments, we find consistently high efficiencies, comparable to or higher than those reported for binary black holes [84]. As a general trend, we observe that higher noise levels (Fig. D.7a) and earlier pre-merger times (Fig. D.7b lead to higher efficiencies. This is because low SNR events generally have broader posteriors, which are simpler to model for DINGO-BNS density estimators. Furthermore, the GW signal morphology is most complicated around the merger, making pre-merger inference a much simpler than inference based on the full signal.

For CE injections with GW170817-like parameters (Fig. D.7c), DINGO-BNS achieves extremely high efficiency for early pre-merger analyses but the performance decreases substantially for later analysis times. This effect can again be attributed to the increase in SNR, which is of $O(10^3)$ 15 minutes before the merger. Improving DINGO-BNS for such high SNR events will likely require improved density estimators [229] that can better deal with tighter posteriors. When limiting the SNR by increasing the distance prior (Tab. D.1), we find good sample efficiencies for an after-merger CE analysis that uses the full 4096 second long signal (Fig. D.7c).

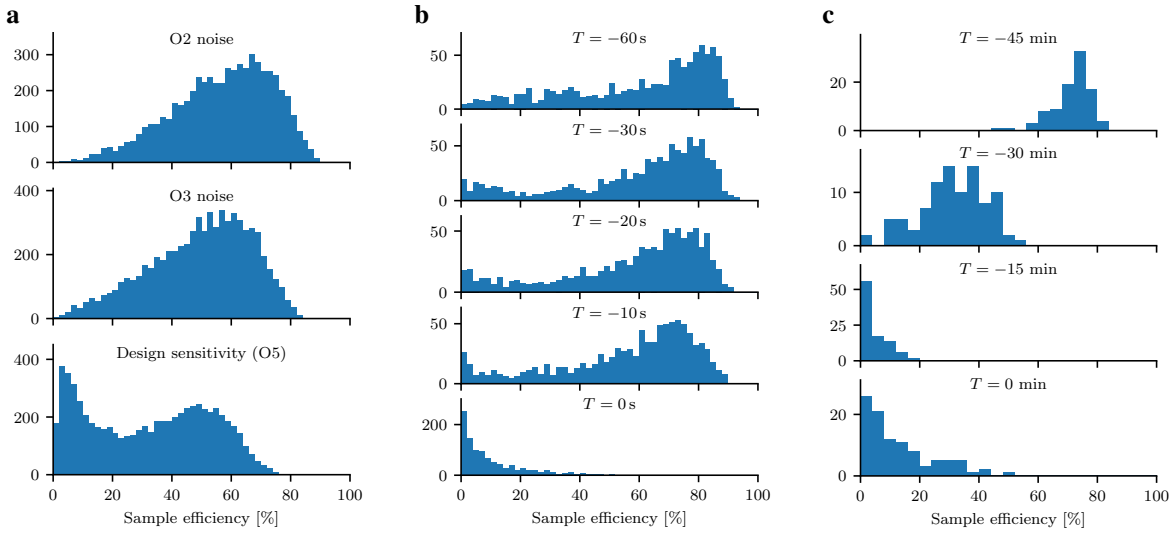


Fig. D.7 Sample efficiencies for the injection studies. (a) GW170817-like injections using different detector noise levels. (b) Injections using LVK design sensitivity PSDs. (c) Injections using CE PSDs.

D.2.2 Inference times

The computational cost of inference with DINGO-BNS is dominated by (1) neural network forward passes to sample from the approximate posterior $\theta \sim q(\theta|d_{\widetilde{\mathcal{M}}}, \widetilde{\mathcal{M}})$ and by (2) likelihood evaluations $p(\theta|d)$ used for importance sampling. For 50,000 samples on an H100 GPU, (1) takes ~ 0.370 seconds and (2) takes ~ 0.190 seconds, resulting in an inference time of less than 0.6 seconds. The speed of the likelihood evaluations is enabled by using JAX waveform and likelihood implementations [96, 232, 234], combined with the heterodyning and multibanding step that we also use to compress the data for the DINGO-BNS network. We extend the open source implementations [96, 234] by combining NRTidalv1 [88, 89] with IMRPhenomPv2 [115, 134, 50] as well as re-implementing the DINGO likelihood functions in JAX. We can jit the likelihood ahead of time since we evaluate a fixed number of waveforms at a fixed number of frequency bins. Thus we leave the jitting time (18 seconds) out of the timing estimate for importance sampling. This is in contrast to previous JAX-based GW works [232, 234] which use a fiducial waveform (determined at inference time via likelihood maximization) to perform heterodyning. Likelihood evaluations can also be done without JAX, which takes less than 10 seconds on a single node with 64 CPUs for 50,000 samples. For the vast majority of DINGO-BNS analyses in this study, the sample efficiency is sufficiently high such that 50,000 samples correspond to several thousands of effective samples after importance sampling, enabling full importance sampling inference in less than a second. Note that these numbers refer to inference times, assuming data have already been provided to DINGO-BNS. Accelerating other aspects of LVK low-latency pipelines is critical for minimizing alert times [63].

D.2.3 PSD tuning

Although most of the networks used in this study are trained with only a single PSD per detector, in practice we would generally train DINGO-BNS with an entire distribution of PSDs to enable instant

tuning to drifting detector noise [82]. (This is not relevant to tests involving, e.g., design sensitivity noise.) Conditioning on the PSD makes the inference task more complicated, and therefore leads to slightly reduced performance. For example, when repeating the first injection experiment (Fig. D.7a) with a DINGO-BNS network trained with a distribution covering the entire second LVK observing run (O2), the mean efficiency is reduced from 57% to 25%. Such networks can in principle also be trained before the start of an observing run, by training with a synthetic dataset designed to reflect the expected noise PSDs [230].