Monthly Notices of the ROYAL ASTRONOMICAL SOCIETY

MNRAS **498,** 4492–4502 (2020) doi:10.1093/mnras/staa2483
Advance Access publication 2020 August 19

Massively parallel Bayesian inference for transient gravitational-wave astronomy

Rory J. E. Smith [1,2]★ Gregory Ashton,[1,2] Avi Vajpeyi[1,2] and Colm Talbot[1,2,3]

[1]*School of Physics and Astronomy, Monash University, Clayton, VIC 3800, Australia*
[2]*OzGrav: The ARC Centre of Excellence for Gravitational Wave Discovery, Clayton, VIC 3800, Australia*
[3]*LIGO, California Institute of Technology, Pasadena, CA 91125, USA*

Accepted 2020 August 13. Received 2020 August 4; in original form 2020 April 26

ABSTRACT

Understanding the properties of transient gravitational waves (GWs) and their sources is of broad interest in physics and astronomy. Bayesian inference is the standard framework for astrophysical measurement in transient GW astronomy. Usually, stochastic sampling algorithms are used to estimate posterior probability distributions over the parameter spaces of models describing experimental data. The most physically accurate models typically come with a large computational overhead which can render data analsis extremely time consuming, or possibly even prohibitive. In some cases highly specialized optimizations can mitigate these issues, though they can be difficult to implement, as well as to generalize to arbitrary models of the data. Here, we investigate an accurate, flexible, and scalable method for astrophysical inference: parallelized nested sampling. The reduction in the wall-time of inference scales almost linearly with the number of parallel processes running on a high-performance computing cluster. By utilizing a pool of several hundreds or thousands of CPUs in a high-performance cluster, the large wall times of many astrophysical inferences can be alleviated while simultaneously ensuring that any GW signal model can be used 'out of the box', i.e. without additional optimization or approximation. Our method will be useful to both the LIGO-Virgo-KAGRA collaborations and the wider scientific community performing astrophysical analyses on GWs. An implementation is available in the open source gravitational-wave inference library `pBilby` (parallel `bilby`).

Key words: gravitational waves – methods: data analysis.

1 INTRODUCTION

Gravitational wave (GW) transients from merging binary black holes (BBHs) and binary neutron stars (BNSs) are now being routinely detected by the Advanced LIGO and Virgo detector network (GraceDB 2019). These compact-binary systems offer unprecedented means to study strong-field gravity (Abbott et al. 2016, 2019c,d), matter at supra-nuclear densities (Abbott et al. 2018), and stellar astrophysics (Abbott et al. 2019e). With data in the public domain (Vallisneri et al. 2015), methods to infer the properties of GWs are of broad interest to various communities in physics and astronomy.

Bayesian inference is the standard framework for performing precision astrophysical measurements in transient GW astronomy (Abbott et al. 2019c). The output of Bayesian inference is two fold: (i) posterior probability densities of astrophysical quantities encoded in GWs, such as the masses and spins of BBHs, and (ii) an estimate of the probability of having observed the data under a particular hypothesis – commonly called the *evidence* – which is used for hypothesis testing. Broadly speaking, there are three key ingredients to Bayesian Inference: (i) experimental data, (ii) a model, e.g. of the signal and noise components in interferometer strain data, and (iii) an algorithm to efficiently explore the parameter space of the models, i.e. the astrophysical parameters of interest. Usually, stochastic sampling algorithms, such as nested sampling Skilling (2006) or Markov chain Monte Carlo (MCMC; Metropolis et al. 1953; Hastings 1970) are used due to the high dimensionality of the parameter spaces (Veitch et al. 2015; Ashton et al. 2019; Biwer et al. 2019).

Generally, there are two classes of astrophysical inferences in transient GW astronomy. One class performs inferences on individual signals, e.g. from compact binary mergers. Here, the source properties of individual transient signals are inferred, such as the masses, spins, source location etc. Another class aims to infer the ensemble properties of particular GW sources. For example, the BNS merger rate, or the mass and spin spectrum of BBHs. This class of takes as input the inferences made on individual events, and is therefore known as *hierarchical inference*, or population inference (Abbott et al. 2019e). Here, we focus on the former class of inferences, though the methods presented in this paper will also be applicable to population inferences.

Despite years of technical advances, Bayesian inference in GW astronomy remains challenging: high-dimensional parameter spaces are difficult to explore efficiently, and the overall computational cost can be high. For example, when performing inferences on individual compact binary merger signals, the end-to-end analysis time – commonly referred to as the *wall time* – can range between several hours to several weeks, months, or even years (Smith et al. 2016). The most expensive cases correspond to some of the most physically realistic and important analyses, but their high cost can be a major hurdle, or even roadblock, to astrophysical discovery. The

★ E-mail: rory.smith@monash.edu

large wall time of Bayesian inference on individual GW signals is the central issue that we address in this paper.

In some cases, approximate methods can mitigate the large wall times. A class of approximations collectively known as reduced order methods (Pürrer 2014, 2016; Canizares et al. 2015; Smith et al. 2016; Blackman et al. 2017; Varma et al. 2019) employ dimensionality reduction techniques to achieve low-latency parameter estimation. They generally find an approximation to signal models that is computationally more tractable than the underlying model. A method known as 'relative binning' (Zackay, Dai & Venumadhav 2018) also exploits a reduced-order decomposition of the likelihood function, and has been shown to reduce the cost of likelihood/mode evaluations by roughly an order of magnitude greater than the most efficient reduced order models. Algorithms such as RIFT (Pankow et al. 2015; Lange, O'Shaughnessy & Rizzo 2018; Wysocki et al. 2019) achieve rapid parameter estimation by pre-computing aspects of the inference problem in parallel and approximating expensive functions using cheaper interpolation methods. Both these methods utilize an 'offline/online' decomposition in which expensive computations are first performed offline – possibly in parallel – in order to facilitate fast online analyses using problem-specific approximations. Generally, these techniques all face the so-called 'curse of dimensionality', i.e. they become exponentially more difficult to apply as the parameter space of models increases. Graphical processor units (GPUs) can, in some cases, reduce the cost of inference by more than an order of magnitude (Talbot et al. 2019). However, their utility is limited to a small class of GW signal models and so they cannot be used for general inference problems.

Our overall aim here is to provide a general framework for performing Bayesian inference in transient GW astronomy that significantly lowers the wall-time of data analysis. We are motivated by three considerations: (i) accuracy, meaning the framework should produce statistically robust inferences; (ii) flexibility, meaning it should be agnostic to the models and data being used; and (iii) scalability, meaning it can handle a growing amount of work by adding computational resources.

We show that wall-time of astrophysical inference on individual GW signals can be significantly reduced using a highly flexible, massively parallel nested sampling algorithm deployed at scale on a high-performance CPU cluster. The reduction in wall-time scales almost linearly with the number of CPUs in the cluster. In some cases, our method reduces the wall-time from several years to around a week using the most physically complete GW signal models. Our method bridges the gap between a lack of available fast approximate methods, and the need to perform timely precision inference on individual GW events. The method meets the three criteria of accuracy, flexibility, and scalability defined above. While our particular application is to inferences on individual GW signals, the method is agnostic to the particular inference problem and so will also be useful for reducing the wall time of hierarchical inferences. This represents a major advancement of techniques to mitigate the wall time of inference in GW astronomy. More broadly, the methods presented here should be useful to other fields in astronomy where the cost of inference is dominated by expensive calls to parametrized models of experimental data. The software used in this paper is available in the open source GW inference library parallel bilby (`pBilby`) Smith et al. (2019). `pBilby` is based on the `bilby` Ashton et al. (2019) GW inference software library, but has been optimized for parallel computing environments.

The remainder of this paper is organized as follows. In Section 2, we give an overview of Bayesian Inference. In Section 3, we describe nested sampling and our parallelization scheme. In Section 4, we benchmark the performance of parallel nested sampling. In Section 5, we discuss how parallel nested sampling compares to alternative methods for reducing the wall time of inference in transient GW astronomy. In Section 6, we describe further applications of parallel nested sampling to GW astronomy. Finally, we give concluding remarks in Section 7.

2 BAYESIAN INFERENCE

Bayesian inference generally consists of two parts: *Parameter estimation* and *hypothesis testing*. Parameter estimation entails computing the posterior probability density of the source parameters given the experimental data, e.g. the masses and spins of BBHs. Hypothesis testing entails computing the Bayesian 'evidence': The probability of the data given an hypotheses. With the evidence, one can quantify the relative probability of the data under competing hypotheses, e.g. *How much more probable is it that the data contain a signal with higher order mode content than a signal with only the leading-order quadrupolar mode?* (The LIGO Scientific Collaboration 2020).

Inferences made about the astrophysics of individual GW transients generally rely on (i) a model for the underlying signal and possibly noise components of the data, and (ii) a statistical description of noise processes in the data (Finn & Chernoff 1993; Cutler & Flanagan 1994). Stochastic sampling algorithms – such as nested sampling (Skilling 2006), or Markov chain Monte Carlo (Metropolis et al. 1953; Hastings 1970) – are employed to search the parameter space of the models and estimate posterior densities and evidences (Christensen & Meyer 2001; Veitch & Vecchio 2010).

Bayesian inference relates the probability of model parameters θ to experimental data d, and an hypothesis for the data $\mathcal{H}$, via Bayes theorem (Bayes & Price 1763)

$$p(\theta|d,\mathcal{H}) = \frac{\pi(\theta|\mathcal{H})\,\mathcal{L}(d|\theta,\mathcal{H})}{\mathcal{Z}(d|\mathcal{H})}\,. \tag{1}$$

Here, $p(\theta|d,\mathcal{H})$ is the *posterior probability density* of the parameters θ given d and $\mathcal{H}$; $\mathcal{L}(d|\theta,\mathcal{H})$ is the *likelihood* of d given θ and $\mathcal{H}$; $\pi(\theta|\mathcal{H})$ is the *prior* probability of θ; and $\mathcal{Z}(d|\mathcal{H})$ is the *evidence* of d given $\mathcal{H}$. The posterior density is the target for *parameter estimation*, while the evidence is the target for *hypothesis testing*. Both the posterior and evidence can be estimated to high accuracy using nested sampling or thermodynamic integration (Veitch et al. 2015; Ashton et al. 2019; Biwer et al. 2019). Assuming the priors can be defined, the primary input to inference algorithms is the likelihood function.

To motivate discussion of the computational cost of inference on modelled coalescing compact binary signals, we consider the usual likelihood function that describes the probability of interferometer data given (i) an hypothesis that the data contain a signal plus Gaussian noise ($\mathcal{H}_S$), and (ii) parameters θ which describe a model of the GW signal signal. This likelihood is the basis for most inferences on individual transient signals in GW astronomy (Veitch et al. 2015; Ashton et al. 2019; Abbott et al. 2019b), and constitutes the dominant cost of inference (Pankow et al. 2015; Smith et al. 2016). The uncertainty on the data is due to the random noise component which is a stationary Gaussian process, coloured by the noise power spectral densities of the interferometers. The likelihood is (Romano & Cornish 2017)

$$\mathcal{L}(d|\theta,\mathcal{H}_S) \propto \prod_j^M \prod_i^{N_{\rm det}} \frac{2}{\pi T S_{ij}} \exp\left(-\frac{2}{T}\frac{|\tilde{d}_{ij}-\tilde{h}_{ij}(\theta)|^2}{S_{ij}}\right) \tag{2}$$

Here, $\tilde{d}$ and $\tilde{h}$ are, respectively, the Fourier transforms of the strain data and signal model, S is the detector noise power spectral density, and T is the duration of the data in seconds. The first product (over j) runs over the number of frequency bins M in the Fourier transformed data/model, the second product (over i) runs over the number of interferometers $N_{\rm det}$.

2.1 Models and parameter spaces

The choice of signal model $h(\theta)$ defines a *particular* signal hypothesis. In practice many different signal models are often used to analyse a given signal (Abbott et al. 2019c). This can be to evaluate the significance of certain physics present in the signals, e.g. higher order modes, by computing the evidence of the data using models with and without higher order mode content (The LIGO Scientific Collaboration 2020). Using multiple models can also serve to estimate systematic uncertainty in inferences that exists due to differences between models. Here, we consider three fiducial signal models. For BBH analyses, we consider the models known as `IMRPhenomPv3HM` (Khan et al. 2020) and `SEOBNRv4PHM` (Ossokine et al. 2020). For BNS analyses, we use an effective-precession model `IMRPhenomPv2NRT` (Hannam et al. 2014; Husa et al. 2016; Khan et al. 2016; Dietrich, Bernuzzi & Tichy 2017; Dietrich et al. 2019). Due to the higher order mode content, these BBH models are crucial for precision physics measurements on GWs from BBHs when the mass ratio of the systems is asymmetric, such as GW190412 (The LIGO Scientific Collaboration 2020). Both models represent the current state-of-the art of BBH models that cover a large mass and spin range. They are also the most expensive BBH models. `IMRPhenomPv2NRT` includes the effect of precessing spin on the heavier of the two bodies, and models the tidal deformability of neutron stars through two tidal deformability parameters. This model was used in the LIGO/Virgo analyses of GW170817 and GW190425, as well as in numerous other studies (Abbott et al. 2019c).

In addition to signal models, it is common to include models for the noise features. Typically, we model uncertainty of the data calibration (Cahillane et al. 2017), and use a point estimate for the power spectral density, which typically is generated either using off-source data or an on-source estimation method (Cornish & Littenberg 2015; Chatziioannou et al. 2019).

The dimensionality of model parameter spaces can be highly variable. Astrophysical BBHs are described by 15 parameters (masses, spins, source location etc.), BNSs are described by an additional two parameters that describe the tidal deformability of the stars. The data-calibration model uses a set of amplitude and phases to model systematic uncertainty in the Fourier-domain data at a set of judiciously chosen frequency nodes. Typically, 10 nodes are used per interferometer data set and the calibration model is described by 20 parameters (10 amplitudes and 10 phases). Thus, data from a three-detector network are described by 75–77 parameters which must be inferred simultaneously.

2.2 The computational cost of inference

There are two scales that determine the overall computational cost of Bayesian inference: (i) The cost of evaluating parametrized models of the data, and (ii) the rate of convergence of the sampling algorithms. We find it convenient to measure the computational cost in terms of CPU time, as this can be used to determine the wall time of the inference process. The cost of (i) generally determines the CPU time of one iteration of a stochastic sampling algorithm, while (ii) determines the overall CPU time required to complete the analysis.

The typical wall time can be estimated by first considering the total CPU time. To leading order, the CPU time $T_{\rm c}$ of Bayesian inference scales like the average call-time of the data model $\langle T_{\rm m}\rangle$, multiplied by the total number of calls to the likelihood function N of the stochastic sampling algorithm

$$T_{\rm c} = N\langle T_{\rm m}\rangle. \tag{3}$$

We treat N as an overall normalization which is typically $N \sim \mathcal{O}(10^7)$. When serial sampling algorithms are used, the CPU time $T_{\rm c}$ is equal to the wall time $T_{\rm w}$. The average call-time $\langle T_{\rm m}\rangle$ is strongly dependent on the complexity of the GW signal models, and possibly models for the noise.

2.2.1 *Coalescing compact binary signal models*

For a given model, $\langle T_{\rm m}\rangle$ scales with the signal's bandwidth multiplied by its duration, which is equal to M in the sum in equation (2) (Canizares et al. 2015). The overall cost is set by the intrinsic complexity of the model (Pürrer 2014; Smith et al. 2016). For signal models defined in the time domain, $\tilde{h}$ is computed by first evaluating the model in the time domain and subsequently taking the discrete Fourier transform. Time-domain signal models can be significantly more computationally expensive than those defined directly in the frequency domain. Many time-domain models require solving coupled ODEs to evaluate the signal at discrete times, see e.g. Pan et al. (2014). Together with the additional cost of the Fourier transform, the relative cost of using time domain models in inference can be between one to two orders of magnitude more expensive than frequency-domain models (Pürrer 2014; Smith et al. 2016).

As a rule of thumb, more sophisticated models have higher $\langle T_{\rm m}\rangle$. In practice, the range is broad: $\mathcal{O}(10^{-3}\,{\rm s}) \lesssim \langle T_{\rm m}\rangle \lesssim \mathcal{O}(1\,{\rm s})$. The lower limit corresponds to approximate frequency-domain signal models on short-duration BBHs, e.g. `IMRPhenomPv2` (Hannam et al. 2014). The upper limit corresponds to frequency-domain BNS signal models, e.g. `IMRPhenomPv2NRT`, or complex time-domain BBH models that include spin precession effects and higher order modes, e.g. `SEOBNRv4PHM` and `IMRPhenomPv3HM`. Hence, for serial sampling algorithms, the wall time roughly ranges between $\mathcal{O}(1\,{\rm d}) \lesssim T_{\rm w} \lesssim \mathcal{O}(1\,{\rm yr})$. The upper limit presents serious hurdles, or possibly roadblocks, to using models with, e.g. higher order mode content, and two-body spin dynamics. This problem will be compounded as GW detectors push their low-frequency sensitivity into the 5−10 Hz range because the in-band duration of observable signals will be up to an order of magnitude longer (The LIGO Scientific collaboration 2019; Reitze et al. 2019a; Maggiore et al. 2020).

3 PARALLEL NESTED SAMPLING

Nested sampling is a stochastic-sampling method designed foremost to estimate the evidence $Z(d|\mathcal{H})$ (Skilling 2006) in equation (1), which is the primary ingredient in Bayesian hypothesis testing. As a byproduct, nested sampling also produces the posterior density $p(\theta|d,\mathcal{H})$. Importantly, nested sampling is scalable to high-dimensional and irregularly shaped parameter spaces (Chopin & Robert 2010). This affords a large degree of flexibility and ensures that nested sampling is well suited to extensions of the likelihood function in equation (2), e.g. by increasing the dimensionality of the

parameter space to include parameters that model features of the noise, or parameters that describe signals in alternative theories of gravity.

The evidence can be computed via the following integral

$$Z(\mathrm{d}|\mathcal{H}) = \int_{\Omega_\theta} \mathrm{d}\theta\, \pi(\theta)\, \mathcal{L}(\mathrm{d}|\theta, \mathcal{H}) \tag{4}$$

$$= \int_{X=0}^{1} \mathrm{d}X\, \mathcal{L}(\mathrm{d}|X, \mathcal{H}) \tag{5}$$

$$\approx \sum_i \Delta X_i \mathcal{L}(\mathrm{d}|X_i, \mathcal{H}). \tag{6}$$

The second line transforms the integral over the multidimensional parameter space θ into a one-dimensional integral over the *prior mass* $\mathrm{d}X = \mathrm{d}\theta\, \pi(\theta)$. The quantity $\mathcal{L}(d|X, \mathcal{H})$ is an iso-likelihood contour (Skilling 2006), i.e. it defines a boundary of constant likelihood within the prior volume X.

In practice, the inverse mapping $\theta(X)$ is not known, and so the integrals in equation (6) cannot be performed analytically. Nested sampling estimates the evidence in equation (6) algorithmically. Here, we are agnostic to particular variants of nested sampling algorithms – see e.g. Handley, Hobson & Lasenby (2015), Speagle (2020) – because our aim is simply to remove one particular bottleneck that occurs due to the high cost of evaluating the models that enter the likelihood function. We therefore will describe parallel sampling in the context of one of the most basic variants of nested sampling known as 'static nested sampling' (Speagle 2020). We will not discuss the theory of nested sampling in depth, and we refer the reader to Skilling (2006) and Speagle (2020). However, we will find it useful to sketch the main algorithmic components of static nested sampling in order to introduce the parallelization scheme. There are three key components: (i) prior sampling, (ii) evidence estimation, and (iii) obtaining posterior samples. The parallelization scheme enters into stage (i). Before we describe the scheme, we briefly describe the three elements below.

3.1 Prior sampling

Nested sampling estimates equation (6) by drawing samples from the prior distribution. Samples are accepted subject to the constraint that those drawn on subsequent iterations have a higher likelihood than those on previous iterations. A key element is the set of *live points*. The algorithm is seeded by drawing a number K live points from the prior. These points are ranked from highest to lowest likelihood. The algorithm then proceeds by drawing samples θ_i from the prior on each iteration i . The aim is to replace the live point with the lowest likelihood $\mathcal{L}_{\rm min}$ on each iteration. Samples θ_i are accepted on each iteration subject to the constraint $\mathcal{L}(d|\theta_i) \geq \mathcal{L}_{\rm min}$. The sample associated with $\mathcal{L}_{\rm min}$ is removed from the list of live points and added to a list of *dead points*, and the new pair $\{\theta_i, \mathcal{L}(d|\theta_i)\}$ is added to the list of live points.

3.2 Evidence estimation

Once a sample has been accepted, the prior volume X_i bounded by the likelihood $\mathcal{L}(d|\theta_i)$ can be estimated as (Skilling 2006) $X_i \approx [K/(K+1)]^i$, or equivalently $\ln X_i \approx -i/K$. With a set of likelihoods and an estimate for the change in prior volume $\Delta X_i = X_i - X_{i-1}$, the Riemann sum in equation (6) can be computed on each iteration. The algorithm terminates when the change in the (log) evidence is below some user-defined threshold: $\Delta \ln Z = \ln Z_i - \ln Z_{i-1} \leq \epsilon$.

3.3 Posterior samples

Once the algorithm has terminated, the posterior can be estimated as follows. Because the evidence is the integral of the *un-normalized posterior* density, we must have

$$Z \approx \sum_i \Delta X_i \mathcal{L}(d|\theta_i) = \sum_i p(\theta_i)\,, \tag{7}$$

where $p(\theta_i)$ is an 'importance weight' which represents an estimate of the un-normalized posterior density at sample point θ_i: $p(\theta_i) \approx \mathcal{L}(d|\theta_i)\pi(\theta)\Delta\theta_i$. The importance weights can then be used to approximate the posterior

$$p(\theta|d, \mathcal{H}) \approx \frac{\sum_i p(\theta_i)\delta(\theta - \theta_i)}{\sum_i p(\theta_i)}\,, \tag{8}$$

$$= Z^{-1} \sum_i p(\theta_i)\delta(\theta - \theta_i). \tag{9}$$

3.4 Parallel prior sampling

In practice, a bottleneck arises when drawing prior samples to update the live points. This is because drawing samples require evaluating the likelihood constraint, and hence the likelihood function, which is computationally expensive. This bottleneck can be alleviated by parallelizing the prior-sampling step.

The parallel variant of static nested sampling is shown in Algorithm 1. The only difference to *serial* static nested sampling is that samples will be drawn from the prior in parallel on each iteration. This is possible because each iteration of the nested sampling algorithm is independent of the state of the algorithm on previous iterations, i.e. a series of draws from the prior is equivalent to the same draws being made simultaneously. Intuitively, if we were able to achieve perfect scaling, we would be able to advance the state of the nested sampling algorithm by exactly a factor of n on each iteration because we could make n live-point updates simultaneously. The parallel sampling procedure is straightforward to implement on $n_{\rm cores}$ CPU cores via Message Passing Interface (MPI; Dalcin et al. 2011). We use a head/worker model[1] where the 'head' node organizes live/dead points, and estimates the evidence, while the $n_{\rm cores} - 1$ 'worker' nodes find new live points. On each iteration i, a CPU j evolves the same lowest likelihood live point $\mathcal{L}_{\rm min}$. A sample $\theta_{i,j}$ is drawn from the prior and is accepted subject to the usual constraint $\mathcal{L}(d|\theta_{i,j}) \geq \mathcal{L}_{\rm min}$, or rejected otherwise. Once the $n^{'} \leq n_{\rm cores} - 1$ samples have been gathered, they can be used to update the list of live and dead points. We can iteratively replace $\{\mathcal{L}_{\rm min}, \theta(\mathcal{L}_{\rm min})\}$ with $\{\mathcal{L}(d|\theta_{i,j}), \theta_{i,j}\}_{j=1}^{n^{'}}$. In principle, we could let each of the workers continue sampling until they all find a valid sample point, however, this would create a sampling bottleneck whereby the list of live points cannot be updated until the least efficient worker returns a sample.

The probability of drawing a point between two iso-likelihood contours scales like the inverse of the volume contained between the contours. As such, parallel prior sampling will not in general draw samples that are guaranteed to be accepted as new live points, and

[1]Note that this is (unfortunately) frequently referred to as a 'master/slave' model

hence we cannot expect to achieve linear scaling with the number of CPUs, $n_{\rm cores}$. We quantify the overall improvement in efficiency by a speedup factor which is a function of the number of live points and cores. The scaling relation for the speedup S is (Handley et al. 2015)

$$S(n_{\rm cores}, n_{\rm live}) = n_{\rm live} \ln(1 + n_{\rm cores}/n_{\rm live}). \quad (10)$$

The expected wall time of inference using parallel nested sampling therefore scales as

$$T_{\rm w}(n_{\rm cores}, n_{\rm live}) = \frac{N}{S(n_{\rm cores}, n_{\rm live})} \langle T_{\rm m} \rangle \quad (11)$$

$$= \frac{T_{\rm c}}{S(n_{\rm cores}, n_{\rm live})}. \quad (12)$$

4 PERFORMANCE TESTS AND RESULTS

Parallel nested sampling is theoretically capable of reducing the wall-time according to the scaling relation in equation (10). We first compare the empirical scaling to the theoretical expectation. Secondly, we determine that our implementation of parallel nested sampling yields unbiased estimates of posterior densities.

In order to determine the speedup scaling, we measure the wall time of the BBH merger event GW150914 as a function of the number of CPU cores, keeping the number of live points fixed. This benchmark test should provide a generic scaling relation for the reduction in wall time, provided the likelihood function dominates the overall cost of inference and other costs are negligible. As such, it should be applicable to determine the speedup and reduction in wall time of analyses on other GW events and, e.g. hierarchical inference studies.

To demonstrate that the method produces unbiased posteriors, we perform a BBH 'injection campaign': we create 100 synthetic BBH merger signals using `IMRPhenomXPHM`. These signals are added into Gaussian, stationary noise coloured with the aLIGO and Virgo PSDs. We then test the quality of the inferred posterior probability densities using a parameter-parameter test (P–P test).

4.1 Scaling relation

4.1.1 Implementation

We use the GW inference package *parallel bilby* (`pBilby`) to analyse the GW event GW150914. Parallel nested sampling (Alg. 1) is implemented in `pBilby` via the `dynesty` nested sampling library. Communication between nodes is accomplished using MPI through the `PYTHON` package `mpi4py` and `schwimmbad`. We analyse 4 s of strain data containing the GW150914 from the LIGO-Hanford and LIGO-Livingston observatories. We use a minimum and maximum frequency of 20 and 1024 Hz, respectively. The data, noise PSD, calibration model and prior ranges were taken from the Gravitational-wave Open Science Center (Abbott et al. 2019a).

For the GW likelihood function, we use two models, one for the BBH merger signal, and another for the calibration of the data. We use the GW signal model `IMRPhenomPv3HM` and a data calibration model from Cahillane et al. (2017). The signal model is a cutting edge BBH model which includes the effects of spin precession, and higher order GW modes. The computational cost is typical of the current generation of signal models. The data calibration model is the standard model used in LIGO/Virgo analyses on compact binary mergers. Thus, the wall-time measurements will be indicative of the actual run times for real LIGO-Virgo-KAGRA analyses. In total, the model parameter space is 55-dimensional: 15 astrophysical parameters describe the GW signal, and 40 describe the data calibration.

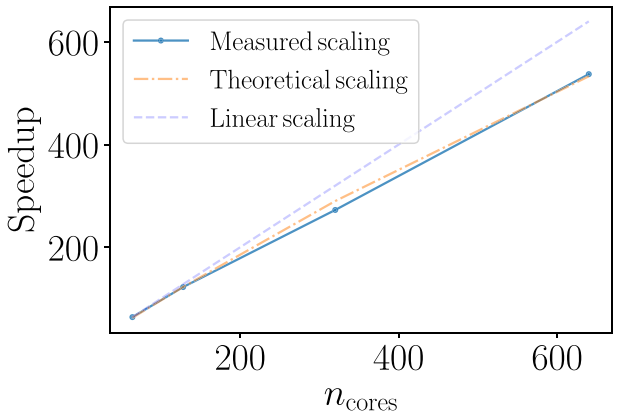

Figure 1. Speed up factor equation (10) versus number of CPUs for a fixed number of live points ($n_{\rm live} = 2000$).

Our analyses use 2000 live points, which we have found is robust for inferences on BBH signals including a data-calibration model. To effectively bound the prior distribution and improve convergence, we use multi-elipsoid bounding distributions (Feroz, Hobson & Bridges 2009) implemented in `dynesty`. To ensure we generate prior samples efficiently, we use a modified version of the MCMC proposal distribution implemented in `dynesty`. We fix the number of random walks in the MCMC to ensure that workers are synchronized and the cores are properly load balanced. The default behaviour of the MCMC in `dynesty` is such that the random walk does not terminate until a sample point satisfying the likelihood constraint is found. In parallel nestsed sampling, this behaviour can create a sampling bottleneck whereby the time it takes to return the list of proposed points is limited by the least efficient random walk. In practice, this can result in a single, or a few, MCMC chains out of many hundreds continuing to sample to find a new point while the remaining processes are idling ready to update the list of live points.

To measure the scaling we record the wall-time of running `dynesty` with parallel prior-sampling on GW150914 using $n_{\rm cores} = (16, 64, 320, 640)$ CPU cores to draw prior samples in parallel on each iteration. Note, that because one CPU process is reserved as the 'head' process while the others draw samples, the number of CPUs drawing samples in parallel is $n_{\rm cores} - 1$. We perform five independent runs for each $n_{\rm cores}$ to get a measure of the typical variation in wall times. From the wall-times, we can directly compute the scaling as a function of $n_{\rm cores}$ which we can compare to equation (10). All of the runs were performed on Intel Xeon E5-2660 (Sandybridge) CPUs with a 2.2 GHz clock rate. Nodes are networked via non-blocking QDR infiniband.

4.1.2 Results

The measured speedup-scaling is shown in Fig. 1. We find excellent agreement with the theoretical prediction for the scaling for a fixed number of live points and variable number of CPU cores. The theoretical scaling curve is computed using equation (10). Because the sampling time is dominated by the cost $\langle T_{\rm m} \rangle$ of evaluating the models that enter into the likelihood function, the scaling can be

Table 1. Wall times for selected events using $n_{\rm cores} = (16, 64, 640)$ CPUs. Measured wall times are non-italicized and estimated wall times are *italicized*.

	IMRPhenomPv3HM			SEOBNRv4PHM			IMRPhenomPv2NRT		
Number of CPUs	16	64	640	16	64	640	16	64	640
GW150914	3.9 d	23.3 h	2.8 h	*83.7* d	*21.2* d	2.5 d	–	–	–
GW190425	–	–	–	—	—	—	*30.7* d	*7.8 d*	22 h
GW190412	*60.3 d*	*15.3 d*	1.8 d	*2.9* yr	*276.1* d	11.53 d	—	—	—

thought of as an effective decrease in the average wall-time of evaluating these models. Provided one is in the regime where the dominant cost of inference is $\langle T_{\rm m} \rangle$, we expect that the scaling will hold when the number of CPU cores and/or the number of live points is increased.

Moreover, as we have argued, the scaling does not depend on the *particular* choice of models, likelihood functions, or even the type of data being analysed. Thus, the scaling relation should be broadly applicable to a range of inference problems in astronomy where the cost of parametrized models dominates the cost of inference, e.g. population/hierarchical inference studies in GW astronomy. We also note that the scaling may be independent of the actual variant of nested sampling algorithm, provided the are compatible with the same or similar parallel prior-sampling methods.

4.2 Example wall-time reduction

In Table 1, we show representative wall times for running parallel nested sampling on the GW events GW150914, GW190425, and GW190412. These events correspond to a short-duration BBH merger, a BNS merger, and a long-duration (∼10 s) BBH merger. Non-italicized wall times in Table 1 are measured and italicized wall times are estimated from the measured values. We consider the waveform families `IMRPhenomPv3HM`, `SEOBNRV4HM`, and `IMRPhenomPv2NRT`. For each event analysis, we determine the wall time using $n_{\rm cores} = (16, 64, 640)$. We use the same data duration and sampling rate as in Section 4.1.1. Prior ranges are taken from Abbott et al. (2019a).

For analyses on GW150914-like systems, we determine that run times can be reduced to around 2.8 h when using 640 CPU cores, down from 3.9 d using 16 cores when using `IMRPhenomPv3HM`. The scaling shown in Fig. 1 is based on these measurements. Analyses on GW190412 are expected to scale similarly as the likelihood function will be an even more dominant cost due to the increased duration of the data. We find that for `IMRPhenomPv3HM` the analysis time can be reduced to 1.8 d on 640 cores, down from 60.3 d on 16 cores. For `SEOBNRv4PHM`, the analysis time can be reduced to 11.5 d on 640 cores, versus 2.9 yr on 16 cores. These cases are particularly relevant to LIGO-Virgo data analysis as reduced order methods are not yet available for these signal models. For BNS analyses on GW190425-like systems, we show that wall times can be reduced to 22 h on 640 cores versus 30.7 d on 16 cores when using `IMRPhenomPv2NRT`.

4.3 Sampling accuracy

Our goal here is to produce a metric that measures the accuracy of the estimates of posterior density produced by the algorithm. Because the output of an analysis is a set of PDFs, any bias in a single set is hard to gauge. We therefore test the quality of an ensemble of posterior PDFs. We quantify sampling accuracy using a P–P test. We generate artificial data sets containing GW signals and LIGO-like noise. Our expectation is that the true parameter values should fall within the X per cent credible region X per cent of the time, signifying that our posterior densities are unbiased. For all estimated parameters, the P–P test computes the fraction of events for which the injected signal parameters fall within the X per cent credible interval (CI), and assigns a p-value to the outcome.

Table 2. Parameters and prior distributions used in the analysis of simulated BBH merger events for the P–P test in Section 4. The parameters are (from top to bottom): luminosity distance D_L, chirp mass $\mathcal{M}_c$, mass ratio q, inclination ι, orbital phase at coalescence $\phi_{\rm c}$, polarization phase ψ, right ascension RA, declination Dec., time at coalescence $t_{\rm c}$, spin magnitude on the heavier BH a_1, spin magnitude on the lighter BH a_2, spin tilt angle on the heavier black hole θ_1, spin tilt angle on the lighter black hole θ_2, the angle between the two spin vectors $\phi_{1,2}$, and the angle between the orbital and total angular momentum $\phi_{J,L}$.

Parameter	Prior	Prior bounds
D_L [Mpc]	Power law ($\alpha = 2$)	[50, 2000]
$\mathcal{M}_c$ [$M_\odot$]	Uniform	[15,69.9]
q	Uniform	[0.125,1]
ι [rad]	sin	$[0,\pi]$
$\phi_{\rm c}$ [rad]	Uniform	$[0, 2\pi]$
ψ [rad]	Uniform	$[0, \pi]$
RA [rad]	Uniform	$[0, 2\pi]$
Dec. [rad]	cos	$[0, \pi]$
$t_{\rm c}$ [s]	Uniform	$[t_{\rm c,\,true} - 0.1\,{\rm s}, t_{\rm c,\,true} + 0.1\,{\rm s}]$
a_1	Uniform	[0,0.88]
a_2	Uniform	[0,0.88]
θ_1	sin	$[0,\pi]$
θ_2	sin	$[0,\pi]$
$\phi_{1,2}$	Uniform	$[0, 2\pi]$
$\phi_{J,L}$	Uniform	$[0, 2\pi]$

We use the likelihood function in equation (2) which contains 15 free parameters which describe the BBH signals. The sampled parameters, together with the associated priors, are listed in Table 2. We do not consider a model of the data calibration. We use the same run configuration described in Section 4.1.1. We analyse 100 synthetic GW signals with parameters randomly drawn from the priors in Table 2. The distribution of SNRs of the signals is shown in left-hand panel of Fig. 2.

The P−P plot is shown in Fig. 2. The x- and y-axes are, respectively, the CI and the fraction of events in a particular CI. Perfect sampling results in all curves falling along the diagonal. The grey region shows the 1σ, 2σ, 3σ uncertainty regions for the distribution of curves. We quantify the quality of our sampling by first assuming a null hypothesis of perfect sampling. We determine the p-value of each sample parameter, and then produce an overall p-value using a KS test according to Biwer et al. (2019). We find that our results are consistent with the null hypothesis at the p-value $p = 66.9$ per cent level, and we do not have reason to reject the null hypothesis.

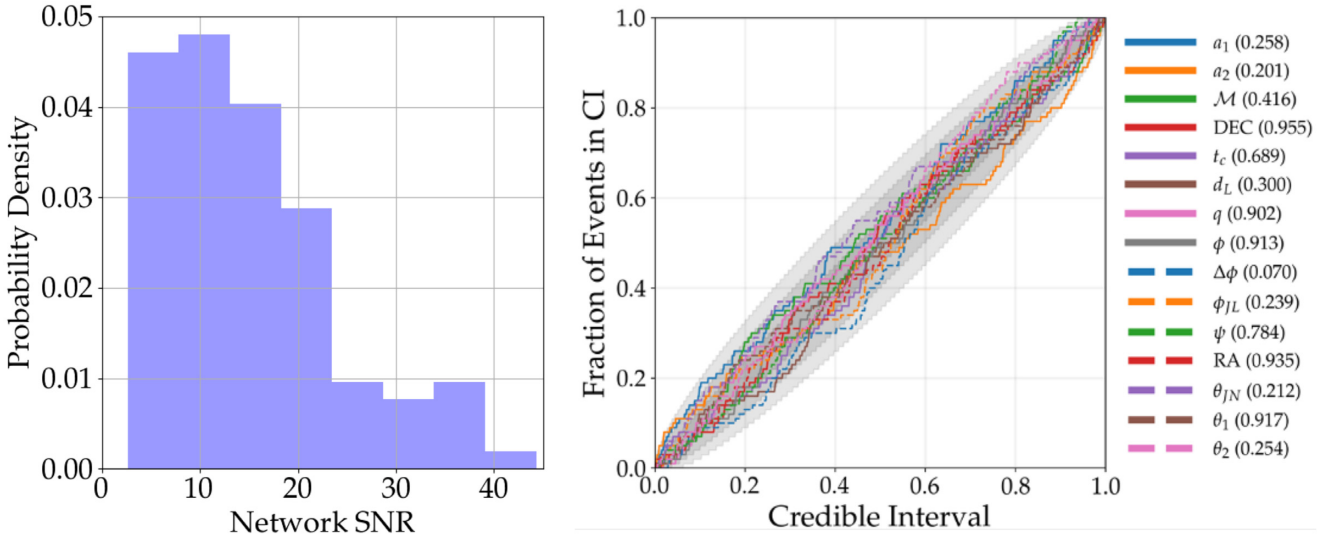

Figure 2. Left, the probability density of the injection set's network signal-to-noise ratio (SNR). Right, the CI versus the fraction of events in a particular CI for the 15 parameters that describe 100 simulated BBH merger signals. Perfect scaling results in all curves falling along the horizontal. The grey region shows the 1σ, 2σ, 3σ uncertainty regions for the distribution of curves. The combined p-value for all parameters, over all tests, is 0.669, and individual parameter p-values are displayed in parentheses in the plot legend.

5 COMPARISON TO COMPLEMENTARY METHODS

A class of techniques known collectively as 'reduced order methods' have been successful at reducing the cost of inference using particular signal models by up to a factor of around 300, and are employed by the LIGO Scientific Collaboration in production-level analyses (Pürrer 2014, 2016; Smith et al. 2016). One class known as a reduced order model, utilizes a problem-specific down-sampling of waveforms at judiciously chosen time or frequency nodes. The full waveform can be reconstructed together with an efficient global interpolation method to up-sample the waveforms to the full time or frequency space. A further class of methods known as reduced order quadratures exploit the reduced order model repesentation to compress the number of terms in the likelihood function equation (2), effectively performing inference on compressed data. However, they are difficult to apply broadly to all classes of signal models, in particular to fully precessing time-domain signal models, due to the curse of dimensionality, see e.g. Field et al. (2014), Blackman et al. (2017), and Varma et al. (2019). Moreover, they typically require highly specialized knowledge to construct and as such may not be readily utilized by the larger physics and astronomy communities. Nevertheless, given the current trajectory of research on reduced order methods, it seems likely that in the future they will exist for fully precessing time-domain models. While we have not explored the possibility of combining parallel nested sampling with reduced order model waveforms or reduced order quadratures, it seems probable that they could further reduce the wall time of inference by between one and several orders of magnitude. Combining reduced order methods with parallel nested sampling therefore merits further attention.

The relative binning Zackay et al. (2018) also exploits a reduced-order decomposition of the likelihood function, and has been shown to reduce the cost of likelihood/mode evaluations by a factor of around 10^4 for BNS mergers – roughly an order of magnitude greater than the most efficient reduced order models. Relative binning utilizes 'summary data' which captures sufficient information about how gravitational waveforms smoothly change over parameter space with respect to a fiducial waveform. This data is around an order of magnitude less than the level of down sampling achieved by reduced order models. The study in Zackay et al. (2018) was limited to frequency-domain waveforms, though in principle it can also be applied to time-domain waveforms making the method fairly flexible. One drawback is that it requires that the summary data – the complex-valued GW strain at well-chosen frequency or time bins – can be directly accessed. While this data can be easily accessed for frequency or time domain waveforms that admit closed-form expressions, e.g. Hannam et al. (2014), it cannot be accessed for waveforms that do not. For example, in order to access the time-domain strain at a set of sparsely separated time bins for models such as `SEOBNRv4PHM` would still require solving the waveform at all intermediate time bins. This is because many time-domain waveform models require evolving the orbital dynamics via a set of coupled ODEs. Nevertheless, assuming that this issue can be overcome, e.g. with reduced order models, relative binning may be able to offer genuinely low-latency inference with or without parallel nested sampling. As with reduced order methods, we believe that research combining relative binning with parallel sampling is warranted.

Other methods offer various degrees of parallelism: Monte Carlo methods in `LALInference` (Veitch et al. 2015) have been employed to facilitate using expensive models such as SEOBNRv3 (Abbott et al. 2019c). Parallelism enters in two ways. First, parallel tempering is used to more efficiently explore the parameter space. Secondly, many independent instances of MCMC can be run in parallel and the outputs combined. However, the efficiency of the algorithm is typically poor *and* the model is expensive. These two issues compound in such a way as to make the wall time *and* CPU time are large.[2] Parallel 'grid-based' methods such as RIFT (Pankow et al. 2015; Lange et al. 2018; Wysocki et al.

[2]For example, the analyses on the GWTC-1 event GW170608 used 120 parallel MCMC chains running continuously for around two months Chase ().

2019) precompute certain aspects of the model/likelihood space in parallel in combination with interpolation techniques to estimate the posterior density and evidence. Importantly, RIFT can be used to estimate posteriors and evidences using, e.g. a relatively sparse set of numerical relativitiy simulations. This is achieved by evaluating the likelihood at specific points in parameter space which can then be interpolated across the domain of high posterior support. While this method offers significant advantages over pure sampling-based methods, the complexity of grid-based interpolation methods scales unfavourably with the dimensionality of the parameter space, i.e. it suffers from the curse of dimensionality. For this reason, it will be difficult to scale grid-based methods to the levels required by general problems in inference in GW astronomy.

An embarassingly parallel method known as likelihood reweighting (Kish 1995; Payne, Talbot & Thrane 2019) produces posterior samples and evidences from a target likelihood function, i.e. the one of interest, by leveraging samples and evidence obtained using a reference likelihood function that is computationally cheaper to evaluate (Elvira, Martino & Robert 2018). Reweighting is efficient when the target and reference posteriors are similar. For instance, analyses using higher order mode models on the GWTC-1 events can generate posterior samples with between 7 per cent and 60 per cent efficiency (Payne et al. 2019). However, there are two drawbacks. First, when the target and reference likelihoods diverge, the overall efficiency can be poor. For loud events, e.g. SNR ~ 50, the efficiency can be around 0.1 per cent (Payne et al. 2019) meaning that many thousands of reference analyses have to be performed to generate a satisfactory number of effective samples through reweighting. In practice this can lead to a very high overall CPU time, though due to the embarassingly parallel nature of the problem, a low wall time. Secondly, the choice of a good reference likelihood is not always obvious. Here, a trade off between accuracy and speed has to be made, and several of the fastest waveform models have restrictions in, e.g. mass ratio and spin (Smith et al. 2016), which could make the target and reference likelihoods diverge in regions of parameter space, thus introducing a source of inefficiency in the reweighting procedure.

Lastly, we consider the use of GPUs. GPUs can accelerate aspects of the inference problem that are embarassingly parallel. In particular, many frequency-domain GW models admit closed form expressions and hence the model at each frequency bin can be evaluated in parallel. In Talbot et al. (2019), the authors demonstrate that the cost of evaluating frequency-domain waveform models can be accelerated by a factor of $\sim$50 using a single GPU. We note that the method described here can be used to distribute sampling over a pool of GPUs to obtain further acceleration. A clear drawback of GPU acceleration is that it is unlikely to be able to accelerate models that are computed by first solving couple ODEs, e.g. most time-domain models. These models must be evolved iteratively, and so the full time series cannot be evaluated in parallel in contrast to their frequency-domain counterparts.

A combination of the techniques described above, together with parallel sampling methods will likely be required to tackle inference problems at the scale required by future GW experiments. For example, third-generation detectors such as Einstein Telescope (Sathyaprakash et al. 2012; Cosmic Explorer Reitze et al. 2019a) will observe signals in their sensitive band for several tens of minutes to hours. Space-based detectors such as *LISA* Cutler (1998), Amaro-Seoane et al. (2012) will observe GW signals in their sensitive band for up to several weeks to months. Compounding the cost of inference due to long signal duration, many overlapping signals will be in band at any one time, with many sub-threshold signals contributing to a non-Gaussian background (Smith & Thrane 2018). Ongoing research combining some of the inference methods described here will be crucial for realizing the full potential of GW detector data.

6 FURTHER APPLICATIONS

We have focused on individual event inference problems where the only free parameters are those of signals described by (approximations to) General Relativity, and data calibration. Nested sampling methods have been shown to be robust for estimating evidences and posteriors in parameter spaces that have many tens to hundreds of parameters (Feroz et al. 2009; Allison & Dunkley 2014). Thus, our results demonstrate that provided the inference problem is dominated by the cost of the likelihood function, then parallelized nested sampling will offer comparable speedups for inferences in which the models and parameter spaces are significantly larger and more complex than those which we have considered. For example, it is increasingly common for analyses to estimate not just signal parameters but also those of models for the noise power spectral density (Cornish & Littenberg 2015). Additionally, signal models in alternative theories of gravity are parametrized by many more than the 15–17 parameters which describe binary-merger signals in general relativity (Abbott et al. 2016). Thus, our method is extendable to a wide class of important (astro)physical analyses on individual GW events.

In addition, to inference on individual GW events, parallel nested sampling will be useful in population (hierarchical) inference studies which estimate ensemble properties of GW events, such as the mass spectrum of BBHs. In population inference, posterior samples from many events are combined self-consistently to infer information about the underlying distribution from which the samples were drawn. Typically, the cost of population inference scales like the number of samples per event multiplied by the number of events (Talbot et al. 2019). In these problems, the cost of evaluating parametrized models dominates the cost of inference, as in inference on individual GW events considered in this paper. This implies that the cost of population inference studies should be reduced according to the scaling in equation (10). As such, our method may be important when the number of events becomes very large, e.g. as LIGO/Virgo/KAGRA observe many hundreds of events. We note that GPU acceleration has already been shown to accelerate population inference by between one and two orders of magnitude (Talbot et al. 2019). As such, parallel nested sampling may not be necessary until the volume of data required for population inference exceeds the memory capacity of GPUs.

Parallelized nested sampling may also serve as a useful tool in tackling inference on signals as seen by third-generation detectors, e.g. Einstein Telescope and Cosmic Explorer Maggiore et al. (2020) and Reitze et al. (2019a). Astrophysical analysis for these instruments will be significantly more complex: many signals will be in-band simultaneously, and signals may be in band for up to several tens of minutes (Sathyaprakash et al. 2012; Abbott et al. 2017; Reitze et al. 2019b). Thus methods that can alleviate aspects of the wall-time of inference will be valuable as the demands and complexity of data analysis increase. As we note in Section 5, a combination of parallel nested sampling together with reduced order, or relative binning techniques, could lead to dramatic performance improvements. These could be crucial to inference with third-generation detectors, or space-based detectors such as *LISA* Amaro-Seoane et al. (2012), where in-band signals may be observable for several hours, to many

days or weeks. It is unlikely that parallelized sampling methods alone will be sufficient to tackle the complex inference challenges posed by future detectors. However, a combination of parallel sampling with reduced order methods could be a stepping stone towards an inference algorithm capable of handling signal-dominated GW data.

Beyond transient GW astronomy, parallel nested sampling and our `pBilby` code should have applications throughout astronomy and astrophysics. For example, the serial variant of `pBilby` has been used in a number of studies in pulsar and radio astronomy (Cho et al. 2020; Jiang et al. 2020; Lower et al. 2020; Zhu & Thrane 2020). The methods presented in this paper should offer greater scalability of data analysis in these fields.

7 CONCLUSION

Parallelized nested sampling, deployed at scale on a high-performance CPU cluster, reduces the wall-time of inference according to equation (10). It does not approximate either GW signal models, or the statistical properties of the data; is accurate, flexible, scalable, and easy to implement. As such, is can be used in a broad variety of inference analyses. We have demonstrated reductions in wall time from several years to several days for realistic LIGO-Virgo analyses that use cutting-edge GW signal and data-calibration models.

We have argued that the measured speedup achieved by parallel nested sampling should apply irrespective of the type of data or models being used, provided the dominant cost of inference stems from expensive calls to likelihood/model functions. As such, our method – and code, `pBilby` – should be useful for other expensive inference problems, such as hierarchical inference in GW astronomy.

While potentially computationally expensive, parallel nested sampling none the less affords greatly expedited inferences on GWs provided one has access to a high-performance computer cluster. Given the increasing availability of clusters, together with cloud-computing resources, parallelized nested sampling should be a useful tool to both the LIGO-Virgo-KAGRA Collaboration, as well as to independent research groups in astronomy more broadly.

ACKNOWLEDGEMENTS

This work is supported through Australian Research Council (ARC) Centre of Excellence CE170100004. The analyses presented in this paper were performed using the supercomputer cluster at the Swinburne University of Technology (SSTAR). This document has LIGO Document number P1900255-v1. We would like to thank Mathew Pitkin, Roberto Cotesta, Simon Stevenson, Serguei Ossokine, and Scott Coughlin for extensive testing of `pBilby`, and Eve Chase for providing information about the GWTC-1 analyses performed using SEOBNRv3. Additional thanks to Colin Capano for reviewing this manuscript. Thanks also to the SSTAR system admins for their support with all things MPI and for their patience. We are grateful for insightful comments from Vivien Raymond, Eve Chase, Richard O'Shaughnessy, Moritz Hubner, Michele Vallisneri, Alessandra Buonanno, Vicky Kalogera, and the LIGO-Virgo Parameter Estimation and Coalescing Compact Binary working groups. Additional thanks to Joshua Speagle for pointing out the scaling relation for parallel nested sampling. 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 is funded by the U.S. National Science Foundation. Virgo is funded by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale della Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by Polish and Hungarian institutes.

DATA AVAILABILITY

The `pBilby` software library is available at https://git.ligo.org/lscsoft/parallel_bilby/.

REFERENCES

Abbott B. P. et al., 2016, Phys. Rev. Lett., 116, 221101
Abbott B. P. et al., 2017, Class. Quantum Gravity, 34, 044001
Abbott B. P. et al., 2018, Phys. Rev. Lett., 121, 161101
Abbott R. et al., 2019a, preprint (arXiv:1912.11716)
Abbott B. P. et al., 2019b, Phys. Rev. X, 9, 011001
Abbott B. et al., 2019c, Phys. Rev. D, 100
Abbott B. P. et al., 2019d, Phys. Rev. Lett., 123, 011102
Abbott B. P. et al., 2019e, ApJ, 882, L24
Allison R., Dunkley J., 2014, MNRAS, 437, 3918
Amaro-Seoane P. et al., 2012, Class. Quantum Gravity, 29, 124016
Ashton G. et al., 2019, ApJS, 241, 27
Bayes T., Price R., 1763, Phil. Trans. R. Soc., 53, 370
Biwer C. M., Capano C. D., De S., Cabero M., Brown D. A., Nitz A. H., Raymond V., 2019, PASP, 131, 024503
Blackman J. et al., 2017, Phys. Rev. D, 96, 024058
Cahillane C. et al., 2017, Phys. Rev. D, 96, 102001
Canizares P., Field S. E., Gair J., Raymond V., Smith R., Tiglio M., 2015, Phys. Rev. Lett., 114, 071104
Chatziioannou K. et al., 2019, Phys. Rev. D, 100, 104015
Cho H. et al., 2020, ApJ, 891, L38
Chopin N., Robert C. P., 2010, Biometrika, 97, 741
Christensen N., Meyer R., 2001, Phys. Rev. D, 64, 022001
Cornish N. J., Littenberg T. B., 2015, Class. Quantum Gravity, 32, 135012
Cutler C., 1998, Phys. Rev. D, 57, 7089
Cutler C., Flanagan E. E., 1994, Phys. Rev. D, 49, 2658
Dalcin L. D., Paz R. R., Kler P. A., Cosimo A., 2011, Adv. Water Resour., 34, 1124
Dietrich T., Bernuzzi S., Tichy W., 2017, Phys. Rev. D, 96, 121501
Dietrich T. et al., 2019, Phys. Rev. D, 99, 024029
Elvira V., Martino L., Robert C. P., 2018, preprint (arXiv:1809.04129)
Feroz F., Hobson M. P., Bridges M., 2009, MNRAS, 398, 1601
Field S. E., Galley C. R., Hesthaven J. S., Kaye J., Tiglio M., 2014, Phys. Rev. X, 4, 031006
Finn L. S., Chernoff D. F., 1993, Phys. Rev. D, 47, 2198
GraceDB, 2019, GraceDB – Gravitational-Wave Candidate Event Database, https://gracedb.ligo.org/superevents/public/O3/
Handley W. J., Hobson M. P., Lasenby A. N., 2015, MNRAS, 453, 4384
Hannam M., Schmidt P., Bohé A., Haegel L., Husa S., Ohme F., Pratten G., Pürrer M., 2014, Phys. Rev. Lett., 113, 151101
Hastings W. K., 1970, Biometrika, 57, 97
Husa S., Khan S., Hannam M., Pürrer M., Ohme F., Jiménez Forteza X., Bohé A., 2016, Phys. Rev. D, 93, 044006
Jiang J.-L., Tang S.-P., Wang Y.-Z., Fan Y.-Z., Wei D.-M., 2020, ApJ, 892, 55
Khan S., Husa S., Hannam M., Ohme F., Pürrer M., Jiménez Forteza X., Bohé A., 2016, Phys. Rev. D, 93, 044007
Khan S., Ohme F., Chatziioannou K., Hannam M., 2020, Phys. Rev. D, 101, 024056
Kish L., 1995, Survey Sampling, 3rd edn. Wiley-Interscience, Oxford
Lange J., O'Shaughnessy R., Rizzo M., 2018, preprint (arXiv:1805.10457)
Lower M. E. et al., 2020, MNRAS, 494, 228
Maggiore M. et al., 2020, J. Cosmol. Astropart. Phys., 2020, 050
Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H., Teller E., 1953, J. Chem. Phys., 21, 1087
Ossokine S. et al., 2020, preprint (arXiv:2004.09442)
Pan Y., Buonanno A., Taracchini A., Kidder L. E., Mroué A. H., Pfeiffer H. P., Scheel M. A., Szilágyi B., 2014, Phys. Rev. D, 89, 084006

Pankow C., Brady P., Ochsner E., O'Shaughnessy R., 2015, Phys. Rev. D, 92, 023002
Payne E., Talbot C., Thrane E., 2019, Phys. Rev. D, 100, 123017
Pürrer M., 2014, Class. Quantum Gravity, 31, 195010
Pürrer M., 2016, Phys. Rev. D, 93, 064041
Reitze D. et al., Bulletin of the American Astronomical Society, , 2019a, 51, 141
Reitze D. et al., 2019b, preprint (arXiv:1907.04833)
Romano J. D., Cornish N. J., 2017, Living. Rev. Relativ., 20, 2
Sathyaprakash B. et al., 2012, Class. Quantum Gravity, 29, 124013
Skilling J., 2006, Bayesian Anal., 1, 833
Smith R., Thrane E., 2018, Phys. Rev. X, 8, 021019
Smith R., Field S. E., Blackburn K., Haster C.-J., Pürrer M., Raymond V., Schmidt P., 2016, Phys. Rev. D, 94, 44031
Smith R., Ashton G., Vajpeyi A., Talbot C., the LIGO Scientific Collaboration, 2019, Parallel Bilby git repository, https://git.ligo.org/lscsoft/parallel_bilby/
Speagle J. S., 2020, MNRAS, 493, 3132
Talbot C., Smith R., Thrane E., Poole G. B., 2019, Phys. Rev. D, 100, 043030
The LIGO Scientific Collaboration, 2020, preprint (arXiv:2004.08342)
The LIGO Scientific collaboration, 2019, preprint (arXiv:1904.03187)
Vallisneri M., Kanner J., Williams R., Weinstein A., Stephens B., 2015, J. Phys. Conf. Ser., 610, 012021
Varma V., Field S. E., Scheel M. A., Blackman J., Kidder L. E., Pfeiffer H. P., 2019, Phys. Rev. D, 99, 064045
Veitch J., Vecchio A., 2010, Phys. Rev. D, 81, 062003
Veitch J. et al., 2015, Phys. Rev. D, 91, 042003
Wysocki D., O'Shaughnessy R., Lange J., Fang Y.-L. L., 2019, Phys. Rev. D, 99, 084026
Zackay B., Dai L., Venumadhav T., 2018, preprint (arXiv:1806.08792)
Zhu X.-J., Thrane E., 2020, preprint (arXiv:2004.10944)

APPENDIX

The nested sampling algorithm pseudo-code described in Section 3 is shown in Algorithm 1. The parameters describing the BBH signal injections, as well as their priors, used in Section 4 are shown in Table 2.

Algorithm 1: Static Parallel Nested Sampling

```
// Initialize a pool of n CPUs
// Initialize live points
do in parallel
  | Draw K "live" points {θ_1, ..., θ_K} from the prior π(θ)
end
// Main sampling loop
while stopping criterion not met do
  | Compute the minimum likelihood L^min among the current set of live points
  | // Parallel sampling step on "worker nodes"
  | do in parallel
  |   | Draw n − 1 samples {θ_i}_{i=1}^{n−1} from the prior
  |   | Accept n′ samples subject to the constraint L(θ_i) > L^min, otherwise discard
  | end
  | // Gather parallel samples on "head node"
  | for i = 1 to n′ do
  |   | Add the k^th live point θ_k associated with L^min to a list of "dead" points
  |   | Replace θ_k with θ′_i
  |   | Compute the minimum likelihood L^min among the current set of live points
  | end
  | // Check whether to stop
  | Evaluate stopping criterion
  | // Check whether to update prior sampling method/parameters
  | Evaluate bounding distribution
end
// Add final live points
while K > 0 do
  | Compute the minimum likelihood L^min among the current set of live points
  | Add the kth live point θ_k associated with L^min to a list of "dead" points
  | Remove θ_k from the set of live points
  | Set K = K − 1
end
```

This paper has been typeset from a TEX/LATEX file prepared by the author.