Expanding RIFT: Improving performance for GW parameter inference

J. Wofford,[1] A. B. Yelikar,[1] Hannah Gallagher,[1] E. Champion,[1] D. Wysocki,[1, 2] V. Delfavero,[1] J. Lange,[3, 1] C. Rose,[2] V. Valsan,[2] S. Morisaki,[4, 2] J. Read,[5] C. Henshaw,[6] and R. O'Shaughnessy[1]

[1] *Center for Computational Relativity and Gravitation, Rochester Institute of Technology, Rochester, New York 14623, USA*
[2] *University of Wisconsin-Milwaukee, Milwaukee, WI 53201, USA*
[3] *University of Texas, Austin, TX 78712, USA*
[4] *Institute for Cosmic Ray Research, The University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba 277-8582, Japan*
[5] *California State University Fullerton, Fullerton, CA 92831, USA*
[6] *Georgia Institute of Technology, Atlanta, GA 30332, USA*

The Rapid Iterative FiTting (RIFT) parameter inference algorithm provides a framework for efficient, highly-parallelized parameter inference for GW sources. In this paper, we summarize essential algorithm enhancements and operating point choices for the RIFT iterative algorithm, including settings used for analysis of LIGO/Virgo O3 observations. We also describe other extensions to the RIFT algorithm and software ecosystem. Some extensions increase RIFT's flexibility to produce outputs pertinent to GW astrophysics. Other extensions increase its computational efficiency or stability. Using many randomly-selected sources, we assess code robustness with two distinct code configurations, one designed to mimic settings as of LIGO/Virgo O3 and another employing several performance enhancements. We illustrate RIFT's capabilities with analysis of selected events.

I. INTRODUCTION

Ground-based gravitational wave (GW) detectors including Advanced LIGO [1] and Virgo [2, 3], now joined by KAGRA [4] continue to identify coalescing compact binaries [5–10]. Many more GW observations are expected as observatories reach design sensitivity [11], with detection rates expected to exceed one per day when detectors reach their design sensitivity. Their properties can be characterized via Bayesian inference, comparing data to the expectations given different potential sources [5–7, 12–18]. At present, a wide variety of phenomenological or interpolated estimates for GW from a merging binary are available [19–24]. Inferences using these models can be very computationally costly, particularly when using the best available models.

The Rapid Iterative FiTting (RIFT) [17] is one of several parameter inference algorithms [18, 25] used to produce the initial interpretation of GW observations [7–10, 14, 15, 26–28]. The most popular approaches for gravitational wave parameter inference rely on Markov chains, within either Markov Chain Monte Carlo or nested sampling codes; see [29] for a recent review. By contrast, RIFT performs Bayesian inference through Monte Carlo quadrature, combined with an iterative algorithm to successively approximate pertinent likelihoods [17, 30]. RIFT's structure offers novel opportunities to efficiently construct and re-use the outputs needed for Bayesian parameter and population inference [17, 31]. RIFT naturally exports a continuous likelihood versus source parameters, valuable for population inference [32] and critical when downstream use employs tightly constrained source populations like a concrete nuclear equation of state [32, 33]. RIFT can use these exported likelihoods to produce low-cost, high-accuracy model evidences, allowing for model selection between different source physics scenarios (e.g., between an aligned or precessing BH binary [34], between a model with and without higher order multipole GW [34], or between different models for the nuclear equation of state [35]). RIFT can natively perform multimodel inference using both interpolated likelihoods and the raw data from which they are generated [31], important given notable modeling systematics. These multimodel inferences enable extension and re-use of previous analyses to incorporate additional modeling as needed. With companion software [36], RIFT's workflow can even identify its own settings, for example expanding mass prior ranges as needed.

RIFT's structure also offers novel opportunities to perform inference using large-scale distributed computing, while mitigating the downside of intermittently unreliable computing environments. RIFT's computational cost is dominated by an embarrassingly parallel exploration phase, where many source parameters are independently compared to the data, allowing it to scale to very large computing resources at need. Inevitably, these large workflows naturally discover any poorly-configured hardware and software, particularly when interpreting many candidate sources. RIFT's iterative structure, however, means that its workflow and settings can be successively adapted to compensate for infrastructure problems, avoiding poorly-behaved nodes [36]. RIFT achieves the aforementioned flexibility with low computational cost, based at root on an efficient re-representation of the GW likelihood [16], combined with GPU-accelerated likelihood evaluation [30]. The RIFT software ecosystem thus provides a robust framework to reduce the overall cost of inferring source parameters, enabling larger-scale analyses and greater scope to probe waveform systematics. Reducing overall evaluation cost and runtime has many potential downstream implications, not least including low-latency parameter inference [37–39] needed to facilitate multimessenger followup observations; see, e.g., [40] and references therein.

In this paper, we introduce several extensions of the original RIFT implementation, all available through its open-source code repository [41]. Several of these features were used to interpret gravitational wave sources during O3, the third observing run of the Advanced LIGO and Advanced Virgo instruments; see, e.g., [8, 9, 27, 28, 34]. This paper is organized as follows. In Section II we briefly review the essential elements of the RIFT algorithm in regular use prior to the O3 analysis. In Section III, we describe essential additions employed in the O3 analysis, and validate our production setup with standard tests. In Section IV, we describe extensions to RIFT's O3-style approach, to improve its efficiency, flexibility, and capability for unsupervised operation. While many of these extensions were first introduced in the RIFT source several years ago, and have been applied in other work, this paper provides the first detailed description of these updates. In Section V, we enumerate the specific prototype RIFT configurations we recommend for regular use and which we assess here. In Section VI, we validate several of the key RIFT elements described above with targeted and statistical tests. Finally, in Section VII we report on reanalysis of selected real observations, to highlight RIFT's improved performance and capability. Several of our demonstrations are performed on real gravitational wave data, available from the Gravitational Wave Open Science center [42]. Our study provides a backward- and forward-looking description of RIFT code development, as needed for long-term sustainable reproducibility of its GW inference results.

II. RIFT REVIEW

A coalescing compact binary in a quasicircular orbit can be completely characterized by its intrinsic and extrinsic parameters. By intrinsic parameters we refer to the binary's masses m_i, spins, and any quantities characterizing matter in the system. By extrinsic parameters we refer to the seven numbers needed to characterize its spacetime location and orientation. We will express masses in solar mass units and dimensionless spins in terms of Cartesian components $\chi_{i,x}, \chi_{i,y}, \chi_{i,z}$, expressed relative to a frame with $\hat{\mathbf{z}} = \hat{\mathbf{L}}$ and (for simplicity) at the orbital frequency corresponding to the earliest time of computational interest (e.g., an orbital frequency of $\simeq 10$Hz). We will use λ, θ to refer to intrinsic and extrinsic parameters, respectively.

As illustrated in Figure 1, RIFT principally consists of a two-stage iterative process to interpret gravitational wave observations d via comparison to predicted gravitational wave signals $h(\boldsymbol{\lambda}, \boldsymbol{\theta})$. In one stage, implemented by many workers in parallel (denoted by ILE in the diagram), RIFT computes a marginal likelihood

$$\mathcal{L}(\boldsymbol{\lambda}) \equiv \int \mathcal{L}_{\rm full}(\boldsymbol{\lambda}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) d\boldsymbol{\theta} \tag{1}$$

from the likelihood $\mathcal{L}_{\rm full}(\boldsymbol{\lambda}, \theta)$ of the gravitational wave signal in the multi-detector network, accounting for detector response; see [16, 17] for a more detailed specification. In the second stage, denoted by CIP in the diagram, RIFT performs two tasks. First, it generates an approximation to $\mathcal{L}(\lambda)$ based on its accumulated archived knowledge of marginal likelihood evaluations $(\lambda_\alpha, \mathcal{L}_\alpha)$. Second, using this approximation, it deduces the (detector-frame) posterior distribution

$$p_{\rm post} = \frac{\mathcal{L}(\boldsymbol{\lambda}) p(\boldsymbol{\lambda})}{\int d\boldsymbol{\lambda} \mathcal{L}(\boldsymbol{\lambda}) p(\boldsymbol{\lambda})}. \tag{2}$$

where prior $p(\boldsymbol{\lambda})$ is the prior on intrinsic parameters like mass and spin.

A. Evaluating the marginalized likelihood

As described in previous work [16], RIFT's likelihood uses physical insight to carry out its evaluation particularly efficiently for binaries with similar intrinsic parameters but different extrinsic parameters. At a high level, RIFT relies on a decomposition of arbitrary gravitational wave signals $h(t)$ into physical basis signals $h_{lm}(t)$, associated with a (spin-weighted) spherical harmonic decomposition of radiation in all possible emission directions. This decomposition allows RIFT to compute cross-correlations between this basis and each detector's data; the likelihood for arbitrary source orientations, sky positions, and distances follows by a weighted average of these cross-correlation timeseries. Recently, Wysocki and collaborators described a very efficient GPU-accelerated implementation of the likelihood, enabling significant speed improvements [30].

Given the likelihood $\mathcal{L}_{\rm full}(\boldsymbol{\lambda}, \boldsymbol{\theta})$, RIFT evaluates the marginal likelihood via an adaptive Monte Carlo integrator:

$$\mathcal{L}(\boldsymbol{\lambda}) \simeq \frac{1}{N} \sum_k \mathcal{L}_{\rm full}(\boldsymbol{\lambda}, \theta_k) p(\boldsymbol{\theta}_k) / p_s(\boldsymbol{\theta}_k) \tag{3}$$

Inherited from its progenitor [16], RIFT performed this Monte Carlo integrator using an (adaptive) sampling prior p_s which has product form, consistent with standard Cartesian adaptive integrators [43–45]. After a large block of evaluations, each one-dimensional marginal sampling prior can be updated to more closely conform to the support of the integrand, based on a (smoothed) fixed-size one-dimensional histogram for each adaptive dimension. While very powerful, this adaptive integrator limits RIFT for two common applications. First, its proposed sampling prior is extremely inefficient when the integrand exhibits strong correlations between many dimensions. Second, its python-based implementation generates random numbers with its CPU, which must be transferred back and forth to and from the GPU when evaluating the likelihood. In this work, we will examine two alternatives which alleviate each limitation in turn.

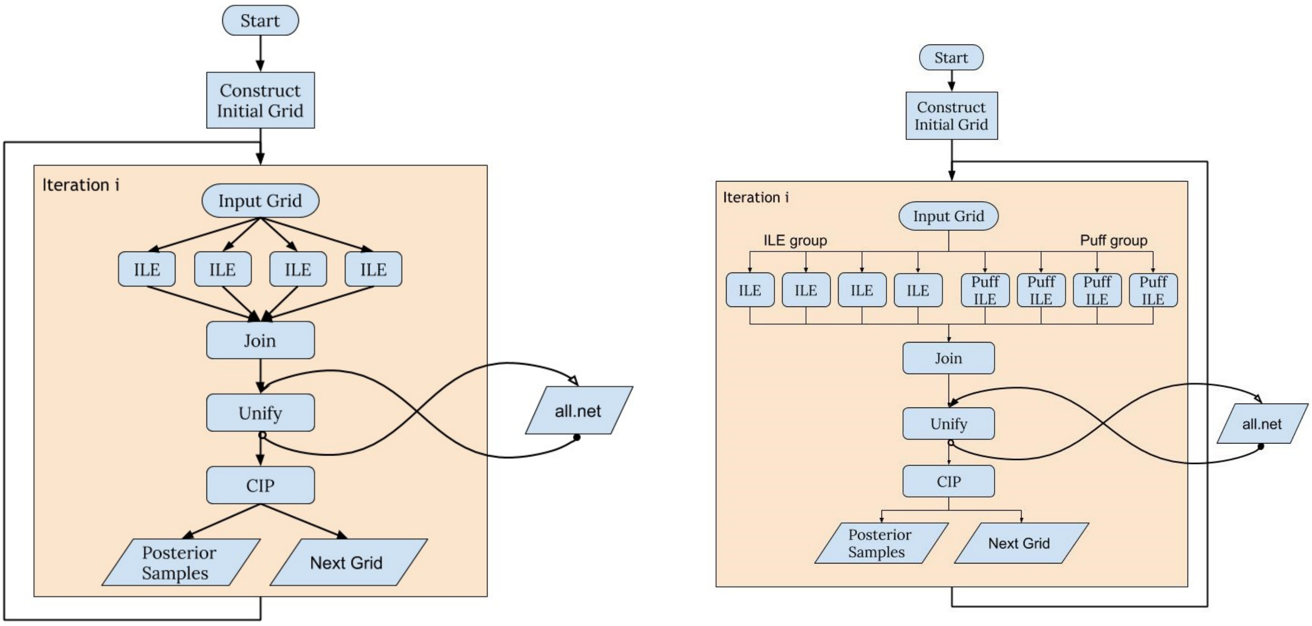

FIG. 1: **RIFT workflow**: A flowchart of the RIFT iterative workflow. The individual worker jobs (ILE, an acronym derived from "integrate likelihood:extrinsic") compute the marginalized likelihood; the likelihood fitting and posterior-generation code (CIP, an acronym derived from "construct intrinsic posterior") uses these marginal likelihoods to estimate the posterior distribution. The join stage combines the results from ILE workers for a specific iteration; the "unify" step accumulates results over multiple iterations into a single file. The iterative workflow starts with an initial grid, encompassing some estimate of initial parameters. Not shown here explicitly are (a) the jittering or "puffball" step, which generates a mirror grid with a subset of parameters randomly perturbed, accordingly to the samples' covariance matrix, whose results are used as the inputs to the "puff group" shown on the right; (b) the option for multiple CIP instances, which work in parallel to more efficiently generate slowly-converging posteriors; and (c) convergence testing, to assess when to terminate the iterative process.

B. Likelihood interpolation and posterior distributions

To estimate $\mathcal{L}$ from discrete samples $\lambda_\alpha, \mathcal{L}_\alpha$, RIFT used Gaussian process regression. Following the RIFT paper, for brevity and to be consistent with conventional notation, in this section we denote $\boldsymbol{\lambda}_\alpha$ by x and $\ln \mathcal{L}_{\mathrm{marg}\alpha}$ by y. In this approach, we estimate the expected value of $y(x)$ from data x_* and values y_* via

$$\langle y(x) \rangle = \sum_{\alpha,\alpha'} k(x, x_{*,\alpha}) (K^{-1})_{\alpha,\alpha'} y_{*,\alpha'} \tag{4}$$

where α is an integer running over the number of training samples in (x_*, y_*) and where the matrix $K = k(x_\alpha, x'_\alpha)$ y_*. We employ a kernel function $k(x, x')$ which allows for uncertainty in each estimated training point's value $y_{*,\alpha}$ due to Monte Carlo integration, as well as a conventional squared exponential kernel to allow for changes in the functions versus parameters:

$$k(x, x') = \sigma_o^2 e^{-(x-x')Q(x-x')/2} + \sigma_n^2 \delta_{x,x'} \tag{5}$$

The hyper-parameters of this kernel (σ_o, σ_n and the positive-definite symmetric matrix Q) are chosen to minimize the likelihood of our training data x_k, y_k with covariance matrix K:

$$\ln \ell(y) = -\frac{1}{2}(y/\sigma)^T K^{-1} (y/\sigma) - \ln \sqrt{\frac{|K|}{(2\pi)^N \sigma^2}} \tag{6}$$

where σ_k are the individual estimated uncertainties in each y_k and $\sigma^2 \equiv \prod_k \sigma_k^2$. We perform all Gaussian process interpolation with widely-available open-source software [46]. The computational cost of full-scale Gaussian process optimization and evaluation increases rapidly with the dimension D of the matrix K, as D^3 and D^2 respectively.

Given the likelihood, fair samples from the posterior distribution are generated by the following two-step process, described in the RIFT paper. First, using the likelihood estimate $\hat{\mathcal{L}}_{\mathrm{marg}}$ and the same adaptive Monte Carlo integrator described above, we perform the Monte Carlo integral $\int d\boldsymbol{\lambda} \hat{\mathcal{L}}_{\mathrm{marg}} p(\boldsymbol{\lambda})$, producing sample locations $\boldsymbol{\lambda}_k$ and associated weights $w_k = \hat{\mathcal{L}}_{\mathrm{marg}} p(\lambda)/p_s(\lambda)$. Second, we make a fair draw from these weighted samples.

C. Exploring the parameter space

For expedient convergence, RIFT has two additional methods to explore the parameter space: dithering and incremental dimensionality.

After the posterior is produced and a candidate grid generated, RIFT can optionally produce a second candidate grid derived from and supplementing the first. In this second grid, points are generated by performing dithering on (or randomization of) arbitrary combinations of parameters, then rejecting unphysical combinations. For example, the candidate points may have small (correlated) offsets in chirp mass, η, and $\chi_{\rm eff}$ added, with offset covariance matrix set by the covariance matrix of the input candidate grid. Particularly after several iterations, this dithering can remedy a significantly-offset initial grid which misses the true likelihood maximum. This dithering also insures good sampling outside the boundaries of the target point. The original RIFT paper [17] only implemented correlated dithering based on sample covariance. Later in this paper, we describe incremental improvements to the dithering process which further improve performance.

RIFT can also employ different parameterizations at each stage. In particular, as explained in the RIFT paper, RIFT can employ likelihood models with increasing numbers of parameters, starting with the dominant degrees of freedom (e.g., $\mathcal{M}_c$, η, and $\chi_{\rm eff}$ for massive BHs) and adding in subdominant degrees of freedom in subsequent iterations. This approach helps address a tradeoff between cost and complexity. For the first few iterations, RIFT needs to identify the peak likelihood, as characterized by the dominant parameters. Using all model parameters can be highly counterproductive, as fits with all degrees of freedom require overwhelming numbers of evaluations $\lambda_k, \mathcal{L}_k$ in order to avoid overfitting/under-resolving. (With too few evaluations and several irrelevant parameters included, the Gaussian process behaves pathologically.) By reducing the number of poorly-constrained parameters early on, we can employ far fewer points early on. Because of the computational cost of GP regression in high dimensions and with many points, this was essential for handling complex sources like precessing BH binaries with the original GP likelihood estimate. The appropriate dimensional hierarchy depends on the physics involved (e.g., configurations with high mass; BHNS with strong precession; NS-NS binaries with tides; et cetera) but is well-motivated from simple Fisher matrix arguments. Specifically, the component masses and a measure of aligned binary spin (e.g., $\chi_{\rm eff}$) approximately characterize the dominant degrees of freedom for nearly-nonprecessing binaries, particularly when organized as the chirp mass $\mathcal{M}_c$ and symmetric mass ratio η. As most observed binaries exhibit nearly no precession, these variables form a natural set to adopt for the first iterations. As transverse and other spin degrees of freedom have a subdominant impact on the marginal likelihood, we can add these incrementally, after obtaining a converged estimate for the behavior for nonprecessing degrees of freedom. Prior to O3, these choices were made by humans, and the iteration plan assembled by hand and adjusted at need.

D. Convergence testing

The RIFT paper [17] introduced a procedure to assess convergence: for each marginal 1d distribution, compute the KL divergence between successive iterations, $D_{KL}(p|q) = \int dxp \ln p/q$ where p, q are 1-dimensional probability densities. A fiducial convergence threshold was $O(10^{-2})$ for each variable. At the time, these KL divergences were evaluated using KDE-based estimates of each 1d marginal distribution. Subsequently, Delfavero [47] introduced and assessed a simpler and more stable 1d convergence diagnostic: the net L^1 difference between each one-dimensional CDF:

$$D_{L^1} = \max_x |\hat{P}(x) - \hat{Q}(x)| \tag{7}$$

where P, Q are empirical CDFs associated with the two sample sizes. The L^1 norm has been well-studied in the context of KS tests. Delfavero proposed a convergence threshold of $-\frac{1}{N}\ln\alpha$, where α was the desired confidence level of the test and N is the common sample size.

Finally, to better capture correlations in our convergence tests, we have also implemented a simplified multi-dimensional convergence test, which compares the empirical means μ_1, μ_2 and covariance matrices Σ_1, Σ_2 associated with two sets of fair samples under the assumption that both characterize a Gaussian distribution:

$$\begin{aligned} D_{KL} = \frac{1}{2} \Big[&(\mu_2 - \mu_1)\Sigma_2^{-1}(\mu_2 - \mu_1) \\ &+ ({\rm Tr}(\Sigma_2^{-1}\Sigma_1) - d) + \ln\frac{|\Sigma_2|}{|\Sigma_1|} \Big] \end{aligned} \tag{8}$$

where d is the dimension of the problem. Like the 1d KL divergence test, we adopt a fiducial convergence threshold of 10^{-2}. The user can select any subset of variables (and any coordinates) with which to evaluate this joint test, though we recommend using $\mathcal{M}_c, \eta, \chi_{\rm eff}$ at a minimum. Unless otherwise noted, we adopt and report on runs using the latter convergence diagnostic below.

E. Limitations

To recap, RIFT organizes Bayesian inference as an iterative two-stage process. In one stage, it rapidly evaluates a marginal likelihood $\mathcal{L}$ for compact binary source parameters λ, via a Monte Carlo integral. In another stage, it uses its accumulated knowledge of previous likelihood evaluations $(\lambda_k, \mathcal{L}_k)$ to estimate $\mathcal{L}$ as a function of arbitrary λ; from this estimate, it draws samples for the posterior for λ, again via a Monte Carlo integral. The output of the second stage is passed back to the first, until the results converge.

While in principle effective, in practice this strategy relied principally on high-dimensional fits and dithering to explore the model space. The Gaussian process fits employed previously, however, were excessively parsimonious outside the previously trained domain, prohibiting exploration. Dithering was an essential but occasionally fragile element of our procedure to explore the parameter space. For this reason, in Section III below we introduced several additional techniques to automate exploration of the binary parameter space, particularly by improved dithering and by systematically hierarchically adding degrees of freedom with increasingly subdominant effects on typical likelihoods.

Additionally, as originally implemented, key elements of the RIFT parameter inference strategy had notable sources of inefficiency. For example, the adaptive Monte Carlo integrator inherited from Pankow et al [16] is both relatively slow and algorithmically inflexible, not well-suited to sample distributions with strong correlations which its adaptive algorithm's built-in assumptions can't efficiently replicate. More painfully, the standard interpolation implementation adopted (Gaussian process regression) scaled very inefficiently with the number of input likelihood evaluations, placing severe limits on the scale of problems that could be usefully addressed. Described at greater length below, these defects are being addressed by the methods first described in this work.

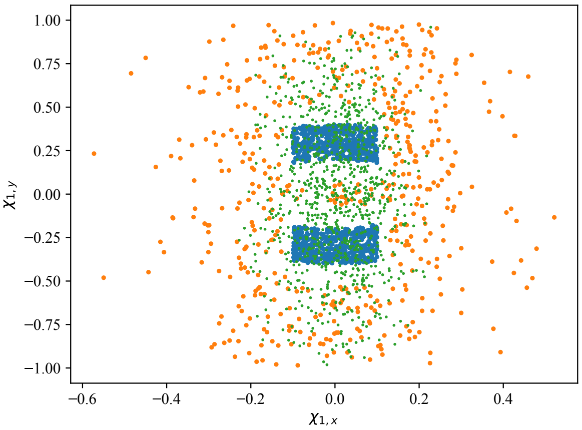

FIG. 2: Demonstration of dithering algorithms: The blue points show synthetic input data which is uniform in $\chi_{1x}, \chi_{1,y}$ in $|\chi_1| \in [0.2, 0.4]$ and $\chi_{1,x} \in [-0.1, 0.1]$. The green points show the orginal dithering procedure with $F = 1$ and $\epsilon = 0$. The orange points show the result with $F = 3, \epsilon = 0.01$. The modified algorithm more efficiently explores the outskirts of the otherwise-sampled area.

III. RIFT DURING O3

In the O3 era, production-scale RIFT calculations employed several additional operating-point choices and features which have not previously been described in the literature.

A. Waveform support

RIFT inference is performed using the spin-weighted spherical harmonic waveforms $h_{lm}(t)$ or $h_{lm}(f)$ [16, 17, 30, 48], usually computed from binary parameters through the `lalsimulation` library. During the O3 analysis era and publications [9, 28], commonly-used estimates for the gravitational waves emitted from quasicircular binary merger included IMRPhenomD [49, 50], IMRPhenomPv2 [19], IMRPhenomXPHM [23], and SEOBNRv4PHM [24, 51]. While the illustrations and tests presented in this work draw upon these established source models, we point out that RIFT's likelihood-based approach enables transparent visualization, calculation, and mitigation of the impact of waveform systematics [17, 31].

B. Dithering and exploration

RIFT originally generated candidate future samples using an estimate $\hat{\mathcal{L}}_{\rm marg}$ of the marginal likelihood. While this method could very efficiently explore the parameter space, it would often only explore within the neighborhood already explored, even with high likelihood on the edge of the previously-explored set. In O3, we therefore added a simple dithering algorithm, to supplement candidate points with a companion set, where each companion point was drawn from the original sample but had added random uncertainty in selected parameters. We chose random uncertainties centered on but with larger than the covariance than the available training data x_α, with the goal of enveloping the posterior and its marginal-significance tails, to stabilize our estimate of the log likelihood in a region not well served by RIFT's normal approach for selecting training data (draws from the posterior). [In a sense, we use an "overdispersed" investigation of training data x_α to avoid an "underdispersed" final posterior.] We further generalized our dithering algorithm in three ways: by rejecting dithered samples based on proximity; by allowing the user to eliminate cross terms in the covariance matrix used for dithering; and by allowing the user to request random candidates in any subset of parameters, instead of simple dithering.

Our first dithering algorithm draws δx_α for a specified set of variables α, based on the empirical covariance matrix $\Sigma_{\alpha\beta}$ for those variables. The offsets are randomly drawn from a multivariate normal distribution with covariance matrix $F^2\Sigma$ with F a user-specified factor (chosen as $F = 3$ by default). Without rejection, this method frequently produces candidate points in extremely close proximity to previously densely-explored regions. We therefore allowed the user to specify a threshold ϵ, such that dithered samples with distance $\delta x_\alpha \delta x_\beta \Sigma^{-1}_{\alpha\beta}/d$ smaller than ϵ to any previously-evaluated point would be removed, where d is the dimension of parameters being dithered. As a result, the dithered samples at late times better explore the outskirts of the posterior. Figure 2 shows an example of the two methods, applied to a toy problem. The modified method does not generate samples inside regions otherwise being explored by the input grid. The current implementation adopts the same ϵ for all iterations, and performs dithering and rejection based on the most recent samples rather than all past history.

RIFT's original dithering algorithm would also extend only along the principal axes of existing correlations in its targeted input variables. While GW observations can produce strong correlations near the peak likelihood, farther away from the peak the likelihood surface can exhibit other correlations. As a result, RIFT's original dithering algorithm would not enable efficient identification of subdominant correlations and extended, correlated tails in the posterior distribution. To address this deficiency, we provide the capability to employ two core modifications, though neither is active by default. In the first, the user can modify the covariance matrix used to dither the input samples by requiring $\Sigma_{\alpha,\beta} = \Sigma_{\beta,\alpha} = 0$ for any list of user-specified pairs (α, β). In the second, the user can request that any specific coordinate α is drawn at random, uniformly over a user-specified range. The latter method is extremely useful for marginally-accessible degrees of freedom (e.g., subdominant tidal parameters), which are prone to overfitting.

C. Physics-inspired iterative architecture

RIFT's final results are produced from a set of likelihood evaluations $\{\lambda_\alpha, \ln \mathcal{L}_{\mathrm{marg}\,\alpha}\}$. Nominally, RIFT uses the same prior and likelihood model to produce final results and accumulate these likelihood evaluations, in its iterative process. However, RIFT can also accumulate these likelihood evaluations with any prior, and using iterations with fit estimates which omit known-subdominant degrees of freedom. These choices, denoted as architectures in the text below, can significantly reduce the latency or even overall computational cost, as noted in Section II C. We used them in O3 because higher-dimensional likelihood models require more input data and often higher computational (gaussian-process) cost; whenever plausible, lower-dimensional likelihood models were desirable for exploratory iterations.

Architectures for binary black holes: For all compact binaries without matter, RIFT's O3-era unsupervised approach was very conservative. Specifically, the O3-era RIFT starts with 3 iterations fitting using $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}$, with a volumetric spin prior; 2 iterations fitting with $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}$ and $\chi_- = (m_1\chi_{1,z} - m_2\chi_{2,z})/M$ and a volumetric spin prior; 2 iterations fitting with $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}, \chi_{i,x}, \chi_{i,y}$ for $i = 1, 2$ with a volumetric spin prior; and (if adopting conventional priors) 3 iterations using the same parameters, but with a spin prior that is uniform in spin magnitude. The validation study for this approach is described in Section VI with Figure 14.

This strategy was designed to characterize the massive BH binaries with $\mathcal{M}_c > 20M_\odot$ that were relatively common in O1O2, and was particularly targeted to identify signatures of strong precession. With relatively few cycles in contemporary ground-based instruments, BH binaries with $\mathcal{M}_c > 20M_\odot$ have posterior distributions which only weakly constrain intrinsic parameters except for $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}$. Particularly at very high mass, the transverse spins in particular usually have minimal impact on the posterior distribution. As a result, when investigating massive and possibly precessing BH binaries, we can adopt an architecture which increases in complexity, where the first few iterations use a fit with only $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}$; the next few iterations add an antisymmetric aligned spin χ_-; and the last few iterations use all spin degrees of freedom.

For most massive BH binaries, this unsupervised architecture was massive overkill. However, this configuration is also robust and efficient when the true signal parameters are not covered by the initial candidate grid. Frequently, real GW searches identify parameters well-seperated from the final posterior distribution. Similarly, due to strong model systematics, the true parameters may be recovered with substantial bias with an alternative model.

Architectures for matter: While this paper will describe all pertinent updates to RIFT, we have chosen to emphasize binary black holes and largely eschew matter effects, for clarity deferring new demonstrations of our current and extended matter-related capabilities to future work. When performing an unsupervised analysis with matter, the O3-era RIFT algorithm adopted the same architecture choices as massive binary black holes, in particular assuming the likelihood at leading order only depends on $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}$ and tides. As with binary black holes, we adopted a lower-dimensional model early on, assuming the marginal likelihood depends on the dimensionless tidal deformabilities Λ_i only through $\tilde{\Lambda}$ for most iterations; see [7, 35] for discussion of these parameters. To account for degeneracies, we perform correlated dithering in $\mathcal{M}_c, \eta, \chi_{\mathrm{eff}}, \tilde{\Lambda}$. To further ensure the low-Λ region is well-explored, in O3 we adopted a non-uniform prior on Λ_i which favors small Λ. RIFT results using this approach have been previously presented, including a novel population study [32].

As with the BH-BH case, this architecture is motivated by the physics of binary inspiral. Working to leading order, we characterize the gravitational effects of compact objects with matter by a dimensionless tidal deformability parameter Λ_i. Following convention when presenting results not conditioned on other observations or theory, we adopt a uniform prior on these Λ_i, extending from 0 to 5000 independent of compact object mass. This prior is not well-suited to exploring the tidal parameter space, because real compact objects are subject to an equation of state and thus $\Lambda(m)$ relation which depends strongly on mass, is typically much less than 5000, and goes to zero at high mass. Additionally, tidal effects are highly subdominant and enter at leading order through a single mass-weighted combination $\tilde{\Lambda}$. Exploring the tidal parameter space using the default prior with RIFT is exceptionally inefficient, because the prior strongly disfavors the low-$\tilde{\Lambda}$ configurations associated with the (weak) peak in the marginal likelihood, particularly for very massive NS with extremely small $\Lambda(m)$. Conversely, the approach described above was demonstrably sufficient to enable a multi-event population analysis to recover a proposed NS equation of state and mass/spin distribution from synthetic GW observations [32].

D. Well-motivated initial grids

When given a good starting grid $\{\lambda_k\}$, RIFT converges well. For unsupervised analysis of binary black holes in O3, we used a hypercube in $\mathcal{M}_c, \eta, \chi_{\rm eff}$ chosen based on the search-reported candidate parameters. The chirp mass region was chosen over a logarithmic region of width $\Delta \ln \mathcal{M}_c = \pm(1.5)^2 0.3(v/0.2)^7/\rho$ centered on the reported chirp mass $\mathcal{M}_{c*}$, where v is the smaller of $(\pi \mathcal{M}_{c*} f_{\rm min})^{1/3}$ or 0.2, ρ is the search-reported signal to noise, and $f_{\rm min}$ is the minimum frequency used for parameter inference. The η extent covers from $\eta_{\rm min}$ to 1/4. If the trigger symmetric mass ratio $\eta_* > 0.1$, then $\eta_{\rm min}$ was (by default) the larger of 0.1 and that value η_1 such that $m_2 = 1M_\odot$ given $\mathcal{M}_{c*}, \eta_1$. If the trigger symmetric mass ratio is more extreme ($\eta_* < 0.1$), then $\eta_{\rm min} = 0.25\eta_*$. Finally, the $\chi_{\rm eff}$ interval was chosen to be $\chi_{{\rm eff},*} \pm 0.3/\rho$. This wide region in mass, mass ratio, and aligned spin helped compensate for the often-large biases between search trigger parameters and the true posterior.

For unsupervised investigations involving matter, motivated by plausible nuclear equations of state, we adopt an initial grid which uniformly covers a region in the neighborhood of a fiducial analytic estimate $\Lambda_{fid}(m)$, assumed to be 20 for $m > 2.2M_\odot$ and $3000((2.2 - m/M_\odot)/1.2))^2$ otherwise. Specifically, we uniformly sample $\Lambda \in [\lambda_{min,fid}, \lambda_{max,fid}]$ where $\Lambda_{min,fid}$ is the smaller of 50 and $0.2\Lambda_{fid}$ and $\Lambda_{max,fid}$ is the smaller of $0.2\Lambda_{fid}$ and 1500. With this starting grid, we can recover tidal parameters for realistic NS over a wide range of masses. For example, these settings were adopted in our detailed systematics study about jointly fitting the nuclear equation of state and BNS population [32].

E. Validation of O3 configuration

Among other tests, we validated RIFT O3-era code configurations against generic, randomly chosen merging binaries using a standard probability-probability (PP) plot test [52, 53]. Using RIFT on each source k, with true parameters λ_k, we estimate the fraction of the posterior distributions which is below the true source value $\lambda_{k,\alpha}$ $[\hat{P}_{k,\alpha}(< \lambda_{k,\alpha})]$ for each intrinsic parameter α. After reindexing the sources so $\hat{P}_{k,\alpha}(\lambda_{k,\alpha})$ increases with k for some fixed α, a plot of k/N versus $\hat{P}_k(\lambda_{k,\alpha})$ for both mass parameters can be compared with the expected result $(P(< p) = p)$ and binomial uncertainty interval.

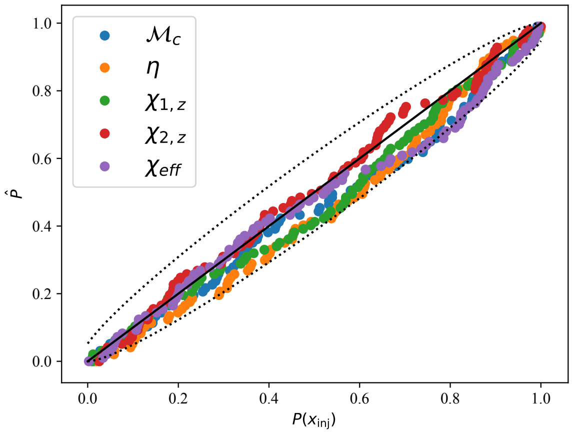

FIG. 3: **Nonprecessing PP plot, early-O3 code**: Probability-probability plot to validate recovery of precessing synthetic signals, following the RIFT methods paper [17]. The synthetic sources and parameter inferences are constructed with IMRPhenomD in gaussian noise with presumed known PSDs, for 3-detector networks, starting the signal at 20 Hz and using a 4096 Hz sampling rate. Detector-frame masses are drawn uniformly in the region bounded by $\mathcal{M}/M_\odot \in [30, 60]$ and $\eta \in [0.2, 1/4]$, and sources are drawn volumetrically between 1.5 and 4Gpc. Both BH dimensionless spins are drawn uniformly in $\chi_{i,z}$. For computational efficiency, all sources in this specific test have a fixed and presumed known sky location.

Other previously-published studies have already reported on comparable PP plot tests, in the context of waveform systematics [54], using the O3-era code. Conversely, Section VI describes more comprehensive tests and PP plots applied to the current edition of the code. However, for completeness, Figure 3 shows the result of one such contemporary O3-era test for intrinsic degrees of freedom.

F. Inefficiencies and Limitations of the O3 configuration

RIFT's development up to O3 was tightly constrained, needing to be completed and assessed well before any O3 analysis, resulting in occasionally fragile and suboptimally efficient but still extremely portable and reproducible tool. RIFT's O3 configurations have many completely arbitrary limitations, introduced both by the difficulties inherent in our software environment and our timeline. The foremost difficulty in operating RIFT remains its organization: multiple independent command-line scripts, communicating information via files, orchestrated into a pipeline via `condor` [55–57]. For O3, we did not have sufficient time to implement more than the most naive control logic: a fixed number of iterations to investigate the intrinsic variables, followed by an (optional) step to extract extrinsic variables, with some fixed $O(1)$ set of extrinsic samples associated to each intrinsic point. Convergence diagnostics were only used post-facto by the end user, to characterize run quality. With convergence tested only by humans in postprocessing, we needed to run every analysis for an extended period, usually employing many more iterations than necessary, to ensure almost all problems would be well-converged without human intervention. The most challenging and unanticipated problems, however, would require human intervention.

The deployment timeline also introduced additional unavoidable development requirements, occurring often simultaneously with ongoing efforts to refine our workflow during O3a. For the first half of O3 (O3a), RIFT had to be refactored into a python package (`pypi` and `conda` in particular), so it could be integrated into the LVK's standard software infrastructure. To support this refactoring specifically and code portability in general, we also had to create a continuous-integration test suite. Prior to the second half of O3 (O3b), RIFT had to be ported to python 3.

Despite its fragility and overkill, RIFT was extremely successful in O3. In O3a, RIFT was extensively used to analyze the GWTC-2 events using models with higher-order modes [28]. In O3b (GWTC-3), RIFT was also used to analyze events with a costly model including higher-order modes [9], and was also operated principally by external groups through large-scale automated software (`asimov`) [58].

Above and beyond the severe limitations introduced by using a fixed number of iterations, without convergence test integration within our control logic, our O3 experience suggested several additional elements of RIFT needed improvement. Figure 4 provides an example of an unpleasant but typical O3 RIFT inference of a low-mass binary, illustrating many of the problems described below. First and foremost, our Gaussian-process fitting implementation using fiducial coordinates became almost unusably slow when trained with many inputs, requiring as much as a day to generate for several challenging problems of astrophysical interest. In fact, in O3, we implemented several workarounds to prevent our fit from ever using too many training points. However, particularly for low-mass binaries with strong inter-parameter correlations in its posterior, our fits were also prone to misidentify suitable length scales [i.e., the diagonal elements of Q in Eq. (5)], leading to patchy and irregular posteriors when the fits were not informed by overwhelmingly large data volumes; see Figure 4 for examples. Since our technique required many iterations of fitting, often one to several weeks could be required to interpret the most interesting precessing binaries.

Second, our adaptive Monte Carlo integration algorithm did not effectively exploit extremely strong and

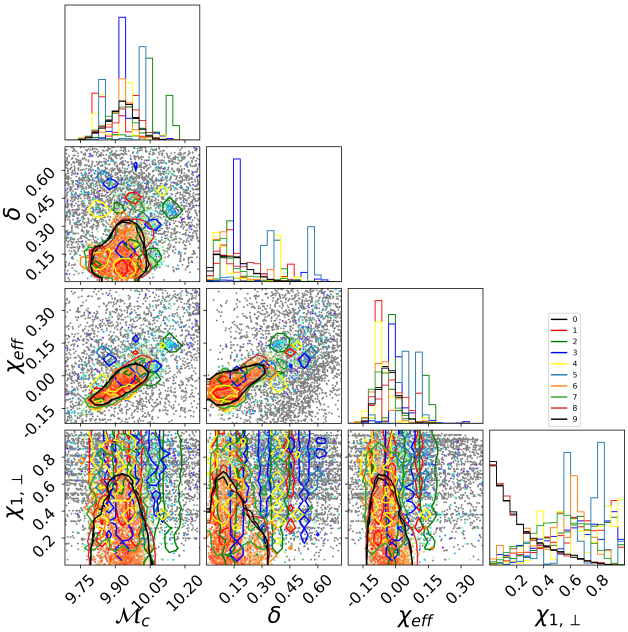

FIG. 4: **Illustration of O3 architecture limitations for unsupervised inference of low-mass binaries**: A corner plot with 90% credible interval quantiles for different RIFT iterations, indicated by different line colors. The color scale shows the likelihood range, over a dynamic range $\Delta \ln \mathcal{L} \leq 15$; gray points indicate likelihood evaluations below this range. This figure shows an unsupervised analysis of GW190707, evaluated using the old O3 convergence architecture (with SEOBNRv4PHM). The panels show marginal distributions in chirp mass $\mathcal{M}_c$, asymmetric mass ratio $\delta = (m_1 - m_2)/M$, inspiral effective spin $\chi_{\rm eff}$, and the magnitude of the transverse dimensionless spin of the primary $\chi_{1,\perp} = \sqrt{\chi_{1,x}^2 + \chi_{1,y}^2}$. Section III F enumerates the many algorithmic limitations of the O3 analysis highlighted by this kind of analysis.

well-understood correlations in the posterior distribution of chirping binaries. For BHNS binaries in particular, the natural error ellipsoids are extremely long and narrow; see, e.g., [59, 60]. For these extreme binaries, our intrinsic posterior Monte Carlo integration (performed in CIP) typically completed with a ratio $n_{\rm eff}/n$, which roughly measures the number of independent sample points per proposed Monte Carlo trial, often smaller than 10^{-8}. Even typical low-mass binaries had low values of $n_{\rm eff}/n$. Combined with the relatively long evaluation time of Gaussian processes, all low-mass binaries were uncomfortably difficult to investigate.

Third, our choice for how to explore precessing DOF was not well-adapted to investigate the low-mass binaries which nature provides, whose overall spins (and transverse spins) seem small. Instead, motivated by discovery potential, in O3 we used a volumetric prior (in a hypercube) for the component spins $\boldsymbol{\chi}_i$ to initially explore both aligned and transverse degrees of freedom. This prior was used in most of the initial iterations, when we adopted a fitting ansatz based only on the *aligned* degrees of freedom, under the assumption that precession effects were small, to populate presumed-subdominant precessing degrees of freedom. We used this prior for several iterations, including early iterations where the likelihood fit did not include and could not adapt to transverse degrees of freedom. However, as became apparent later in O3, none of the low mass events had significant support for nonzero transverse spin. By contrast, our prior frequently generated points with large transverse spins, which fit poorly (because these configurations would have many easily-observable precession cycles at low mass). At best, our choice of volumetric prior wasted time that could have been spent exploring the transverse degrees of freedom more efficiently. At worst, the volumetric prior points actively impeded convergence early on. In practice, we sometimes needed the final fully-precessing iterations even to get a plausible posterior at all for low-mass binaries such as NSBH [8].

Figure 4 provides an example showing how our inference only stabilized after adopting a uniform-spin-magnitude prior in the final iterations. In this analysis, the first several iterations incorrectly adopted a likelihood completely independent of transverse spin, despite frequently sampling large transverse spins which corresponded to substantial precession. This O3-era combination of extreme-spin prior and no-transverse likelihood model was extremely difficult to fit and sample, particularly in the old coordinate system which lacked awareness of the strong $\mathcal{M}_c, \eta, \chi_{\rm eff}$ correlations expected from leading-order post-Newtonian inspiral. The last few iterations (here labelled 6,7,8) however employ transverse spins in their fits and, combined with a more suitable spin prior, finally recover a smooth posterior. For this and similar posteriors, where only the last few iterations are well-behaved, manual additional investigation was required, extending the existing run to assess if RIFT had indeed converged.

Finally, given the many inefficiencies already limiting our performance and limited development time, we left many elements of RIFT in highly unoptimized forms. For example, the coordinate conversions within CIP used to implement generic chart transformations between fitting and sampling coordinates used a generic but slow data structure instead of fast vectorized once-and-for-all transformations. CIP workers were operated such that, if any one failed, all were rerun. This poor choice required considerably more processing and longer latency when very large numbers of workers were needed to handle BHNS binaries, for example. Some of our integration algorithms were insufficiently overflow-protected, causing errors when even modest-amplitude signals' likelihoods were evaluated directly (as opposed to only as a logarithm). Finally, users had few guarantees about the effective sample size of their output. For a handful of low-significance events in particular, the small and unpredictable sample size was intermittently a challenge in O3b, during which several postprocessing resampling stages were applied to RIFT's output via `asimov` to change the distance prior and add calibration marginalization.

IV. UPDATES

In this section, we describe several extensions to the way RIFT was used during O3, improving its likelihood approximation; integration; and workflow. Where appropriate, we also provide simple (PP) tests to validate specific modules. After describing these many possible additions, in the next Section VI we describe how we downselect between these options: by measuring RIFT perfomrance when interpreting two fiducial sources: a synthetic binary black hole and GW190412. Having downselected between the many available configurations, that section also provides targeted validation studies using synthetic binary black hole and binary neutron star sources.

A. Added coordinate systems (and priors)

All algorithms used within RIFT – interpolation, posterior generation, grid placement, dithering, convergence tests, et cetera – perform better in coordinate systems which are well-adapted to the likelihoods of real gravitational wave sources. For example, all our current and new unstructured interpolation algorithms inherit some implicit or explicit dependence on the coordinate system used to formulate them.

Rotated Inspiral-Phase (RIP) coordinates Particularly for low-mass systems, the neigborhood of near-peak likelihood is best characterized (and quite ellipsoidal) in well-chosen, instrument-dependent coordinates. While optimal local coordinates can always be derived by Fisher matrix methods, in general the orientation of this optimal ellipsoid relative to an underlying generic coordinate chart is highly source-dependent. However, using a fiducial contemporary ground-based network to eliminate (minimal) ambiguity about the appropriate detector network, recently Lee and collaborators [61] introduced a global coordinate system remapping $\mathcal{M}_c, q, \chi_{1,z}, \chi_{2,z}$ into a *global* coordinate system which is well-suited to model the likelihood for *arbitrary* sources.

The RIP coordinate system is motivated by the leading-order post-Newtonian expressions for the gravitational wave strain emitted by a nonprecessing binary in the $\ell, |m| = (2, 2)$ mode. Using standard techniques [62, 63], the frequency-domain gravitational wave phase $\tilde{h} = |h(f)| \exp(-i\Psi(f)))$, evaluated at some reference frequency can be expressed in terms of several post-

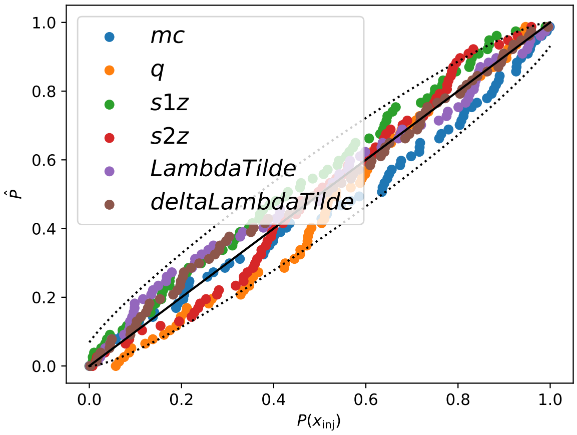

FIG. 5: **RIP PP plot**: Probability-probability plot to validate the rotated inspiral phase coordinates. The synthetic sources and parameter inferences are constructed with NRHybSur3dq8Tidal ($\ell = 5$) in gaussian noise with presumed known PSDs, for 3-detector networks, starting the signal at 30 Hz and using a 4096 Hz sampling rate. Detector-frame masses are drawn uniformly in the region bounded by $\mathcal{M}/M_\odot \in [1.2, 1.4]$ and $\eta \in [0.2, 1/4]$, and sources are drawn volumetrically between 90 and 240Mpc. Both NS dimensionless spins are drawn uniformly and volumetrically from [0,0.05]. For computational efficiency, all sources in this test have a fixed and presumed known sky location.

Newtonian parameters

$$\psi_1 = \frac{3}{128}(\pi \mathcal{M}_c f_{\rm ref})^{-5/3} \tag{9a}$$

$$\psi_2 = \frac{55}{384}(\eta + \frac{743}{924})\eta^{-2/5}(\pi \mathcal{M}_c f_{\rm ref})^{-1} \tag{9b}$$

$$\psi_3 = \frac{3}{32}(\beta - 4\pi)\eta^{-3/5}(\pi \mathcal{M}_c f_{\rm ref})^{-2/3} \tag{9c}$$

where for convenience we adopt $G = c = 1$ units in our expressions, where the PN parameter β is defined by [64]

$$\beta = \frac{1}{12}\sum_{i=1}^{2}\left[113(m_i/M)^2 + 75\eta\right]\chi_{i,z} \tag{10}$$

and where $\chi_{i,z} \equiv \hat{L}\cdot\mathbf{S}_i/m_i^2$ is the projection of the dimensionless spin along the instantaneous angular momentum axis. The rotated inspiral-phase coordinates X_i follow from a coordinate transformation $\boldsymbol{X} = U\boldsymbol{\psi}$ where U an instrument-dependent 3×3 matrix derived from the Fisher matrix expressed in terms of these coordinates. To be concrete, we follow Lee et al and adopt a single fiducial choice for U:

$$U = \begin{bmatrix} 0.97437198 & 0.20868103 & 0.08397302 \\ -0.22132704 & 0.82273827 & 0.52356096 \\ 0.04016942 & -0.52872863 & 0.84783987 \end{bmatrix} \tag{11}$$

Unless otherwise noted, we employ a nominal $f_{\rm ref} = 200$Hz to define this transformation for all masses. Despite the relatively high reference frequency, this transformation remains well-behaved even for very massive black hole binaries, implying the coordinates can be employed throughout the observed (detector-frame) space of compact binary parameters.

Following Lee et al (their Section III.C), we define the RIP coordinate system $\mu_1, \mu_2, \delta, \chi_{2,z}$. Within the framework described so far, this coordinate system can be employed within the fitting and posterior generation stage (CIP) in two ways. On the one hand, we can use RIP as a coordinate system well-adapted to *fitting* the likelihood. In this approach, after re-expressing our training points $\boldsymbol{\lambda}$ in the RIP coordinate system, our unstructured interpolation code produces an approximation $\hat{\mathcal{L}}(\boldsymbol{\lambda})$ in terms of those coordinates. Aside from this modest change, CIP can be used the same way, for example using any coordinate system and Monte Carlo integration technique to perform posterior generation. Figure 5 shows an end-to-end validation study of RIFT when RIP coordinates are employed within CIP, using a suite of many synthetic aligned-spin sources drawn with random intrinsic and extrinsic parameters.

On the other hand, we can also use RIP as a coordinate system well-adapted to exploring and *sampling* the likelihood. To do so we must define an effective sampling prior $p_{s,eff}(\boldsymbol{\lambda})$ for the RIP coordinates $\boldsymbol{\lambda}$. Though nominally simple, the complicated nonseperable boundaries associated with this sampling prior have so far complicated our ability to employ this seemingly simple prior within our existing frameworks based on purely seperable physical priors. We defer use of these accelerated coordinates to future work on very-low-latency analysis.

Rotated detector-network-frame sky coordinates To more efficiently sample the sky, we provide users the option to use a coordinate system for the sky where the nominal north pole corresponds to a vector connecting two of the interferometers. As discussed in many previous implementations of this transformation (see, e.g., [18, 65]), in this coordinate system the posterior distribution will be aligned with lines of constant nominal declination, enabling more efficient adaptive sampling.

Pseudo-cylindrical coordinates for spheres: Posterior generation of precessing spins can be computationally costly in the most straightforward spin coordinate system; spherical polar coordinates for each spin. For context, for nonprecessing binaries we can adopt carteisan aligned spins $\chi_{i,z}$, and use an adaptive integration method which captures correlations between $\chi_{1,z}, \chi_{2,z}$. By contrast, in spherical polar coordinates, these simple and strong correlations are distributed among many more parameters. Similarly, for most massive BH binaries observed so far, the *transverse* spin components $\chi_{i,\perp}$ are extremely weakly constrained. However, in spherical polar coordinates, the transverse and aligned spin components are strongly mixed.

To improve the prospects for our adaptive integrators to better reflect the correlations among spin parameters, we introduce a coordinate transformation mapping a sphere to a cylinder: $(R, z) = (\bar{R}\sqrt{1-\bar{z}^2}, \bar{z})$ for $\bar{R}, \bar{z} \in [0,1]$. In these coordinates, the overall spherical volume element $d\phi \wedge RdR \wedge dz$ can be recovered with the seperable sampling priors $p_V(\bar{R}) = 2\bar{R}$ and $p_V(\bar{z}) = 3(1-\bar{z}^2)/4$. These coordinates enable efficient sampling of the unit sphere with a volumetric prior $p_V(\bar{R})p_V(z)/2\pi$ using ccoordinates well-adapted to the typical constraints afforded by GW observations.

While sufficient for volumetric sampling, however, the discussion in Appendix B suggests more singular sampling priors will enable better sampling of the fiducial uniform-spin-magnitude spin prior (i.e., $d^3\chi/3|\chi|^2$). In this common scenario, we adopt a more singular pseudo-radial sampling prior $p_s(\bar{R}) = \bar{R}^{-3/4}/4$, or equivalently a uniformly-sampled radial coordinate $\bar{u} = \bar{R}^{1/4}$. Numerical experiments similar to those in Appendix B demonstrate improved scaling relative to naively reweighting volumetric samples. To be concrete, these modified pseudo-cylindrical coordinates represent a dimensionless spin vector $\boldsymbol{\chi}$ as

$$\boldsymbol{\chi} = \hat{\mathbf{z}}\chi_z + \bar{\chi}_u^4\sqrt{1-\chi_z^2}[\cos\phi\hat{\mathbf{x}} + \sin\phi\hat{\mathbf{y}}] \qquad (12)$$

In terms of these coordinates $\bar{\chi}_u, , \phi, \chi_z$, a volumetric prior follows from p_V by change of coordinate and jacobian: $p_{UV} = 15\bar{\chi}_u^7(1-\chi_z^2)/8\pi$. The corresponding uniform spin magnitude prior is $1/3|\boldsymbol{\chi}|^2$ times this function.

Generalized precession coordinate: The originally proposed precession parameter $\chi_p \equiv \max\left(\chi_1\sin\theta_1, \tilde{\Omega}\chi_2\sin\theta_2\right)$ characterizes the largest dynamical spin in the binary, but fails to account for the effects of dual-misalignment. This deficiency, which manifests in systems with equal mass ratio and large transverse spins, is resolved by the parameter $\langle\chi_p\rangle$, which averages over all spin angles on the precession timescale [66]. The initial implementation of $\langle\chi_p\rangle$ is detailed in [67], where events from O3b were analyzed as a post-processing step using existing samples to compute posteriors for both χ_p and $\langle\chi_p\rangle$. Additionally this parameter is a constant of motion at 2PN order on the spin-precession timescale (and nearly conserved on the radiation-reaction timescale), making it a good candidate for a fitting coordinate when computing posteriors for analyses that assume precession. By computing an approximate $\mathcal{L}(\lambda)$ from the archived marginal likelihood calculations in the $\langle\chi_p\rangle$ coordinate, we then assign a uniform prior in the domain $0 \leq \langle\chi_p\rangle \leq 2$ to compute the posterior distribution. Note that although the $1 < \langle\chi_p\rangle \leq 2$ domain is exclusive to binaries with two misaligned spins, there are spin morpholigies in the $0 \leq \langle\chi_p\rangle < 1$ domain for which $\langle\chi_p\rangle$ differs strongly from χ_p. Allowing the prior to cover this space leaves the analysis agnostic to the fully precessing behavior. This functionality has now been implemented in RIFT as part of the CIP subroutine, and the efficacy of this parameter is currently being tested via injection study, the results of which will be discussed in a forthcoming publication; see also [68].

B. More robust and efficient likelihood approximations

In this section, we summarize several different techniques to approximate the marginal likelihood.

1. Random forests

First used in RIFT for interpolation in [69], random forests interpolate generic functions by constructing a family of many random decision trees, with piecewise constant approximations of the form $y(x) = \sum_k w_k \mathbf{I}_k(x)$ where $\mathbf{I}_k$ is unity inside the selected volume and zero elsewhere [70–72]. Customarily, each choice in the decision tree decides between one (randomly selected) coordinate in the variable x; as a result, each decision tree selects a sequence of rectangular cartesian regions. Random forests construct an ensemble of trees, each with randomly chosen decision points. We employ the Extra-Trees algorithm [72], as implemented in scikit-learn [46]. In the limit of extremely deep and random trees, this algorithm converges to a piecewise linear and continuous approximation [72].

Because random forests' basis functions $\mathbf{I}_k$ are step functions aligned with the coordinate axes of x, random forests can be sensitive to the choice of coordinates, particularly when the posterior exhibits strong correlations between multiple parameters. Our investigations suggest RF fits robustly perform well in all coordinates for sources with broad, uncorrelated posteriors (e.g., massive binary black holes). By contrast, for high-mass-ratio sources in particular RF fits should only be used with specialized coordinate systems like the RIP coordinates above. Even more so than gaussian processes, random forests do not extrapolate well outside of their domain, and as a result posteriors which extend to sharp prior boundaries can introduce undersampling or even pathologicar behavior. As a concrete example, RF-based posterior generation for binary neutron star observations with nonprecessing binaries with uniform priors on $\chi_{i,z} \in [-1, 1]$ can behave extremely poorly; uniform spin magnitude sampling for nonprecessing BNS with RF fits should always employ a tightly restricted spin prior. As a second example, RF-based posteriors for the transverse spin require extensive sampling near $\chi_{i,\perp} \simeq 0$ to explore this region well, hence the pseudo-cylindrical coordinates of Eq. (12). As a third example, RF-based posterior generation near the equal-mass line can be prone to under-predicting the region near $q \simeq 1$, though suitable mass ratio sampling coordinates could mitigate this effect.

2. Sparse gaussian processes

RIFT initially adopted conventional Gaussian Process (GP) regression to estimate the marginal likelihood versus intrinsic parameters, with a full rank (squared exponential) kernel $k(x, x')$ as provided by SCIKIT-LEARN. Straightforward GP regression techniques are costly since they involve matrix inverses, with nominal cost scaling as n^3 for a full-rank matrix [73]. This scaling severely limited our ability to increase model dimension or to use more training data. Sparse kernels or approximations have been widely explored in the GP literature [74–77]. To perform GP regression more efficiently, we have implimented a piecewise polynomial covariance function with compact support [73]. These basis functions are guarenteed to be positive definite, and the covariance between points becomes zero as their distance increases, and are given as $K_{ppD,q}(r)$.

$$K_{ppD,0}(r) = (1-r)_+^j \tag{13}$$

$$K_{ppD,1}(r) = (1-r)_+^{j+1}((j+1)r+1) \tag{14}$$

$$K_{ppD,2}(r) = \frac{(1-r)_+^{j+2}((j^2+4j+3)r^2+(3j+6)r+3)}{3} \tag{15}$$

$$\begin{aligned} K_{ppD,3}(r) = (1-r)_+^{j+3}\,&\big((j^3+9j^2+23j+15)r^3 \\ &+(6j^2+36j+45)r^2 \\ &+(15j+45)r+15)\big) \div 15 \end{aligned} \tag{16}$$

Where $j = \lfloor \frac{D}{2} \rfloor + q + 1$, D is the dimensionality of your data set. q is chosen such that the sample function is $2q$ times differentiable. We have chosen $q = 1$, and added a whitenoise kernel as well. We have seen that the sample time for this function scales only with n for high n.

3. Quadratic and gaussian estimates for placement

During initial exploration the posterior for tightly constrained events, particularly for precessing binaries, relatively few points $\boldsymbol{\lambda}_\alpha$ will have high likelihood $\mathcal{L}_\alpha$. With limited training data in these iterations, our most flexible and efficient interpolation methods in practice can spuriously identify overly-complicated likelihood estimates, with complex isocontours and mutiple extrema. For several future applications, we introduce two simple likelihood approximations, both using some pre-determined threshold $\mathcal{L}_{cut}$ to identify the subset of training data $\boldsymbol{\lambda}_\alpha$ with $\mathcal{L}_\alpha > \mathcal{L}_{cut}$. In the mean-covariance approximation, we compute the sample mean $\bar{\boldsymbol{\lambda}}$ and sample covariance $\bar{\boldsymbol{\Sigma}}$, then adopt the ansatz

$$\ln \hat{\mathcal{L}}_{cov} = \ln \mathcal{L}_{max} - \frac{1}{2}(\lambda - \bar{\lambda})_p(\lambda - \bar{\lambda})_q \bar{\Sigma}^{-1}_{pq} \tag{17}$$

In the quadratic approximation, by contrast, we perform a least-squares quadratic form fit to $\ln \mathcal{L}$ versus $\boldsymbol{\lambda}$, then use the expression

$$\ln \hat{\mathcal{L}}_{cov} = \ln \mathcal{L}_{maxfit} - \frac{1}{2}(\lambda - \lambda_*)_p(\lambda - \lambda_*)_q \Gamma_{pq} \tag{18}$$

where $\ln \mathcal{L}_{\rm maxfit}, \lambda_*, \gamma$ are all identified by the quadratic fit. We provide these simple approximations for testing, for potential use in ultra-low-latency analysis, and to better extract simple approximate results (e.g., Gaussian approximations) from detailed analyses. These two approximations are not included in the operational recommendations for long-latency offline inference presented later in this work.

C. Improved integration

RIFT uses Monte Carlo integration in both stages of its iterative process, for posterior generation (CIP) and likelihood marginalization (ILE). Beacuse of the dynamic range, sometimes sharp features, and strong correlations present in the likelihood integrand, RIFT uses custom implementations of adaptive Monte Carlo integration. In this section, we primarily describe alternative Monte Carlo integration implementations which meet some of our design goals. Appendix B describes how RIFT and other codes characterize sampling size: RIFT customarily uses $n_{\rm eff}$ while $n_{\rm ESS}$ is used by many other inference codes. Figure 6 illustrates how the two new Monte Carlo integration methods compare to our previous approach, for the purposes of estimating posterior distributions via weighted samples. The code used to generate this figure (and thus test the integrators at a variety of target resolutions) is disseminated with the source and run as part of our continuous integration suite. Also disseminated with the RIFT source is the code used to generate Figure 7, the inferred sample distribution implied by the Rosenbrock likelihood [78, 79].

RIFT's low-level likelihood evaluation can use direct quadrature over some extrinsic degrees of freedom. In this work, we specifically describe how RIFT can now use a fast numerical quadrature over distance.

1. Adaptive sampling with gaussian mixture models

Because many of our integrands have strongly correlated dimensions, seperable sampling priors are often very inefficient. To identify correlations, we provide an alternative adaptive sampler, such that p_s is built from Gaussian mixture models.

In the simplest and default form, we continue to assume a seperable sampling prior. For adaptive dimensions, we adopt one-dimensional sampling distributions

$$p_{s,k}(\theta_k) = \sum_\alpha w_\alpha p_n(\theta_k | \mu_\alpha, \sigma_\alpha) \tag{19}$$

where p_n is a standard normal distribution with mean μ_α and standard deviation σ_α; where w_α are weights associated with each gaussian component; and where for simplicity we fix the number of components a priori. Each adaptive iteration, we use the expectation-maximization algorithm to re-assess the weights and covariance [80–82], organizing the calculation to enable fast iterative updates; see the Appendix for details. For example, each iteration the integral result I and variance σ_I^2 are updated with a running average using the previous values I, σ_I^2 and the values over the current subsample I_{new}, σ_{Inew}^2, according to

$$I' = \frac{n_s I + I_{new}}{n_s + 1} \tag{20}$$

$$\sigma_{I'}^2 = \frac{n_s \sigma_I^2 + \sigma^2}{n_s + 1}. \tag{21}$$

where n_s is the number of previous iterations. To gracefully handle finite boundaries, we use truncated normal distributions $p_{n,t}$ in place of normal distributions p_n in our mixture model.

More commonly, we employ correlated sampling in subsets of dimensions: p_s is no longer seperable. The same algorithm applies. To handle finite boundaries, we use fast rejection sampling to identify valid configurations; see the Appendix. Our implementation allows the user to specify at runtime which (if any) dimensions will use correlated sampling.

To illustrate how this new sampler compares to the original implementation in controlled circumstances, we employ both to produce independent samples from an underlying correlated three-dimensional gaussian likelihood function. Figure 6 shows the true and estimated one-dimensional cumulative distribution functions, after a fixed number of likelihood evaluations. As expected, the new GMM-based integrator recovers the true distribution more accurately at fixed cost. More extensive tests with a wider range of sample sizes and reference distributions corroborates this anecdotal example.

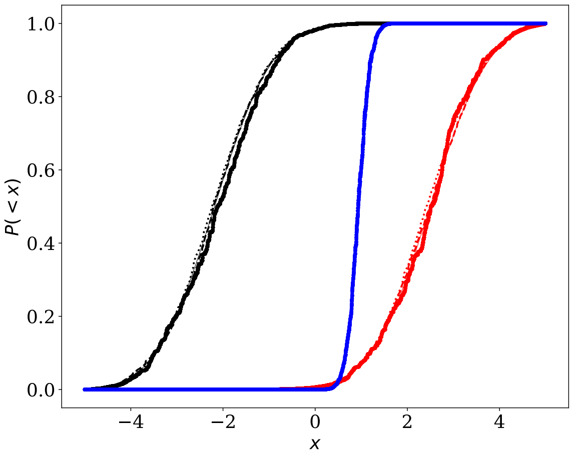

FIG. 6: **Integrating random multidimensional gaussian**: Cumulative distribution functions inferred for a random uncorrelated three-dimensional gaussian likelihood, different adaptive weighted Monte Carlo integration techniques. The horizontal axis indicates the dimension x_i and the vertical axis our estimate of $P(< x_i)$. Colors indicate the dimension being rendered (e.g., x_1, x_2 or x_3), line styles indicate the method used to generate the CDF. Solid lines show the true CDF; dashed lines show a CDF obtained with the original MCMC integrator; thin dotted lines show the corresponding estimate from the gaussian mixture model (GMM) integrator; and heavy dots show the results with the adaptive cartesian integrator. All three estimates are derived from the same number of function evaluations (2×10^4) and produce comparable $n_{\rm eff} \simeq O(50)$; both are chosen to be small enough (and hence our resolution poor enough) so that the reader can differentiate between the different curves in the figure.

2. *GPU-accelerated Monte Carlo integration*

The ILE likelihood is dramatically more efficient when implemented on GPUs. The previous adaptive integrator, however, performed all random number generation with a CPU, then transferred large numbers of random samples to the likelihood evaluator on the GPU. The overhead associated with the Monte Carlo integrator can limit ILE's performance. We therefore re-implemented a simplified version of the previous Monte Carlo integration algorithm, using CUPY/NUMPY to allow the same source code to drive both CPU-only and GPU-enhanced mode. The end-user can request this integration algorithm in both CIP (in CPU-only mode) and ILE.

As in the initial implementation, we assume a seperable sampling prior $p_s(\boldsymbol{\theta}) = \prod_k p_{s,k}(\theta_k)$. For dimensions that do not benefit from adaptive refinement, we use fixed priors. For adaptive dimensions, $p_{s,k}(\theta)$ is revised based on the recent past history of $N_{adapt} = n_{adapt} * n_{chunk}$ samples. The adapted sampling distribution after refinement is a histogram

$$p_{s,k}(x) = \frac{1}{\Delta\Theta_k N_{adapt}} \sum_\alpha n_\alpha \mathcal{S}(x|x_\alpha, \Delta X/N_{adapt}) \tag{22}$$

where n_α is the number of samples in the past history with θ_k between θ_α and $\theta_\alpha + \Delta\Theta/N$, so $\sum_\alpha n_\alpha = N_{adapt}$; and where $S(x|x_*, \Delta x)$ is a unit step function equal to 1 between x_* and $x_* + \Delta x$ and zero elsewhere. To minimize fine-tuning and the need for costly conditional statements, following the original implementation we we employ a fixed number $n_{bins} = 100$ bins in adaptive dimensions.

When combined with RIFT's GPU-native likelihood function, all elements of the Monte Carlo integration can be performed on the GPU board, with minimal data transfer as needed to orchestrate the integration. As a result, this fully-GPU marginal likelihood evaluates very quickly, even for models involving many higher-order

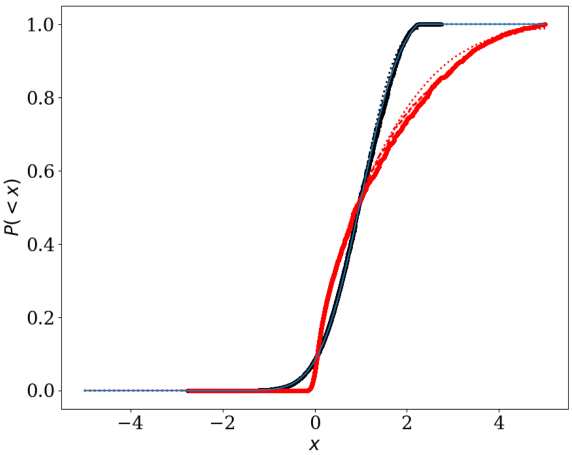

FIG. 7: **Integrating the rosenbrock likelihood**: One-dimensional marginal distributions for the two-dimensional rosenbrock likelihood inferred using our three standard integrators with a fixed number 10^6 random adaptive likelihood evaluations using 10^4 samples per adaptive batch. The solid line shows an independent solution derived from one-dimensional quadrature [79]. All three integrators predict an integration error and recover the quadrature-based evidence ($\simeq -5.804$) to within $\simeq 10^{-2}$ for this fixed sample size. For these settings we find $n_{\rm ESS} \simeq 5800, 1400, 5800$ ($n_{\rm eff} \simeq 3000, 130, 300$) for the default, adaptive cartesian (AC), and gaussian mixture model (GMM) integrations respectively. The one-dimensional Jensen-Shannon (JS) divergence between inferred marginal distributions for each parameter is comparable to or smaller than a fiducial threshold [83] of $10/\max[n_{\rm ESS,A}, n_{\rm ESS,B}]$ for the integrators A,B respectively (e.g., for the default and AC integrators it is $< 10^{-3}$, well below this threshold; for any other integrator and the GMM integrator, it is less than 5×10^{-2}, or modestly above the target). Note that since RIFT convergence is assessed using $n_{\rm eff}$, the GMM integration would be run many times longer in a normal RIFT run.

modes. All of the essential coordinate transformations described previously which accelerate ILE are compatible with this implementation, including distance marginalization and rotated sky coordinates. Figure 8 shows an end-to-end validation study of RIFT when adaptive cartesian integration is employed within ILE, using a suite of many synthetic zero-spin sources drawn with random intrinsic and extrinsic parameters.

3. *Distance marginalization*

Following previous work [84], Morisaki developed a concrete technique to directly marginalize over distance[85]. An implementation of this technique by Morisaki and Wysocki is now available within ILE. Figure 9 shows a large-scale end-to-end test of this code, to demonstrate it preserves the statistical purity of our recovered intrinsic parameter distributions. Directly marginalizing in distance reduces the computational overhead of the Monte Carlo integration step, allowing notably faster performance at fixed target accuracy. Figure 9 shows an end-to-end validation study of RIFT when distance marginalization is employed within ILE, using a suite of many synthetic zero-spin sources drawn with random intrinsic and extrinsic parameters.

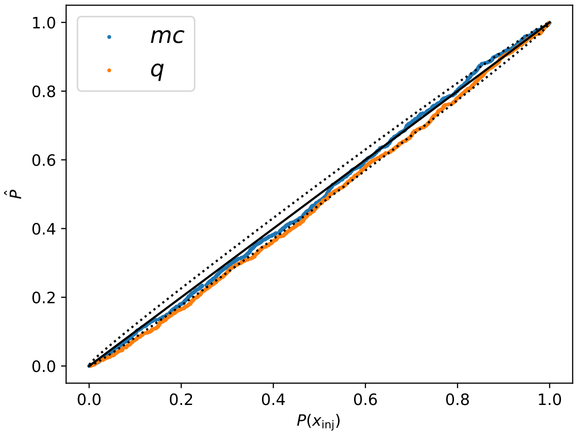

FIG. 8: **End-to-end test with GPU-accelerated Monte Carlo integration** Similar to Figure 3, a probability-probability plot constructed with synthetic zero-spin injections. In this figure, the underlying calculations used marginalized likelihood calculations evaluated using the GPU-accelerated Monte Carlo integration code.

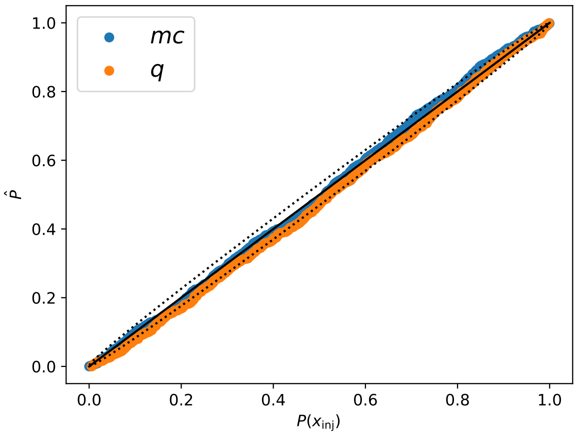

FIG. 9: **End-to-end test with distance marginalization** Similar to Figure 3, a probability-probability plot constructed with synthetic zero-spin injections. In this figure, the underlying calculations used marginalized likelihood calculations evaluated using the distance marginalization code.

4. Gaussian resampling of gaussian likelihoods

When suitable, a gaussian likelihood approximation [Eq. (18)] allows us to refactor our Monte Carlo integration technique: rather than draw samples x_k from a *sampling prior* p_s and computing the expectation of $\mathcal{L}(x)p(x)/p_s(x)$, we instead draw samples from the *normal likelihood* [modulo boundary truncation effects] and compute the expectation of $\mathcal{L}_{max}p(x)/p_g(0)$ where p_g is the appropriate truncated normal distrbution evaluated at its peak. The posteriors deduced with a Gaussian likelihood can be surprisingly close to the full answer, even allowing for large model dimensions [86]. This reweighting-based technique can also be performed extremely quickly, with the corresponding calculations generally limited by infrastructure (e.g., starting up an interpreter and loading libraries; file input and output). A subsequent companion study will outline the reliability and performance of various ultra-low-latency strategies, including the reliability and efficiency of this approach.

D. Updated convergence architecture

As described in Section III C, RIFT adopts different settings in different iterations, to leverage our experience with hierarchically exploring compact binary parameter space. Particularly for unsupervised operation, RIFT's initial grids often only explore a three-dimensional subset of nonprecessing binary parameters. We therefore adopt a sequence of settings for each iteration's use of the CIP code, which both performs fits and generates the posterior via weighted Monte Carlo integration. In O3, these settings were chosen to gradually increase the sampling and fitting dimensionality, with the hope of identifying and characterizing many strongly precessing BH-BH binaries. However, as described in Section III F, the overly conservative and inflexible choices adopted for unsupervised operation in O3 were extremely inefficient for low-mass or highly-asymmetric binaries.

In this work, we introduce a new architecture which (conbined with the previously-reported coordinates and integrators) efficiently and reliably recovers the properties of low-mass and asymmetric binaries. Specifically, we first perform two iterations using $\mathcal{M}_c, \delta, \chi_{\rm eff}$ as fitting parameters, sampling uniformly in mass; uniformly in $\chi_{i,z}$; and with a modified prior for $\chi_{i,\perp}$. These iterations capture the dominant aligned-spin degrees of freedom for most BH binaries, while populating the transverse spins. We next perform two iterations using $\mathcal{M}_c, \delta, \chi_{\rm eff}, \chi_p$ as fitting parameters, with the same priors as before. By adding transverse spin dependence, we capture the (dominant) impact of transverse spin, particularly important for low masses or asymmetric binaries when transverse spins are so frequently constrained to be nearly zero. Finally, we iterate to convergence, using $\mathcal{M}_c, \delta, \chi_{\rm eff}, \chi_-, \chi_{i,\perp}$ as fitting parameters, and uniform-in-spin-magnitude sampling parameters. Extrinsic parameters are extracted from the final converged iteration. For comparison, we will also describe results derived using an otherwise similar architecture, but adopting the pseudo-cylindrical coordinates for spin provided by Eq. (12). In these alternative analyses, the first four iterations adopt a uniform prior on $\bar{\chi}_{i,u}$, which densely samples the region with $\boldsymbol{\chi}_{i,\perp} \simeq 0$.

This new approach is enabled by adding a notable missing feature for RIFT: iteration until convergence. RIFT users can now request a specific CIP configuration be used repeatedly, in a recursively-generated subworkflow, until the posterior converges according to the user-specified convergence diagnostic.

Figure 10 illustrates unsupervised operation for GW190425, interpreted with IMRPhenomPv2. The left panel shows our default new architecture, where the initial prior over transverse spins is well-adapted to discovering and characterizing large transverse spins, while the right panel employs a more concentrated initial spin prior (i.e., the uniform-in-$\bar{\chi}_{i,u}$ prior). Both demonstrations perform dramatically better than the low-mass analysis shown in Figure 4, with a steady increase in understanding as our exploration and likelihood model adapts as necessary to model the posterior given the adopted priors. However, these two analyses' small differences highlight the importance of adopting initial priors well-suited to the event and final objective.

In the left panel of Figure 10, the RIP and spin coordinate systems enable the first two iterations to rapidlly identify pertinent aligned degrees of freedom. The next two iterations then correctly contrain the transverse spin, while refining an estimate for masses and $\chi_{i,z}$ that is appropriate for our initial prior. However, when we adopt the final spin prior and iterate to convergence, the code (correctly) increasingly identifies an extended region with higher spin and mass ratio, smoothly connected to the main posterior but now identified as pertinent given the new prior. Iterations cease when the code converges.

In the right panel of Figure 10, we repeat our analysis using the alternative configuration above, differing only in the spin coordinates used throughout the analysis and in the transverse spin prior adopted for the first few iterations. As exemplified by the analysis from the left panel, nature so far has provided binary black holes consistent with zero transverse spin. For low mass binaries, the transverse spins are well constrained to be near zero. As a result, the analysis shown on the right converges much more quickly to our final result.

The contrasting performance of the two analyses shown in Figure 10 highlights the dangers of simply reweighting an existing result to a new prior; see also Appendix B for further discussion. Figure 11 provides another way to quantify the impact of our initial prior choices on convergence, using the multiple Monte Carlo estimates of the evidence $Z = \int \mathcal{L}_{\rm marg} d\lambda$ and their error reported by

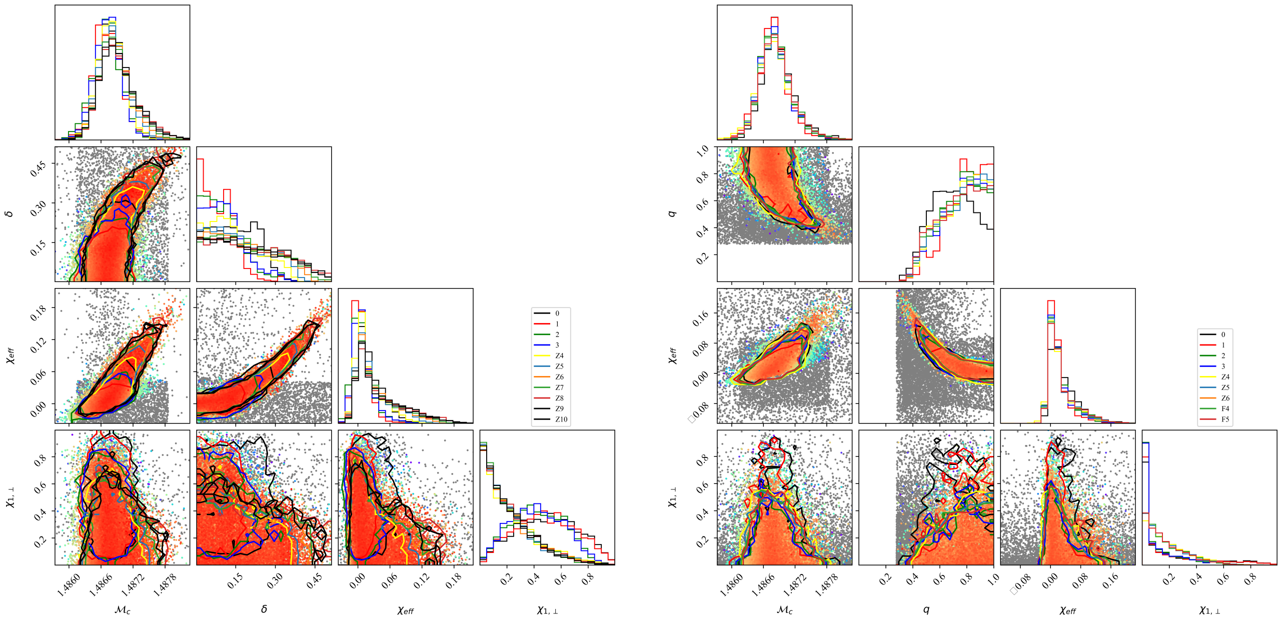

FIG. 10: **GW190425, inferred with different choices for initial priors**: Both panels show corner plots with 90% credible interval quantiles inference for GW190425 using IMRPhenomPv2 using binary black hole priors (i.e., $|\chi_i|$ uniform in magnitude between 0 and 1) for different RIFT iterations, indicated by different line colors; the color scale shows the likelihood range, over a dynamic range $\Delta \ln \mathcal{L} \leq 15$. The two panels adopt the same architecture, differing only in the coordinates used for spin sampling [Eq. (12)] and in the prior adopted for transverse spins in the first four iterations.

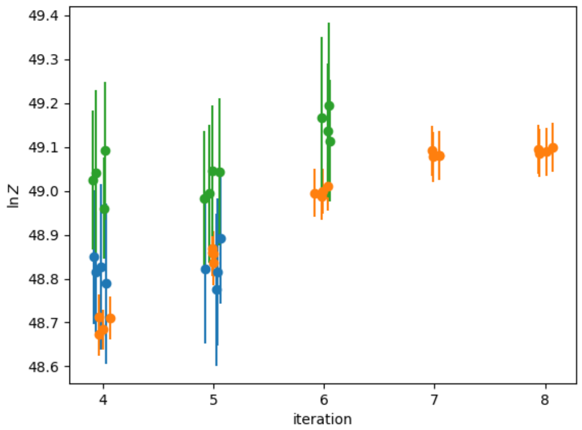

FIG. 11: **Intermediate evidence estimate versus iteration**: For the RIFT analyses of 190425 shown in Figure 10, this figure shows our *internal* evidence estimates $Z = \int \mathcal{L}_{\rm marg} p(\lambda) d\lambda$ produced by each CIP worker versus iteration number, while iterating towards convergence. Points show the Monte Carlo estimate, and error bars are estimated 90% *statistical* confidence intervals (i.e., 1.64 times the Monte Carlo integration error estimate). We have slightly jittered the x coordinates in this figure to better differentiate overlapping error bars. Orange and blue indicate the analyses performed in the left and right panels of Figure 10 respectively.

each CIP worker.[1] In the orange points, showing the analysis using our default transverse prior for the first initial iterations, we see the evidence estimate systematically evolves upward as the posterior approaches our final converged result. As expected given Figure 10, the statistical errors estimated from Monte Carlo integration substantially understate the systematic error in the evidence. By contrast, the green traces immediately identify the final overall evidence, again as expected given Figure 10.

E. Automated information transfer between analyses

Due to its iterative nature and reliance on archived likelihood evaluations $\{\lambda_k, \mathcal{L}_k\}$, RIFT has unique capabilities to use information from previous or even concurrent analyses with different models and configurations [17]. Though these capabilities are particularly powerful when adopting the same waveform model and data analysis settings (i.e., the likelihoods themselves can be

[1] This internal-use evidence during these intermediate iterations has substantially larger statistical errors than final evidence, which is evaluated using much longer iterations during the final iteration.

re-used), they can also be very powerful tools even between waveforms. As a concrete example, RIFT analyses performed using multiple waveform models can efficiently marginalize over waveform uncertainty [54]. As another example, rapid analyses with simpler waveform physics (e.g., no precession) or faster waveform models can feed directly into an ongoing RIFT analysis, by supplying additional target points for likelihood evaluation.

The RIFT workflow has always had natural stages where external information can be conveniently inserted (e.g., adding likelihood evaluations, or proposing new points for next-step evaluation). Where previously we had ad hoc procedures to manually edit or supply the necessary files, with the latest generation of RIFT we introduce the `fetch` process, designed to retrieve candidate points (or likelihoods) from any external run. [In fact, we even retrieve information from the recursive iterate-to-convergence stage via this same framework.]

We foresee three natural use cases for the `fetch` framework. First, this framework enables a particularly efficient run hierarchy for modest-latency analysis over the first few minutes, hours, and days. Fast analyses (with RIFT and other codes) using simplified physics (e.g., without precession) seed longer-timescale analysis with more physics. Within and between stages, RIFT supplies an approximate posterior distribution. Second, building on this approach for offline followup, this framework enables efficient analysis with multiple approximations, where these approximations' analyses may have different timescales owing to their computational cost. The two analyses can inform each other, if simultaneous, or the fast analysis can inform the slower one if computational costs are significantly different (e.g., due to the incorporation of many higher-order modes). Finally, by fetching from previous work, RIFT can most efficiently complete final production-quality analyses, building on previous experience.

F. Adaptive mesh refinement

In conventional RIFT, the fitting- and posterior-generation stage is the most serial and time-consuming, particularly for low-mass sources. Rose et al [87] introduced an adaptive mesh refinement (AMR) for gravitational wave parameter inference, a strategy which very efficiently finds and explores the (intrinsic, marginal) likelihood over modest dimensionality (i.e. $d \leq 4$, corresponding to the nonprecessing intrinsic degrees of freedom). Below, we describe one way that RIFT can use AMR without employing additional external information supplied by searches or precomputed overlap tables.

Our default AMR approach is initiated with a coordinate hypercube in one of a few blessed groups of parameters, such as $\mathcal{M}_c, \delta, \chi_{i,z}$. The AMR engine then successively retrieves information about likelihoods on grid nodes; assesses grid cells which require refinement; and identifies new node centers for subsequent evaluation. At each specific grid level, cells are identified as needing refinement based on a threshold p. A specific cell n out of N cells is selected if (after sorting all cell likelihoods $\mathcal{L}_k$ and forming the cumulative sum $\mathcal{S}(n) = \sum_{k=1}^{n} \mathcal{L}_k$) the sum satisfies $S(n)/S(N) > 1 - p$. Roughly speaking, this threshold associates each cell (at any refinement level) with equal AMR probability mass, and performs refinement of the most significant fraction p of the nominal AMR probability mass. Each successive grid level fully refines all areas requested for refinement. No prior coordinate-dependent or refinement-level information is used to guide the refinement choices.

To assess convergence of our refinement, we have two natural diagnostics: the integrated likelihood and the distribution of likelihoods. For the first, at each level we can estimate the unweighted evidence $Z = \int \mathcal{L} dx$ with successive Riemann integral estimates $Z_\ell = \sum_k \mathcal{L}_{l,k} \prod_\alpha \Delta x_\alpha / \ell$ where $\mathcal{L}_{\ell,k}$ are the likelihood values in level ℓ with a top-level grid spacing of Δx_α for each dimension α. For the second, we can use the distribution of $2 \ln \mathcal{L}_{\ell,k}$ at each ℓ. When AMR is nearly converged, the inter-evaluation seperations will be small, and the distribution should be roughly consistent with a χ^2 distribution with roughly d degrees of freedom, depending on the number of well-constrained parameters being simultaneously explored.

Lacking the need to interpolate the likelihood or sample a posterior, the AMR engine operates within seconds. Operationally, the AMR engine behaves like a drop-in replacement for CIP: the code can effectively iterate to convergence using just AMR. Thus, the AMR engine provides an extremely rapid way to explore the likelihood. As desired, we can also run conventional CIP in parallel, during postprocessing, or even as part of a parallel non-AMR analysis with more degrees of freedom to identify a fully-interpolated posterior distribution. This latter approach in particular offers an extremely powerful technique to bootstrap inference for the most challenging low-mass, high-mass ratio sources.

G. Single-event EOS inference with pretabulated equation of state

RIFT already has at least two frameworks to constrain the nuclear equation of state (EOS). On the one hand, given any tabulated EOS, RIFT can efficiently compute an evidence for that EOS, based on integrating the interpolated marginal likelihood $\mathcal{L}(\lambda)$ while accounting for the unique relationship between NS mass m and tidal deformability Λ that this EOS allows [88]. On the other hand, RIFT can also similarly constrain a parameterized equation of state, constructing a posterior for its hyperparameters [17]. However, single-event inferences have two significant limitations. First, near-future measurements must simultaneously constrain the EOS and NS mass and spin distribution, to avoid introducing biases into the recovered EOS. Second and more pertinent here, any single-event inference ignores substantial prior knowledge about the nuclear EOS obtained from previous analyses. The extension described below provides a simple remedy to this situation suitable for near-future investigations.

Several studies have adopted nonparametric approaches to EOS inference, relying on concrete tables of many EOS realizations [89–92]. These EOS libraries can be weighted to better fit any observation (e.g., gravitational wave, NICER, or pulsar mass constraint) and as needed resampled to impose desired priors (e.g., uniform in maximum mass, $R_{1.4}$, et cetera). The most precise but computationally intensive RIFT strategy involves brute force: compute the EOS evidence for each tabulated EOS. A simpler albeit more approximate strategy involves an order statistic S_α defined for every tabulated EOS α. We have adopted $S = \lambda(\mathcal{M}_c * 2^{1/5})$ as our ordering statistic: the tidal deformability of each neutron star in a symmetric binary, such that the chirp mass is consistent with the observed (detector-frame) chirp mass. Because in practice the mass ratio of NS binaries can't be differentiated from unity, this quantity is a good estimate for the dominant impact ($\tilde{\Lambda}$) that the EOS has on the inspiralling binary, evaluated at masses appropriate for the binary. Each iteration, CIP can construct a posterior in $X = (m_i, \boldsymbol{\chi}_i, S)$ and thus proposed synthetic binaries λ_k, where the binary tidal deformabilities Λ_i associated with each X_k are estimated using the EOS with the closest order statistic S_α to S_k (i.e., $\Lambda_i(S_\alpha)$). This approach allows us to quickly employ any EOS tabulation conditioned on any previous measurements as part of our usual iterative inference technique.

V. SELECTING FIDUCIAL RIFT CONFIGURATIONS

RIFT's modular organization offers immense operational flexibility. Before providing detailed validation studies for selected configurations, in this section we briefly describe several code configurations and report on their performance, to illuminate our choices behind the specific configurations.

A. Selecting between integration and fitting algorithms: A matrix of configurations

RIFT has several modules for integration and interpolation. To simplify the process of discriminating between and validating all of the principal code configurations, we for simplicity focus the most well-behaved scenario: massive binary black holes, similar to those frequently identified by binary black hole searches in advanced LIGO and Virgo data during O3. This choice for fiducial profiling tests, anecdotal examples, and PP plots allows us to assess these configurations in the best possible light, and is appropriate for most observed sources.

Specifically, we summarized three integration techniques for CIP (default, GMM, and adaptive cartesian or AC) and three fitting methods (gaussian process (gp), random forest (rf), and sparse gaussian process). Additionally, some of these CIP techniques can be employed with multiple refinements (e.g., different parameter correlations allowed for GMM; different coordinate systems; et cetera). Both new integration techniques can potentially also be used in and accelerate ILE. Being GPU-accelerated, the AC method is particularly well-suited for ILE, since its other costly likelihood-evaluation operations are already performed on-GPU. In subsequent sections, we will exclusively employ AC integration within ILE. With a focus only on seleting between different CIP configurations, in this section we fix our ILE settings, employing the previous default Monte Carlo integrator, and only report the impact on overall CIP runtime.

Table I illustrates changing overall code resource use from an analysis of GW190620 with a straw-man configuration: a nonprecessing IMRPhenomD model with uniform priors on $\chi_{i,z} \in [-1, 1]$. All configurations adopt the same architecture: RIP coordinates for spin; an initial grid of 1500 points; two iterations omitting the subdominant spin, followed by iteration to convergence with the subdominant spin included. As demonstrated by Figure 12, all analyses converge to a comparable-quality result. Considering all possible pairs of these 5 analyses, the mean one-dimensional JS divergence for $\mathcal{M}_{cz}, q, \chi_{\rm eff}$ are $(2, 1.8, 1.9) \times 10^{-3}$, respectively, dominated by comparisons with the O3 configuration (GP/default) and consistent with the target threshold and sample size produced by this experiment: 4500 samples produced from each analysis, based on $n_{\rm eff}$. Despite intentionally adopting the most favorable circumstances for the default configuration, with very few points and model complexity for the gaussian process interpolator and fiducial integrator, this intentionally simplified example shows that even in this simplest of cases, using our new interpolation and integration methods produce overall better performance.

Figure 13 illustrates code performance on inference of a single zero-spin binary black hole using a nonprecessing IMRPhenomD model.

B. Fiducial production-quality configuration

Motivated by the above, we recommend the following settings for our production analysis. For marginal likelihoods (ILE), we use the AC integrator with distance marginalization, using a target $n_{\rm eff} \simeq 10$. Only the skymap is adapted; other degrees of freedom are sampled by brute force. When assessing batches of points by a single ILE worker, we freeze the skymap after the first iteration. For posterior generation (CIP), we use the GMM sampler with an RF fit, using the previously-described convergence architectures including iterating to convergence, with at least 3 CIP workers contributing to the overall posterior in each iteration. We use

Fit	Integrator	Total (h)	CIP (h)	ILE (h)	T_W(100,3)
GP	Default	14.25	3.8	10.5	1.4
GP	GMM	11.8	3.5	8.25	1.2
RF	Default	7.5	0.17	7.5	0.13
RF	GMM	7.3	0.17	7.1	0.13
RF	AC	5.5	0.17	5.3	0.11

TABLE I: **Example of total analysis costs: GW190620**: Runtime costs (in hours) for a complete, converged analysis of GW190620 using IMRPhenomD and five different RIFT configurations: two different interpolators and three different integrators. The first two columns indicate the CIP interpolation and integration configuration; the third is the total CIP resource usage (here, over all 3 simultaneous CIP instances); and the fourth is the total ILE resource usage, which is smaller for more efficient posterior exploration and modeling. The last column [Eq. (C1), evaluated with $N_I = 100$ and $N_C = 3$] estimates the average wait time if 100 ILE instances and 3 CIP instances are always available for RIFT inference.

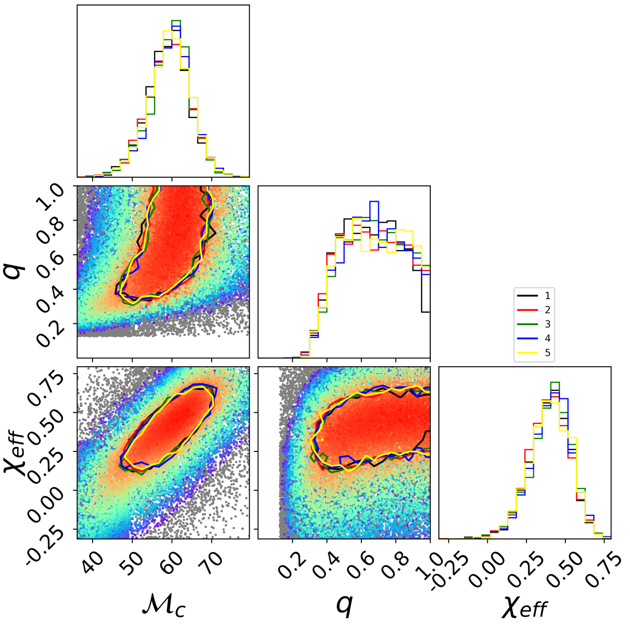

FIG. 12: **Example of results from fiducial GW190620 analysis:** Posteriors derived from all five benchmark analyses of GW190620 enumerated in Table I using IMRPhenomD and uniform priors on $\chi_{i,z}$. All agree. The run labels 1,2,3,4,5 matches the row order used in Table I.

correlated sampling among $\mathcal{M}_c, \delta$ and the cartesian spin components early on, to accelerate sampling.

VI. TESTS

A. Anecdotal end-to-end unsupervised operation

To insure that these alternative algorithmic components do not change RIFT's inferences, and to obtain profiling information to characterize their performance, we performed a large suite of analyses on real events throughout and after the development process. Figure 10 shows a concrete example: an analysis of GW190425 with IMRPhenomPv2 with $\ell \leq 2$ modes. Our default test suite included GW151226, GW170829, GW190412, GW190425, GW190814, GW190620, and GW200115. Almost all worked without human supervision throughout the development process; in our final code configuration, GW190814 converges quickly as well. We systematically tested ILE with GPU acceleration and distance marginalization; CIP with a random-forest fit and correlated GMM sampling, with 3 workers; and a workflow with a convergent subdag and customary precessing iterative structure. Selected examples from these validation studies appear elsewhere in this work.

B. Illustrative example

To provide systematic, controlled, quantitative tests of our algorithmic changes, we employed two fiducial sources: GW190620, as discussed with Figure I and Table 12 above, and a fiducial synthetic zero-spin source, shown in Figures 13. We employ six variants of our algorithms, changing the fitting method (line colors) and the MC integrator (line styles). Likelihood interpolation methods shown are the original aproach (black), our sparse GP code (blue), and the random forest code (green). Monte Carlo integration options for CIP are the original MC method (solid) and the GMM adaptive integrator (dashed). Unlike the previous example, which used a contemporary adaptive architecture, the synthetic source used an O3-style configuration: a fixed number of iterations, using a fixed (and complete) coordinate system to characterize the likelihood.

This anecdotal example consists of a zero-spin BH, shown as the first panel in Figure 13. In this example, RIFT uses the same setup as the ILE-GPU paper [30]: we perform 7 iterations, starting with a 100-point uniform grid in $\mathcal{M}_c, \delta$; each iteration has 5000 evaluation points. Each of these tests use a jittering factor of 3 and force-away parameter of 0.05. We show posterior distributions obtained with RIFT using each combination of settings. All agree.

C. Component performance on many randomly-selected sources

We have also validated several of the new code configurations with probability-probability (PP) plot tests,

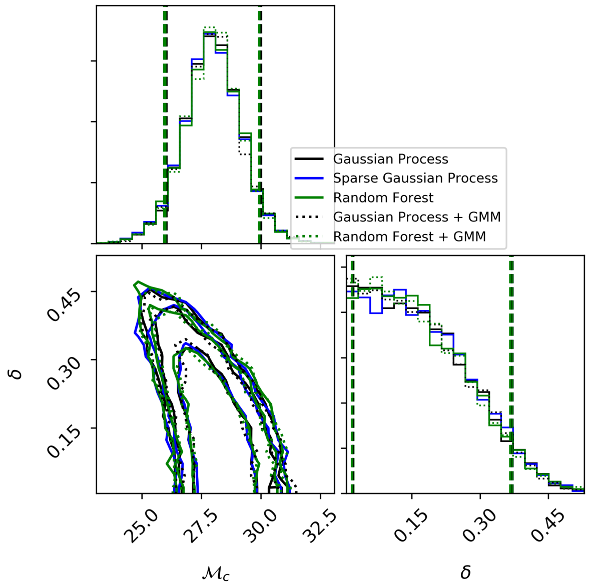

FIG. 13: **Recovering a fiducial posterior distributions: binary black hole**: Results of an analysis of a binary black hole model with aligned spins. Posterior distribution for $\mathcal{M}_c$ and $\delta = (m_1 - m_2)/M$ obtained using RIFT with different interpolation methods (colors) and integration methods (line styles). All source parameters and analysis settings agree precisely with the original RIFT-GPU paper [30]. The black curves indicate conventional GP interpolation with a squared exponential kernel, as in the original RIFT paper; the blue, green, and red lines indicate sparse-GP interpolation (SGP), random-forest interpolation (RF). Colors and line styles indicate the integration and fitting methods used.

using models of varying complexity. Several of these PP plot tests have already appeared earlier, in sections describing and validating individual module components: Figure 5, a PP test for RIP coordinates and nonprecessing PE for NS with tides (and higher-order modes); Figure 8, a PP test for the AC integrator; and Figure 9, a PP test for distance marginalization. Figure 14 shows yet another test of multiple new code components – here, random forest fits and GMM integration in CIP. For this figure, we constructed 200 random synthetic sources with precessing black hole spins, then estimated their parameters using the IMRPhenomPv2 waveform model.

To more sharply validate that our different code configurations for CIP produce identical results on a large sample of synthetic sources, we compared two code configurations on 100 random synthetic injections with zero spin in distinct realizations of random gaussian noise: the fiducial code configuration used in O3, and a version using a random forest fit and GMM sampler in CIP. We extended each analysis until our KL-divergence-based diagnostic on $\mathcal{M}_c$ fell below 10^{-2}. We find that each pair of analyses of the same data produces the same results, as measured by our KL-divergence-based diagnostic Figure 15 shows another measure of agreement between the two algorithms: the difference between the quantiles $\hat{P}(x_j)$, versus $\hat{P}$. Small random differences $\simeq 1/\sqrt{10^4}$ between the two inferred $\hat{P}$ are expected because both probabilities are derived from 10^4 posterior samples. By construction, this test shows both codes produce indistinguishable PP plots. In short, all the extensions described produce indistinguishable results, differing only in their efficiency.

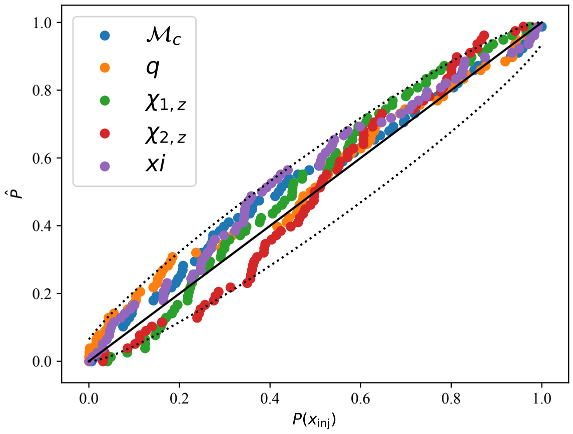

FIG. 14: **Precessing PP plot**: Probability-probability plot to validate recovery of precessing synthetic signals, following the RIFT methods paper [17]. The synthetic sources and parameter inferences are constructed with IMRPhenomPv2 in gaussian noise with presumed known PSDs, for 3-detector networks, starting the signal at 20 Hz and using a 4096 Hz sampling rate. Detector-frame masses are drawn uniformly in the region bounded by $\mathcal{M}/M_\odot \in [30, 60]$ and $\eta \in [0.2, 1/4]$, and sources are drawn volumetrically between 1.5 and 4Gpc. Both BH dimensionless spins are drawn uniformly and volumetrically from the unit sphere. For computational efficiency, all sources in this test have a fixed and presumed known sky location.

VII. ANALYSIS OF RECENT EVENTS

RIFT has been extensively used to analyze GW observations in O1 [93], O2 [15], and O3 [9, 10, 27, 28, 34]. In this section, we briefly reanalyze some recent notable observations with RIFT, to highlight the performance advantages of the configurations and extensions described in this paper. Additionally, we also examine selected events which other groups have prioritized for their own reanalysis. While we adopt largely consistent data conditioning settings and priors as used in previously published work, we do not attempt to rigorously reproduce any previously published work, for simplicity adopting the algorithms described above without any added non-RIFT extensions. (For example and by contrast, the headline results presented in recent LVK analysis of O3 adopt a different fiducial distance prior and attempt to

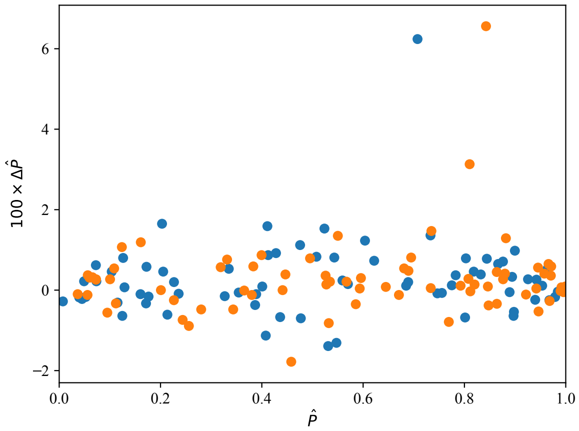

FIG. 15: **Recovering many random posterior distributions**: For the two code configurations a plot of inferred $\Delta\hat{P} \equiv \hat{P}_1(x) - \hat{P}_2(x)$ versus $\hat{P}_1(x)$ for j indexing the synthetic event, where $\hat{P}$ corresponds to the empirical estimate of the posterior CDF deduced from posterior samples (here using 10^4 samples), $1, 2$ refer to the two code configurations, and where x refers to chirp mass (blue) or mass ratio (orange). The two codes assign nearly the same quantile to the injected value for all injections.

marginalize over calibration uncertainties [9, 10]. In this work we adopt the customary d_L^2 distance prior and do not include marginalization over data processing uncertainties.) Rather, the illustrative results presented below in part reflect the reasonable differences expected between groups adopting different analysis choices.

A. GW190814, GW190412

The two events GW190814 and GW190412 are asymmetric compact binary black hole mergers, whose posteriors exhibit strong correlations between $\mathcal{M}_c, \eta$, and component compact object spins. As a result, the revised RIFT configuration enables significantly more efficient performance for these events. For context, during O3 both events required months of wallclock time and extensive human oversight, at least two orders of magnitude more effort than other contemporary O3 RIFT analyses. Now, both events can be analyzed automatically with substantially reduced computational and wallclock time, with minimal human oversight. While detailed timing depends strongly on the waveform model and optimization settings used, we can consistently produce results for both within days (for slow models with complex physics) to even tens of minutes (for simple physics and fast models). As our replication study adds no new scientific insight about these three events, we do not illustrate them here. Instead, Figure 16 shows the estimated run duration assuming no resource congestion: the cumulative CIP evaluation time, divided by the number of CIP workers used simultaneously (here, 3). While this histogram shows only runtimes for IMRPhenomPv2, RIFT's computational cost should be comparable for more costly waveforms; see Appendix C for further discussion.

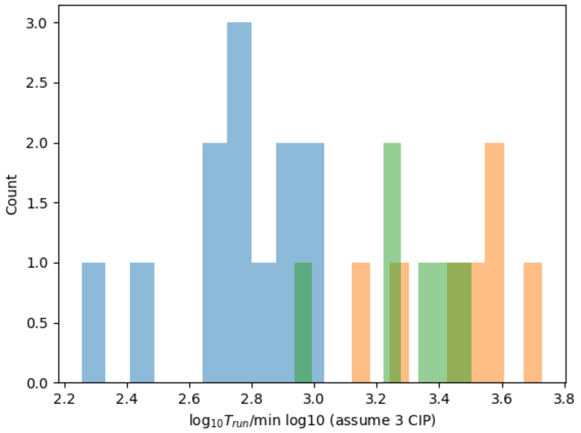

FIG. 16: **Projected runtime for exceptional events** A histogram of the runtime $T_{\rm run}$ (in minutes) for RIFT using IMRPhenomPv2 for GW190425 (blue), GW190814 (orange), and GW190412 (green), run multiple times for each event, with successive runs using contemporary and near-contemporary code configurations.

B. GW200115

Based on the inferred mass of its secondary, the low mass asymmetric merger GW200115 is expected to be a neutron star-black hole merger. As with GW190814 and GW190412, the revised RIFT extensions and configurations presented in this work enable dramatically more efficnet analysis, without human intervention. The left panel in Figure 17 shows an analysis with IMRPhenomPv2 of GW200115 plotted against the SEOBNRv4PHM production run (black solid). The new run uses a faster set of interpolators (rf) and samplers (GMM) and a better and new coordinate system, which reduces the runtime to a matter of a couple of days compared to weeks.

The right panel in Figure 17 shows a similar reanalysis of GW200115 with IMRPhenomXPHM. For comparison, the solid black and blue contours and distributions illustrate previously-reported results, which incorporate calibration marginalization and an alternative distance prior, and were performed with a different analysis code. As expected, the RIFT analysis presented here conforms as expected to the marginal likelihoods shown in color scale. This reanalysis favors a higher secondary mass and a more negatively aligned spin. All differences between these calculations are modest, with largely overlapping support.

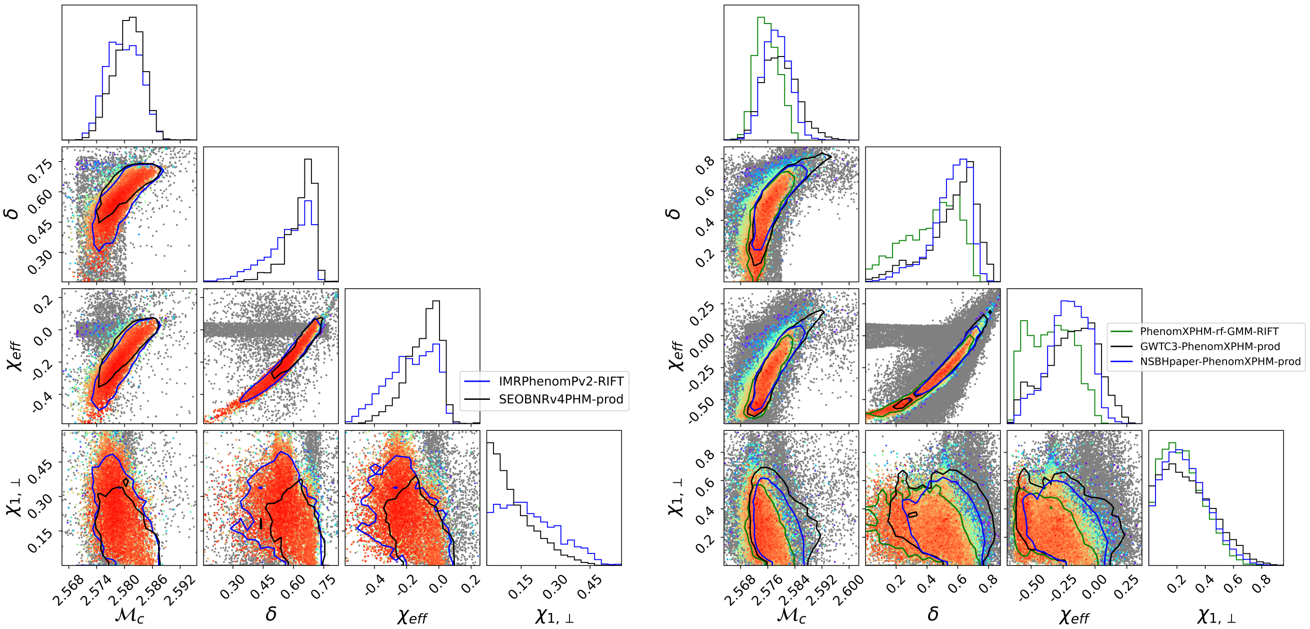

FIG. 17: **Analysis of GW200115**: Corner plot with likelihoods (points and color scale) for GW200115, analyzed with RIFT and IMRPhenomPv2 (left panel) and IMRPhenomXPHM (right panel), showing the parameters $\mathcal{M}_c$, $\delta = (m_1 - m_2)/M$, $\chi_{\rm eff}$ and the magnitude of the transverse component of the primary spin $\chi_{1,\perp} = \sqrt{\chi_{1,x}^2 + \chi_{1,y}^2}$. Color scale shows RIFT marginal likelihood evaluations versus intrinsic parameters; solid contours show 90% credible intervals for the two-dimensional marginal distributions; and the diagonal panels show one-dimensional marginal distributions of these parameters. In the left panel, for comparison the black solid curves and distributions show similar results derived with the published SEOBNRv4PHM analysis. In the right panel, for comparison the green and blue curves show previously-reported results derived with IMRPhenomXPHM with different analysis settings and inference.

C. GW151226

The original published analyses of GW151226 favored comparable binary masses, with a nominal posterior for the two ordered variables $m_1 > m_2$ as close to equal mass as would be expected given the strong degeneracy along lines of constant $\mathcal{M}_c$. These results were corrobrated in GWTC-2 with reanalysis including direct comparison to numerical relativity simulations including higher-order modes, albeit at the time limited only to nonprecessing simulations [94]. Several groups have published reanalyses of these events (e.g., [95–97]), including a recent LVK reanalysis [10].

One followup reinvestigation of this event using models with recent semianalytic waveform models have found modest support for higher mass ratio [97]. They suggest the high-mass-ratio configurations ($1/q > 5$) could be consistent with strong orbital precession.

The two panels of Figure 18 shows an unsupervised RIFT reanalysis of this event with IMRPhenomXPHM, using our contemporary architecture: adaptive convergence, RIP and pseudo-cylindrical coordiantes, et cetera as described in Section V B. As previously, contours indicate 90% credible intervals, while the colorscale indicates $\ln \mathcal{L}$; points colored in light gray have $\ln \mathcal{L}$ farther than 15 away from the peak value. All analyses use settings comparable to our original analysis of this event: C02 data, with identical noise power spectra estimates. We have both reanalyzed these events with our customary current pipeline and also performed a targeted analysis to densely evaluate the likelihood for $\eta \in [0.02, 0.08]$. For our analysis with IMRPhenomXPHM, we adopt a reference frequency of $f_{\rm ref} = 100$Hz, comparable to [97]. We have corroborated our conclusions with a focused investigation of high-mass-ratio region. In neither analysis do we find the posterior strongly supports a high q, strongly precessing interpretation of GW151226.

Looking more closely at the underlying marginal likelihoods, in both cases we find *some* high-q, strongly-precessing configurations with significant (but nonexceptional) marginal likelihood. In both our unsupervised and targeted analyses, we find that, *conditioned on the requirement of high mass ratio*, the posterior distribution for transverse spin $\chi_{1,\perp}$ suggests well-localized spin orientation, consistent with the statements in [97]. In other words, the high-q region of the posterior modestly favors transverse spins, and the specific orientation of these spins is better constrained with $f_{\rm ref} = 100$Hz than the fiducial 10 Hz. At this reference frequency, the one-dimensional distribution of ϕ_i and the two-dimensional distributions $\chi_{1,\perp}$ show some modest asymmetries, in this mass region. While we concur with [97] that the

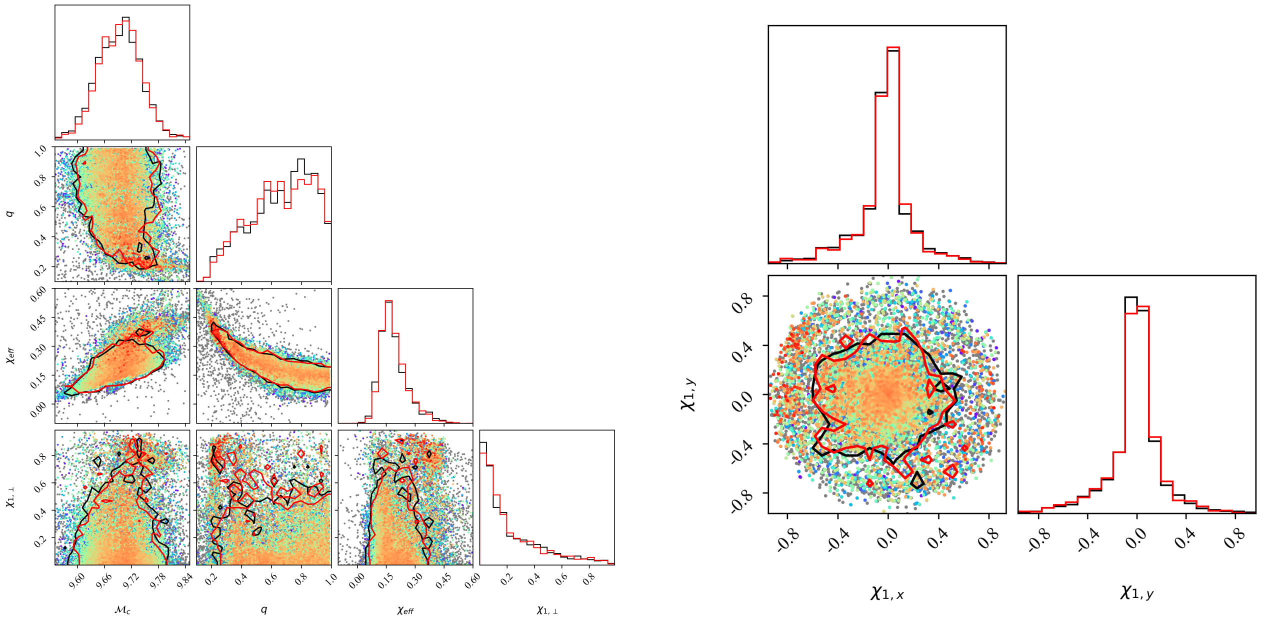

FIG. 18: **Reanalysis of GW151226 with IMRPhenomXPHM**: Two corner plots with 90% credible interval quantiles for the last two RIFT iterations, indicated by different line colors. The left panel shows $\mathcal{M}_c, q = m_2/m_1, \chi_{\rm eff}$ and the magnitude of the transverse dimensionless spin of the primary $\chi_{1,\perp} = \sqrt{\chi_{1,x}^2 + \chi_{1,y}^2}$. The right panel shows the two transverse components of the dimensionless spin of the primary $\chi_{1,x}, \chi_{1,y}$. The color scale shows the likelihood range, over a dynamic range $\Delta \ln \mathcal{L} \leq 15$; gray points indicate likelihood evaluations below this range. While a small region of marginally higher likelihood exists at asymmetric mass ratio, large transverse spins, and positive $\chi_{\rm eff}$, overall our inferences still favor a conventional interpretation for this event.

choice of prior strongly suppresses the significance of the transverse spins in the *posterior*, the lack of exceptional precessing configurations with with high *marginal likelihoods* demonstrates that these are at best comparably likely to the well-explored comparable-mass component of the posterior.

The differences in interpretation between this analysis and those of Chia et al can easily arise from relatively small details, For a relatively weak event like GW151226, small perturbations to a nearly-flat, low-significance likelihood introduced by (for example) different choices in data conditioning can easily produce large changes in the posterior. Closely examining their work, their figures suggest no support for spin-orbit-induced modulations during the *insprial* phase: their Fig. 6 suggests that their early-time maximum likelihood waveforms are consistent with low transverse spin. In other words, their results are consistent with a viewing angle consistent with $\pm\mathbf{J}$, the total angular momentum direction, along which minimal modulation is expected.

VIII. CONCLUSIONS

We have described extensions of the RIFT parameter inference software ecosystem, including the specific choices adopted during O3 and many new extensions proposed for post-O3 work. In this work specifically, we introduce new coordinate systems, fitting techniques, integration techniques, and pipeline architectures which together significantly accelerate the performance and scientific return of RIFT as a parameter inference tool. To highlight RIFT's capabilities, we briefly report on reanalyses of several pertinent GW observations.

The improvements highlighted here do not exhaust RIFT's capability, both in general or for ultra-low-latency analysis. For example, previous studies proposed direct quadrature of both intrinsic and extrinsic variables using surrogate models [48]. RIFT could likewise benefit from normalizing flows to accelerate its Monte Carlo integration, in effect accelerating our importance sampling via approximate inference and benefitting from several group's efforts to provide such preliminary estimates [98–102]. Additionally and more broadly, RIFT can benefit from many additional improvements in its implementation details, including more optimal coordinates; more use of GPU-accelerated computation; and streamlined pipeline architecture. Even with the existing codebase, RIFT offers novel low-latency capability, even with costly models. For example, RIFT could use a small number of very large iterations, after loose targeting with preliminary estimates (e.g., from nonprecessing inference). Even the existing framework can complete inference within a handful of minutes for simple models, with appropriate operating point choices. We defer discussion of specific low-latency analysis frameworks to a dedicated publication.

Acknowledgments

The authors thank Ben Farr for many helpful comments on the manuscript. JW is supported by NSF PHY-2110460. ROS, VD, and AY are supported by NSF PHY-2012057; ROS is also supported via NSF PHY-1912632 and AST-1909534. DW is supported by NSF PHY-1912649, NSF PHY-2207728. JR is supported by NSF RUI-2110441 and AST-2219109. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation. The authors are grateful for computational resources provided by the LIGO Laboratory, supported by National Science Foundation Grants PHY-0757058 and PHY-0823459, and by the International Gravitational Wave Observatory Network.

APPENDIX A: GAUSSIAN MIXTURE MODEL IMPLEMENTATION

Gaussian mixture models (GMMs) and expectation-maximization (EM) have been thoroughly described in the literature [80–82]. In this appendix, we summarize our implementation, emphasizing features needed for our work.

Expectation-maximization (EM) for GMMs: The EM algorithm fits K Gaussians, each described by its mean $\boldsymbol{\mu}_k$, its covariance $\boldsymbol{\Sigma}_k$, and its mixture weight π_k, such that the likelihood of the model $\mathscr{L}$ is maximized. For each sample $\boldsymbol{x}_n$ and its corresponding sample weight w_n,

$$\mathscr{L} = \prod_n P(\boldsymbol{x}_n). \tag{A1}$$

$P(\boldsymbol{x})$ is the probability of a point $\boldsymbol{x}$ under the current model, or

$$P(\boldsymbol{x}_n) = w_n \sum_k N(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k, \tag{A2}$$

where $N(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is the multivariate Gaussian density. $P(\boldsymbol{x})$ can be split into the K individual probabilities for each $\boldsymbol{x}$:

$$p_{nk} = \frac{N(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k w_n}{P(\boldsymbol{x}_n)} \tag{A3}$$

The above equations describe the expectation step (E-step) of the EM algorithm. The means, covariances, and mixture weights are estimated from p_{nk} in the maximization step (M-step) as follows.

$$\boldsymbol{\mu}_k = \sum_n p_{nk}\boldsymbol{x}_n \Big/ \sum_n p_{nk} \tag{A4}$$

$$\boldsymbol{\Sigma}_k = \sum_n p_{nk}(\boldsymbol{x}_n - \boldsymbol{\mu}_k)(\boldsymbol{x}_n - \boldsymbol{\mu}_k) \Big/ \sum_n p_{nk} \tag{A5}$$

$$\pi_k = \frac{1}{N}\sum_n p_{nk} \tag{A6}$$

$$p_{nk} = \frac{N(\boldsymbol{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\pi_k w_n}{P(\boldsymbol{x}_n)} \tag{A7}$$

The iterative EM algorithm is initilized by guessing initial values for the means, covariances, and mixture weights. Alternating E-steps and M-steps are done until the change in $\mathscr{L}$ between iterations is below a predetermined threshold.

The standard expectation-maximization algorithm has been modified to allow iterative, online updates of the model with a new set of samples and weights. A new GMM is first trained using the new data. Each component i in the new model is then matched to a component j in the old model in such a way as to minimize the total Mahalanobis distance M_D between all of the means:

$$\begin{aligned}\sum M_D = \sum_{i,j=0} (&\sqrt{(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)^T \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\mu}_j - \boldsymbol{\mu}_i)} \\ &+ \sqrt{(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)})\end{aligned} \tag{A8}$$

The combination of components which minimizes $\sum M_D$ is the one that is kept.

Once the components have been matched, they are combined. N is the total number of points the old model has been trained on and M is the number of samples in the new batch.

$$\boldsymbol{\mu} = \frac{N\pi_j\boldsymbol{\mu}_j + M\pi_i\boldsymbol{\mu}_i}{N\pi_j + M\pi_i} \tag{A9}$$

$$\boldsymbol{\Sigma} = \frac{N\pi_j\boldsymbol{\Sigma}_j + M\pi_i\boldsymbol{\Sigma}_i}{N\pi_j + M\pi_i} + \frac{N\pi_j\boldsymbol{\mu}_j\boldsymbol{\mu}_j^T + M\pi_i\boldsymbol{\mu}_i\boldsymbol{\mu}_i^T}{N\pi_j + M\pi_i} - \boldsymbol{\mu}\boldsymbol{\mu}^T \tag{A10}$$

$$\pi = \frac{N\pi_j + M\pi_k}{N + M} \tag{A11}$$

Accounting for finite domains: Monte Carlo sampling for parameter estimation requires samples from a finite, rectangular domain. When sampling from a GMM, therefore, we must truncate the infinite-domain multivariate Gaussians to our finite domain. There is no widely-used implementation of a truncated multivariate Gaussian, but we can take advantage of `Scipy`'s univariate `truncnorm` function.

In general, to generate a sample from a multivariate Gaussian from a distribution with covariance $\boldsymbol{\Sigma}$ and mean $\boldsymbol{\mu}$, we first generate a sample $\boldsymbol{x}$ from a Gaussian centered at the origin with covariance 1. Our final sample is then

$$\boldsymbol{x}' = \boldsymbol{\lambda}^{1/2}\boldsymbol{\phi}\boldsymbol{x} + \boldsymbol{\mu}, \tag{A12}$$

where $\boldsymbol{\lambda}$ is a diagonal matrix of the eigenvalues of $\boldsymbol{\Sigma}$ and $\boldsymbol{\phi}$ is a matrix containing the corresponding eigenvectors of $\boldsymbol{\Sigma}$.

To generate truncated Gaussian samples, we then simply take our initial sample (with mean 0 and covariance 1) using `truncnorm`, and transform them to have the desired mean and covariance. The problem with this approach, however, is that the bounds are transformed along with the samples, resulting in a parallelogram-shaped domain. Our solution to this problem is to sample from the smallest rectangualar region that, when transformed, will contain the desired sampling domain; any samples that end up outside of this region are simply thrown out.

Each corner of our desired domain is transformed by $\boldsymbol{r}$, where

$$\boldsymbol{r} = [\boldsymbol{\lambda}^{1/2}\boldsymbol{\phi}]^{-1}. \tag{A13}$$

For each dimension, the minimum and maximum transformed corner points are used as the bounds for our univariate `truncnorm` samples. After transforming the samples, we throw out any that fall outside our original bounds.

APPENDIX B: MONTE CARLO INTEGRATION AND INDEPENDENT SAMPLES

RIFT relies heavily on Monte Carlo integration. This appendix provides a brief review, highlighting pertinent subtle issues about results and convergence that arise in real applications. In this section, we will consider an integral $S = \int Lp(x)dx$ over some volume in x relative to a normalized probability $p(x)$, and its alternative expression $S = \int L[p(x)/p_x(x)]p_x(x)dx$ relative to another probability density $p_x(x)$ over x. We will define the random variable $w = Lp/p_x$, such that $S = \langle w \rangle$ (averaging over the distribution from p_s). The Monte Carlo approach to this integral involves drawing many samples x_k, evaluating w_k, and evaluating the sample mean $\bar{w} = \sum_{k=1}^{N} w_k/N$. According to the weak (and strong) laws of large numbers, the sample mean will converge to S, so long as $\langle w \rangle$ is finite (even if higher-order moments do not exist). In the special case that w has finite and known variance $V(w) \equiv \sigma_w^2$, the distribution of the sample mean will be asymptotically normal, with a mean of $\langle w \rangle$ and a variance of σ_w^2/N. If the variance exists and can be well-approximated by the sample variance, then the samples themselves provide an estimate for the integral and its error [45].

RIFT both uses the Monte Carlo integral (ILE) and, when appropriate (CIP), the associated weighted samples themselves. To introduce notation, the weighted Monte Carlo integration methods output points x_k, likelihoods $L_k \geq 0$, and weights $w_k = L(x_k)p(x_k)/p_s(x_k)$, where x_k are fair draws from the sampling prior $p_s(x)$. Using these outputs, we evaluate overall Monte Carlo integrals; estimate marginal distributions; and resample to produce fair-draw outputs. As each outcome involves a different expression of these samples, several measures of convergence and hence the "number of independent samples" have been adopted, expressed in terms of normalized sample probabilities $p_k = w_k/\sum_q w_q$.

Historically, RIFT adopts a very conservative account of the number of independent samples [16]:

$$n_{\rm eff} = \frac{\sum p_k}{\max\{p_k\}} \tag{B1}$$

The value $1/n_{\rm eff}$ is the largest discontinuous jump in the estimator $\hat{P}(< x) = \sum_k p_k \theta(x_k - x)$ for any one-dimensional cumulative probability distribution $P(< x)$ derived from the full samples. Alternatively, the sample size can be defined using the estimated moments of the weight distribution. For example, one estimate of the effective sample size is [103]

$$n_{\rm eff,ESS} = \frac{(\sum_k p_k)^2}{\sum_k p_k^2} \tag{B2}$$

An alternative choice grounded in the Monte Carlo integral error standardizes the sample size to the sample variance. As the natural count of independent samples scales as $N/V(w)$, an alternative estimate for the number of independent samples based on the Monte Carlo variance estimate [104]:

$$\frac{\langle w \rangle^2}{n_{\rm var}} \equiv \frac{1}{N}\left[-\langle w \rangle^2 + \frac{1}{N-1}\sum_k w_k^2 \right] \tag{B3}$$

In other words, $n_{\rm eff}$ is the ratio of the sample mean (squared) and the sample variance, times the number of points drawn: $n_{\rm eff} = N\bar{w}^2/s^2$ where s^2 denotes the sample variance. This accounting of the number of independent points can be dramatically larger than the conservative estimate of Eq. (B1), depending on the integrand, or nearly zero for scenarios where the variance diverges, as discussed below. Finally, the entropy of the probability weights $H(p) = \sum_k p_k \ln(1/p_k)$ is maximized at $\ln n$ when all the probability weights are equal. Motivated by the maximum value of entropy, we define

$$n_{\rm eff,H} \equiv \exp(H(p)) \tag{B4}$$

This information-theory-based estimate of the number of independent evaluations can be slightly less conservative than $n_{\rm eff}$.

As a practical illustration of these sample size conventions, we introduce a one-dimensional toy model: $L = 1$, $p(x) = 1$ and $p_x(x) = \alpha x^{\alpha-1}$ for $x \in [0, 1]$, with $\alpha > 0$. For example, these pair of priors might represent an attempt to rescale a single spin's volumetric sampling density (i.e., the case $\alpha = 3$ and $x = |\chi_1|$) to reflect a physical uniform spin magnitude prior. In this scenario, $w = p/p_s = 1/\alpha x^{\alpha-1}$, which is defined over $[1/\alpha, \infty)$ for $\alpha > 1$ and over $[0, 1/\alpha)$ for $\alpha < 1$. All comments below are easily verified by simple numerical experiments.

- *Monte Carlo integral*: As required by the strong and weak law of large numbers, $\sum_k w_k/N$ is nearly unity almost always for large N. At fixed sample size but changing α, the standard naive Monte Carlo uncertainty estimate $s/\sqrt{N}$ increases, reflecting the rarity of sample points sufficiently close to the small region near $x \simeq 0$ which dominates the integral.

  Nominally we would expect the Monte Carlo uncertainty to scale as $V(w)/N$. For $\alpha \in (0, 2)$ the variance is integrable, but for $\alpha > 2$ the lower limit diverges:

  $$\langle w^2 \rangle = \int_0^1 dx/p_s(x) = \frac{1}{\alpha(2-\alpha)} x^{2-\alpha}\Big|_0^2$$

  Therefore, the analytic expression $V(w) = (1-\alpha)^2/\alpha(2-\alpha)$ for the variance is only well-defined for $\alpha < 2$ – in particular, excluding the highly-desirable scenario of reweighting from a volumetric to a uniform spin magnitude prior! Nonetheless, above this threshold and in the regime of a formally divergent variance, the conventional estimate for Monte Carlo error based on the *sample* variance is a reasonable estimate of the true error scale for many $\alpha < 10$. These divergences in the moments of w do *not* limit the efficacy of Monte Carlo integration, whose convergence is asymptotically protected by the law of large numbers.

- *Moment-based size*: We proposed two sample size estimates based on (sample) means and variances of the distribution. As noted above, for $\alpha > 2$, the true second moment and variances diverge. Nonetheless, in empirical experiments with our toy problem with $\alpha > 2$, both $n_{\rm eff}$ and $n_{\rm var}$ exhibit qualitative consistent behavior relative to the other two sample size estimates discussed below.

- *Default (max-sample) size*: The single-most-significant sample provides a conservative estimate for the effective sample size which manifestly must remain finite and comprehensible. For non-normalized draws w_k such that $\langle w \rangle = 1$, our default estimate for $N_{\rm eff}$ is approximately $N/\max_k w_k$. For the scenario with $\alpha > 1$, roughly speaking since the nearest sample has probability $1/N$, this means $n_{\rm eff} \simeq N\alpha x_{\rm min}^{\alpha-1} \simeq \alpha N x^\alpha/x \simeq \alpha/x$, and thus $n_{\rm eff} = N/w_{\rm max} = \alpha N^{1/\alpha}$ – in other words, a few times the natural number $N^{1/\alpha}$ of points expected nearby.

  More formally for $\alpha > 1$, the cumualtive distribution of the largest value $w_{\rm max}$: out of N samples of w is $P(< w_{\rm max})^N$. Because of reordering between x and w for $\alpha > 1$, the cumulative distribution function for w is easily expressed in terms of the cumulative distribution of $u = x^\alpha : P(< w) = P(> u(w)) = 1 - u = 1 - (1/\alpha w)^{(\alpha/(1-\alpha))}$. In the limit of large N, the median value of $w_{\rm max}$ can be estimated by solving $1/2 = P(< w_{\rm max})^N$, leading to a simple approximate expression for the median value of $n_{\rm eff}$ for this one-dimensional rescaling:

$$\text{median}_x(n_{\rm eff}) \simeq N\alpha \left(\frac{N}{\ln 2}\right)^{-(\alpha-1)/\alpha} \simeq O(N^{1/\alpha}) \quad \text{(B5)}$$

- *Entropy-based size*: Finally, because our entropy size estimate involves a Monte Carlo estimate of the distribution entropy, we anticipate that the finiteness of the entropy ensures the entropy-based sample size is stable and well-posed for all $\alpha > 0$.

As illustrated by the discussion above, these different sample size estimates can have dramatically different behavior, including different scaling with N! For our toy problem, empirically $n_{\rm eff,H}$ is a nearly constant fraction of N; $n_{\rm eff}$ scales as $N^{1/\alpha}$, with a very unfavorable prefactor; and $n_{\rm eff,var} \simeq n_{\rm eff,ESS}$ scale roughly in between (e.g., comparable to $N^{2/\alpha}$). In particular, these expressions suggest that both our probability distributions (with error scale set by $1/n_{\rm eff}$ in their cumulative) and evidence (with error scale set by $1/\sqrt{n_{\rm var}}$) have uncertainties scaling as $N^{-1/\alpha}$ for this toy problem. This expected but extremely unfavorable scaling has straightforward implications for our reweighting strategies: in short, avoid spin reweighting whenever possible, unless drawing samples from a distribution with similar singular behavior near the origin.

APPENDIX C: HOW RESOURCES DETERMINE POSSIBLE OPERATING POINTS

As synthetic and real sources accumulate, RIFT users typically need to perform extremely large numbers of source inferences. The rate, latency, and accuracy of these inferences depend on the available resources, waveform model cost, and population of signals being analyzed. Different science objectives and available resources can produce dramatically different choices for how to operate the RIFT pipeline. In this section., we briefly outline how these choices impact RIFT analysis throughput and overall cost, highlighting a few expected use cases.

Generally, RIFT involves two sets of calculations, potentially provided by distinct pools of resources: ILE evaluations, provided by N_I resources (e.g., low-cost GPUs); and CIP posterior generation, provided by N_C resources (e.g., usually modestly memory-rich CPUs). Both ILE and CIP are characterized by a typical runtime. Each ILE marginal likelihood evaluation for a specific model (i.e., approximation, mode list, starting frequency) and at a fiducial accuracy ϵ (i.e., the relative error in their marginal likelihood) requires a time τ_I, ranging from a second to a few minutes. Each CIP posterior generation worker requires a time τ_C that depends strongly on the approach used, dimensionality of the space, and model complexity. In this appendix we consider CIP configurations with runtimes from seconds to hours. Uusally, both ILE and CIP involve Monte Carlo integration, so their runtime nominally increases as $1/\epsilon^2$. RIFT employs many instances of ILE and CIP simultaneously.

The typical wait time and total pipeline resource usage follows by accounting for the total cost needed for all stages of the analysis. We assume a full analysis requires $n_I \simeq O(10^3)$ likelihood evaluations, organized roughly into n_{it} chunks of size n_I/n_{it}. After each chunk, a number n_c CIP workers will each independently generate a fraction of the overall posterior; the total time needed to complete posterior generation can be appreciably reduced by employing many CIP workers simultaneously. When many RIFT analyses are performed simultaneously, the overall resource usage per event can be estimated ignoring the pipeline's serialization of ILE and CIP stages. In this circumstance, the resource usage T_R and average analysis time T_W

$$T_R = \tau_I n_I + \tau_C n_C n_{it} \tag{C1a}$$

$$T_R/10^4{\rm s} \simeq (\tau_i/{\rm s})(n_I/10^4) + 10(\tau_c/{\rm h})(n_C/3)(n_{\rm it}/10) \tag{C1b}$$

$$T_W = \frac{\tau_I n_I}{N_I} + \frac{\tau_C n_C n_{it}}{N_C} \tag{C1c}$$

where the first expression provides the total time needed to perform an analysis, while the second estimates the effective analysis duration given the resources available. The total resource usage needed to perform $\mathcal{N}$ analyses is just $\mathcal{N} T_R$. The two types of resources contribute equally to the average analysis wait time when

$$\begin{aligned} N_{C,\rm match} &\equiv N_I \frac{\tau_C n_C n_{it}}{n_i \tau_I} \\ &\simeq 10 N_I \frac{(\tau_c/{\rm h})(n_C/3)(n_{\rm it}/10)}{(\tau_i/{\rm h})(n_I/10^4)} \end{aligned} \tag{C1d}$$

More concretely, a RIFT analysis pool could need roughly $N_{c,\rm match}/N_I \simeq$ ten times as many CPUs as GPUs, to maintain a steady state, given these fiducial timescales τ_I, τ_C. Finally, ignoring resource contention limits and recognizing that each ILE job in fact evaluates $w \simeq O(10)$ likelihood evaluations in series, the user time needed to complete a single targeted analysis with a larger number n'_C of CIP instances could be as short as

$$T_U = n_{it}\left[\tau_I w + \frac{\tau_C}{n'_C/n_C}\right] \tag{C1e}$$

where we assume each worker performs a fraction $\tau_C n_C/n'_C$ of the overall work of generating the posterior. The runtime τ_I depends strongly on the maximum mode order $\ell_{\rm max}$, if waveform generation costs are subdominant to the costs of evaluating the likelihood many times. Because the RIFT likelihood depends on matrix multiplications over arrays of modes, in this regime the ILE runtime will scale roughly as the square of the number of waveform spherical harmonic modes h_{lm} used in the analysis:

$$\tau_I \simeq \tau_{I,\rm ref}\frac{1}{2}\sum_{\ell=2}^{\ell_{\rm max}}(2\ell+1) = \tau_{I,\rm rmref}\frac{2}{3}(\ell_{\rm max}+1)^3 - \frac{L+31}{6} \tag{C2}$$

$$\simeq \tau_{I,\rm ref} 80(\ell_{\rm max}/4)^3 \tag{C3}$$

For contemporary hardware and GPU-accelerated integration in ILE, we observe τ_I between 30-90 seconds for $\ell_{\rm max} = 4$ and $\tau_{I,\rm ref}$ less than one second for a simple nonprecessing model.

In practice, users will not achieve even these modest benchmarks on T_W, T_U due to resource contention, queuing time, and cluster mismaps. For example, with typically 5000 likelihood evaluations used in the first iteration and $w \simeq 20$ likelihood evaluations per worker, only a small fraction of the $5000/w \simeq 250$ ILE jobs needed can be queued simultaneously, as usually $N_I \ll 250$. As these first short likelihood evaluations finish, the time needed to queue new jobs to replace them often substantially exceeds their duration unless τ_I is exceptionally long or w large, both factors contributing to overall run latency.

1. Default operating choices

Our default operating point choices reflect the fiducial scalings in Eq. (C1), appropriate to precessing black hole binaries analyzed with a fast waveform approximation. Conflating the impact of our hardware and queue priority environments, we effectively have access to relatively many low-cost GPUs (e.g., tens of GPUs per user), but have less frequent access to the high-memory nodes we usually use for CIP (e.g., tens of non-GPU cores per user). Otherwise, our typical analyses' inputs are compraable to the fiducial scalings above: $N_I \simeq 2 \times 10^4$ marginal likelihood evaluations to achieve our target accuracy, with $n_{\rm it}$ between 5 and 10. As a result, our analyses' wait times are invariably CIP-constrained, as $N_C \ll N_{C,\rm match}$; total resource usage is likewise CIP dominated, with $T_R \simeq N_C \tau_c n_{\rm it}$ between a few tens to $O(100)$ hours per run; and effective wait times $T_W \simeq \tau_C n_C n_{it}/N_C$ of order a few to several hours, or even tens of hours for larger τ_C. The user wait time T_U for any specific analysis will be smaller in direct proportion to the number of workers employed. An individual with access to these resources can maintain roughly $N_C/n_{it} n_C$ analyses simultaneously in a steady state; for our fiducial single user, this number is of order unity. While we scaled the discussion above to individual users, a large organization with more resources (e.g., $N_I \simeq 200, N_C > 2000$) and control over queue priority can achieve correspondingly higher throughput simply by allocating more resources and priority to RIFT operations. Such high resources should be sufficient *in principle* to complete even costly analyses with larger values of τ_I, τ_C in roughly tens of minutes on average [30].

In these circumstances, operating point choices which maximize N_C, ε and minimize $\tau_C n_{it} n_C$ have immediate return on overall cost and latency. For example, the number of matching CIP-capable resources N_C can be enhanced with lower memory requirements or alternative computing pools (e.g., the open science grid). The number of iterations and τ_C can be reduced by well-adapted coordinates and prior settings. Three extreme examples of low τ_C involve nonprecessing binaries (for which CIP can often complete within minutes); Gaussian-based posterior generation (for which CIP can complete within about one minute); and AMR-based grid placement (for which τ_C completes in seconds). The user efficiency ε can be increased with careful planning and extensive automation. In a resource-saturated environment, increasing the number n_C of workers per job does not change throughput, just the latency T_U of each analysis.

Finally, we emphasize that user mishaps, poor planning, and cluster mischance usually dominate unused time. A typical single user will usually complete only a small fraction ε of intended analyses in their final form, with the overwhelming majority associated with exploratory work, preliminary analysis, and validation.

2. High-resource, low-latency configuration

If a highly-resourced organizations targets large-scale automated low-latency analysis with the conventional RIFT pipeline, the achievable latency would nominally eventually be limited by the first term in Eq. (C1e): $n_{it}\tau_I w$, associated with the runtime needed to serially perform n_{it} instances of ILE in series, each evaluating the likelihood w times. In practice, however, several sources of pipeline overhead will contribute to added lag, such as the startup time for CIP and ILE.

3. Extremely low-latency configurations

The lowest possible latencies T_U can be achieved using a simple waveform model (i.e., low τ_I) with limited waveform physics (i.e., small τ_C), small numbers of evaluations w per ILE worker, and few iterations $n_{\rm it}$ needed to achieve the target accuracy goal. As an example, the AMR-based strategy [87] referenced above is designed to have $w \simeq 1$ and likelihood evaluation times τ_I of order tens of seconds (i.e., larger than the steady-state limit due to startup and file access overhead). Using high-priority queuing and with AMR grid placement requiring τ_C of order seconds, conceivably an AMR approach should perform followup within a minute or less.

4. GPU-limited configuration

A configuration with large τ_I or relatively small N_I can produce an unusual ILE-limited configuration. These circumstances can arise for analyses with many higher order modes, as $\tau_I \propto \ell^3_{\rm max}$, or with few available high-speed GPU resources needed to achieve accelerated integration. These circumstances also require that N_I/τ_I is larger for a GPU configuration (small N_I but also small τ_I) is still large compared to the corresponding product for a CPU configuration (larger N_I but much larger τ_I). In these circumstances, the typical analysis time will be dominated by likelihood evaluations ($T_W \simeq \tau_I n_I/N_I$). As a concrete example, a user performing analyses of higher-order-mode models with a small GPU pool ($N_I \simeq 10$) could have $\tau_I \simeq 1$min, implying a typical analysis wait time of $T_W \simeq 50$h, ignoring the smaller contribution from CIP to the overall analysis time.

5. Extremely high-cost waveforms

RIFT has in the past operated successfully with waveforms requiring hours to generate. Even for relatively modern waveform generators, the generation of waveforms for very low-mass binaries could be costly and produce large data products, owing to the signal's duration and the potential need to adopt a high sampling rate to resolve high-frequency higher-order modes. When the waveform generation cost dominates all other considerations, RIFT should employ the largest possible pool of resources for N_I: both GPU and non-GPU resources should be included. Similarly, each worker should evidently only analyze one observation ($w \simeq 1$) to reduce latency $T_U \simeq n_{it}\tau_I$. In this configuration, the cost per analysis can substantially increase: $T_R \simeq n_{\rm it}\tau_I \simeq 2\times 10^3$h $(\tau/4{\rm min})(n_{it}/10)$. We emphasize that a high waveform cost does not preclude low-latency analysis, if τ_I is sufficiently small compared to the target latency.

APPENDIX D: NUMERICAL AND ADAPTATION APPROACHES NEEDED FOR STRONG SIGNALS

The main text describes our customary recommendations for RIFT, appropriate to the vast majority of sources with signal-to-noise ρ below $\simeq 30$. In this section, we address additional numerical, operating point, and algorithmic choices more appropriate to signals with high or very high amplitudes.

1. Estimates of signal strength

The intrinsic source signal-to-noise ratio has a well-understood impact on the complexity and scale of the likelihood $\mathcal{L}_{\rm full}$ and posterior. In this subsection, we will use ρ to denote the true signal amplitude, defined such that *in the absence of noise* $\rho^2/2 = \max_{\lambda,\theta}\mathcal{L}_{\rm full}$; $\rho_{\rm hint}$ will be an estimated signal amplitude, provided by the search pipelines which discovered the event candidate; and $\rho_{\rm guess}(\lambda)$ is a guess described below designed to estimate $\max_\theta \sqrt{2\mathcal{L}_{\rm full}(\lambda,\theta)}$ for a specific set of source parameters λ.

Our estimate $\rho_{\rm guess}(\lambda)$ is expressed in terms of the factors entering into the full likelihood used within ILE [16]:

$$\ln\mathcal{L}_{\rm full}(\lambda,\theta) = -\frac{1}{2}\sum_k \langle h_k(\lambda,\theta) - d_k|h_k(\lambda,\theta) - d_k\rangle_k - \langle d_k|d_k\rangle_k \tag{D1}$$

$$= \sum_k \sum_{lm} (F_{k\,-2}Y_{lm})^* Q_{k,lm}(\lambda, t_k) \tag{D2}$$

$$- \frac{(D_{\rm ref}/D)^2}{4}\sum_k \left[|F_k|^2[{}_{-2}Y_{lm}]^*{}_{-2}Y_{l'm'}U_{k,lm,l'm'}(\lambda) + {\rm Re}\left(F_k^2\,{}_{-2}Y_{lm}\,{}_{-2}Y_{l'm'}V_{k,lm,l'm'}\right)\right] \tag{D3}$$

where the pertinent factors are expressed in terms of inner products of the signal modes h_{lm} with each other or with the data:

$$Q_{k,\ell m}(\lambda, t_k) \equiv \langle h_{\ell m}(\lambda, t_k)|d\rangle_k$$
$$= 2\int_{|f|>f_{\rm low}} \frac{df}{S_{n,k}(|f|)} e^{2\pi i f t_k}\tilde{h}^*_{\ell m}(\lambda; f)\tilde{d}(f)\ , \tag{D4a}$$

$$U_{k,\ell m,\ell' m'}(\lambda) = \langle h_{\ell m}|h_{\ell' m'}\rangle_k\ , \tag{D4b}$$

$$V_{k,\ell m,\ell' m'}(\lambda) = \langle h^*_{\ell m}|h_{\ell' m'}\rangle_k\ . \tag{D4c}$$

Our order-of-magnitude estimate $\rho_{\rm guess}$ follows by approximating this likelihood expression, omtting V; eliding the impact of extrinsic angular factors F_+, Y_{lm}; and ignoring timing-related triangulation effects:

$$\ln\mathcal{L}_{\rm full} \simeq \frac{D_{\rm ref}}{D}Q(\lambda) - \frac{(D_{\rm ref}/D)^2}{4}U(\lambda) \tag{D5}$$

Maximizing this expression over the single remaining extrinsic parameter D produces an order-of-magnitude estimate for the maximum value:

$$\max_\theta \ln\mathcal{L}(\lambda,\theta) \simeq O(1)\frac{U}{Q^2} \tag{D6}$$

To account for all pertinent interferometers and modes symmetrically, we therefore define $\rho_{\rm guess}$ as follows:

$$(\rho_{\rm guess} * 2.3)^2 \simeq \sum_k\sum_{lm}\frac{|U_{k,lm,lm}(\lambda)|}{\max_| |Q_{k,lm}(t)|^2} \tag{D7}$$

In this expression, the factor of 2.3 has been chosen empirically, to produce estimates which correspond closely to cases with known ρ.

2. Choices for finite-precision floating point arithmetic and overflow

RIFT performs Monte Carlo integrals such as Eq. (3) over functions of order $e^{\rho^2/2}$. For loud signals, these large integrands can easily produce numerical overflow. For example, since a conventional 64-bit double-precision floating point number can express numbers between $\simeq e^{\pm 308}$, while a conventional 128-bit quad-precision floating number can express numbers over roughly twice that dynamic range, a source with amplitude louder than $\rho > \sqrt{2\ln(10)\times 308} \simeq 37.5$ (for single precision) or 53 (for quad precision) would produce a peak likelihood $\mathcal{L}_{\rm full}(\lambda,\theta)$ which overflows the precision of available arithmetic.

RIFT's Monte Carlo integration suite offers a range of solutions, balancing stability against speed. The two new integrators (GMM and AC) both can operate in a conventional overflow-protected mode, where all likelihoods are expressed as logs and all sums appearing in integrals like Eq. (3) are performed via the "logsumexp" function $g(\mathbf{x}) = \ln\sum_k e^{x_k}$. This stability comes at increased cost, primarily for the AC integrator which may need to transfer data between the CPU and GPU to perform this calculation. For most soures with modest amplitudes, however, RIFT can safely operate all its integrators with raw floating point numbers. To mitigate the impact of overflow, the user can choose to offset the floating point precision window, multiplying the likelihoods by a factor $e^{-\mathcal{O}}$. Customarily, we choose $\mathcal{O} \simeq \rho^2_{\rm hint}/2 - O({\rm few})$ or $\mathcal{O} \simeq \rho^2_{\rm guess}/2 - O({\rm few})$, to ensure evaluations in the support of the posterior avoid overflow. This workaround allows us to mildly stretch the window available for analysis with raw floating point operations. The pertinent limits for CPU-enabled operation of all our Monte Carlo integrators in raw floating-point mode, both in CIP and ILE, are usually set by quad-precision floating point arithmetic. However, when using GPU acceleration, the AC integrator is currently limited by double-precision arithmetic, a constraint which limits GPU-accelerated AC integration in ILE with raw floating-point numbers to signals of $\rho \lesssim 35$.

3. More flexible sampling models

As described in the text, our default extrinsic integration strategy *does not adapt* in several dimensions, limiting adaptation usually to sky location. This brute-force approach ensures ILE and RIFT will correctly cover the complex, correlated, often multimodal extrinsic posteriors arising ubiquitously for weak sources. For strong sources, however, our brute-force approach becomes untenable. Rather, to have any chance to find the small fraction $\simeq \rho^{-d_{\rm eff}}$ of the prior extrinsic volume where the posterior has support, where $d_{\rm eff}$ counts the number of extrinsic dimensions, we must adapt our sampling distribution in all dimensions simultaneously, using well-chosen coordinates.

For these reasons, four the loudest signals, we recommend GMM sampling, using sky- and phase-rotated coordinates, with distance marginalization.

APPENDIX E: TARGETS FOR FUTURE IMPROVEMENT

While we've substantially extneded RIFT relative to the O2 and O3 editions, RIFT could be easily improved in several ways.

1. Conventional convergence criteria

While most other inference codes have standardized on a target $n_{\rm eff,var}$, RIFT's hodepodge of convergence tests and diagnostics can produce uneven-quality posteriors over parameter space. We should report and use evidence-based convergence diagnostics for the iterate-to-convergence step, and consistently report $n_{\rm eff,var}$ from CIP at all stages. Both of these updates require architectural changes: our pipeline currently only passes samples to our convergence tests, not evidences (or evidence histories).

RIFT should also adopt a much longer, user-selected iterate-to-convergence cap. Our experience suggests that 10 iterations will be more than enough; if more iterations are required, the user should reconsider their choices, as they've probably made an error or adopted options that are poorly suited to their problem. However, most end-users want a black-box framework which will iterate to convergence no matter how long it takes.

2. Miscellaneous technical improvements

Better coordinates: Our interpolations and thus RIFT can be prone to under-exploring regions near the hard $q \simeq 1$ boundary. Initial grids and mass ratio coordinates that further emphasize this region should be explored.
Better integration (general): Two of our adaptive Monte Carlo integrators (AC and default) adopt largely ad hoc choices for the number of sampling bins (i.e., 100 bins for each adaptive coordinate). This arbitrary dimension-independent choice places severe limits on our ability to adaptively sample in many dimensions d_a. The GMM integrators adopt ad-hoc choices for the number of components, and those components are initiallzed randomly without information deduced from previous analysis or function data. For example, the adaptive CIP integrators are re-initialized and independently adapt for each iteration and for each worker, not efficiently exploiting the many previous iterations to initialize an adaptive sampler. Particularly for the GMM sampler and during the convergence phase, such initialization could help improve convergence.
Better integration (ILE): Despite heavy use of GPU optimization, our Monte Carlo integration of the extrinsic likelihood could be substantially improved. For example, recent work [105] strongly suggests that the posterior (and hence our Monte Carlo integration) can be substantially simplified by suitable coordinates. They demonstrate that careful use of reference frequency, polarization coordinate, and emission polar angle can dramatically simplify the phase posterior. By contrast, we're presently sampling uniformly over these two angles, introducing substantial inefficnecy at high amplitude. ILE integration generally only adapts in a small subset of the available dimensions.
Integration target (CIP,convergence): The integration sample size target n_{eff} for each individual CIP worker and the overall output isn't self-consistently chosen with the target accuracy threshold used to assess convergence. The fiducial threshold of 10^{-2} applied to Eq. (8), or equivalent thresholds applied to other metrics like the JS divergence, should be user-adjustable, using some clearly understood empirical relationship between this threshold and a target accuracy goal for the final posterior. As several other groups have adopted JS divergence to assess convergence, we should adjust our convergence criteria to use this diagnostic. The target accuracy threshold should be adaptively tightened, and the number of raw Monte Carlo samples N increased, over the course of an analysis, rather than fix the threshold and maximum number of evaluations for iterations.
Overall infrastructure: ILE and CIP should use a task-based parallelism architecture, offloading startup costs and management to the scheduler and better-enabling ongoing use of resources. Too often nodes are under-used for ILE integration, while too few CIP instances are instantiated given integration needs.

We should generalize our approach to allow for conditional priors, such as a mass-dependent prior on the tidal deformability λ or mass ratio.

3. Caveats and stability considerations

RIFT's code settings and use cases are carefully tailored to match the capabilities of the fitting and Monte

Carlo integration algorithms used. Previously in Section III F, we described several inefficiencies and limitations of the code elements used in RIFT's O3-era operation. In this section, we briefly highlight ways in which the *new* components of RIFT could be misused or misbehave, as an aid to diagnosing potential analysis problems.

GMM integrator stability with correlated sampling: The GMM integrator was designed to adapt efficiently to *correlated* dimensions, including multiple correlated components. However, this flexibility if employed unchecked can easily wildly overfit, with the EM algorithm producing singular covariance matricies. For this reason, at present we hardcode the number of components for both ILE and CIP, depending on the use case. Additionally, for ILE we only allow pairwise correlation, using physics-based motivation.

Since our production configuration uses correlated GMM sampling, we emphasize the ways in which this configuration can misbehave. For low-mass and high-mass ratio binaries, the strongly correlated posteriors can produce singular covariance matricies, requiring the sampler to reset. If this sampler reset occurs at an inopportune time, just prior to the end of a run, an individual worker's output is more likely to be "spoiled".

[1] LIGO Scientific Collaboration, J. Aasi, B. P. Abbott, R. Abbott, T. Abbott, M. R. Abernathy, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., Classical and Quantum Gravity **32**, 074001 (2015), 1411.4547.

[2] T. Accadia and et al, Journal of Instrumentation **7**, P03012 (2012), URL `http://iopscience.iop.org/1748-0221/7/03/P03012`.

[3] F. Acernese, M. Agathos, K. Agatsuma, D. Aisa, N. Allemandou, A. Allocca, J. Amarni, P. Astone, G. Balestri, G. Ballardin, et al., Classical and Quantum Gravity **32**, 024001 (2015), 1408.3978.

[4] T. Akutsu, M. Ando, K. Arai, Y. Arai, S. Araki, A. Araya, N. Aritomi, Y. Aso, S. Bae, Y. Bae, et al., Progress of Theoretical and Experimental Physics **2021**, 05A101 (2021), 2005.05574.

[5] The LIGO Scientific Collaboration and the Virgo Collaboration, Phys. Rev. Lett. **16**, 061102 (2016).

[6] B. Abbott et al. (The LIGO Scientific Collaboration and the Virgo Collaboration), PRX **6**, 041015 (2016), 1606.04856, URL `https://journals.aps.org/prx/abstract/10.1103/PhysRevX.6.041015`.

[7] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., Phys. Rev. Lett. **119**, 161101 (2017).

[8] R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. X. Adhikari, V. B. Adya, C. Affeldt, et al., ApJL **915**, L5 (2021), 2106.15163.

[9] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, et al., Available as LIGO-P2000318 (????), URL `https://dcc.ligo.org/LIGO-P2000318`.

[10] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, et al., Available as LIGO-P2100063 (2021), URL `https://dcc.ligo.org/LIGO-P2100063/public`.

[11] B. P. Abbott, R. Abbott, T. D. Abbott, M. R. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, et al., Living Reviews in Relativity **19**, 1 (2016).

[12] B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. X. Adhikari, V. B. Adya, et al., Physical Review Letters **118**, 221101 (2017), 1706.01812.

[13] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., Phys. Rev. Lett. **119**, 141101 (2017), 1709.09660.

[14] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., ApJL **851**, L35 (2017).

[15] The LIGO Scientific Collaboration, The Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., PRX **9**, 031040 (2018).

[16] C. Pankow, P. Brady, E. Ochsner, and R. O'Shaughnessy, Phys. Rev. D **92**, 023002 (2015), URL `http://adsabs.harvard.edu/abs/2015PhRvD..92b3002P`.

[17] J. Lange, R. O'Shaughnessy, and M. Rizzo, Submitted to PRD; available at arxiv:1805.10457 (2018).

[18] J. Veitch, V. Raymond, B. Farr, W. M. Farr, P. Graff, S. Vitale, B. Aylott, K. Blackburn, N. Christensen, M. Coughlin, et al., Phys. Rev. D **91**, 042003 (2015), URL `http://link.aps.org/doi/10.1103/PhysRevD.91.042003`.

[19] M. Hannam, P. Schmidt, A. Bohé, L. Haegel, S. Husa, F. Ohme, G. Pratten, and M. Pürrer, Phys. Rev. Lett. **113**, 151101 (2014).

[20] S. Khan, K. Chatziioannou, M. Hannam, and F. Ohme, Phys. Rev. D **100**, 024059 (2019), 1809.10113.

[21] A. Bohé, L. Shao, A. Taracchini, A. Buonanno, S. Babak, I. W. Harry, I. Hinder, S. Ossokine, M. Pürrer, V. Raymond, et al., Phys. Rev. D **95**, 044028 (2017), 1611.03703.

[22] V. Varma, S. E. Field, M. A. Scheel, J. Blackman, D. Gerosa, L. C. Stein, L. E. Kidder, and H. P. Pfeiffer, Physical Review Research **1**, 033015 (2019), 1905.09300.

[23] G. Pratten, C. García-Quirós, M. Colleoni, A. Ramos-Buades, H. Estellés, M. Mateu-Lucena, R. Jaume, M. Haney, D. Keitel, J. E. Thompson, et al., Phys. Rev. D **103**, 104056 (2021), 2004.06503.

[24] S. Ossokine, A. Buonanno, S. Marsat, R. Cotesta, S. Babak, T. Dietrich, R. Haas, I. Hinder, H. P. Pfeiffer, M. Pürrer, et al., Phys. Rev. D **102**, 044055 (2020), 2004.09442.

[25] G. Ashton, M. Hübner, P. D. Lasky, C. Talbot, K. Ackley, S. Biscoveanu, Q. Chu, A. Divakarla, P. J. Easter,

B. Goncharov, et al., ApJS **241**, 27 (2019), 1811.02042.

[26] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, et al., Phys. Rev. Lett. **125**, 101102 (2020).

[27] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, et al., ApJL **900**, L13 (2020), 2009.01190.

[28] The LIGO Scientific Collaboration, the Virgo Collaboration, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. X. Adhikari, et al., Physical Review X **11**, 021053 (2021), 2010.14527.

[29] N. Christensen and R. Meyer, Rev. Mod. Phys. **94**, 025001 (2022), URL `https://link.aps.org/doi/10.1103/RevModPhys.94.025001`.

[30] D. Wysocki, R. O'Shaughnessy, J. Lange, and Y.-L. L. Fang, Phys. Rev. D **99**, 084026 (2019), 1902.04934.

[31] A. Z. Jan, A. B. Yelikar, J. Lange, and R. O'Shaughnessy, Phys. Rev. D **102**, 124069 (2020), 2011.03571.

[32] D. Wysocki, R. O'Shaughnessy, L. Wade, and J. Lange, Submitted to PRD; available as arxiv:2001.01747 (2020), URL `https://arxiv.org/abs/2001.01747`.

[33] M. Al-Mamun, A. W. Steiner, J. Nättilä, J. Lange, R. O'Shaughnessy, I. Tews, S. Gandolfi, C. Heinke, and S. Han, Phys. Rev. Lett. **126**, 061101 (2021), 2008.12817.

[34] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, V. B. Adya, et al., Phys. Rev. D **102**, 043015 (2020).

[35] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, and et al, Classical and Quantum Gravity **37**, 045006 (2020), 1908.01012.

[36] R. Udall, J. Brandt, G. Manchanda, A. Arulanandan, J. Clark, J. Lange, R. O'Shaughnessy, and L. Cadonati, arXiv e-prints arXiv:2110.10243 (2021), 2110.10243.

[37] N. J. Cornish, Phys. Rev. D **104**, 104054 (2021), URL `https://link.aps.org/doi/10.1103/PhysRevD.104.104054`.

[38] N. J. Cornish, Phys. Rev. D **103**, 104057 (2021), URL `https://link.aps.org/doi/10.1103/PhysRevD.103.104057`.

[39] S. Morisaki and V. Raymond, Phys. Rev. D **102**, 104020 (2020), URL `https://link.aps.org/doi/10.1103/PhysRevD.102.104020`.

[40] A. B. Yelikar, V. Delfavero, and R. O'Shaughnessy, arXiv e-prints arXiv:2301.01337 (2023), 2301.01337.

[41] L. J. O'Shaughnessy, R. and others (????), URL `http://git.ligo.org/rapidpe-rift/rift`.

[42] R. Abbott, T. D. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. X. Adhikari, V. B. Adya, C. Affeldt, M. Agathos, et al., SoftwareX **13**, 100658 (2021), 1912.11716.

[43] G. P. Lepage, Newman Laboratory of Nuclear studies report CLNS-80/447 (1980), URL `https://lib-extopc.kek.jp/preprints/PDF/1980/8006/8006210.pdf`.

[44] G. P. Lepage, Journal of Computational Physics **439**, 110386 (2021), 2009.05112.

[45] Press, Teukolsky, Flannery, and Vetterling, *Numerical recipies* (????), URL `http://www.nr.com`.

[46] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Journal of Machine Learning Research **12**, 2825 (2011), URL `https://arxiv.org/abs/1201.0490`.

[47] M. Delfavero, *Assessing the Convergence of Iterative Parameter Estimation* (2019), mS thesis for RIT, available as https://scholarworks.rit.edu/theses/10153/.

[48] R. O'Shaughnessy, J. Blackman, and S. Field, CQG (2017), URL `http://iopscience.iop.org/article/10.1088/1361-6382/aa7649`.

[49] S. Husa, S. Khan, M. Hannam, M. Pürrer, F. Ohme, X. J. Forteza, and A. Bohé, Phys. Rev. D **93**, 044006 (2016), 1508.07250.

[50] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Pürrer, X. J. Forteza, and A. Bohé, Phys. Rev. D **93**, 044007 (2016), 1508.07253.

[51] R. Cotesta, A. Buonanno, A. Bohé, A. Taracchini, I. Hinder, and S. Ossokine, Phys. Rev. D **98**, 084028 (2018), 1803.10701.

[52] S. Cook, A. Gelman, and D. Rubin, Journal of Computational and Graphical Statistics **15**, 675 (2006), URL `https://www.tandfonline.com/doi/abs/10.1198/106186006X136976`.

[53] T. Sidery, B. Aylott, N. Christensen, B. Farr, W. Farr, F. Feroz, J. Gair, K. Grover, P. Graff, C. Hanna, et al., Phys. Rev. D **89**, 084060 (2014), 1312.6013.

[54] A. Z. Jan, A. B. Yelikar, J. Lange, and R. O'Shaughnessy, Phys. Rev. D **102**, 124069 (2020), 2011.03571.

[55] D. Thain, T. Tannenbaum, and M. Livny, Concurrency - Practice and Experience **17**, 323 (2005).

[56] B. Bockelman, M. Livny, B. Lin, and F. Prelz, Journal of Computational Science (2020), ISSN 1877-7503, URL `http://www.sciencedirect.com/science/article/pii/S1877750320305147`.

[57] B. Bockelman, T. Cartwright, J. Frey, E. M. Fajardo, B. Lin, M. Selmeci, T. Tannenbaum, and M. Zvada, Journal of Physics: Conference Series **664**, 062003 (2015), URL `http://stacks.iop.org/1742-6596/664/i=6/a=062003`.

[58] D. Williams, Available at https://git.ligo.org/asimov/asimov (????), URL `https://git.ligo.org/asimov/asimov`.

[59] H. Cho, E. Ochsner, R. O'Shaughnessy, C. Kim, and C. Lee, Phys. Rev. D **87**, 02400 (2013), URL `http://xxx.lanl.gov/abs/arXiv:1209.4494`.

[60] R. O'Shaughnessy, B. Farr, E. Ochsner, H.-S. Cho, V. Raymond, C. Kim, and C.-H. Lee, Phys. Rev. D **89**, 102005 (2014), URL `http://link.aps.org/doi/10.1103/PhysRevD.89.102005`.

[61] E. Lee, S. Morisaki, and H. Tagoshi, Phys. Rev. D **105**, 124057 (2022), URL `https://link.aps.org/doi/10.1103/PhysRevD.105.124057`.

[62] L. Blanchet, Living Reviews in Relativity **17**, 2 (2014), 1310.1528, URL `http://xxx.lanl.gov/abs/arXiv:1310.1528`.

[63] K. G. Arun, A. Buonanno, G. Faye, and E. Ochsner, Phys. Rev. D **79**, 104023 (2009), 0810.5336.

[64] E. Poisson and C. M. Will, Phys. Rev. D **52**, 848 (1995).

[65] I. M. Romero-Shaw, C. Talbot, S. Biscoveanu, V. D'Emilio, G. Ashton, C. P. L. Berry, S. Coughlin, S. Galaudage, C. Hoy, M. Hübner, et al., MNRAS **499**, 3295 (2020), 2006.00714.

[66] D. Gerosa, M. Mould, D. Gangardt, P. Schmidt, G. Pratten, and L. M. Thomas, Physical Review D **103** (2021), ISSN 2470-0029, URL `http://dx.doi.org/10.1103/PhysRevD.103.064067`.

[67] C. Henshaw, R. O'Shaughnessy, and L. Cadonati, Classical and Quantum Gravity **39**, 125003 (2022), 2201.05220.

[68] V. De Renzis, D. Gerosa, G. Pratten, P. Schmidt, and M. Mould, arXiv e-prints arXiv:2207.00030 (2022), 2207.00030.

[69] D. Wysocki, R. O'Shaughnessy, L. Wade, and J. Lange, arXiv e-prints arXiv:2001.01747 (2020), 2001.01747.

[70] L. Breiman, Machine Learning **45**, 5 (2001).

[71] K. P. Murphy, *Machine Learning: A Probabilistic Perspective* (The MIT Press, 2012), ISBN 0262018020, 9780262018029.

[72] P. Geurts, D. Ernst, and L. Wehenkel, Machine learning **63**, 3 (2006).

[73] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning* (The MIT Press, 2006).

[74] M. Bauer, M. van der Wilk, and C. E. Rasmussen, arXiv e-prints (2016), 1606.04820.

[75] M. A. Álvarez, D. Luengo, M. K. Titsias, and N. D. Lawrence, arXiv e-prints (2009), 0912.3268.

[76] J. Hensman, N. Fusi, and N. D. Lawrence, arXiv e-prints (2013), 1309.6835.

[77] M. Jankowiak, G. Pleiss, and J. R. Gardner, arXiv e-prints (2019), 1910.07123.

[78] H. H. Rosenbrock, The Computer Journal **3**, 175 (1960).

[79] A. Fowlie, W. Handley, and L. Su, MNRAS **497**, 5256 (2020), 2006.03371.

[80] A. Dempster, N. Laird, and D. Rubin, Journal of the Royal Statistical Society, Series B **39**, 1 (1977).

[81] G. M.R. and Y. Chen, Foundations and Trends in Signal Processing **4**, 223 (2010).

[82] G. McLachlan and T. Krishnan, *The EM Algorithm and Extensions* (John Wiley and Sons, 2008).

[83] G. Ashton and C. Talbot, MNRAS **507**, 2037 (2021), 2106.08730.

[84] E. Thrane and C. Talbot, PASA **37**, e036 (2020).

[85] S. Morisaki, LIGO DCC T2100485 (2021), URL `https://dcc.ligo.org/LIGO-T2100485`.

[86] V. Delfavero, R. O'Shaughnessy, D. Wysocki, and A. Yelikar, arXiv e-prints arXiv:2107.13082 (2021), 2107.13082.

[87] C. A. Rose, V. Valsan, P. R. Brady, S. Walsh, and C. Pankow, arXiv e-prints arXiv:2201.05263 (2022), 2201.05263.

[88] The LIGO Scientific Collaboration, the Virgo Collaboration, B. P. Abbott, R. Abbott, T. D. Abbott, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, et al., Phys. Rev. Lett. **121**, 161101 (2018).

[89] C. D. Capano, I. Tews, S. M. Brown, B. Margalit, S. De, S. Kumar, D. A. Brown, B. Krishnan, and S. Reddy, Nature Astronomy **4**, 625 (2020), 1908.10352.

[90] P. Landry and R. Essick, Phys. Rev. D **99**, 084049 (2019), 1811.12529.

[91] I. Legred, K. Chatziioannou, R. Essick, S. Han, and P. Landry, Phys. Rev. D **104**, 063003 (2021), 2106.05313.

[92] T. Gorda, O. Komoltsev, and A. Kurkela, arXiv e-prints arXiv:2204.11877 (2022), 2204.11877.

[93] B. Abbott et al. (The LIGO Scientific Collaboration and the Virgo Collaboration), Phys. Rev. D **94**, 064035 (2016), URL `http://link.aps.org/doi/10.1103/PhysRevD.94.064035`.

[94] LIGO Scientific Collaboration, Available as LIGO-P1900124 from dcc.ligo.org (????), URL `https://dcc.ligo.org/LIGO-P1900124/public`.

[95] M. Mateu-Lucena, S. Husa, M. Colleoni, H. Estellés, C. García-Quirós, D. Keitel, M. de Lluc Planas, and A. Ramos-Buades, arXiv e-prints arXiv:2105.05960 (2021), 2105.05960.

[96] A. Vajpeyi, R. Smith, and E. Thrane, arXiv e-prints arXiv:2203.13406 (2022), 2203.13406.

[97] H. S. Chia, S. Olsen, J. Roulet, L. Dai, T. Venumadhav, B. Zackay, and M. Zaldarriaga, Phys. Rev. D **106**, 024009 (2022), 2105.06486.

[98] M. Dax, S. R. Green, J. Gair, J. H. Macke, A. Buonanno, and B. Schölkopf, Phys. Rev. Lett. **127**, 241103 (2021), 2106.12594.

[99] A. J. K. Chua and M. Vallisneri, Phys. Rev. Lett. **124**, 041102 (2020), 1909.05966.

[100] A. Delaunoy, A. Wehenkel, T. Hinderer, S. Nissanke, C. Weniger, A. R. Williamson, and G. Louppe, arXiv e-prints arXiv:2010.12931 (2020), 2010.12931.

[101] H. Gabbard, C. Messenger, I. S. Heng, F. Tonolini, and R. Murray-Smith, Nature Physics **18**, 112 (2022), 1909.06296.

[102] P. G. Krastev, K. Gill, V. A. Villar, and E. Berger, Physics Letters B **815**, 136161 (2021), 2012.13101.

[103] L. Kish, *Survey sampling* (John Wiley & Sons, Inc, London, 1965), ISBN 0-471-10949-5.

[104] W. M. Farr, Research Notes of the American Astronomical Society **3**, 66 (2019), 1904.10879.

[105] J. Roulet, S. Olsen, J. Mushkin, T. Islam, T. Venumadhav, B. Zackay, and M. Zaldarriaga, arXiv e-prints arXiv:2207.03508 (2022), 2207.03508.