PyROQ: a Python-based Reduced Order Quadrature Building Code for Fast Gravitational Wave Inference

Hong Qi[1,*] and Vivien Raymond[1]

[1] *School of Physics and Astronomy, Cardiff University, Cardiff, UK, CF24 3AA*

(Dated: September 30, 2020)

The low frequency sensitivity on the orders of a few Hz in future gravitational-wave observatories will enable the detection of gravitational wave signals of very long duration. The runtime of parameter estimation with these long waveforms can be months even years, which make it impractical with existing Bayesian inference pipelines. Reduced order modelling and reduced order quadrature integration rules have recently been exploited as promising techniques that can greatly reduce parameter estimation computational costs. We describe a Python-based reduced order quadrature building code named PyROQ that builds the reduced order quadrature data needed to accelerate the parameter estimation of gravitational waves. The infrastructure of the code is also directly applicable to gravitational wave inference for space-borne gravitational-wave detectors such as the Laser Interferometer Space Antenna (LISA). In addition, the techniques are broadly applicable to other research fields where fast Bayesian analysis is necessary.

I. INTRODUCTION

Gravitational-wave observatories such as LIGO [1], Virgo [2], and KAGRA [3] are now frequently detecting signals from binary black holes [4–7], binary neutron stars [8] and neutron star - black hole binaries [8–10]. Their sensitivity keeps improving, with new generations of detectors expected to come online in the near future, such as A+, Voyager [11–13], the Einstein Telescope [14], Cosmic Explorer [15]. The current low frequency limit of gravitational wave detector is about 10 Hz [16], but improvements on the current facilities mean that a lower limit of 5 Hz is feasible, and lower limit of 1 Hz is envisioned for future generations. In order to take full advantage of this sensitivity, waveform signal models are required to span the whole sensitivity window, starting from the lowest frequency possible. It can take a prohibitively long time to compute the likelihood for such signals, unless efficient representation techniques are used, such as Singular Value Decomposition [17–19], Reduced Order Modelling (ROM) [20–23], and Relative Binning [24]. We focus in this work on the application of ROM techniques as applied to the calculation of the likelihood integrand, building Reduced Order Quadrature (ROQ) rules. On the longest waveform signal used, speed-ups of many orders of magnitude are possible. The main computational cost is moved in an off-line basis building stage, which can be done once per waveform model and ahead of time. However, as new waveform models are being developed at a faster rate [25–28], in order to fully exploit those even more complex models, a user-friendly and easy-to-use ROQ basis building code developed in Python as the language of choice of the LIGO community is desirable.

In addition, the most advanced waveform models that are able to extract the most astrophysical information from the gravitational-wave signals are also the most computationally expensive. Many of the most impactful studies will involve waveform models with more detailed physics, such as eccentricity or higher-mode effects. The high computational costs of parameter estimation with these long waveforms with more physics make it almost impossible with standard Bayesian inference pipelines. For example, it can take several months to finish a parameter estimation run on a BNS signal using standard methods, whereas techniques like ROQ rule can reduce the runtime to about 24 hours [22, 29] and under certain conditions a couple of hours [30]. Furthermore, in practice reruns are often needed due to fine-tuning and errors, as well as using different stages and types of calibrated data such as cleaned data [4].

In this paper, we present a method that can be used to search for reduced bases and construct ROQ data for gravitational waveforms and inference on gravitational waves using a modified algorithm to the greedy method that was adopted by the C++ code GreedyCpp [20, 23]. We also showcase the implementation of the method, the PyROQ code written in Python. We show that even though Python is in general slower than C++ [31], with the modified algorithm PyROQ is comparable in speed to the GreedyCpp in the ROQ data constructions.

The paper is organized as follows. In Section II we introduce the basics of ROQ rules for gravitational wave parameter estimation, as well as the algorithm we use to produce the ROQ data. In Section III we describe our validation of the PyROQ code with an application to IMRPhenomPv2 [32, 33] waveform, checking against the ROQ from GreedyCpp. Using the Bilby pipeline, in Section IV we show the likelihood comparisons and the posterior comparisons, respectively, for simulated NSBH injections with the standard method and the ROQ method where the ROQ data were constructed from PyROQ; we also compare the inference on a few real BBH detections in LIGO data using the ROQs constructed by PyROQ and GreedyCpp; we also briefly introduced the applica-

* hong.qi@ligo.org

tion in GW inference with LISA with supermassive black hole binaries (SMBHBs). At last in Section V, we summarize what has been presented in the paper.

The PyROQ code is publicly available at http://pypi.org/project/PyROQ.

II. METHODOLOGY

A. Basics of inference on gravitational wave

Inference on gravitational waves is also known as gravitational wave parameter estimation. It provides the posterior probability density functions (PDFs) of a set of parameters $\vec{\Lambda}$ that characterize the source properties and are used to model the gravitational wave signal, $h(\vec{\Lambda})$, from the detector's collected strain data d. A widely accepted method to calculate the PDFs is Bayesian inference in which Bayes theorem is used

$$p(\vec{\Lambda}|d) = \frac{\mathcal{P}(\vec{\Lambda})\ \mathcal{L}(d|\vec{\Lambda})}{E(d)}\,, \tag{1}$$

where $p(\vec{\Lambda}|d)$ is the posterior probability of the model parameters given the data, $\mathcal{P}(\vec{\Lambda})$ is the prior probability on the model parameters, $\mathcal{L}(d|\vec{\Lambda})$ is the likelihood of the data at given model parameters, and $E(d)$ is the model evidence which describes the probability of the data given the model. The evidence is the same for all waveform models for a certain hypothesis, thus it enters only as an overall scaling factor in parameter estimation. The prior depends on what one believes in advance. Therefore, the most computationally consuming part is the likelihood.

Suppose the detector data d is composed of a GW signal $h(\vec{\Lambda}_{\texttt{true}})$ and noise n, i.e., $d = h(\vec{\Lambda}_{\texttt{true}}) + n$. The log-likelihood function can be computed by

$$\log \mathcal{L}(d|\vec{\Lambda}) = -\frac{1}{2}(d - h(\vec{\Lambda}), d - h(\vec{\Lambda}))\,, \tag{2}$$

where (a, b) is an *overlap* integral:

$$(d, h(\vec{\Lambda})) = 4\Re\ \Delta f \sum_{k=1}^{L} \frac{\tilde{d}^*(f_k)\tilde{h}(f_k;\vec{\Lambda})}{S_n(f_k)}\,. \tag{3}$$

Here $\tilde{d}(f_k)$ and $\tilde{h}(f_k;\vec{\Lambda})$ are the discrete Fourier transforms at frequencies $\{f_k\}_{k=1}^{L}$ and $S_n(f_k)$ is the power spectral density (PSD) of the detector's noise. For a given observation time $T = 1/\Delta f$ and detection frequency window from $f_{\texttt{high}}$ to $f_{\texttt{low}}$, there are $L \sim \texttt{int}\left(\left[f_{\texttt{high}} - f_{\texttt{low}}\right] T\right)$ sampling points in (3). When L is large, and $\vec{\Lambda}$ gets larger as more physics is considered in new waveform models and must be sampled extensively, the evaluation of the model at each f_k and the repeated evaluation of the likelihood have become bottlenecks in gravitational wave parameter estimation.

B. ROQ rules for gravitational wave models

Here we describe succinctly the procedure of ROQ rules building, following the conventions in Ref. [22]. The model of a gravitational-wave strain signal $h(f)$ detected by a ground-based interferometer and its overlap with itself have their empirical interpolants, which can be written as (cf. Eq. (7) of Ref. [22])

$$\tilde{h}_{\rm A}(f_i;\vec{\lambda}) \approx \sum_{j=1}^{N_{\texttt{L}}} B_j(f_i)\tilde{h}_{\rm A}(F_j;\vec{\lambda})\,, \quad \text{with A} \in \{+,\times\}\,, \tag{4a}$$

$$\Re\left[\tilde{h}_{\rm A}(f_i;\vec{\lambda})\tilde{h}^*_{\rm B}(f_i;\vec{\lambda})\right] \approx \sum_{k=1}^{N_{\texttt{Q}}} C_k(f_i)\Re\left[\tilde{h}_{\rm A}(\mathcal{F}_k;\vec{\lambda})\tilde{h}^*_{\rm B}(\mathcal{F}_k;\vec{\lambda})\right]\,, \quad \text{with A, B} \in \{+,\times\}\,, \tag{4b}$$

that accurately approximate both the polarization states and their products that are required to compute the log-likelihood in Eq. (2). The $\{B_j\}_{j=1}^{N_{\texttt{L}}}$ is the reduced basis (RB) for the two polarization states and $\{C_k\}_{k=1}^{N_{\texttt{Q}}}$ is the reduced basis for the inner product of the waveform with itself. The nodes $\{F_j\}_{j=1}^{N_{\texttt{L}}}$ are the *empirical interpolation nodes*, which are uniquely selected to produce accurate waveform interpolation with the basis $\{B_j\}_{j=1}^{N}$. The $\tilde{h}_{\rm A}(\vec{\lambda};F_j)$ is an *empirical interpolant* of the A-polarization state at those empirical nodes. Similarly for the products of polarization states. Substituting the approximation (4) into (3) gives a *reduced order quadrature* (ROQ) rule. Sec. II C describes the algorithms we use to build the bases in (4).

All extrinsic parameters (defined here as the sky-position RA and Dec, the polarisation angle ψ, the distance D and the coalescence time t_c), do not affect the frequency evolution of the binary and simply scale the inner product, thereby sharing the same ROQs, except for the coalescence time t_c. The coalescence time does require special treatment: following previous work (see [21, 22], we build a unique set of ROQ weights for equally

spaced values of t_c (see below).

The full likelihood can be approximated by the ROQ likelihood, which can be decomposed into parts [22]

$$2\log\mathcal{L} \approx 2F_+(d,h_+)_{\rm ROQ} + 2F_\times(d,h_\times)_{\rm ROQ} - |F_+|^2\,(h_+,h_+)_{\rm ROQ} - |F_\times|^2\,(h_\times,h_\times)_{\rm ROQ} - 2F_+F_\times(h_+,h_\times)_{\rm ROQ} - (d,d)\,,$$

where the linear part and its corresponding ROQ weights

$$(d, h_{\rm A}(\vec{\lambda}))_{\rm ROQ} \approx \sum_{j=1}^{N_{\rm L}} \omega_j(t_c)\tilde{h}_{\rm A}(F_j;\vec{\lambda})\,, \tag{5a}$$

$$\omega_j(t_c) = 4\Re\ \Delta f \sum_{i=1}^{L} \frac{\tilde{d}^*(f_i)B_j(f_i)}{S_n(f_i)}{\rm e}^{-2\pi{\rm i}t_c f_i} \tag{5b}$$

and the quadratic part and its weights

$$(h_{\rm A}(\vec{\lambda}), h_{\rm B}(\vec{\lambda}))_{\rm ROQ} \approx \sum_{k=1}^{N_{\rm Q}} \psi_k \tilde{h}_{\rm A}(\mathcal{F}_k;\vec{\lambda})\tilde{h}^*_{\rm B}(\mathcal{F}_k;\vec{\lambda})\,, \tag{6a}$$

$$\psi_k = 4\Re\ \Delta f \sum_{i=1}^{L} \frac{C_k(f_i)}{S_n(f_i)}\,, \tag{6b}$$

Once the weights are computed, evaluating the ROQ likelihood only requires $N_{\rm L} + N_{\rm Q}$ terms, hence reducing the cost of (3) by a factor of $L/(N_{\rm L}+N_{\rm Q})$. The ROQ rules are similar to the standard evaluation pattern, thereby allowing existing codes to easily implement these tools.

The quality of a basis is measured by the *empirical interpolation errors*[20], denoted by $\sigma_{\rm EI}$'s, in the interpolation of millions of test waveforms with the basis.

C. Numerical algorithm of PyROQ

It is unnecessary to have a one-fits-all basis that can represent all the waveforms in the whole parameter space of a gravitational wave model, because the basis would be too large to save time for parameter estimation. Also, the searching pipelines (such as in [34, 35]) can estimate the chirp mass at 10% accuracy or higher precision so it is sufficient to use a basis built from smaller parameter space range. Therefore, before searching for reduced bases to build ROQ data, we first divide the entire parameter space of a waveform model into several chunks. The chunks are split based on the chirp mass values that correspond to a signal length T, whose values are traditionally 4 second, 8 second, ..., and 256 second in the integer power of base 2 [22]. We leave overlaps between adjacent chunks to cover the situation when the parameters of a signal are around the division boundaries of chirp mass. The reduced linear and quadratic bases are then searched for to build ROQ data for each of these chunks.

PyROQ makes significant modifications to the greedy algorithm of GreedyCpp, and exploited a different enrichment strategy. As a Python-based code, PyROQ has natural challenges in computing speed compared to the C++ based code GreedyCpp at the computer language level. If PyROQ simply copies GreedyCpp's algorithm, the ROQ data construction is expected to be slower. We need a code in Python to be at least as speedy if not faster. This urges the modifications in the algorithm of PyROQ.

Bear in mind that the basis construction is to find in the waveform space a set of basis elements that are most different from each other and can span waveform space. Note that the basis that can interpolate the chunk of waveforms is not unique, because the basis searching process is similar to choosing a basis of two elements for a 2-D vector space where there are many choices for the two elements. Also note that the order of the basis elements in the basis can be reordered and still represent test waveforms to the same accuracy (cite the linear interpolation equation), so it makes the basis searching process more flexible. The task then simplifies to find a basis out of infinite possible bases that are roughly the same size and can span the waveform space accurately.

Before we do any searches, it should be made clear that we aim at a reduced basis that can span the parameter space accurately enough for parameter estimation usages. Typically it requires that the maximum empirical interpolation error by the reduced basis for the waveforms in the parameter space of interest should be less than 5×10^{-7}. We use the letter ϵ for this threshold (maximum empirical interpolation error). This is equivalent to that the overlap between any represented waveform and its original waveform is larger than $1-2\times10^{-6}$. This can ensure that the fractional error in logarithmic likelihood is less than 10^{-6} [21, 36] which makes the parameter estimation with ROQ method yield the same posteriors as the standard non-ROQ method. Another constraint on the threshold ϵ comes from the waveform accuracy. The interpolation errors should be smaller than at least one order of magnitude of the accuracy of a waveform model. For example, if the waveform is accurate up to 10^{-8}, then the maximum empirical interpolation error ϵ of represented waveforms by the reduced basis should be less than 10^{-9} instead of 5×10^{-7}. In other words, the threshold ϵ (maximum empirical interpolation error) should be less than the minimum of 5×10^{-7} and one order of magnitude smaller than the waveform accuracy. One can always build ROQs more accurate than the min-

imum requirement at their interest, though it is not always necessary.

Algorithm 1 Greedy+Enrichment algorithm for reduced basis search

Input: $\{\mathbf{e}_{\text{corner},j}\}_{j=1}^{N_c}$, ϵ

Set $i = 0$
RB $= \{\mathbf{e}_{\text{corner},j}\}_{j=1}^{N_c}$
while $\sigma_{\text{EI}} \geq \epsilon/10$ **do** (Greedy loop)
    $i = i + 1$
    $\mathcal{T}_i$ = Generate N_{sub} random $\boldsymbol{\lambda} \in \mathcal{T}$
    $\boldsymbol{\lambda}_{new} = \text{argmax}_{\boldsymbol{\lambda}\in\mathcal{T}_i} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{P}_j h(\cdot;\boldsymbol{\lambda})\|^2$ (Gram-Schmidt using parallelization)
    $e_{i+1} = h(\cdot;\boldsymbol{\lambda}_{new}) - \mathcal{P}_i h(\cdot;\boldsymbol{\lambda}_{new})$
    $e_{i+1} = e_{i+1}/\|e_{i+1}\|$
    RB = RB $\cup\, e_{i+1}$
    $\mathcal{T}_j$ = Generate 1000 random $\boldsymbol{\lambda} \in \mathcal{T}$
    $\sigma_{\text{EI}} = \max_{\boldsymbol{\lambda}\in\mathcal{T}_j} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{I}_M(h(\cdot;\boldsymbol{\lambda}))\|^2$ (Using parallelization)
end while
while $\mathcal{T}_{\text{outliers}} \neq \varnothing$ **do** (Enrichment loop)
    $\mathcal{T}_k$ = Generate 1000000 random $\boldsymbol{\lambda} \in \mathcal{T}$ (Using parallelization for this line and the next)
    $\mathcal{T}_{\text{outliers}} = \boldsymbol{\lambda} \in \mathcal{T}_k$ and $\|h(\cdot;\boldsymbol{\lambda}) - \mathcal{I}_M(h(\cdot;\boldsymbol{\lambda}))\|^2 > \epsilon$
    while $\sigma_{\text{EI}} \geq \epsilon$ **do**
        i=i+1
        $\sigma_{\text{EI}} = \max_{\boldsymbol{\lambda}\in\mathcal{T}_{\text{outliers}}} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{I}_M(h(\cdot;\boldsymbol{\lambda}))\|^2$
        $\boldsymbol{\lambda}_{new} = \text{argmax}_{\boldsymbol{\lambda}\in\mathcal{T}_j} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{I}_M(h(\cdot;\boldsymbol{\lambda}))\|^2$
        $e_{i+1} = h(\cdot;\boldsymbol{\lambda}_{new}) - \mathcal{P}_i h(\cdot;\boldsymbol{\lambda}_{new})$
        $e_{i+1} = e_{i+1}/\|e_{i+1}\|$
        RB = RB $\cup\, e_{i+1}$
        $\mathcal{T}_{\text{outliers}} = \mathcal{T}_{\text{outliers}} - \boldsymbol{\lambda}_{new}$
    end while
end while

Output: RB $\{e_i\}_{i=1}^m$

The reduced order quadrature rule is trained on dynamic and relatively small training sets $\mathcal{T}_i$ of waveforms, which are equivalent to a large and dense training set $\mathcal{T}$, using an algorithm implemented in Python. The algorithm is shown in Algorithm 1, where for the notations that are not specified they follow the work in [20]. The initiative basis elements are straightforward to find. Because the basis must interpolate the waveforms at the corners of waveform space, we use those of the lowest and highest boundary values in the corners of the parameter space as the initial basis waveforms, where we extract out the first few basis elements using normalization and Gram-Schmidt. The number of corner initial waveforms is N_c. Our practises show that using the corner waveforms rather than random waveforms as initiative basis elements, the basis size can be reduced by $\sim 20\%$. We proceed with small training waveform sets that each has waveforms on the order of $N_{sub} = 10^3 \sim 10^4$. In each small set, we use Gram-Schmidt to select the most different basis element to the already found basis elements that have been selected and accumulated in previous steps, and add it to the found reduced basis.

Enrichment of the basis is applied in the process of interpolating test waveforms with the found basis. The found basis is first validated on 1000 test waveforms. If all waveforms can be interpolated accurately (with an interpolation error less than one tenth of the threshold ϵ) then we move to validate it on 10^6 test waveforms. In this step, if all those one million waveforms can be interpolated accurately (with an interpolation error less than the threshold ϵ) then the reduced basis has been found and the basis searching task is completed. However, usually a fraction of the 10^6 test waveforms cannot be represented accurately by the found basis. We call them the outlier test waveforms, $\mathcal{T}_{\text{outliers}}$. The outlier waveform with the highest empirical interpolation error will be added as a new basis element to form an enriched basis. This enriched basis element is calculated using Gram-Schmidt projection. The enriched basis, which contains the previously found basis elements and the enriched basis element, is then applied on the remaining outlier test waveforms. If some of them still cannot be interpolated accurately, then the one with the highest empirical interpolation error will be added as another enriched basis element. The process is repeated until all the outlier waveforms are represented accurately.

Note that unlike GreedyCpp, we do not add all the outlier test waveforms above a certain empirical interpolation error threshold to enrich the training waveform set. Rather, we choose the one with the largest interpolation error to enrich the basis. This is because we observed that some outliers with high empirical interpolation errors above the threshold can be from the same narrow region of the parameter space, and adding one of them as a new basis element can interpolate the rest of the outliers accurately.

Enrichment is critical in finishing up an ROQ basis search. Through enrichment we are able to add more effective basis elements not only to represent those "outlier" waveforms that cannot be interpolated by existing reduced basis but also many more training and test waveforms. The question is where to add the enrichment. In Algorithm 1 we show that in the Greedy loop, the threshold is set to be 0.1ϵ. This threshold can be increased, and theoretically it can be infinity and thus the enrichment is introduced from the beginning of reduced basis construction. Recall that the set of basis elements to represent a waveform space is not unique. The advantage to do the enrichment earlier is to decrease the Gram-Schmidt projections needed on the order of N^2 to reduce computation time.

Smaller training waveform sets mean more searches are required and a larger basis size. An extreme scenario is when each small training set has one waveform, the size of the basis is equal to the number of training waveforms. In this case all the training waveforms can be accurately interpolated but the basis size is too large to be used in parameter estimation. Larger training waveform sets do result in smaller basis size, but each new basis element search costs more time and hence it takes longer time to

finish the basis searches. Fish (fast basis searching) and bear palm (small basis size) cannot be obtained at the same time. What makes a balance then? We find that the basis can be obtained at a reasonable computation time when each small training waveform set has ~ 10 times the basis size N_L, taking linear basis for example. Figure 1 shows that for a chunk of waveform space, the size of the linear basis as a function of the size of the training waveforms in each small training set. For computing speed consideration we are generous about the number of the basis elements for the chunk and do not require it to be the least (perfect), but rather compromise at a potential 25% more basis elements, which result in an increase in the running time of gravitational wave parameter estimation to be less than 25%, which is acceptable.

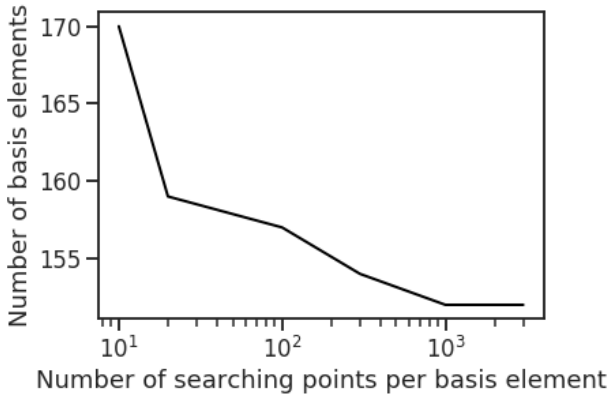

FIG. 1. The number of basis elements as a function of searching points in each small subset for the PhenomPv2 waveform model. The waveform parameter range has $M_c \in [20, 21]M\odot$, mass ratio $q \in [1, 1.2]$, and both spins up to 0.1.

For a given chunk of waveform space, the number of basis elements that are required to precisely represent all waveforms in it depends on the following factors: (a). The nature of the waveforms. For example, how different are the wiggles in each waveform imagining we finely grid the waveform space to the extent that all the features in the wiggles and wobbles of the waveforms are caught. The more different the features, the bigger the size of the basis needed. (b). The duration of the waveforms. Longer duration requires a larger basis.

Knowing these factors are important for PyROQ users to estimate the sizes of the bases and thus the construction time before running the code. For a new waveform, we would not be able to know the exact values of $N_{\mathtt{L}}$ and $N_{\mathtt{Q}}$ before carrying out a basis search with a large waveform training set. However, searching for the basis with a sole large training set is what we want to avoid due to the low computation speed in Python compared to C++. The solution is to use PhenomPv2 as a baseline. We first use a small test to decide the level of complexity of the waveform nature compared to PhenomPv2 under a similar waveform parameter space. We calculate the number of basis elements required for the small training set and compare it to that of PhenomPv2. Then we estimate $N_{\mathtt{L}}$ and $N_{\mathtt{Q}}$ to get proper setups to run PyROQ. With a pre-evaluation using IMRPhenomPv2, we can skip some loops in the Greedy loop in Algorithm 1.

We applied trivial parallel computation in PyROQ to reduce the wall time for waveform generations and the Gram-Schmidt process in each small group of waveforms for a new basis element searching. Specifically, Multiprocessing in Python is used, see Algorithm 1 for where it weighs in. With 50 processes running at the same time, the wall time is reduced by a factor of ~ 40. As the number of processes increase the wall time drops accordingly but not linearly.

D. Waveform models

A variety of waveform models are available for compact binary coalescence. PyROQ has included the ones that have been released in the LALSuite software library [37] and keeps up-to-date with their new released waveform approximants. However, it is easy to adapt the code to work for other waveforms, providing that the waveforms are frequency or time series. For the ROQs to be used in LIGO parameter estimation pipelines, the frequency domain is preferable even though there is no such requirement in ROQ data constructions.

III. CODE VALIDATION WITH IMRPHENOMPV2 WAVEFORM

Three aspects should be checked carefully for any code developed to construct reduced bases and reduced order quadrature data: the accuracy of the represented waveforms by the reduced basis, where it should reach the requirements imposed by parameter estimation; the size of the reduced linear and quadratic bases, where a smaller size provides higher speedups in a ROQ run than a standard run; the time it takes to build the basis, where our philosophy is a maximum of two weeks for any basis construction. In this section we first briefly compare the performances of PyROQ with that of GreedyCpp to showcase that the basis sizes and the construction time are both fine using PyROQ. We then demonstrate that while the other two aspects are considered, the bases generated with PyROQ can represent waveforms to the accuracy typically required by parameter estimation and therefore PyROQ can serve as an ROQ building code for LIGO gravitational waveforms.

The waveform model used for the code validation is a phenomenological waveform model known as IMRPhenomPv2, which is implemented in the LIGO Algorithm Library (LAL) [38]. This model describes an approximate IMR signal of precessing binary black holes by appropriately rotating the waveforms of an aligned-spin system by means of Euler rotations into the mode. The-

oretically, any waveform can be used for the validation of the code, but we chose the precessing waveform IMRPhenomPv2 for two reasons. First, it is a vanilla waveform that is frequently used in LIGO data analysis especially the PE runs with real data. Second, its bases have been built using GreedyCpp, so the performances of both GreedyCpp and PyROQ can be more easily and directly compared.

A. Performances compared with GreedyCpp

We built the linear and the quadratic bases for the 4-second IMRPhenomPv2 waveforms to showcase the performances of PyROQ compared to GreedyCpp. The parameter space and range were chosen to be the same as Case A in [22] which used GreedyCpp to make proper comparison when we built the ROQs with PyROQ. The comparisons are listed in Table I. The built time for the ROQs in this example was less than 10 hours on a 4-GB RAM computing node for the linear basis, and even less time for the quadratic basis because it has a smaller size than the linear basis. We impose the philosophy that it should take no more than two weeks to build a basis for any chunk of waveform parameter space range, otherwise further code optimisation should be applied. When the bases construction time is less than two weeks, we have the freedom to increase the searching points for a new basis element and thus reduce the basis size. Refer to Figure 1 and the section it is in for more details.

The example linear basis size by PyROQ is larger than GreedyCpp by 14%. This means that a ROQ run for a 4-second signal takes roughly 14% more time to finish if using the example ROQ data built by PyROQ than GreedyCpp, and thus about 12% less speedup the ROQ method can gain compared to the standard method. However, notice that in this example the bases construction aimed at building the ROQs in short amount of time and did not use parallelization, which was in a few hours. Because the construction time was much less than two weeks, we can use more points for every new basis element searching, or introduce enrichment earlier, and increase the construction time accordingly but still maintain it under two weeks, for which we expect smaller-sized linear basis and quadratic basis. In sum, the performances of PyROQ in terms of basis size and construction time are at least acceptable.

B. Accuracy of interpolated waveforms

Parameter estimation has requirements on the accuracy of interpolated waveforms for the ROQs to be applied to get the same estimation results as the standard method. It requires that the threshold ϵ, i.e. the maximum empirical interpolation error of interpolated waveforms in the parameter space, to be less than the minimum of 5×10^{-7} and one order of magnitude smaller than the accuracy of the waveform model, such that the fractional error in logarithmic likelihood is tiny enough to produce visually indistinguishable posteriors.

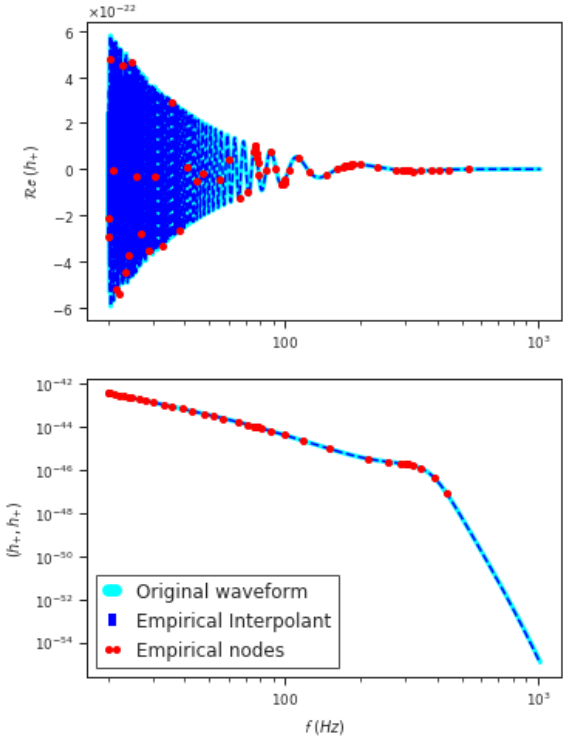

FIG. 2. Empirical interpolants of a random 16-second test waveform (top panel) and its overlap with itself (bottom panel). The light blue thick lines are the original waveform (top) and the original overlap (bottom). Their corresponding interpolants are shown in deep blue dashed lines. The red dots are the empirical nodes, the number of which is equal to that of the reduced basis elements that are used to represent the waveform or the overlap. The empirical interpolation errors for both interpolants are less than 5×10^{-12} and 1×10^{-12}, respectively.

An instance of the linear and the quadratic empirical interpolants of a random waveform and its overlap with itself is demonstrated in Figure 2. We randomly drew one million samples and calculated their empirical interpolation errors with the basis in Table I to check if accuracy requirement can be satisfied. The empirical interpolation errors of those one million test waveforms are smaller than 5×10^{-12}, and their distribution are shown in Figure 3. Recall that the machine epsilon in the code is 2.22×10^{-16}.

IV. APPLICATIONS ON PARAMETER ESTIMATION

Now that quality waveform bases are obtained, it is important to study if the ROQs generated by PyROQ can be reliably applied to inference on gravitational waves.

Code	Frequency range (Hz) Min	Max	Waveform duration T	$\Delta f(Hz)$	$M_c(M_\odot)$ Min	Max	Basis size Linear	Quadratic	Maximum error	Test waveforms
GreedyCpp	20	1024	$1.5s < T < 4s$	1/4	12.3	23	300	197	6×10^{-9}	15 millions
PyROQ	20	1024	$1.5s < T < 4s$	1/4	12.3	23	342	203	1×10^{-10}	2 millions
PyROQ	20	1024	$1.5s < T < 4s$	1/4	12.3	23	420	240	5×10^{-12}	2 millions

TABLE I. Comparisons of GreedyCpp and PyROQ in building a ROM for 4-second signals using IMRPhenomPv2 waveform. We use the same parameter values as in reference [22] such as limiting the magnitudes of the spin-related parameters (χ_1, χ_2, χ_p) to lie within the range $(-0.9, -0.9, 0) \leq (\chi_1, \chi_2, \chi_p) \leq (0.9, 0.9, 0.9)$ and we use the full range for the spin angles: $(0, 0) \leq (\theta_J, \alpha_0) \leq (\pi, 2\pi)$. The differences in the setups are the maximum interpolation error and the number of test waveforms, which are shown in the last two rows.

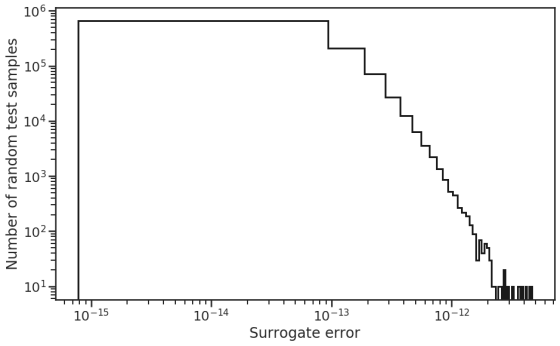

FIG. 3. Empirical interpolation error distribution for one million randomly drawn samples from the parameter space that the ROQs were built from. All test samples have interpolation errors less than 5×10^{-12}, which is the threshold for the ROQ construction. The machine epsilon is 2.22×10^{-16}.

The detectors of interest include but are not limited to the ground-based LIGO-Virgo-KAGRA (LVK) detectors and the LISA space detector. Simulated gravitational wave injections are required to compare the parameter estimation between the standard method and the ROQ method using the ROQs built with PyROQ. We do not start with real detections because they contain noise and we want to have more variables under control to pinpoint the effects from waveform interpolation. Real detections using the ROQ method are studied later in this section after the effects from waveform interpolation are addressed.

A. Simulated gravitational wave injections

Simulated gravitational waves injections are needed to investigate the effects of empirical interpolation errors on parameter estimation. For the questions that are concerned, any gravitational waves from the compact binary coalescence (CBC) category can fulfill the purposes. However, just to be chic, we chose neutron star-black hole mergers, which are the last type of CBCs that can be detected by LIGO and currently a hot topic [9, 10, 39]. Therefore, simulated signals modeled by IMRPhenomPv2 waveform and randomly drawn from the parameter space of the 16-second signals were injected coherently in the two LIGO detectors.

Each simulated signal was injected into three types of noise: zero noise, O4 noise, and the LIGO designed noise. Injections made with zero noise have no noise in the simulated strain data. These injections can eliminate the unclear influences on likelihood calculations from the noisy real data, so that they can get the possible differences that are resulted from the representations of the waveforms by the reduced bases. The O4 noise and the LIGO designed noise cases make it possible to see the effects that come from the noise.

Particularly, the studies with one randomly chosen waveform, which corresponded to three injections, are elaborately explained in sections IV B and IV C.

B. Point-by-point Likelihood Comparisons

We have shown that with the bases constructed by PyROQ, the errors of the empirically interpolated IMRPhenomPv2 waveforms are smaller than 5×10^{-12}. Now we determine how those tiny interpolation errors affect parameter estimation using the NSBH injections that are described in Section IV A. Before skipping to check the posteriors, we shall first examine the likelihoods. We performed point-by-point comparisons between the likelihoods calculated from both the standard *Full* likelihood function and the ROQ likelihood function. The likelihoods were evaluated with `Bilby` , which is one of the gravitational wave inference tools used in the LIGO-Virgo Collaboration.

Under the assumption that the ROQ is an approximation of the *Full* likelihood, the two methods are required to yield the same likelihood for the same set of waveform parameter values for the ROQ to qualify a valid substitute to the *Full* likelihood function for parameter estimation. Figure 4 shows the *Full* and the ROQ likelihoods calculated for an example NSBH waveform which is injected into zero-noise, designed noise, and O3 noise. In this example the chirp mass parameter is varied and the other parameters are fixed for the likelihood calculations and comparisons, but it can be generalised to cases where all the parameters vary simultaneously. It can be seen that the likelihoods from the two methods are indistinguishable, for the zero noise and the LIGO designed

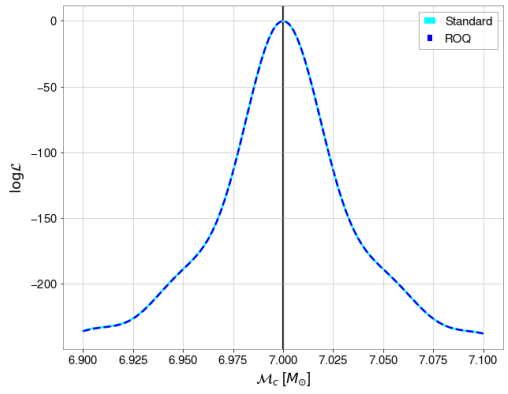

FIG. 4. Point-by-point likelihood comparisons between the standard Full likelihood and the ROQ likelihood using Bilby inference pipeline for an injection of a NSBH signal into zero noise. The vertical black line shows the chirp mass value ($7M_\odot$) of the injected waveform.

noise respectively, up to a tiny fractional difference

$$1 - \frac{\log \mathcal{L}_{\texttt{Full}}}{\log \mathcal{L}_{\texttt{ROQ}}} \leq 10^{-6}. \tag{7}$$

This shows that the waveforms represented by the bases generated by PyROQ can represent the original ones to the likelihood accuracy needed by the parameter estimation that applies the ROQ rules. Aside from the example case shown here, we have observed this fractional difference to be in all injection examples. This demonstrates that PyROQ can build qualified ROQ data for parameter estimation on detections with LIGO or similar detector sensitivity.

Note that historically, the accuracy requirement for waveform interpolations placed by PyROQ was that the empirical interpolation errors are less than 5×10^{-12}, but our studies indicate that a 1×10^{-10} threshold is already accurate enough in point-by-point likelihood comparisons in the regime of LIGO detectors for this specific waveform model and parameter space range. Therefore ROQs of less accuracy (1×10^{-10} in empirical interpolation error) can provide the same GW inferences as the standard cases. A comprehensive study of the waveform accuracy and the ROQ accuracy required for LIGO gravitational wave parameter estimation will be presented in *Waveform and ROQ Accuracy Requirements For LVK Parameter Estimation*, which is a paper under preparation.

C. Standard and ROQ Posterior Comparisons

We now move to compare the posterior probability distributions for the three NSBH injections discussed in Section IV B. We performed the parameter estimation for those injections using `Bilby`.

Table II lists the injected and the recovered parameter values with the two methods for the three injections of the same waveform. Both the standard method using the Full likelihood and the ROQ method using the ROQ likelihood can recover the injected gravitational wave parameters very accurately and precisely. The two methods agree with each other to the fourth significant figures for the 68% credible intervals for each injection. Figure 5 show the corner plots of posterior distributions of chirp mass M_c and mass ratio q for the three injections. The corner plots illustrate that the posterior distributions are consistent between the two methods for the injection.

		$M_c(M_\odot)$	q	SNR
injection		7.00	0.07143	16.4
Zero-nosie	standard	$7.00^{0.003}_{0.003}$	$0.07143^{+0.0001}_{-0.0001}$	16.4
Zero-nosie	PyROQ	$7.00^{0.003}_{0.003}$	$0.07143^{+0.0001}_{-0.0001}$	16.4

TABLE II. The injected and recovered values of chirp mass M_c, mass ratio q, and Signal-to-Noise Ratio (SNR) of the analysis from Figure 5. Modes and 68% credible intervals are read from both the standard likelihood and the ROQ compressed likelihood.

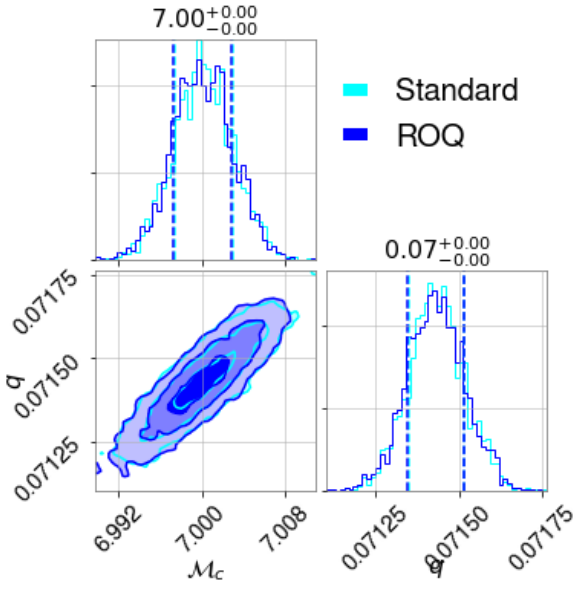

FIG. 5. Corner plots of probability density function for the chirp mass M_c and mass ratio q of a simulated zero noise injection. In cyan as obtained in 50 hours by the standard likelihood, and in cyan as obtained in 1 hour with the ROQ. The injection values are on the top of the subplots, and are listed in Table II.

While the posteriors from the two methods are visually indistinguishable, the running time is different for the two methods. The ROQ method produces speedups of about 50 times for the injection.Recall that the speedup depends on the size of the reduced basis compared to the frequency nodes used in the waveform model, as well as

the other parts of parameter estimation. The more the parameter estimation is dominated by waveform generations, the more speedup the ROQ method gains.

D. Consistency Comparisons with GreedyCpp

It is of interest to examine whether the parameter estimation results are consistent using the bases constructed by PyROQ and those by GreedyCpp.

We analysed the real observed data that contained the two gravitational wave events GW150914 and GW151226 in the LIGO's first observing run. For each GW event, we set up two identical ROQ runs using the IMRPhenomPv2 waveform and the ROQ likelihood function evaluated with `LALInference` [40], except that one used the bases constructed by PyROQ and the other used those by GreedyCpp as in Table I . We make comparison plots between the recovered posterior PDFs for the chirp mass in Figure 6. Other parameters show similar consistency.

The inferences of the astrophysical parameters from both runs are consistent for both events, respectively. The figure clearly shows that the ROQ method with the PyROQ reduced bases and the GreedyCpp reduced bases both recover visually identical posterior PDFs. Besides, the running time is similar for both runs for each event. This is expected because the reduced basis sizes are comparable using the two basis construction codes. These provide further supports for PyROQ to be used as a qualified ROQ data building code.

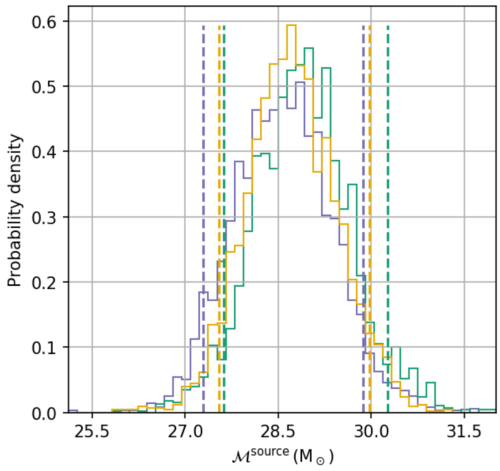

FIG. 6. Comparisons of ROQ PE runs for GW150914 using bases constructed with GreedyCpp (yellow) and PyROQ (blue). The green is standard run without ROQs.

The analyses of the other LVK detections using the ROQs generated by PyROQ are out of the scope of this paper. When time allows it is useful to revisit those old detections with the new tool because it can provide more benchmarking that benefits inference on future LVK gravitational wave detections using PyROQ.

E. Application of standard and ROQ methods to static LISA

We briefly illustrate an application of the ROQ method onto a simulated toy detection scenario with the space detector LISA. Specifically, first with PyROQ we built the ROQs for days-long ($M_c \in [10^6, 10^7] M_\odot$) signals of supermassive black hole binaries using IMRPhenomPv2 waveform. The frequency node separation is $1/1638400-$ *second* (about once per 19 days).

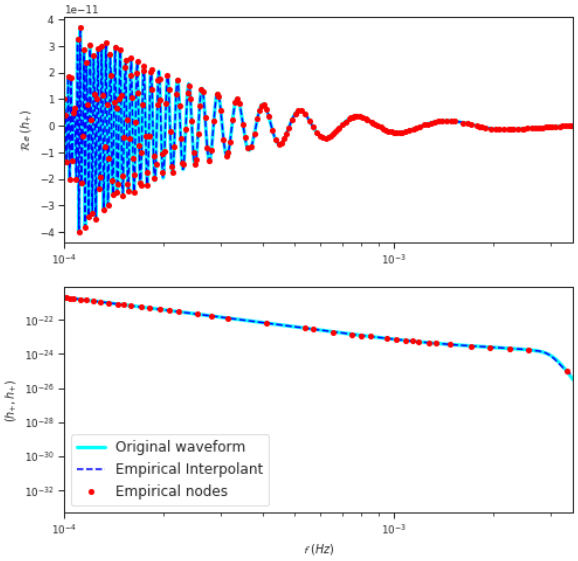

FIG. 7. Empirical interpolants of a random 1638400-second (about 19-day) test waveform (top panel) and its overlap with itself (bottom panel). The chirp mass is $2.8 \times 10^6 M_\odot$, and the mass ratio is 1.25. The light blue thick lines are the original waveform (top) and the original overlap (bottom). Their corresponding interpolants are shown in deep blue dashed lines. The red dots are the empirical nodes.

It is shown in Figure 7 the empirical interpolants for a randomly chosen example waveform in the parameter range in which the ROQs were built. The chirp mass is $2.8 \times 10^6 M_\odot$, up to a factor of 10^5 than the first detected BBH; and the mass ratio is 1.25, which is similar to that of the first detected BBH. With Bilby, we created a detector that has an arm length of $2.5 \times 10^{-6} km$, which is about the size of LISA, and set it to be located on the surface of the Earth and has the PSDs similar to that of LISA. We call it static LISA, in the sense that we ignore the motion of LISA throughout a signals duration, which can be days, months, and even years. Then we made injections of SMBHB mergers into the static LISA detector and recover the gravitational wave parameters using both the standard method and the ROQ method. The overall speedup was about 5 times using the ROQ method for all the injections and detections we simulated. For the example frequency-domain signal that spans a few mHz, the signal size is actually pretty small with $\sim 10^4$ data points. For years-long signal in LISA, more inference speedups

are expected from the ROQ method. Therefore, we estimate that there are non-negligible speedups for the LISA parameter estimation using the ROQ method after the time delay interferometery (TDI) data that are collected by the real, moving LISA are used.

V. CONCLUSION

We have described a method to search for reduced bases of gravitational waveforms and build the reduced order quadrature data for fast parameter estimation. We subsequently illustrated using IMRPhenomPv2 waveform that the code PyROQ, which implemented the method, can find reduced bases for accurate waveform representation for the waveforms in the LALSuite and build ROQs for fast and accurate parameter estimation on LVK's data. Point-by-point comparison are made between Full likelihood and the ROQ likelihood. Their fractional differences are less than 10^{-6} as shown in Figure 4. We carried out simulated NSBH injections to show that the inference on simulated signals using those ROQs can produce posterior distributions of the parameters at the same accuracy as with the standard non-ROQ method, see Figure 5.

Because the ROQs are built without any noise, they can be used for other ground-based gravitational wave detectors of similar sensitivity range to the LVK detectors. The ROQs can also be used in science scenario studies where simulated gravitational wave signals are injected into the PSDs of the current and the future LVK observations. They are particularly useful when the studies incorporate a large number of simulated events where computation speed becomes an demanding requirement.

For the 4-second IMRPhenomPv2 waveforms, we compared the ROQs generated using PyROQ with those using GreedyCpp in Table I both in the basis size and the computation time. Then we used both the ROQs to infer a few real gravitational wave detections in LIGO's first observing run and showed that the parameter estimation results are barely distinguishable, see Figure 6. We concluded that PyROQ can be safely used in LVK fast parameter estimation.

PyROQ can be easily adapted to build ROQ data for any other gravitational waveforms that are not currently included or released in the LALSuite. We also showcased that PyROQ can be adapted and applied for space detectors such as LISA, see Section IV E. It can also be modified to work for more generalized time and frequency series from experiments in other research fields that require faster parameter estimation.

ACKNOWLEDGMENTS

We wish to acknowledge the useful discussions with Carl Halster, Michael Pürrer, Rory Smith, and Scott Field. We thank Zu-Cheng Chen for helping review the code and useful discussions. Hong especially thanks the members in 53 the Parade for brainstorming up the neat name PyROQ in the spring of 2019. We also thank our LIGO Presentation and Publication reviewers for their feedback on this manuscript. The authors are grateful for computational resources provided by the LIGO Laboratory and supported by National Science Foundation Grants PHY-0757058 and PHY-0823459. We are grateful for the computational resources provided by the Leonard E. Parker Center for Gravitation, Cosmology and Astrophysics at University of Wisconsin-Milwaukee and supported by NSF-0923409. We thank the Hawk supercomputing system provided by Cardiff University. The project is supported by the STFC grant ST/T000147/1.

[1] J. Aasi et al., "Advanced ligo," CQG **32**, 074001 (2015).
[2] F. Acernese et al., "Advanced virgo: a second-generation interferometric gravitational wave detector," CQG **32**, 024001 (2015).
[3] T. Akutsu et al., "2.5 generation interferometric gravitational wave detector," Nat Astron **3**, 35–40 (2019).
[4] B. P. Abbott et al., "A gravitational-wave transient catalog of compact binary mergers observed by ligo and virgo during the first and second observing runs," Phys. Rev. X **9**, 031040 (2019).
[5] Abbott, R. et al., "GW190412: Observation of a Binary-Black-Hole Coalescence with Asymmetric Masses," Phys. Rev. D **102**, 043015 (2020).
[6] Abbott, B. P. et al., "GW190425: Observation of a compact binary coalescence with total mass $\sim$ 3.4 m $\odot$," Astrophysical Journal Letters **892**, L3 (2020).
[7] Abbott, R. et al., "GW190521: A Binary Black Hole Merger with a Total Mass of 150 $M_\odot$," Phys. Rev. Lett. **125**, 101102 (2020).
[8] Abbott, R. et al., "GW190814: Gravitational Waves from the Coalescence of a 23 Solar Mass Black Hole with a 2.6 Solar Mass Compact Object," The Astrophysical Journal Letters **896**, 2 (2020).
[9] LIGO Scientific Collaboration & Virgo Collaboration, "LIGO/Virgo S200105ae: A subthreshold GW compact binary merger candidate. ," (2020), GRB Coordinates Network:26640.
[10] LIGO Scientific Collaboration & Virgo Collaboration, "LIGO/Virgo S200115j: Identification of a GW compact binary merger candidate," (2020), GRB Coordinates Network, Circular Service, No. 26759, #1 (2020/Jan-0):26759.
[11] R. Adhikari, E. K. Gustafson, S. Hild, S. Ballmer, and K. Arai, "the Third Generation LIGO Strawman Design Meeting Tech. Report," https://dcc.ligo.org/cgi-bin/private/DocDB/ ShowDocument?docid=85610 (2012).
[12] LIGO Scientific Collaboration, "The LSC-Virgo White Paper on Instrument Science (2016-2017 edition)," https://dcc.ligo.org/LIGO-T1600119/public (2016).

[13] R. X. Adhikari, P. Ajith, Y. Chen, Y. et al., "Astrophysical science metrics for next-generation gravitational-wave detectors," (2019), arXiv:1905.02842.

[14] Punturo1, M. et al., "The third generation of gravitational wave observatories and their science reach," Classical and Quantum Gravity **27**, 8 (2010).

[15] D. Reitze et al., "Cosmic Explorer: The U.S. Contribution to Gravitational-Wave Astronomy beyond LIGO," (2019), arXiv:1907.04833 [astro-ph.IM].

[16] Abbott B. P. et al., "Prospects for Observing and Localizing Gravitational-Wave Transients with Advanced LIGO, Advanced Virgo and KAGRA," (2013), arXiv:1304.0670.

[17] Kipp Cannon, Adrian Chapman, Chad Hanna, Drew Keppel, Antony C. Searle, and Alan J. Weinstein, "Singular value decomposition applied to compact binary coalescence gravitational-wave signals," Phys. Rev. D **82**, 044025 (2010).

[18] R. J. E. Smith, K. Cannon, C. Hanna, D. Keppel, and I. Mandel, "Towards rapid parameter estimation on gravitational waves from compact binaries using interpolated waveforms," Phys. Rev. D **87**, 122002 (2013).

[19] P. G. Constantine, D. F. Gleich, Y. Hou, and J. Templeton, "Model reduction with mapreduce-enabled tall and skinny singular value decomposition," SIAM Journal on Scientific Computing **36**, 166 (2014).

[20] S. E. Field, C. R. Galley, J. S. Hesthaven, J. Kaye, and M. Tiglio, "Fast prediction and evaluation of gravitational waveforms using surrogate models," Phys. Rev. X **4**, 031006 (2014).

[21] P. Canizares, S. E. Field, J. Gair, V. Raymond, R. Smith, and M. Tiglio, "Accelerated gravitational-wave parameter estimation with reduced order modeling," Phys. Rev. Lett. **114**, 071104 (2015).

[22] R. Smith, S. E. Field, K. Blackburn, C. Haster, M. M. Prrer, V. Raymond, and P. Schmidt, "Fast and accurate inference on gravitational waves from precessing compact binaries," Phys. Rev. D **94**, 044031 (2016).

[23] H. Antil, D. Chen, and Field S. E., "A note on qr-based model reduction: Algorithm, software, and gravitational wave applications," Computing in Science & Engineering **20**, 10 (2018).

[24] Barak Zackay, Liang Dai, and Tejaswi Venumadhav, "Relative Binning and Fast Likelihood Evaluation for Gravitational Wave Parameter Estimation," (2018), arXiv:1806.08792 [astro-ph.IM].

[25] Jonathan E. Thompson, Edward Fauchon-Jones, Sebastian Khan, Elisa Nitoglia, Francesco Pannarale, Tim Dietrich, and Mark Hannam, "Modeling the gravitational wave signature of neutron star black hole coalescences: PhenomNSBH," Phys. Rev. D **101**, 124059 (2020).

[26] Cecilio Garca-Quirs, Marta Colleoni, Sascha Husa, Hctor Estells, Geraint Pratten, Antoni Ramos-Buades, Maite Mateu-Lucena, and Rafel Jaume, "IMRPhenomXHM: A multi-mode frequency-domain model for the gravitational wave signal from non-precessing black-hole binaries," Phys. Rev. D **102**, 064002 (2020).

[27] Hctor Estells, Antoni Ramos-Buades, Sascha Husa, Cecilio Garca-Quirs, Marta Colleoni, Lela Haegel, and Rafel Jaume, "IMRPhenomTP: A phenomenological time domain model for dominant quadrupole gravitational wave signal of coalescing binary black holes," (2020), arXiv:2004.08302.

[28] Geraint Pratten, Cecilio Garca-Quirs, Marta Colleoni, Antoni Ramos-Buades, Hctor Estells, Maite Mateu-Lucena, Rafel Jaume, David Keitel Maria Haney, Jonathan E. Thompson, and Sascha Husa, "Let's twist again: computationally efficient models for the dominant and sub-dominant harmonic modes of precessing binary black holes," (2020), arXiv:2004.06503.

[29] Liang Dai, Tejaswi Venumadhav, and Barak Zackay, "Parameter Estimation for GW170817 using Relative Binning," (2018), arXiv:1806.08793.

[30] Soichiro Morisaki and Vivien Raymond, "Rapid Parameter Estimation of Gravitational Waves from Binary Neutron Star Coalescence using Focused Reduced Order Quadrature," (2020), arXiv:2007.09108.

[31] J. Akeret, L. Gamper., A. Amara, and A. Refregier, "Hope: A python just-in-time compiler for astrophysical computation," Astronomy and Computing **10**, 1 (2015).

[32] S. Husa, S. Khan, M. Hannam, M. Prrer, F. Ohme, X. J. Forteza, and A. Boh, "Frequency-domain gravitational waves from nonprecessing black-hole binaries. I. New numerical waveforms and anatomy of the signal," Phys. Rev. D **93**, 044006 (2016).

[33] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Prrer, X. J. Forteza, and A. Boh, "Frequency-domain gravitational waves from nonprecessing black-hole binaries. II. A phenomenological model for the advanced detector era," Phys. Rev. D **93**, 044007 (2016).

[34] C. Messick et al., "Analysis Framework for the Prompt Discovery of Compact Binary Mergers in Gravitational-wave Data," Phys. Rev. D **95**, 042001 (2017).

[35] Bruce Allen, Warren G. Anderson, Patrick R. Brady, Duncan A. Brown, and Jolien D. E. Creighton, "FINDCHIRP: An algorithm for detection of gravitational waves from inspiraling compact binaries," Phys. Rev. D **85**, 122006 (2012).

[36] P. Canizares, Field S. E., J. R. Gair, and M. Tiglio, "Gravitational wave parameter estimation with compressed likelihood evaluations," Phys. Rev. D **87**, 124005 (2013).

[37] LIGO Scientific Collaboration, "LSC Algorithm Library Suite," (2018).

[38] LIGO/LSC Algorithm Library Suite, https://www.lsc-group.phys.uwm.edu/daswg/projects/lalsuite.html (2020).

[39] S. Anand et al., "Optical follow-up of the neutron star-black hole mergers s200105ae and s200115j," Nat Astron (2020), 10.1038/s41550-020-1183-3.

[40] J. Veitch, V. Raymond, B. Farr, et al., "Robust parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library," Phys. Rev. D **91**, 042003 (2015).