Gravitational wave parameter estimation with compressed likelihood evaluations

Priscilla Canizares,[1] Scott E. Field,[2] Jonathan R. Gair,[1] and Manuel Tiglio[3,4]

[1] *Institute of Astronomy, Madingley Road, Cambridge, CB30HA, United Kingdom*
[2] *Department of Physics, Joint Space Sciences Institute, Maryland Center for Fundamental Physics. University of Maryland, College Park, MD 20742, USA*
[3] *Center for Scientific Computation and Mathematical Modeling, Department of Physics, Joint Space Sciences Institute, Maryland Center for Fundamental Physics. University of Maryland, College Park, MD 20742, USA*
[4] *TAPIR, MC 350-17, California Institute of Technology, Pasadena, CA, 91125, USA*

One of the main bottlenecks in gravitational wave (GW) astronomy is the high cost of performing parameter estimation and GW searches on the fly. We propose a novel technique based on Reduced Order Quadratures (ROQs), an application and data-specific quadrature rule, to perform fast and accurate likelihood evaluations. These are the dominant cost in Markov chain Monte Carlo (MCMC) algorithms, which are widely employed in parameter estimation studies, and so ROQs offer a new way to accelerate gravitational wave parameter estimation. We illustrate our approach using a four dimensional gravitational-wave burst model. We build an ROQ for this model and perform four-dimensional MCMC searches with both the standard and ROQs quadrature rules, showing that, for this model, the ROQ approach is around 25 times faster than the standard approach without any loss of accuracy. The speed-up from using ROQs is expected to increase for more complex GW signal models and therefore has significant potential to accelerate parameter estimation of GW sources such as compact binary coalescences.

I. MOTIVATION AND CONTEXT

Computing correlations between data and models described by large dimensional parameter spaces is an important aspect of many scientific disciplines. Obtaining estimates of the parameters of observed signals is crucial to extract the most from multi-billion dollar experiments such as GW detectors (i.e., advanced LIGO, advanced Virgo, Indigo, and KAGRA) [1–4]. However, carrying out parameter estimation on large dimensional parameter spaces can be computationally expensive. Costs grow further if several different models or alternative theories of gravity (see, for example, [5–12] and [13–15], respectively) are used to analyse the data as a prelude to Bayesian model selection. It is therefore of great importance to develop efficient methods for analysing the data to ensure that all the desired science can be extracted from the data in a reasonable time.

One of the primary methods for computing the probability distribution for the parameters of a given signal in a data set is Markov chain Monte Carlo (MCMC). This requires evaluating the posterior probability of the model parameters throughout parameter space. When the likelihood and hence posterior probability are expensive to evaluate, MCMC algorithms can become computationally prohibitive. In such cases, approximate methods such as the Fisher matrix are widely used because they are significantly cheaper than a full Bayesian analysis. Several rather optimistic assumptions, however, such as high signal-to-noise ratios are often not satisfied in practice. Recently, other sampling approaches [16] for computing the maximum likelihood estimator have been proposed for low signal-to-noise scenarios.

An alternative way to improve the speed of MCMC algorithms is to reduce the cost of calculating a posterior or likelihood at each parameter space point. This strategy has motivated work on directly interpolating the likelihood [17–20] and training a neutral network to "learn" likelihood data [21]. At least in the case of direct interpolation there could be technical obstacles for waveforms with many cycles and/or higher dimensionality [18, 20]. In this paper we describe a novel technique for fast, accurate calculations of correlations between a signal and a catalog of templates, fine tuned for applications such as MCMC. The approach is based on Reduced Order Modeling (ROM) and, as such, aims to significantly reduce the problem's dimensionality by exploiting redundancies. The result is a compressed representation of the likelihood thereby minimizing the cost of each evaluation. Generalizations to higher dimensions and/or many cycles are readily handled within the method's existing framework [22].

Within typical gravitational wave physics applications, the number of templates quickly grows with the number p of physical parameters and the number of GW cycles. For example, the number of templates scale as $\sim (1 - MM)^{-p/2}$ [23], where MM is the minimal match of the catalog. For a compact binary coalescence with $p = 8$ parameters intrinsic parameters lasting for 10^5 cycles one could need up to $\sim 10^{40}$ templates for a fully coherent search [24]. In light of these scalings there is an obvious need for reducing the cost of each correlation.

Correlation costs typically scale with the length N of the data, which depends on both the observation time and sampling rate. Furthermore, for noisy (non-smooth) data standard fast converging numerical integration rules in general do not apply. In this paper we show how integrals with noisy data can be computed with a cost not set by the Nyquist sampling rate or observation time [25], but rather the "information content" of the GW templates themselves. Similar to Gaussian quadratures, the integration converges exponentially fast with the number of sparse data samples m drawn from the full data set. The overall likelihood cost is thereby reduced to $m \ll N$.

Our approach for speeding up correlation computations is based on a recently proposed Reduced Order Quadrature (ROQ) for parametrized functions [22]. ROQ combines dimensional reduction with the Empirical Interpolation Method (EIM) [26, 27] to produce a nearly optimal quadrature rule for parametrized systems. To do so, it exploits smooth dependence with respect to parameter variation, when available, to achieve very fast convergence with the number of ROQ nodes. In many cases ROQs outperform the best known quadrature rule (Gaussian quadratures) for generic smooth functions [22]. The key aspect of this apparent super-optimality is to exploit information about the space of functions in which one is interested.

In the context of GW parameter estimation, the use of ROQs can significantly improve the performance of the numerical algorithms employed for GW parameter estimation and searches by significantly reducing the computational cost of computing a waveform overlap (correlation) with the data without loss of accuracy. We illustrate this application of ROQs to GW parameter estimation in this work using a simple model of a sine-Gaussian GW burst waveform. This model is chosen as a toy model to illustrate the method only. Although such waveforms are used in GW data analysis for ground based detectors (see, for example, [30]), the cost of waveform evaluation is not significant in that case, so we are not proposing that this application is one for which ROQs are required. However, we demonstrate that even for such a simple model the speed up from ROQs is significant and we expect that comparable or greater speed-ups will be possible for more complex GW signal models [22].

This paper is organised as follows. In section II we present an overview of the proposed approach. In sections III, IV and V we introduce the building blocks of the method; namely, Reduced Order Modelling, the Empirical Interpolation Method and Reduced Order Quadratures, as well as the gravitational wave burst model. Finally, in Section VI, we apply the ROQ approach to perform a MCMC search using this burst model, showing that ROQ can considerably speed up MCMC computations. Among the new aspects that we address compared to [22] are how to deal with the time of arrival of the GW signal, and the application of the technique to noisy data. In Appendices A and B, we summarise the greedy approach for generating a Reduced Basis, and the Empirical Interpolation Method, respectively.

II. METHODOLOGY

In this paper we are interested in improving the performance of GW parameter estimation by using ROQs. We assume that the detected data stream is given by $s(t) = h(t;\boldsymbol{\lambda}) + n(t)$, where $h(t;\boldsymbol{\lambda})$ is the GW signal that we want to characterise, which depends on a multi-dimensional set of source parameters $\boldsymbol{\lambda}$, and $n(t)$ is instrumental noise.

In the context of Bayesian parameter estimation the posterior probability distribution function (PDF) provides complete information about the parameters of the signal:

$$p\left(\boldsymbol{\lambda}|s\right) := \mathcal{C} p\left(\boldsymbol{\lambda}\right) P(s|\boldsymbol{\lambda}) \tag{1}$$

Here $p\left(\boldsymbol{\lambda}\right)$ is the prior probability density, $\mathcal{C}$ an overall normalization constant, and $P(s|\boldsymbol{\lambda})$ is the likelihood that the true parameter values are given by a particular $\boldsymbol{\lambda}$, or in other words, the likelihood that our signal is present in the data stream. For Gaussian, stationary noise the likelihood is given by

$$P\left(s|\boldsymbol{\lambda}\right) \propto \exp\left(-\chi^2/2\right), \tag{2}$$

where

$$\chi^2 := \langle n|n\rangle = \langle s(t) - h\left(t;\boldsymbol{\lambda}\right)|s(t) - h\left(t;\boldsymbol{\lambda}\right)\rangle \tag{3}$$

is the weighted norm of the noise realization $n(t)$, defined by the weighted inner product (see e.g. [24])

$$\langle a\,|b\rangle = 4\Re \int_{f_{\min}}^{f_{\max}} \frac{\tilde{a}(f)\tilde{b}^*(f)}{\tilde{S}_n(f)} df, \tag{4}$$

with * denoting complex conjugation and $\tilde{S}_n(f)$ the power spectral density of the detector's noise. Owing to the form of $\tilde{S}_n(f)$ in GW physics, the lower limit of integration in (4) is often $f_{\min} > 0$.

When dealing with high dimensional problems, the process of mapping the likelihood (or the posterior) surface can become very expensive. MCMC algorithms are a useful technique for searching through such large spaces, by following a random walk in parameter space, with the probability of a sample being chosen at any point being proportional to the posterior probability. However, since a MCMC search depends on the number of sampling points, as well as the dimensionality of the problem, it can still be a very expensive algorithm and in many cases prohibitively so.

This paper proposes application and data-specific quadrature rules for scenarios such as GW parameter estimation, where correlations between noisy data and a family of functions have to be repeatedly evaluated. The quadrature rules employed here are a variation of the ROQ introduced in Ref. [22] for the case $n(t) = 0$, and their construction follows several layers of dimensional reduction that are explained in the different Sections of this paper, namely:

1. **Construct a basis for the space of waveforms of interest. Offline stage**.

   Described in Section III. A Reduced Basis-greedy approach has several advantages, including an approximation to the most relevant points in parameter space, but the proposed ROQ can use any choice of a "good" basis.

2. **Identify the Empirical Interpolation points associated with the above basis. Offline stage**.

   Described in Section IV. This step, also through a greedy approach, provides the set of most relevant points in the physical dimension(s), and a nearly optimal global interpolant associated with the basis constructed in Step 1. These EIM nodes are to be used as integration points in the ROQ rule.

3. **Given any stream of data, construct the weights of the ROQ. Startup stage**.

   Described in Section V. These weights are linear combinations of correlations between the data and the basis elements of Step 1.

4. **Fast evaluation of correlations between the data stream and waveforms. Online stage**.

   Described in Section VI. The ROQ uses the nodes computed in Step 2 and the weights computed in Step 3 to perform fast and accurate evaluations of overlaps between the data and any waveform within the model.

Section VI discusses the results of putting the above pieces together into MCMC simulations for parameter estimation of mock data corresponding to the burst model family of waveforms described below in Eq. (7). These waveforms have been used in GW searches, see for example [28–30]. From these simulations, in particular, we quantify the significant speedups that are obtained even for such a simple GW model when using the proposed ROQ.

III. REDUCED ORDER MODELING

Roughly speaking, ROM deals with data which can be represented, with or without loss of accuracy, by fewer degrees of freedom than those of the full problem. For a given problem there are many available methods for revealing a reduced representation. Classical methods such as Principal Component Analysis, Proper Orthogonal or Singular Value Decompositions (SVD) [31], which are related to each other, were introduced as early as the 1800's (see [32] for a review of their history) and reveal low-rank approximations within existing data. Other approaches such as Reduced Basis (RB) (see, as a sample, [33–40] or [41] for a recent review), are specifically designed for parametrized problems whose solution is expensive to evaluate but also have advantages when dealing with "big data" problems (e.g. if the data cannot be fit into memory or the SVD cost becomes prohibitive).

Both RB-greedy and SVD are projection-based ROM algorithms. If the waveforms are known at the training points

$$\mathcal{T} := \{\boldsymbol{\lambda}_i\}_{i=1}^{M}$$

with $\boldsymbol{\lambda}$ some parametrization of the samples, a projection-based method identifies a basis $\{e_i\}_{i=1}^{m}$ such that

$$h(\cdot;\boldsymbol{\lambda}) \approx \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot)\,, \quad \text{for } \boldsymbol{\lambda} \in \mathcal{T} \tag{5}$$

with $m \leq M$ and the coefficients c_i are given by Eq.(A5) (see Appendix A for more details). If the problem is amenable to ROM, then $m < M$ or even $m \ll M$.

To be more concrete, in the GW case $\boldsymbol{\lambda}$ would represent the (intrinsic, extrinsic, or both) parameters of the problem, and M the number of available samples; say, the number of waveforms in a catalog or even the continuum, $M \to \infty$. A generic waveform with associated parameter $\boldsymbol{\lambda}$ would be a function of time or frequency,

$$h = h(t;\boldsymbol{\lambda}) \quad \text{or} \quad h = h(f;\boldsymbol{\lambda})\,.$$

In what follows, we will refer to $\boldsymbol{\lambda}$ as the parameter dimension and f or t as the physical one.

For methods such as SVD the basis elements e_i are linear combinations of all elements in the original training set; in general they do not correspond to any subset of parameters. That is, the elements of the basis are not physical waveforms themselves.

Methods such as Reduced Basis, on the other hand, generate a basis whose elements are waveforms and provide a nested, hierarchal set of the "most important" points in parameter space, in a precise mathematical sense. This can be useful when one of the goals is to build the basis on the fly, without full knowledge of the training set.

A. Generating a basis

Suppose for any $\boldsymbol{\lambda}$ the GW template $h(\cdot;\boldsymbol{\lambda})$ has an accurate approximation (5) in some basis $\{e_i\}_{i=1}^{m}$. Recent work [42–45] has shown that for fixed but arbitrary physical and parameter ranges, a small number of basis functions is sufficient to accurately represent any waveform of the same physical model in that range. Furthermore, when the basis is generated through a RB-greedy algorithm (described in Appendix A), the approximation error is guaranteed to yield a nearly optimal solution of the so-called n-width approximation problem [46, 47]. In the cases of interest this means exponential convergence of the representation error defined below in Eq. (6) with respect to the number of basis functions, resulting in a very compact basis. In addition, the number of basis elements often exhibits negligible increase as the dimensionality of the problem grows [44]. Last but not least, the most relevant points in parameter space are selected.

Of the basis set $\{e_i(\cdot)\}_{i=1}^m$ we require m to be small and the approximation to satisfy

$$\sigma_m := \max_{\boldsymbol{\lambda}} \min_{c_i \in \mathbb{C}} \left\| h(\cdot;\boldsymbol{\lambda}) - \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot) \right\|^2 \leq \epsilon \,, \qquad (6)$$

where ϵ is a user defined bound for the error (in our cases, typically $\sim 10^{-12}$, see for example Fig. 2), the coefficients $\{c_i\}$ are chosen so as to optimize the approximant (see Appendix A), and the largest error in the parameter region of interest is taken. That is, σ_m quantifies the error of the "worst best" approximation by the basis.

Many possible basis choices, including traditional ones such as Legendre or Chebyshev polynomials, could satisfy the above required criteria. In practice, application-specific bases usually provide better accuracy for a given m and also lead to a well-conditioned global interpolation procedure, as defined in Section IV.

We have mentioned the RB-greedy algorithm as one approach to generate a good basis. For definiteness, in the simulations of this paper our basis is constructed with such an algorithm (described in Appendix A). Our proposed ROQ rule is, however, directly applicable to any projection-based ROM basis, including SVD [45, 48, 49].

B. An example of RB: burst waveforms

In order to illustrate our approach, throughout this paper we will consider a four parameter GW-burst waveform, given by the following sine-Gaussian

$$h(t;\boldsymbol{\lambda}) := A e^{-(t-t_c)^2/(2\alpha^2)} \sin(2\pi f_0 (t-t_c)) \,, \qquad (7)$$

with $t \in [-\infty, \infty]$. The Fourier transform of the $t_c = 0$ waveform is given by

$$\tilde{h} = i2A\alpha\sqrt{2\pi}\sinh(4\pi^2\alpha^2 f_0 f) e^{-2\pi^2\alpha^2(f_0^2+f^2)}. \qquad (8)$$

This family of waveforms is described by four free parameters $\boldsymbol{\lambda} = (\alpha, f_0, t_c, A)$. We will build the Reduced Basis over just two parameters (α, f_0), since the others are extrinsic and can be handled differently, as discussed in section V C.

We build the RB for these burst waveforms over the parameter space defined by

$$\alpha = [.02, 2]\,\text{sec} \quad , \quad f_0 = [.01, 1]\text{Hz} \,, \qquad (9)$$

sampled with 180 equally spaced training points in each dimension. Unless otherwise stated, the range given in Eq. (9) will be the default range for all our experiments and the units will always be in seconds and Hertz. To represent any burst waveform drawn from Eq. (9) we take

$$T = 32\,\text{sec} \quad , \quad f_{\rm s} = 64\text{Hz} \,, \qquad (10)$$

to be our default observation time and sampling rate. Similarly, for the injected signals our default parameters will be

$$\alpha = 1 \quad , \quad f_0 = 0.25 \quad , \quad t_c = 0.1 \,, \qquad (11)$$

with t_c the arrival time. We will also present results for a two parameter model in which t_c is fixed at $t_c = 0$ and where A is chosen to give a specified signal-to-noise ratio (SNR), ρ, with $\rho^2 = \langle h|h \rangle$ for the inner product defined by Eq. (4).

Figure 1 shows the points selected by the greedy algorithm to include in the RB, and the order in which they are picked, while Fig. 2 shows the representation error of the training set as a function of the number of RB elements. Consistent with previous experience, we have found that if the training set is dense enough (and for this model, one of 180 × 180 samples is) then a-posteriori simulations injecting waveforms not present in the training set also yield similarly small representation errors by the basis; see for example [43, 44] for more details.

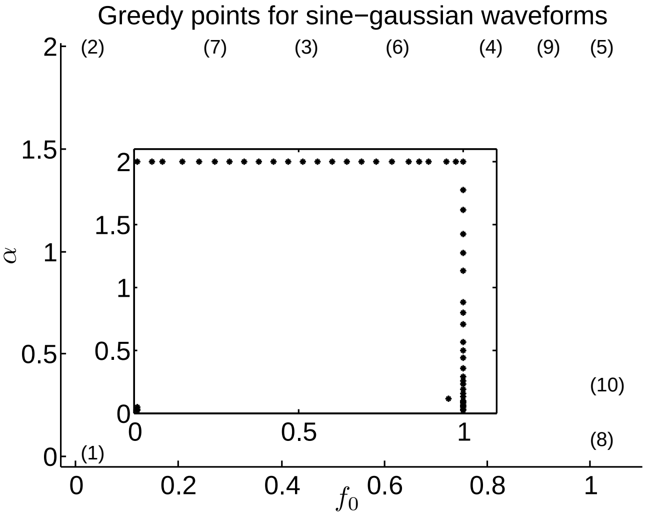

FIG. 1: Points selected by the greedy algorithm for the model family of burst waveforms Eq. (7) with the default range Eq. (9) for its parameters. The first 10 greedy points are represented with markers indicating the order of selection, with parenthesis serving as a visual aid. The inset figure shows all 54 selections with black asterisks.

So far we have described the generation of basis elements for the parameter range Eq. (9). The next step is the *prediction* (as opposed to projection) of waveforms from a sparse set of well chosen frequency samples.

IV. EMPIRICAL INTERPOLATION

Within a projection-based approximation one has

$$h(x) \approx \sum_{i=1}^{m} c_i e_i(x) \,, \qquad (12)$$

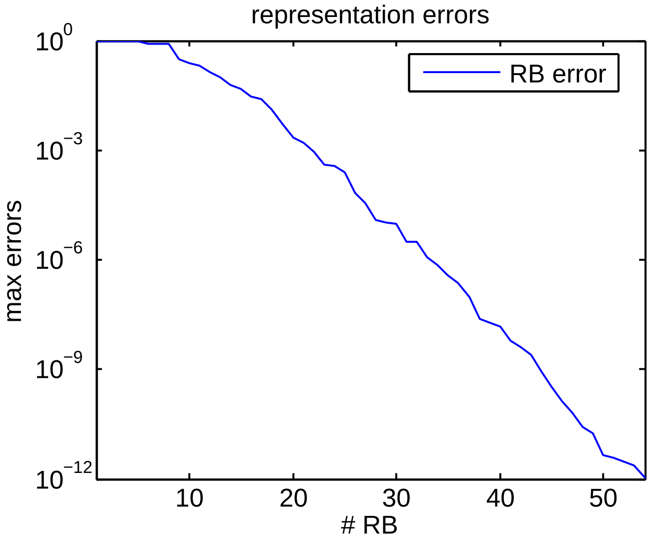

FIG. 2: Approximation error as a function of the number of basis generated with a greedy algorithm from the previous figure. The error is computed as the maximum of $\|h-P_m[h]\|^2$ within the parameter region considered, as defined by Eq. (6).

where the coefficients c_i are given by Eq. (A5). To compute the projection coefficients c_i requires full knowledge of the function h (see Appendix A for more details).

In the interpolation problem, given a basis and partial sampling of h one is interested in predicting the underlying function. We will first review classical interpolation using a polynomial basis before discussing empirical interpolation with application-specific basis functions. We finish this section with an example for burst GWs.

A. Classical interpolation with polynomials

The classical interpolation problem for a function $h(x)$ is the following. Given a set of m nodes $\{x_i\}$, known function evaluations $\{h_i := h(x_i)\}$, and a polynomial basis $e_i = p_i(x)$ a polynomial of degree $i \leq m-1$, find an approximation (the interpolant)

$$\mathcal{I}_m[h](x) = \sum_{i=1}^{m} c_i p_i(x) \approx h(x) \qquad (13)$$

such that

$$\mathcal{I}_m[h](x_i) = h(x_i) \quad \text{for} \quad i = 1 \ldots m\,. \qquad (14)$$

That is, the approximant is required to agree with the function at the set of m nodes where the latter is known.

One can show that the problem Eq. (13) has a unique solution in terms of Lagrange polynomials. Given a convergence rate for the projection-based approximation Eq. (12) one might wonder how much accuracy is lost by trading it for the interpolation Eq. (13) and how to optimally choose the node points x_i. When the relevant error measurement is the maximum pointwise error, Chebyshev nodes are known to be nearly-optimal, bringing an additional error which grows like like $\log(m)$ [50, 51]. Since the reduced basis is application-specific an optimal set of interpolation points is in principle not known. Next we describe an approach for identifying such a set which is nearly-optimal.

B. Empirical interpolation with RB

Similar to the polynomial case, in ROM one seeks to find an empirical (that is, problem-dependent) *global* interpolant

$$\mathcal{I}_m[h](f;\boldsymbol{\lambda})] := \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(f)\,, \qquad (15)$$

where the c_i coefficients are defined as solutions to the interpolation problem

$$\mathcal{I}_m[h](F_k;\boldsymbol{\lambda}) = h(F_k;\boldsymbol{\lambda}), \qquad \forall\, k = 1,\ldots,m. \qquad (16)$$

In general, a well-posed interpolation problem for m basis functions will require m interpolation points $\{F_i\}_{i=1}^{m}$. Additionally, these points must ensure an accurate approximation.

The Empirical Interpolation Method was proposed in 2004 [26] as a way of identifying a good set of interpolation points for arbitrary basis sets and has since found numerous applications [27, 52–55]. Recently, the EIM was shown to dramatically speed up parameterized inner product (overlap) computations in the absence of noise [22]. Crucially, the EIM algorithm selects the interpolation points as a subset of the data samples, say (in the frequency domain case) $\{F_i\} \subset \{f_i\}$.

For the moment we shall assume that the EIM points are known (the precise way of finding them is explained in Appendix B) and proceed to describe how one uses them to find the EIM interpolant. Eq. (16) is equivalent to solving an m-by-m system $A\vec{c} = \vec{h}$ for the coefficients $\vec{c}$, where

$$A := \begin{pmatrix} e_1(F_1) & e_2(F_1) & \cdots & e_m(F_1) \\ e_1(F_2) & e_2(F_2) & \cdots & e_m(F_2) \\ e_1(F_3) & e_2(F_3) & \cdots & e_m(F_3) \\ \vdots & \vdots & \ddots & \vdots \\ e_1(F_m) & e_2(F_m) & \cdots & e_m(F_m) \end{pmatrix}.$$

The EIM algorithm ensures that the matrix A is invertible, with $\vec{c} = A^{-1}\vec{h}$ the unique solution to Eq. (16). As A is parameter independent we have, for all values of $\boldsymbol{\lambda}$,

$$\mathcal{I}_m[h](f;\boldsymbol{\lambda}) = \vec{e}^{\,T}(f)\left[A^{-1}\vec{h}(\boldsymbol{\lambda})\right], \qquad (17)$$

where $\vec{e}^{\,T} = [e_1(f),\ldots,e_m(f)]$ denotes the transpose of the basis vectors, which we continue to view as functions.

The empirical interpolant is nearly optimal in the sense that it satisfies

$$\|h(f;\boldsymbol{\lambda}) - \mathcal{I}_m[h(f;\boldsymbol{\lambda})]\|^2 \leq \Lambda_m^2 \sigma_m \,, \qquad (18)$$

where σ_m characterizes the representation error of the basis as defined in Eq. (6) and Λ_m is a computable Lebesgue constant. For more details and in the context of gravitational waves, see, for example, [22]. For problems with smooth dependence with respect to parameter variation one can expect exponential decay of σ_m with respect to m and therefore of the EIM error Eq. (18) as well.

C. An example of EIM: burst waveforms

We now provide a qualitative outline of the EIM algorithm, with more details given in Appendix B. As input the algorithm takes the basis set $\{e_i\}_{i=1}^m$ and an arbitrary number and choice of data samples $\{f_i\}$ from which the empirical interpolation points $\{F_i\}_{i=1}^m$ are to be selected.

The first point is simply chosen to maximize the value of $|e_1(f_i)|$; that is, $|e_1(F_1)| \geq |e_1(f_i)|$ for all data samples. Next, consider an empirical interpolant for the second basis function using only the first basis function. From Eqs. (15,16) or, equivalently, Eq. (17) we have $\mathcal{I}_1[e_2](f)] = c_1 e_1(f)$ where $c_1 = e_2(F_1)/e_1(F_1)$ has been found from Eq. (16) with $k = 1$. The second empirical interpolation point is chosen to maximize the value of the pointwise interpolation error of $\mathcal{I}_1[e_2](f) - e_2(f)$; that is, $|\mathcal{I}_1[e_2](F_2) - e_2(F_2)| \geq |\mathcal{I}_1[e_2](f_i) - e_2(f_i)|$ for all data samples. This procedure is then repeated to select the remaining $m - 2$ points.

As described, the EIM follows a greedy approach, albeit somewhat different from that one we used to build our basis. While a greedy algorithm to build a RB selects the most relevant points in parameter space, the EIM selects the most relevant points in the *physical dimension(s)*.

Figures 3, 4, and 5 provide a graphical illustration of the EIM algorithm and outputs for the family of sine-Gaussian burst waveforms Eq. (8), using the RB described in Sec. III B.

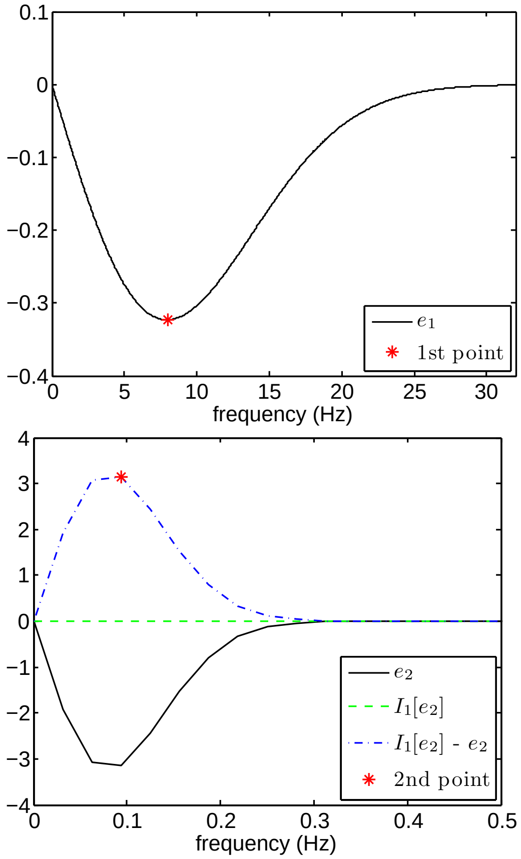

FIG. 3: Iterations 1 (top) and 2 (bottom) of the EIM algorithm. The first EIM point is defined by the location of $\max(|e_1|)$. To identify the second point we i) build the empirical interpolant $I_1[e_2]$ of e_2 using e_1 and the sample point F_1 (cf Eq. (17)), ii) compute the pointwise error $I_1[e_2] - e_2$; iii) the second EIM point is then defined by the location of $\max(|I_1[e_2] - e_2|)$. The process continues until all m empirical interpolation points are found.

V. REDUCED ORDER QUADRATURES

As anticipated and summarized in Section II, building an ROQ has *offline* and *startup* costs, with the advantage of very fast *online* evaluations. In the offline stage we construct the basis and EIM points. This stage is independent of any data/signal. The startup stage, in turn, is data-dependent and completes the ROQ which preserves the accuracy of any quadrature rule of interest, with a number of quadrature nodes which equals the number of basis functions. Roughly speaking, the accuracy of the resulting ROQ is comparable to that of the basis, with the nodes chosen as a subset of the data points at which the signal has been sampled, serving effectively as a downsampling criteria.

The details of how to construct an ROQ rule mimic well known quadratures rules. Let us briefly recall how these standard quadratures for the integration of a real function $h(x)$ are derived: the function is approximated by its polynomial interpolant (cf. Sec. IV) and the latter integrated exactly to compute the weights of the rule. Namely, given the interpolation approximation

$$h(x) \approx \sum_{i=1}^{m} h(x_i)\ell_i(x) \,,$$

where $\ell_i(x)$ are Lagrange polynomials (see Section IV A), standard quadratures are derived as

$$\int h(x)dx \approx \sum_{i=1}^{m} h(x_i)\alpha_i \qquad \alpha_i := \int \ell_i(x) \,.$$

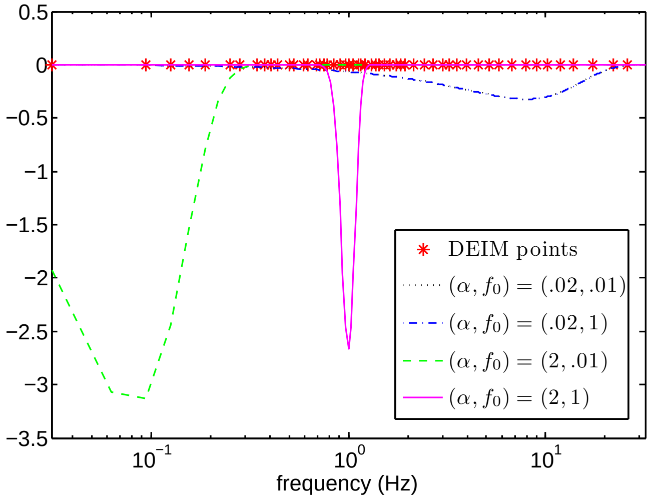

FIG. 4: Empirical interpolation points (red asterisks) selected by the EIM algorithm for the sine-Gaussian waveforms Eq. (8). These points are a subset of the original data (which in this case has equidistant spacing Δf, see Section V A) and cluster towards lower f, as expected. Four representative waveforms are depicted for all possible combinations of max/min values of f_0 and α. Greater diversity in waveform features is evident at lower frequencies.

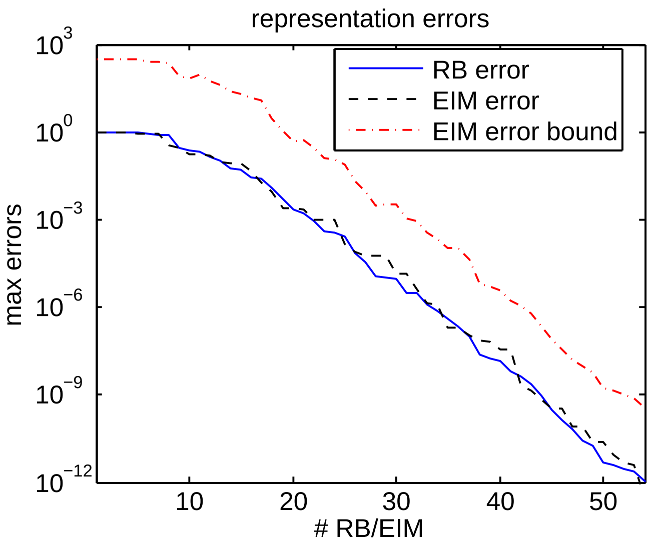

FIG. 5: Approximation error as a function of the number of Reduced Basis generated with a greedy algorithm (solid blue), and for the Empirical Interpolant (dashed black), defined as $\|h - P_n[h]\|^2$ and $\|h - I_n[h]\|^2$, respectively. The dashed red line shows the error bound Eq. (18).

Interpolation at equally spaced points for $m = 1$ leads to the trapezoidal rule, for $m = 2$ to Simpson's rule, etc. By additionally choosing the location of the interpolation points one can maximize the exactness of the quadrature rule for polynomials, leading to Gaussian quadratures.

A. Riemann sum with uniform sampling

In general, the output of a GW detector is comprised of data segments of duration T, which are uniformly sampled every Δt seconds. Then, for $N = T/(\Delta t)$ data samples

$$h(j\Delta t; \boldsymbol{\lambda}) \,, \quad j = 0 \ldots N \,,$$

the discrete Fourier transform $\tilde{h}(f_i; \boldsymbol{\lambda})$ is known at the frequencies $\{f_i\}_{i=0}^{N/2} = \{0, f_0, 2f_0, \ldots, (N/2)\, f_0\}$ where $f_0 = 1/T = (N\Delta t)^{-1} = \Delta f$ is the fundamental frequency and $f_{\rm max} = (N/2) f_0$.

Due to the fact that the data taking procedure dictates the instants of time at which the (non-smooth and noisy) signal is known, an obvious numerical approximation to Eq. (4) is a low order discrete Riemann sum,

$$\langle a\,|b\rangle \approx \langle a\,|b\rangle_{\tt d} := \frac{4}{N\Delta} \Re \sum_{i=0}^{N/2} \left[\frac{\tilde{a}(f_i)\tilde{b}^*(f_i)}{\tilde{S}_n(f_i)} \right] . \qquad (19)$$

The computational cost of Eq. (19) depends on N, which in turn depends on the data sampling rate.

Whether performing searches or parameter estimation studies, the numerical integral Eq. (19) is repeatedly evaluated for a variety of GW templates $h(f, \boldsymbol{\lambda})$. Next we show how such integrals can be computed with a cost not set by the Nyquist sampling rate but rather the "information content" of the GW templates themselves; namely, the number of basis functions, m, in Section III. This is similar in spirit to the fact that compressed sensing can beat Nyquist downsampling [56].

B. Building the ROQ

Consider a discrete approximation $\langle\cdot|\cdot\rangle_{\tt d}$ to the continuum scalar product of Eq. (4). If we have T seconds of noisy data $s(t)$ uniformly sampled at Δt second intervals, the Riemann sum Eq. (19) is a natural choice in data analysis studies, whether for Bayesian parameter estimation or searches with matched filtering.

Given the discrete Fourier transform of a data set $\tilde{s}(f_i)$ (one can similarly build an ROQ in the time domain), and specializing to white noise $\tilde{S}_n = 1$ without loss of generality (one can absorb $\tilde{S}_n$ into the definition of $\tilde{s}$), ROQ inner products between data and templates $h(f; \boldsymbol{\lambda})$ can be computed as

$$\begin{aligned}
\langle h(\boldsymbol{\lambda})|s\rangle_{\mathtt{d}} &= 4\Re\sum_{k=0}^{N/2} s^*(f_k)h(f_k;\boldsymbol{\lambda})\Delta f \\
&\approx 4\Re\sum_{k=0}^{N/2} s^*(f_k)\mathcal{I}_m[h(f_k;\boldsymbol{\lambda})]\Delta f = 4\Re\sum_{k=0}^{N/2} s^*(f_k)\left[\vec{e}^{\,T}(f_k)A^{-1}\vec{h}(\boldsymbol{\lambda})\right]\Delta f \\
&= 4\Re\left[\sum_{k=0}^{N/2} s^*(f_k)\vec{e}^{\,T}(f_k)\Delta f A^{-1}\right]\vec{h}(\boldsymbol{\lambda}) = 4\Re\sum_{k=1}^{m}\omega_k h(F_k;\boldsymbol{\lambda}) \\
&=: \langle h(\boldsymbol{\lambda})|s\rangle_{\mathtt{ROQ}}\,,
\end{aligned}$$

where

$$\omega_j := \sum_{k=0}^{N/2} s^*(f_k)e_j(f_k)\Delta f A^{-1}\,. \tag{20}$$

The vector

$$\sum_{k=0}^{N/2} s^*(f_k)\vec{e}^{\,T}(f_k)\Delta f$$

is composed of inner products between all the basis elements and the data. We refer to $\{\omega_k\}_{k=0}^m$ as *data-specific weights*; their generation comprises the ROQ *startup cost.* Defining the scalar product between the data and the j^{th} basis function by

$$E_j := \sum_{k=0}^{N/2} s^*(f_k)e_j(f_k)\Delta f\,,$$

the data-specific weights are simply

$$\vec{\omega}^{\,T} = \vec{E}^{\,T}A^{-1}\,. \tag{21}$$

Notice that the ROQ nodes are exactly the EIM points which, together with the weights (20), completes our ROQ approximation

$$\langle h(;\boldsymbol{\lambda})|s\rangle_{\mathtt{ROQ}} := 4\Re\sum_{k=1}^{m}\omega_k h(F_k;\boldsymbol{\lambda})\,. \tag{22}$$

The ROQ rule's accuracy only depends on the interpolant's accuracy to represent $h(f;\boldsymbol{\lambda})$ and the accuracy of the original quadrature $\langle\cdot|\cdot\rangle_{\mathtt{d}}$. In particular, the method does *not* assume s to be well approximated by the basis (i.e. neither waveform modeling assumptions nor details about the noise realization are important). Since, as discussed, the error of the interpolant, (18) can be expected to decay exponentially for the cases of interest, in practice the ROQ replaces the original quadrature rule by a less expensive one with the same accuracy (within, say, machine precision). How much smaller m is compared to $N/2$ is model-dependent; in Section VI we quantify this for the family of burst waveforms described in Eq. (8).

Figure 6 shows the nodes chosen by the EIM in the frequency domain for the model burst waveforms Eq. (8), and the ROQ weights from Eq. (21), for the case of mock data with and without noise. Figure 7, in turn, shows the error

$$|\langle h(;\boldsymbol{\lambda})|s\rangle_{\mathtt{ROQ}} - \langle h(;\boldsymbol{\lambda})|s\rangle_{\mathtt{d}}|$$

in the computation of the overlap between burst waveforms and data with and without noise. Here the *max errors* label on the vertical axis refers to the maximum error found in a thorough sampling of the parameter range Eq. (9), and the error is relative to a standard Riemann-sum integration Eq. (19) with 1025 points.

C. Extrinsic parameters

The ROQ are built over the intrinsic parameters characterizing the waveform signal, but there are also extrinsic parameters which affect the signal. These include the arrival time of the signal, which we take to be the time at the midpoint of the burst, t_c, the phase of the waveform at this time, and parameters such as the sky position, orientation and distance to the source. The phase of the waveform affects the model simply as multiplication by a complex constant, which keeps the waveform in the RB space. Similarly, sky position, orientation and distance

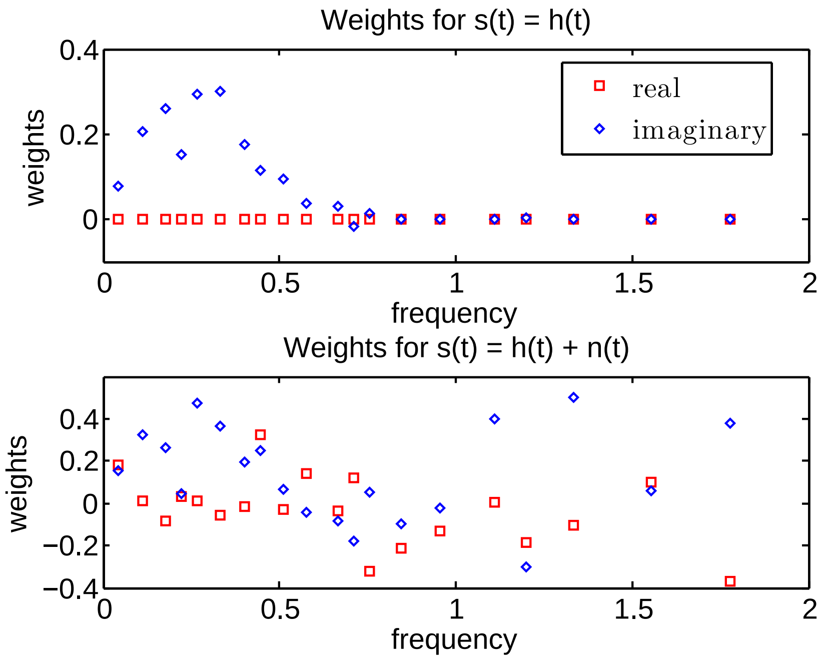

FIG. 6: Real (blue squares) and imaginary (red diamonds) ROQ weights computed from Eq. (21) for the test burst family waveforms Eq. (7) in the range Eq. (9) and the injected signal with default parameters Eq. (11). The top figure is for the noise-free case when $s(t) = h(t)$ while the bottom figure shows weights when $s(t) = h(t) + n(t)$, where $n(t)$ is a particular noise realization.

just affect the amplitude of the source and the projection of the plus and cross polarizations of the waveform into a detector response and also do not take the waveform out of the RB space. The arrival time requires some more discussion.

If we denote by $\boldsymbol{\lambda}$ the set of parameters excluding t_c and by $h_0(t;\boldsymbol{\lambda})$ the waveform computed with $t_c = 0$ then

$$h(t;t_c,\boldsymbol{\lambda}) = h_0(t-t_c;\boldsymbol{\lambda}). \qquad (23)$$

and its Fourier transform is

$$\tilde{h}(f;t_c,\lambda) = \tilde{h}_0(f;\lambda)\mathrm{e}^{2\pi\mathrm{i}ft_c}. \qquad (24)$$

For parameter estimation one computes integrals of the form

$$O(t_c,\lambda) := \int_0^\infty \frac{\tilde{h}(f;t_c,\lambda)\tilde{s}^*(f)}{S_n(f)}\,\mathrm{d}f. \qquad (25)$$

The simple dependence of the Fourier transform is exploited in GW searches by defining the function $\tilde{I}_0(f;\lambda)$ via

$$\tilde{I}_0(f;\lambda) = \frac{\tilde{h}_0(f;\lambda)\tilde{s}^*(f)}{S_n(f)} \qquad (26)$$

for which

$$O(t_c,\lambda) = \int_0^\infty \tilde{I}_0(f;\lambda)\mathrm{e}^{2\pi\mathrm{i}ft_c}\mathrm{d}f = 2\pi I_0(-t_c;\lambda)\,, \qquad (27)$$

where $I_0(t;\lambda)$ is the inverse Fourier transform of $\tilde{I}_0(f;\lambda)$. Since FFTs are efficient, one can search over t_c cheaply by doing this inverse Fourier transform.

The ROQ rule that we have computed for waveforms $h_0(t;\lambda)$ enables us to compute the integral of $\tilde{I}_0(f;\lambda)$ cheaply. However, we now need to compute the integral of $\tilde{I}_0(f;\lambda)\exp(2\pi\mathrm{i}ft_c)$ and so the existing ROQ rule is in principle not guaranteed to work. However, if the ROQ is being used for follow-up parameter estimation, this will normally be triggered by the detection of a candidate event in the data stream of one or more detectors. These triggers will normally be able to localise the event to within a time interval comparable to a couple of cycles of the signal. The simplest approach to handling t_c, which works well in practice, is to build an ROQ rule for the estimated value of t_c (which can be $t_c = 0$ without loss of generality) and use it for other arrival times within a reasonable window around that value. One can include the arrival time information when building the weights with Eq. (20) *at no extra cost*. Figure 8 plots the error that arises from using the $t_c = 0$ ROQ rule for non-zero t_c's in the burst waveforms (8).

D. Computing the likelihood

To evaluate the likelihood we need to compute Eq. (3) as

$$\langle s|s\rangle + \langle h\left(\boldsymbol{\lambda}\right)|h\left(\boldsymbol{\lambda}\right)\rangle - 2\Re\langle s|h\left(\boldsymbol{\lambda}\right)\rangle\,, \qquad (28)$$

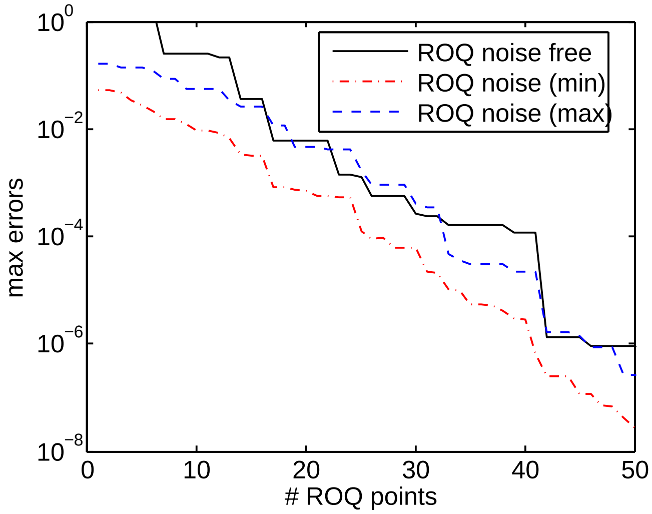

FIG. 7: Integration error $|\langle h(;\boldsymbol{\lambda})|s\rangle_{\texttt{ROQ}} - \langle h(;\boldsymbol{\lambda})|s\rangle_{\texttt{d}}|$ versus number of ROQ nodal points (m) for 10,000 randomly selected values of $\boldsymbol{\lambda}$. The solid black curve depicts the noise-free case $s = h$. The last data point corresponds to the rule used to construct Figure 8. The blue and red curves show the maximum and minimum, over 100 realizations of pure noise data, $s = n$, of the error maximised over all parameter values $\boldsymbol{\lambda}$. *Note that the ROQ shows exponential convergence with respect to the number of ROQ nodes, even for pure noise data.*

where the last term is handled with the ROQ rule (22) and the first term needs to be computed once. In the case of sine-gaussian waveforms in white noise we can compute a closed-form expression for the norm

$$\langle h(\boldsymbol{\lambda})\,|h(\boldsymbol{\lambda})\rangle = 4A^2\alpha\sqrt{\pi}\left(1 - e^{-4\pi^2 f_0^2\alpha^2}\right) \tag{29}$$

when $f_{\min} = 0$ and $f_{\max} = \infty$. When closed-form expressions are unavailable we have a few options. One possibility is to build an ROQ rule for the norm which requires additional offline computations. Here we consider an alternative. Notice that the norm

$$\langle h(\boldsymbol{\lambda})\,|h(\boldsymbol{\lambda})\rangle = \sum_{i=1}^{m} c_i^2 \tag{30}$$

is expressible in terms of the EIM coefficients $\vec{c} = A^{-1}\vec{h}$. Explicit computation of these coefficients carries an $\mathcal{O}(m^2)$ cost which is larger than the ROQ count of $\mathcal{O}(m)$. However, in most applications the waveforms themselves are very expensive to compute and so this cost will still be much smaller than the full likelihood evaluation.

VI. RESULTS

When dealing with high dimensional problems, the process of mapping the likelihood (or the posterior) surface can become very expensive. MCMC algorithms are some of the most efficient methods to compute posterior distributions, but even so higher dimensional problems present a challenge. MCMC methods are likely to be used to carry out parameter inference on gravitational wave data in order to provide estimates of instrumental noise and parameter errors and to enable comparison between different model descriptions of the data. In this section we illustrate how MCMC methods can be accelerated by using ROQ likelihoods.

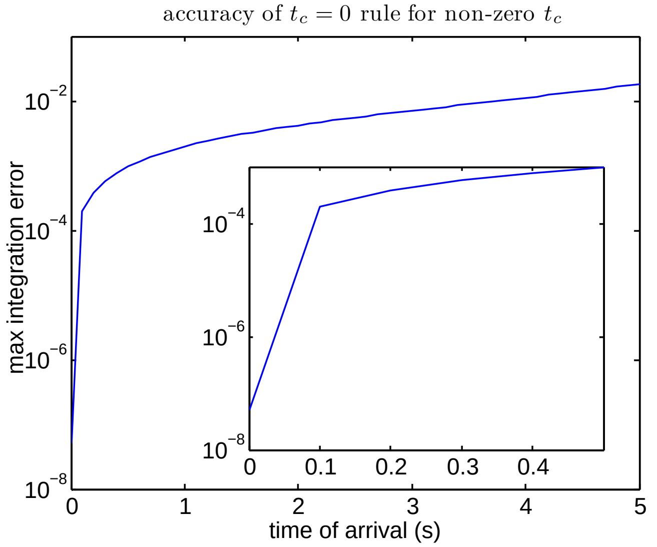

FIG. 8: Errors in computing $\langle s\,|h(\boldsymbol{\lambda}, t_c)\rangle$ using an ROQ rule built for $t_c = 0$ with accuracy better than $\sim 10^{-6}$. Empirically we find that this rule continues to work well for non-zero values of t_c. Looking ahead to Sec. VI we anticipate evaluating the likelihood function for $t_c \leq 0.5\,$sec.

An MCMC algorithm aims to find a chain of $N_{\texttt{mcmc}}$ samples, $\{\mathbf{x}_i\}$, that are distributed according to the target probability distribution, $p_{\rm t}(\mathbf{x}_i)$, such that integrals over the probability distribution can be approximated by sums over the points in the chain

$$\int p_{\rm t}(\mathbf{x}) f(\mathbf{x}) \mathrm{d}\mathbf{x} \approx \sum_{i=1}^{N_{\texttt{mcmc}}} f(\mathbf{x}_i). \tag{31}$$

The chain of points can be obtained using the Metropolis-Hastings algorithm [57]. The first point, $\mathbf{x}_1$, is chosen at random from the prior. At iteration i a new point $\mathbf{y}_i$ is drawn from a proposal distribution $q(\mathbf{y}_i|\mathbf{x}_i)$ and the Metropolis-Hastings ratio, r, evaluated

$$r = \frac{p_{\rm t}(\mathbf{y}_i) q(\mathbf{x}_i|\mathbf{y}_i)}{p_{\rm t}(\mathbf{x}_i) q(\mathbf{y}_i|\mathbf{x}_i)}. \tag{32}$$

A random number $u \in U[0,1]$ is drawn and if $u < r$ the move is accepted, $\mathbf{x}_{i+1} = \mathbf{y}_i$; otherwise the move is rejected and $\mathbf{x}_{i+1} = \mathbf{x}_i$.

In our case, the target distribution is the posterior probability distribution given by Eq. (1), which depends on the likelihood and can therefore be approximated using the ROQ. To illustrate the method, we will consider the problem of recovering the parameters of a burst signal of the form given in Eq. (7) from a noisy data stream.

We include Gaussian white noise with unit power spectral density, $\tilde{S}_n(f) = 1$, and take the parameters of the true signal to be our default ones (11). We assume the observation is 32 sec long and the data is sampled at 64Hz. We use a symmetric Gaussian proposal distribution

$$q(\mathbf{y}_i|\mathbf{x}_i) \propto \exp\left[-\Gamma_{jk}(x_i^j - y_i^j)(x_i^k - y_i^k)/2\right] ,$$

where $\Gamma_{jk} = \langle \partial_j h | \partial_k h \rangle$ is the Fisher information matrix. We use priors on f and α that span the range over which the RB and ROQ were built, given by Eq. (9), and priors for the other parameters of $t_c \in [-2, 2]$, and $A \in [0.1, 10]$. In order to compare the cost and accuracy of the *standard*, or *full* MCMC computation vs the ROQ one, we repeat the analysis using the same data, number of MCMC points, proposal distribution and priors but changing from the standard, full expensive likelihood to the ROQ likelihood one. The results are presented in the following sections.

A. Two parameter search

As a first test we restrict the search to two parameters — $\{f, \alpha\}$ — while fixing t_c and A to the injected values. In Table I we compare the parameter values recovered using the full data set and Riemann sums with those recovered from ROQ likelihoods in one particular noise realization for each of four different SNRs of the injected source. The values are quoted as $\mu_\mathrm{i} \pm \sigma_\mathrm{i}$, where the one dimensional marginalised posterior mean, μ_i, and standard deviation, σ_i, in parameter i are defined from the set of MCMC samples $\{\mathbf{x}_j\}$ by

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} x_j^i, \qquad \sigma_i^2 = \frac{1}{N-1}\sum_{j=1}^{N}\left(x_i^j - \mu_i\right)^2 . \quad (33)$$

It is clear that in all cases the statistics of the posterior distribution are completely consistent between the full likelihood and ROQ likelihood computations. The only differences are beyond the significant digits quoted in the Table and are much smaller than the corresponding uncertainty in the parameter values arising from noise in the data stream. In this case, the ROQ likelihood is extremely accurate, with differences of 10^{-6} or smaller, so it is not surprising that the results are indistinguishable.

We can also ask whether the full posterior distributions are consistent between the two likelihoods. This can be achieved by using a Kolmogorov-Smirnov (KS) test [50] to compare the 1D and 2D marginalised posteriors obtained using the two different likelihoods. Figure 9 shows the 1D marginalised posteriors for f and α computed using the two likelihoods. These are indistinguishable by eye and, more precisely, the p-value of the KS test that the distributions agree are 1.0 for both f and α, so there is no evidence of a difference in the recovered posteriors. This is to be expected because of the high accuray of the ROQ likelihood in this case.

		Recovered values	
SNR	Method	f	α
5	Full	0.189 ± 0.095	0.831 ± 0.194
	ROQ	0.189 ± 0.095	0.831 ± 0.194
10	Full	0.172 ± 0.081	0.803 ± 0.136
	ROQ	0.172 ± 0.081	0.803 ± 0.136
20	Full	0.168 ± 0.075	0.800 ± 0.108
	ROQ	0.168 ± 0.075	0.800 ± 0.108
40	Full	0.212 ± 0.051	0.872 ± 0.091
	ROQ	0.212 ± 0.051	0.872 ± 0.091

TABLE I: Parameter values recovered using the full and ROQ likelihoods. Values quoted are the mean and standard deviation estimated from the posterior for a particular noise realisation. The same noise realisation is used for the full and ROQ likelihood calculation for each SNR.

B. Four parameter search

We now consider a search over the full four dimensional parameter space of $\{f, \alpha, t_c, A\}$. The 1D and 2D marginalised posteriors for a typical noise realisation computed using both the full and ROQ likelihoods are shown in Figure 10, while Table II lists the posterior means and standard deviations found in a particular noise realisation using both techniques for a variety of SNRs of the true source. As in the two parameter case we find that the statistics derived from the posterior distributions are completely consistent between the full and ROQ likelihoods. The few small differences are in the last significant figures and are much smaller than the typical width of the posterior distributions. The 1D and 2D marginalised distributions also pass the KS test with similar p-values to the two parameter case.

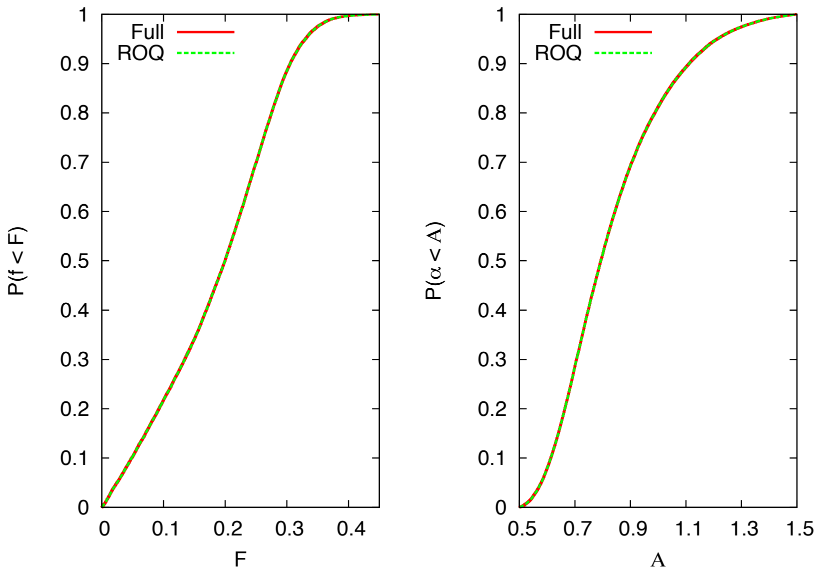

FIG. 9: Marginalised cumulative probability distributions for f (left panel) and α (right panel) for a true source with SNR $\rho = 5$. Each panel contains two curves which lie on top of each other, one computed using the Full likelihood and one using the ROQ likelihood. The KS test confirmed the two distributions were the same.

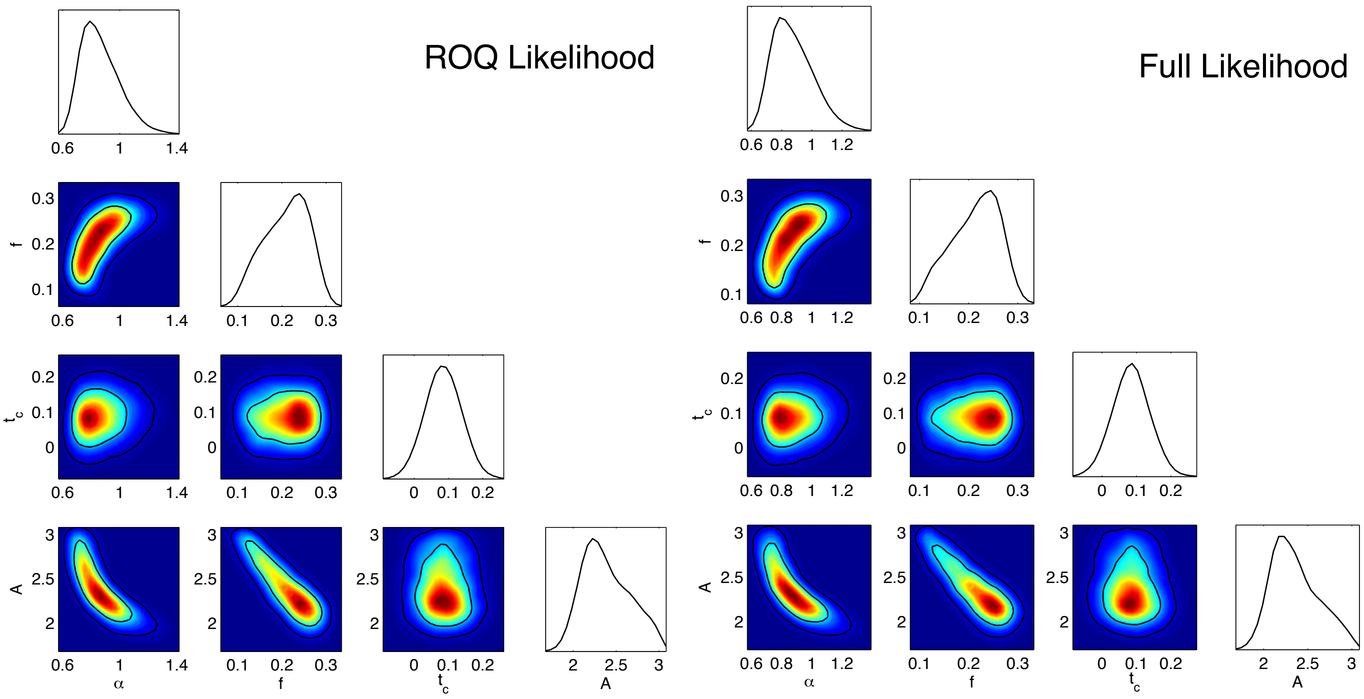

FIG. 10: PDFs obtained for an injected source with SNR $\rho = 10$, employing standard and ROQ MCMC computations in a four-parameter space, namely A, t_c, f and α. The figures show the agreement between the two techniques.

Having established the equivalence of the results for the full and ROQ likelihoods, we can now compare the run time. The ROQ likelihood has a greater initial cost, since the data-specific weights have to be computed using Eq. (21) prior to beginning the MCMC. In general this start up cost is a tiny fraction of the total run time of the MCMC algorithm. In the case of these burst waveforms, the total time taken to compute the weights is $\sim$ 10ms, which is approximately the time it takes to do a $\sim$ 100 point MCMC search using the full likelihood. In Figure 11 we show the time taken to run the MCMC search, i.e., after the initial set-up time, using the full likelihood

		Recovered values			
SNR	Method	f	α	t_c	A
5	Full	0.217 ± 0.069	0.896 ± 0.194	0.068 ± 0.104	1.704 ± 0.379
	ROQ	0.217 ± 0.068	0.897 ± 0.196	0.069 ± 0.104	1.702 ± 0.375
10	Full	0.212 ± 0.048	0.875 ± 0.132	0.084 ± 0.053	2.362 ± 0.278
	ROQ	0.209 ± 0.050	0.866 ± 0.132	0.085 ± 0.052	2.387 ± 0.287
20	Full	0.225 ± 0.029	0.891 ± 0.093	0.092 ± 0.028	2.944 ± 0.176
	ROQ	0.224 ± 0.029	0.892 ± 0.093	0.093 ± 0.028	2.944 ± 0.177
40	Full	0.248 ± 0.009	0.981 ± 0.041	0.097 ± 0.016	3.471 ± 0.157
	ROQ	0.248 ± 0.009	0.981 ± 0.042	0.097 ± 0.016	3.471 ± 0.157

TABLE II: As Table I but for searches over the full set of four parameters.

and using the ROQ likelihood. Figure 12 shows the ratio of the runtimes for the ROQ and full searches.

It is clear from these figures that the cost of the MCMC search grows linearly with the number of MCMC points, as we would expect, since the run-time is determined primarily by the cost of likelihood evaluations. The speed-up from using the ROQ is, in this case, a factor of ~ 25. This factor will of course be problem and implementation specific but is roughly the ratio between the total number of frequency samples $N/2$ and the number of EIM subsamples m. This ratio will depend on various aspects of the problem — the sampling cadence, total observation time, the allowed range for the parameters and the waveform model under consideration. For example, if we know in advance the frequency and duration of the burst then carefully choosing a sampling rate and observation time just large enough for the source in question reduces the speed-up to ~ 10. Such tuning of the cadence and observation time is effectively a compression of the likelihood, and is very effective for a simple model of this type. The fact that even after such tuning the ROQ rule can show a significant speed-up illustrates the power of the method. In other problems, speed-up factors of 10–100 are typical and factors of 1000 are possible, but these have to be computed on a case by case basis and will be reported elsewhere. An investigation of the speed-ups for inspiral waveforms is currently underway.

VII. SUMMARY

In this paper we have proposed using a modification of the Reduced Order Quadratures of Reference [22] for fast, accurate online evaluations of the correlation between a given data stream and a family of gravitational waveforms. The modification is designed for Markov chain Monte Carlo simulations for parameter estimation and as such, it is adapted to a particular stream of (noisy) data. The resulting speed-up is not at the expense of reduced accuracy but, instead, Reduced Order Modeling is used to build application and data-specific quadratures for the problem at hand.

For the particular application considered here as an illustration of the concept, models of burst gravitational waves, we have found speed-ups of $\sim \times 25$, depending on settings such as the central frequency of the wave, the damping factor, observation period and sampling rate. These speedups are expected to increase with the complexity and fidelity of the modeling. For example, in [44] it was found that the number of Reduced Basis waveforms needed to represent the space of inspiral waveforms in the post-Newtonian stationary phase approximation barely increases when (non-precessing) spins are taken into account. Since ROQ by design uses the same number of nodal points as the number of basis functions needed to represent the space of waveforms within a given accuracy, the approach holds the promise of beating the curse of dimensionality. There is also evidence that the case of precessing binaries is amenable to dimensional reduction [58].

VIII. ACKNOWLEDGMENTS

This work was supported in part by NSF Grants PHY0801213 and PHY1005632 to the University of Maryland. PC's work is supported by a Marie Curie Intra-European Fellowship within the 7th European Community Framework Programme (PIEF-GA-2011-299190), and thanks the University of Maryland for hospitality while part of this work was completed. JG's work is supported by the Royal Society. MT thanks the National Institute for Theoretical Physics (NiTheP) at Stellenbosch University, South Africa, where part

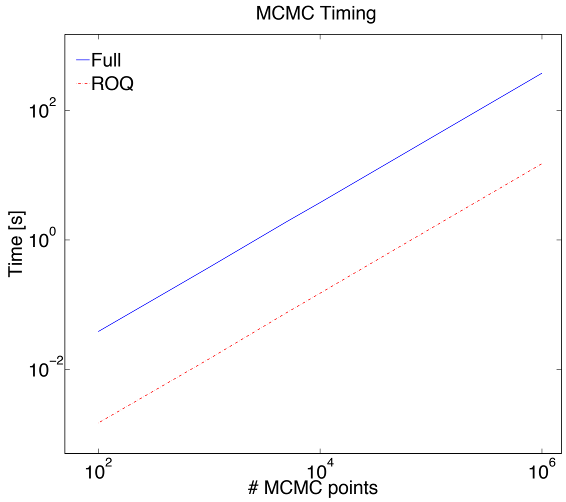

FIG. 11: Runtime as a function of MCMC chains for $N = 10^6$ samples in the frequency domain. The red (dotted) line shows the timing for standard MCMC computations and the blue (dashed) line the timing for the ROQ computations.

of this work was done, for its hospitality. We thank Chad Galley for helpful discussions and comments on the manuscript. We also thank Harbir Antil, Collin Capano, Philip Graff, Frank Herrmann, Jason Kaye, Tyson Littenberg, Ilya Mandel, Evan Ochsner, Ricardo Nochetto and Rory Smith for helpful and insightful discussions.

Appendix A: Reduced Basis

In its simplest form, such as when the waveforms are inexpensive to compute, the greedy algorithm for building a Reduced Basis has as input a set of parameter values

$$\mathcal{T} := \{\boldsymbol{\lambda}_i\}_{i=1}^{M} \tag{A1}$$

usually called *training points*, and associated waveforms $\{h(\cdot;\boldsymbol{\lambda}_i)\}_{i=1}^{M}$, usually called the *training set*.

Part of the output is a hierarchical set of parameter values $\{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \cdots, \boldsymbol{\lambda}_m\} \subseteq \mathcal{T}$ (with $m \leq M$, and $m < M$ or even $m \ll M$ if the problem is amenable to dimensional reduction) called the greedy points, and associated waveforms, which constitute the Reduced Basis,

$$\text{RB} := \{e_1(\cdot) := h(\cdot, \boldsymbol{\lambda}_1), \cdots, e_m(\cdot) := h(\cdot, \boldsymbol{\lambda}_m)\}\,. \tag{A2}$$

The RB serves as a representation of the waveforms in the training set and, if the latter is dense enough, of the whole continuum. The optimal representation by a basis is known to be the orthogonal projection P_m onto its span. This result is a standard linear algebra result, independent of Reduced Basis or Reduced Order Modeling. That is, the approximation

$$h(\cdot;\boldsymbol{\lambda}) \approx \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot) \tag{A3}$$

minimizes the error,

$$\left\| h(\cdot;\boldsymbol{\lambda}) - \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot) \right\|^2 ,$$

when the coefficients c_i are chosen such that the approximant satisfies

$$\left\langle h(\cdot;\boldsymbol{\lambda}) - \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot) \middle| e_j(\cdot) \right\rangle = 0 \quad \forall \quad e_j \in \text{RB}\,. \tag{A4}$$

The solution to (A4) is

$$c_i(\boldsymbol{\lambda}) = \sum_{j=1}^{m} (G^{-1})_{ij} \langle h(\cdot;\boldsymbol{\lambda}) | e_j(\cdot)\rangle\,, \tag{A5}$$

where G^{-1} is the inverse of the *Grammian* or *Gram matrix* G, with entries

$$G_{ij} := \langle e_i | e_j \rangle\,.$$

If the basis is orthonormal, this matrix is the identity and one recovers the familiar expression

$$h \approx \sum_{i=1}^{m} \langle h | e_i \rangle e_i\,.$$

In general the RB waveforms selected by the greedy algorithm will not be orthonormal. Then at each greedy iteration one can use a Gram-Schmidt (GS) procedure to orthonormalize the RB or, equivalently, simply invert the Gram matrix. In either case, for any given basis, the optimal approximation (A3) is given by

$$h(\cdot;\boldsymbol{\lambda}) \approx \mathcal{P}_m h(\cdot;\boldsymbol{\lambda}) := \sum_{i=1}^{m} c_i(\boldsymbol{\lambda}) e_i(\cdot)\,, \tag{A6}$$

with the coefficients c_i given by (A5). Notice that since the approximant (A6) is defined in a completely geometric way, as the orthogonal projection onto the *span* of the RB elements, it is independent of whether a GS procedure is carried out or not. The RB Eq. (A2), at the same time, is composed of a set of the "most relevant" *physical waveforms*.

The precise algorithm to choose the greedy points is described in Alg. 1. Given an arbitrary user-defined tolerance error ϵ, the algorithm stops when the approximation Eq. (A6) meets the tolerance,

$$\|h(\cdot;\boldsymbol{\lambda}) - \mathcal{P}_m h(\cdot;\boldsymbol{\lambda})\|^2 \leq \epsilon \;\; \forall\; \boldsymbol{\lambda} \in \mathcal{T}.$$

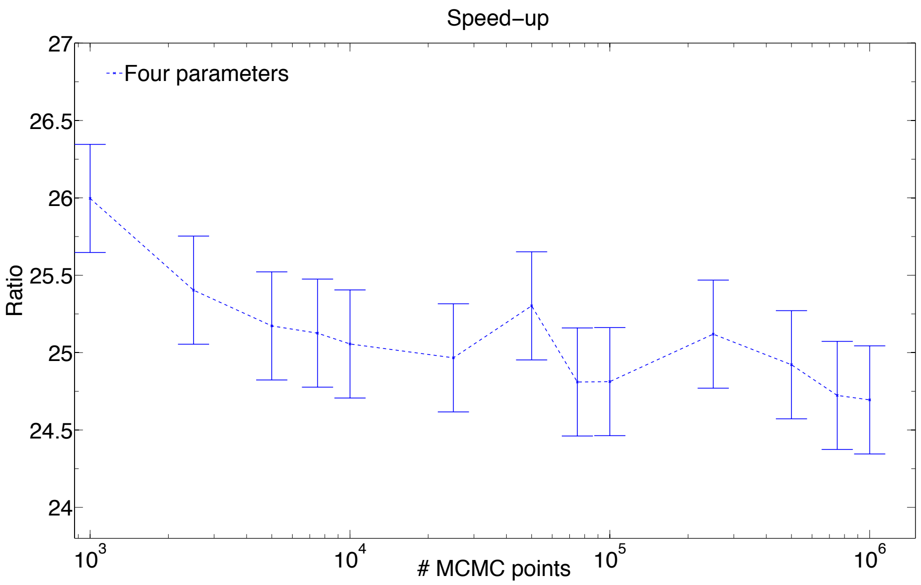

FIG. 12: Time ratio (speed-up) of MCMC simulations using a standard quadrature rule and the ROQ one. The figures show the mean of the speed-up obtained by performing simulations with different seed values for the MCMC.

In all expressions the scalar product $\langle | \rangle$ and its associated norm might be weighted. In the context of gravitational wave physics a natural choice is that one given by Eq. (4), but any other choice is possible.

Algorithm 1 Brief description of the Greedy Algorithm

```
**Input:** $\{\boldsymbol{\lambda}_i\,, h(\cdot;\boldsymbol{\lambda}_i)\}_{i=1}^{M}$, $\epsilon$

**Seed choice** (arbitrary): $\boldsymbol{\lambda}_1$
RB $= \{h(\cdot;\boldsymbol{\lambda}_1)\}$
$i = 1$ and $\sigma_1 = 1$
**while** $\sigma_i \geq \epsilon$ **do**
    $i = i + 1$
    $\sigma_i = \max_{\boldsymbol{\lambda} \in \mathcal{T}} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{P}_{(i-1)} h(\cdot;\boldsymbol{\lambda})\|^2$
    $\boldsymbol{\lambda}_i = \text{argmax}_{\boldsymbol{\lambda} \in \mathcal{T}} \|h(\cdot;\boldsymbol{\lambda}) - \mathcal{P}_{(i-1)} h(\cdot;\boldsymbol{\lambda})\|^2$
    RB $=$ RB $\cup\, h(\cdot, \boldsymbol{\lambda}_i)$
**end while**

**Output:** RB and greedy points
```

Appendix B: The Empirical Interpolation Method

As mentioned, the EIM approach is very different, in goals and scope, to any variation of standard polynomial interpolation, which was described for completeness in Section IV A. The goal of EIM is to deal with parametrized problems, and as such the set of EIM points is nested and hierarchical as one would want when solving equations, the EIM serves as a downsampling criteria, it is nearly optimal (see Eq. 18), designed for unstructured meshes in several dimensions, etc.

The input of EIM is a basis $\{e_i(\cdot)\}_{i=1}^{m}$ whose span accurately approximates the functions of interest. For definiteness we will denote by x the physical dimension(s) and $\boldsymbol{\lambda}$ the parametrization of the functions $h(x;\boldsymbol{\lambda})$ we seek to approximate. For example, if h is a gravitational wave x could denote time or frequency, and $\boldsymbol{\lambda}$ the intrinsic or extrinsic parameters of the system. Let a boldface $\mathbf{x} = (x_1, x_2, \ldots, x_N)^T$ denote a length N vector of points and $\mathbf{h} = h(\mathbf{x};\boldsymbol{\lambda})$ denote the gravitational waveform sampled at these N points. Similarly, $\mathbf{e_i}$ denotes the i^{th} basis function evaluated at N time or frequency samples.

Given an input of m *evaluated* basis functions $\{\mathbf{e_i}\}_{i=1}^{m}$ the output of the EIM algorithm is a set of m EIM points

$$\{\mathbf{X}_i\}_{i=1}^{m} \tag{B1}$$

selected as a subset of $\mathbf{x}$. If a function $h(x;\boldsymbol{\lambda})$ is known at the EIM points $\{\mathbf{X}_i\}_{i=1}^{m}$, the EIM interpolant can predict with high accuracy the function at any other value of $\mathbf{x}$. It is an interpolant in the usual sense, meaning that it agrees with the interpolated function at the interpolation points,

$$\mathcal{I}_m[h](\mathbf{X}_i, \boldsymbol{\lambda}) = h(\mathbf{X}_i, \boldsymbol{\lambda}) \quad \text{for } i = 1, \ldots, m\,.$$

The EIM interpolant is given by Eq. (17), while the selection of the EIM points is described in Algorithm 2. To assist with the description of the EIM algorithm we define the j-term empirical interpolant built from the first j basis functions and points to be

$$\mathcal{I}_j[h](x;\boldsymbol{\lambda}) := \sum_{i=1}^{j} c_i(\boldsymbol{\lambda}) e_i(x)\,, \tag{B2}$$

where the c_i coefficients are solutions to the j-point interpolation problem

$$\mathcal{I}_j[h](X_k;\boldsymbol{\lambda}) = h(X_k;\boldsymbol{\lambda}), \qquad \forall\, k = 1,\ldots,j. \tag{B3}$$

Algorithm 2 Selection of EIM Points

```
Input: Basis vectors $\{\mathbf{e_i}\}_{i=1}^m$ and points $\mathbf{x}$

$i = \text{argmax}|\mathbf{e_1}|$ Comment: here argmax takes a vector and returns the index of its largest entry.
Set $\mathbf{X}_1 = \mathbf{x}_i$

for $j = 2 \to m$ do
    Find $\mathcal{I}_{j-1}[e_j](\mathbf{x})$
    Compute the point-wise error $\mathbf{r} = \mathcal{I}_{j-1}[e_j](\mathbf{x}) - \mathbf{e_j}$
    $i = \text{argmax}|\mathbf{r}|$
    Set $\mathbf{X}_j = \mathbf{x}_i$
end for
Output: EIM points $\mathbf{X}$
```

Comments

1. In standard polynomial interpolation the interpolant is a linear combination of polynomials and function values, as in Eq (13). In the EIM the interpolant is a linear combination of (in the case of interest for this paper), waveforms and function values in the physical dimension(s), as given more precisely by Eq. (15). Parametrization and "physical" dimensions play a dual role.

2. Unlike Gaussian (e.g., Chebyshev) interpolation nodes, EIM nodes are nested and hierarchical. Given a hierarchical basis

$$\{e_1(\cdot)\} \subset \{e_1(\cdot), e_2(\cdot)\} \subset \ldots \subset \{e_i(\cdot)\}_{i=1}^m$$

an associated set of EIM points

$$\{\mathbf{X}_1\} \subset \{\mathbf{X}_1, \mathbf{X_2}\} \subset \ldots \subset \{\mathbf{X}_i\}_{i=1}^m$$

is defined. Each set of p EIM nodes is included within the set of p' EIM nodes whenever $p' > p$ and only depends on the basis of dimension p.

[1] LIGO - http://www.ligo.caltech.edu/.

[2] Virgo - http://www.virgo.infn.it/.

[3] GEO600 - http://www.geo600.uni-hannover.de/.

[4] KAGRA - http://gwcenter.icrr.u-tokyo.ac.jp/en/.

[5] LIGO Scientific, J. Abadie *et al.*, Class. Quantum Grav. **27**, 173001 (2010), arXiv:1003.2480.

[6] D. A. Brown, A. Lundgren, and R. O'Shaughnessy, (2012), arXiv:1203.6060.

[7] P. Ajith, Phys. Rev. **D84**, 084037 (2011), arXiv:1107.1267.

[8] J. M. Centrella, J. G. Baker, B. J. Kelly, and J. R. van Meter, Ann.Rev.Nucl.Part.Sci. **60**, 75 (2010), arXiv:1010.2165.

[9] A. Buonanno, Y. Chen, Y. Pan, and M. Vallisneri, Phys. Rev. D **70**, 104003 (2004).

[10] A. Buonanno, Y.-b. Chen, and M. Vallisneri, Phys. Rev. **D67**, 104025 (2003), arXiv:gr-qc/0211087.

[11] Y. Pan, A. Buonanno, Y. Chen, and M. Vallisneri, Phys. Rev. D **69**, 104017 (2004).

[12] T. B. Littenberg, J. G. Baker, A. Buonanno, and B. J. Kelly, (2012), arXiv:1210.0893.

[13] K. Chatziioannou, N. Yunes, and N. Cornish, (2012), arXiv:1204.2585.

[14] N. Yunes and F. Pretorius, Phys.Rev. **D80**, 122003 (2009), arXiv:0909.3328.

[15] P. Canizares, J. R. Gair, and C. F. Sopuerta, Phys.Rev. **D86**, 044010 (2012), arXiv:1205.1253.

[16] M. Vallisneri, Phys. Rev. Lett. **107**, 191104 (2011), arXiv:1108.1158.

[17] S. Mitra, S. V. Dhurandhar, and L. S. Finn, Phys. Rev. **D72**, 102001 (2005), arXiv:gr-qc/0507011.

[18] J. Kaye, *The interpolation of gravitational waveforms*, PhD thesis, Scientific Computing Group, Brown University, 2012.

[19] K. Cannon, J. Emberson, C. Hanna, D. Keppel, and H. Pfeiffer, (2012), arXiv:1211.7095.

[20] R. Smith, K. Cannon, C. Hanna, D. Keppel, and I. Mandel, (2012), arXiv:1211.1254.

[21] P. Graff, F. Feroz, M. Hobson, and A. Lasenby, Monthly Notices of the Royal Astronomical Society (2012).

[22] H. Antil, S. Field, F. Herrmann, R. Nochetto, and M. Tiglio, (2012), arXiv:1210.0577 [cs.NA], Submitted to the Journal of Scientific Computing.

[23] B. J. Owen, Phys. Rev. **D53**, 6749 (1996), arXiv:gr-qc/9511032.

[24] S. Babak, J. R. Gair, and E. K. Porter, Class.Quant.Grav. **26**, 135004 (2009), arXiv:0902.4133.

[25] B. Allen, W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. Creighton, (2005), arXiv:gr-qc/0509116.

[26] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, Comptes Rendus Mathematique **339**, 667 (2004).

[27] Y. Maday, N. C. Nguyen, A. T. Patera, and S. H. Pau, Communications on Pure and Applied Analysis **8**, 383 (2009).

[28] LIGO Scientific Collaboration, B. Abbott *et al.*, Phys.Rev. **D80**, 102001 (2009), arXiv:0905.0020.

[29] LIGO Collaboration, Virgo Collaboration, J. Abadie *et al.*, Phys.Rev. **D81**, 102001 (2010), arXiv:1002.1036.

[30] LIGO Scientific Collaboration, Virgo Collaboration, J. Abadie *et al.*, Phys.Rev. **D85**, 122007 (2012), arXiv:1202.2788.

[31] R. Pinnau, Model reduction via proper orthogonal decomposition, in *Model Order Reduction: Theory, Research Aspects and Applications*, edited by W. H. A.Schilders, H. A. van der Vorst, and J. Rommes, , Mathematics in Industry Vol. 13, pp. 95–109, Springer Berlin Heidelberg, 2008.

[32] G. W. Stewart, SIAM Rev. **35**, 551 (1993).

[33] Y. Maday, A. T. Patera, and G. Turinici, J. Sci. Comput. **17**, 437 (2002).

[34] K. Veroy, C. Prud'homme, and A. T. Patera, Comptes Rendus Mathematique **337**, 619 (2003).

[35] C. Prud'homme *et al.*, Journal of Fluids Engineering **124**, 70 (2002).

[36] K. Veroy and A. T. Patera, International Journal of Numerical Methods in Fluids **47**, 773 (2005).

[37] A. Patera and G. Rozza, Arch. Comput. Methods Eng. **15**, 229 (2008).

[38] N.-C. Nguyen, G. Rozza, and A. T. Patera, Calcolo **46**, 157 (2009).

[39] Y. Chen, J. S. Hesthaven, Y. Maday, and J. Rodríguez, SIAM J. Sci. Comput. **32**, 970 (2010).

[40] D. J. Knezevic and J. W. Peterson, Computer Methods in Applied Mechanics and Engineering **200**, 1455 (2011).

[41] A. Quarteroni, G. Rozza, and A. Manzoni, Journal of Mathematics in Industry **1** (2011).

[42] S. E. Field *et al.*, Phys. Rev.Lett. **106**, 221102 (2011), arXiv:1101.3765.

[43] S. Caudill, S. E. Field, C. R. Galley, F. Herrmann, and M. Tiglio, Class. Quant. Grav. **29**, 095016 (2012), arXiv:1109.5642.

[44] S. E. Field, C. R. Galley, and E. Ochsner, Phys. Rev. D **86**, 084046 (2012).

[45] K. Cannon, C. Hanna, and D. Keppel, Phys.Rev. **D84**, 084003 (2011), arXiv:1101.4939.

[46] P. Binev *et al.*, SIAM J. Math. Analysis **43**, 1457 (2011).

[47] R. DeVore, G. Petrova, and P. Wojtaszczyk, Arxiv preprint arXiv:1204.2290 (2012).

[48] D. Keppel, K. Cannon, M. Frei, and C. Hanna, http://www.gravity.phys.uwm.edu/conferences/gwpaw/posters/keppel.pdf.

[49] K. Cannon *et al.*, Phys. Rev. **D82**, 044025 (2010), arXiv:1005.0012.

[50] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes*, 2nd ed. (Cambridge University Press, New York, 1992).

[51] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics* (Springer, Berlin, 2010).

[52] S. Chaturantabut and D. C. Sorensen, SIAM Journal on Scientific Computing **32**, 2737 (2010).

[53] S. Chaturantabut and D. Sorensen, Discrete empirical interpolation for nonlinear model reduction, in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pp. 4316 –4321, 2009.

[54] J. L. Eftang and B. Stamm, International Journal for Numerical Methods in Engineering **90**, 412 (2012).

[55] T. O. Aanonsen, *Empirical Interpolation with Application to Reduced Basis Approximations*, PhD thesis, Norwegian University of Science and Technology, 2009.

[56] S. da Costa Ribeiro, M. Kleinsteuber, A. Möller, and M. Kranz, A compressive sensing scheme of frequency sparse signals for mobile and wearable platforms, in *Proceedings of the 13th international conference on Computer Aided Systems Theory - Volume Part II*, EUROCAST'11, pp. 510–518, Berlin, Heidelberg, 2012, Springer-Verlag.

[57] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, The Journal of Chemical Physics **21**, 1087 (1953).

[58] C. R. Galley, F. Herrmann, J. Silberholz, M. Tiglio, and G. Guerberoff, Class. Quantum Grav. **27**, 245007 (2010), arXiv:1005.5560.