

Chapter 1

Model Reduction for Compact Binary Coalescence

The highly anticipated detection of gravitational waves in 2015 from the coalescence of two stellar mass black holes heralded the beginning of a new era of astronomy. For the foreseeable future, compact binary coalescences (CBCs) will remain the most detectable sources of gravitational waves. The detection of these events requires high fidelity waveforms for use in the matched filtering pipeline. However, calculating these waveforms can take weeks to months of supercomputer resources, so adequately surveying the high dimensional parameter space is a computationally infeasible undertaking.

A gravitational waveform is represented in terms of its two polarisations $h_+(t; \lambda)$ and $h_\times(t; \lambda)$ by $h(t; \lambda) = h_+(t; \lambda) + ih_\times(t; \lambda)$, where $\lambda \in \mathcal{P} \subset \mathbb{R}^8$ is a vector describing the binary source parameters (masses, spin magnitudes, spin directions). We denote the space of gravitational waveforms by X . An inner product on this space is given by the complex scalar product

$$\langle h(\cdot; \lambda_1), h(\cdot; \lambda_2) \rangle = \int_{t_{min}}^{t_{max}} h^*(t; \lambda_1) h(t; \lambda_2) dt \quad (1.1)$$

with an inherited norm given by $\|h(\cdot; \lambda)\|^2 = \langle h(\cdot; \lambda), h(\cdot; \lambda) \rangle$. We may write $h(\lambda)$ to denote the full waveform, i.e. $h : \mathcal{P} \rightarrow X$, or alternatively $h(t; \lambda)$, in which case $h(t; \lambda)$ denotes the value at time t , i.e. $h : \mathbb{R} \times \mathcal{P} \rightarrow \mathbb{C}$.

Question What is the space X ? What space do h_+, h_\times reside in?

The space X may have very high dimension, however the space of gravitational waveforms resides on a parametrically induced manifold $\mathcal{M} = \{h(\lambda) : \lambda \in \mathcal{P}\}$, which may be of comparatively low dimension. Thus, for arbitrary λ we seek to represent $h(\lambda)$ as a linear combination of some basis for $\text{span}\{\mathcal{M}\} \subset X$, that is

$$h(t; \lambda) \approx \sum_{i=1}^N A_i(\lambda) e_i(t) \quad (1.2)$$

for some basis e_i , and as small an N as possible. This raises some immediate questions.

1. How do we optimally choose the the best N -dimensional subspace
2. How do we choose a basis for this subspace
3. How do we best combine this basis to approximate $h(\lambda)$?

To answer the first question, we set the benchmark by defining the Kolmogorov spaces, X_N^{Kol} as

$$X_N^{Kol} = \arg \inf_{X_N \subset X} \left(\sup_{\lambda \in \mathcal{P}} \inf_{h_N \in X_N} \|h(\lambda) - h_N\|_X \right) \quad (1.3)$$

where X_N denotes any N -dimensional subspace of N . The Kolmogorov N -width is then

$$\epsilon_N^{Kol} = \sup_{\lambda \in \mathcal{P}} \inf_{h_N \in X_N^{Kol}} \|h(\lambda) - h_N\|_X. \quad (1.4)$$

Conceptually, the Kolmogorov spaces are those that minimise the error between $h(\lambda)$ for any $\lambda \in \mathcal{P}$ and the “closest” element of that space, and the Kolmogorov N -width is the supremum of that error. This is a very strong definition, however actually finding such a space is impractical.

1.1 Empirical Interpolation Method

We seek to find approximations to elements of \mathcal{M} by developing an operator \mathcal{I} to interpolate the waveforms $h(\cdot; \lambda)$. The method takes as input a tolerance ϵ , and optionally a maximal number of iterations N_{max} . After N iterations, the output is a set of N basis functions $\{e_i\}_{i=1}^N$ and interpolation points $\{T_i\}_{i=1}^N$. Given these, the interpolant $\mathcal{I}_N h(\cdot; \lambda)$ of $h(\lambda)$ can be expressed as

$$\mathcal{I}_N h(t; \lambda) = \sum_{i=1}^N A_i(\lambda) e_i(t). \quad (1.5)$$

The interpolant satisfies the N interpolation constraints

$$\mathcal{I}_N h(T_i; \lambda) = h(T_i; \lambda), \quad (1.6)$$

leading to the following matrix equation

$$\mathbb{V} \mathbf{A}(\lambda) = \mathbf{h}(\lambda), \quad (1.7)$$

where $(\mathbb{V})_{ij} = e_j(T_i)$, $(\mathbf{A}(\lambda))_i = A_i(\lambda)$, and $(\mathbf{h})_i = h(T_i; \lambda)$. The algorithm proceeds as follows.

Algorithm 1.1 The Empirical Interpolation Method (Continuous)

Input: N_{max}, ϵ

- 1: $N = 0, \epsilon' = \epsilon + 1, \mathcal{I}_0 h(t; \lambda) = 0$
- 2: $\lambda_1 = \operatorname{argmax}_{\lambda \in \mathcal{P}} \|h(\cdot; \lambda)\|_{L^\infty(\mathbb{R})}$
- 3: **while** $N < N_{max}$ and $\epsilon' > \epsilon$ **do**
- 4: $N \leftarrow N + 1$
- 5: $r(t) = h(t; \lambda_N) - \mathcal{I}_{N-1} h(t; \lambda_N)$
- 6: $T_N = \operatorname{argmax}_{t \in \mathbb{R}} \|r(t)\|$
- 7: $e_N(t) = r(t)/r(T_N)$
- 8: $[\epsilon', \lambda_{N+1}] = \operatorname{argmax}_{\lambda \in \mathcal{P}} \|h(\cdot; \lambda) - \mathcal{I}_N h(t; \lambda)\|_{L^\infty(\mathbb{R})}$
- 9: **end while**

Output: $\{e_i\}_{i=1}^N, \{T_i\}_{i=1}^N$

This produces a sequence of hierarchical spaces $X_1 \subset X_2 \subset \dots \subset X_N$ where $X_n = \operatorname{span} e_{i=1}^n$.

We start with a training set of parameter values $\Xi_{train} = \{\lambda_i\}_{i=1}^M \subset \mathcal{P}$ and associated waveforms $\{h(\lambda_i)\}$. The EIM method proceeds by identifying a set of parameter values $\{\Lambda_1, \dots, \Lambda_N\}$ and associated waveforms $\{h_1(t), \dots, h_N(t)\}$ where $h_i(t) = h(t; \Lambda_i)$ to form a basis for the greedy subspace X_N .

The spaces constructed are hierarchical, that is, for $n < n'$, $X_n \subset X_{n'}$.

An ideal greedy algorithm would

Explore elementary effects.

Chapter 2

Reduced basis methods for parameterised PDEs

2.1 Forward model

Let $\lambda \in \mathcal{P} \subset \mathbb{R}^N$ be a parameter vector, and $\Omega \in \mathbb{R}^d$ be our function domain. The forward model $\mathcal{M} : \mathcal{P} \times \Omega \rightarrow$

2.2 Statistical Inverse Problems

In a deterministic setting, regularisation and optimisation techniques are used to find a single point estimate of the parameter \mathbf{p} . A statistical formulation returns a probability density function over parameter space describing the relative likelihood of observation-consistent parameters, called the ‘posterior distribution’, $\pi(\mathbf{p}|\mathbf{y}_d)$.

Here we infer properties of the probability distribution of model parameters $\mathbf{p} \in \mathbb{R}^N$ from observations $\mathbf{y}_d \in \mathbb{R}^m$. We adopt a Bayesian approach where a prior probability density $\pi(\mathbf{p})$ of \mathbf{p} is given. We also assume that our model of the observations provides us with a likelihood function $\pi(\mathbf{y}_d | \mathbf{p})$ of the data \mathbf{y}_d . The data \mathbf{y}_d is treated as a random vector \mathbf{y} with probability distribution $\pi(\mathbf{y} | \mathbf{p})$ and we denote the expectation as

$$\mathbf{y}(\mathbf{p}) := E(\mathbf{y} | \mathbf{p}).$$

Bayes' law leads to a formula for the posterior probability density of \mathbf{p} as

$$\pi(\mathbf{p} \mid \mathbf{y}_d) \propto \pi(\mathbf{p}) \pi(\mathbf{y}_d \mid \mathbf{p}). \quad (2.1)$$

This probability density could then be further explored for example, in order to find marginals or moments of the posterior or even the MAP (maximum a posteriori) estimate which is of the form

$$\mathbf{p}_{\text{MAP}} = \arg\max_{\mathbf{p} \in \mathbb{R}^N} \pi(\mathbf{p} \mid \mathbf{y}_d). \quad (2.2)$$

In any case, the exploration of the posterior $\pi(\mathbf{p} \mid \mathbf{y}_d)$ requires its numerical evaluation typically many times. In our case each evaluation of the likelihood (and thus the posterior) requires an expensive numerical simulation.

2.2.1 Reduced basis method

In order to decrease the computational load required we reduce the parameter space from \mathbb{R}^N to \mathbb{R}^{N_r} as any exploration or computation in a lower dimensional space is cheaper. For this we generate a sequence of parameter vectors $\mathbf{p}_1, \mathbf{p}_2, \dots$ in \mathbb{R}^N which are defined using a greedy algorithm

$$\mathbf{p}_{k+1} = \arg\max_{\mathbf{p} \in \mathbb{R}^N} \left(\frac{1}{2} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}(Q_k \mathbf{p})\|^2 + \beta \log \pi(\mathbf{p}) \right), \quad (2.3)$$

where the $Q_k : \mathbb{R}^N \rightarrow \mathbb{R}^N$ are orthogonal projections with $\text{range}(Q_k) = \text{span}\{\mathbf{p}_1, \dots, \mathbf{p}_k\}$ for $k \geq 1$ and $Q_0 := 0$. This choice introduces the new \mathbf{p}_{k+1} to make sure that a large number of observations \mathbf{y}_d can be approximated in the range of Q_{k+1} while exploring parameter vectors which are sufficiently likely according to the prior.

Numerical methods are used to solve PDEs, however for complicated problems the number of parameters required may be quite large, resulting in prohibitive computational cost. The reduced basis method is a means of reducing the number of degrees of freedom in order to reduce costs and make solving the problem feasible. Sometimes we are interested in the field that is the solution to the PDE, perhaps a temperature function. Other times we are interested in a function of this field, perhaps the maximum temperature. This method evaluates the relationship between the input parameters and the output function, hopefully bypassing many of the expensive computations. This is useful in many real-time and multiple query contexts.

Consider a parameterised model $\mathcal{M} : \mathcal{P} \rightarrow \mathcal{U} \rightarrow \mathcal{Y}$ of some phenomenon, where given a parameter value \mathbf{p} in the parameter space $\mathcal{P} \subset \mathbb{R}^N$, the model predicts a state $\mathbf{u}(\mathbf{p})$ in the state space $\mathcal{U} \subset \mathbb{R}^K$ from which we can extract output $\mathbf{y}(\mathbf{p}) \in \mathcal{Y} \subset \mathbb{R}^M$. Suppose in an experimental setting we obtain noisy observations $\mathbf{y}_d \in \mathcal{Y}$. The inverse problem is to estimate \mathbf{p} given \mathbf{y}_d , a problem that is often ill-posed.

In order to decrease the computational load required we reduce the parameter space from \mathbb{R}^N to \mathbb{R}^{N_r} as any exploration or computation in a lower dimensional space is cheaper. For this we generate a sequence of parameter vectors $\mathbf{p}_1, \mathbf{p}_2, \dots$ in \mathbb{R}^N which are defined using a greedy algorithm

$$\mathbf{p}_{k+1} = \arg\max_{\mathbf{p} \in \mathbb{R}^N} \left(\frac{1}{2} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}(Q_k \mathbf{p})\|^2 + \beta \log \pi(\mathbf{p}) \right), \quad (2.4)$$

where the $Q_k : \mathbb{R}^N \rightarrow \mathbb{R}^N$ are orthogonal projections with $\text{range}(Q_k) = \text{span}\{\mathbf{p}_1, \dots, \mathbf{p}_k\}$ for $k \geq 1$ and $Q_0 := 0$. This choice introduces the new \mathbf{p}_{k+1} to make sure that a large number of observations \mathbf{y}_d can be approximated in the range of Q_{k+1} while exploring parameter vectors which are sufficiently likely according to the prior. This method is discussed in detail in [?].

2.3 One dimensional wire

Consider a simple one dimensional case. We have a wire fixed at ends $x = 0$ and $x = 1$, under constant tension $T = 1$, and subject to a transverse load $f(x; p)$, where $p \in \mathcal{P}$ is our parameter. Let $u(x, t; p)$ denote the transverse displacement of the wire at point x and time t , with boundary condition $u(0, t; p) = u(1, t; p) = 0$. We find the initial condition $u_0(x; p) = u(x, 0; p) \in H_0[0, 1]$ by solving the one dimensional Poisson equation

$$-\frac{\partial^2 u_0}{\partial x^2}(x) = f(x; p). \quad (2.5)$$

The system time evolution is governed by the one dimensional wave equation, where the wire is assumed to be released from rest

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{\partial x^2} \quad (2.6)$$

The observations $\mathbf{y}_i \in \mathbb{R}^j$ are a set of time series vectors of the displacement amplitudes at some points $x_i \in [0, 1]$ at j sample times.

2.3.1 Exact Solution

In this example we can explicitly compute the operator \mathcal{M} . The solution to (2.5) is

$$u_0(x; p) = \int_0^1 G(x, \xi) f(\xi; p) d\xi, \quad (2.7)$$

Where $G(x, \xi)$ is Green's function given by

$$G(x, \xi) = \begin{cases} \xi(1-x) & \text{if } 0 \leq \xi \leq x \\ x(1-\xi) & \text{if } x \leq \xi \leq 1. \end{cases} \quad (2.8)$$

The general solution to (2.6) is

$$u(x, t; p) = \sum_{n=1}^{\infty} \beta_n(p) \sin(n\pi x) \cos(cn\pi t). \quad (2.9)$$

We find by substituting $t = 0$ that the β_n are the Fourier sine series coefficients of u_0 :

$$\beta_n(p) = 2 \int_0^1 u_0(x; p) \sin(n\pi x) dx. \quad (2.10)$$

The output \mathbf{y} is found by substituting the observation points \mathbf{x} into (2.7):

$$\mathbf{y}(t) = \sum_{n=1}^{\infty} \beta_n(p) \cos(cn\pi t) \sin(\mathbf{x}n\pi). \quad (2.11)$$

2.3.2 Numerical Solution

We now develop the operator \mathcal{M} for the numerical solution to the system.

Initial condition

Consider a finite differencing scheme. Our domain $[0, 1]$ is divided into n subintervals of length $h = \Delta x = 1/n$ so that the endpoints of the intervals are given by $x_j = j\Delta x = jh$ for $j = 0, 1, \dots, n$. Using a Taylor expansion of u around x we find

$$\begin{aligned} -u_{xx}(x) &= \frac{-u(x+h)+2u(x)-u(x-h)}{h^2} = f(x) \text{ by eq 2.5} \\ \rightarrow \frac{-u_{j+1}+2u_j-u_{j-1}}{h^2} &= f(x_j) \end{aligned} \quad (2.12)$$

where $u_i = u(x_i)$. By setting the endpoints according to our boundary conditions $u_0 = u_n = 0$ and defining

$$\hat{u} = [u_1, \dots, u_{n-1}]^T, \hat{f} = [f(x_1), \dots, f(x_{n-1})], A = \begin{bmatrix} 2 & -1 & 0 & \dots & 0 \\ -1 & 2 & -1 & & \vdots \\ 0 & -1 & 2 & \ddots & 0 \\ \vdots & & & \ddots & \ddots & -1 \\ 0 & \dots & 0 & -1 & 2 \end{bmatrix} \quad (2.13)$$

the initial value problem then becomes finding the solution to the matrix eq

$$A\hat{u} = h^2\hat{f} \quad (2.14)$$

the solution of which is

$$\hat{u}_0(p) = A^{-1}h^2\hat{f}(p) \quad (2.15)$$

Evolution

Defining $\hat{u}(x, t)$ similarly to above, 2.6 becomes

$$\frac{d^2\hat{u}}{dt^2} + c^2 A\hat{u} = 0 \quad (2.16)$$

with initial conditions $\hat{u}(0, p) = \hat{u}_0(p)$ and $\hat{u}'(0, p) = \hat{u}_1(p)$. This admits the solution

$$\hat{u}(t; p) = \cos(cA^{\frac{1}{2}}t)A^{-1}\hat{f}(p) \quad (2.17)$$

2.3.3 Reduced basis construction

To start, let us consider just the evolution equation (2.6). We construct the initial condition as follows

$$u_0 = \sum_{n=1}^N p_n \sin(2\pi nx)/n^2, \quad (2.18)$$

where the parameter $\mathbf{p} = [p_n]$ and (2.17) becomes

$$\hat{u}(t; p) = \cos(cA^{\frac{1}{2}}t) \begin{bmatrix} \sin(2\pi\mathbf{x}) & \dots & \sin(2N\pi\mathbf{x})/N^2 \end{bmatrix} \mathbf{p}. \quad (2.19)$$

The number of gridpoints is fixed at $n = 100$. Let us take the output to be the displacement over times t_1, \dots, t_m just at one gridpoint x_s .

$$\mathbf{y} = \begin{bmatrix} \mathbf{e}_s^T \cos(cA^{\frac{1}{2}}t_1) \sin(2\pi\mathbf{x}) & \cdots & \mathbf{e}_s^T \cos(cA^{\frac{1}{2}}t_1) \sin(2N\pi\mathbf{x})/N^2 \\ \vdots & \ddots & \vdots \\ \mathbf{e}_s^T \cos(cA^{\frac{1}{2}}t_m) \sin(2\pi\mathbf{x}) & \cdots & \mathbf{e}_s^T \cos(cA^{\frac{1}{2}}t_m) \sin(2N\pi\mathbf{x})/N^2 \end{bmatrix} \mathbf{p} \equiv M\mathbf{p} \quad (2.20)$$

where $\mathbf{e}_s = [0, \dots, 1 \dots, 0]$ with the 1 in the s^{th} position.

We investigate how well we are able to approximate the output $\mathbf{y}(\mathbf{p})$ by reduced models $\mathbf{y}(Q_k\mathbf{p})$. We define approximations to the parameter vectors \mathbf{p}_k defined in equation (2.4) by

$$\mathbf{p}_{k+1} = \arg \max_{\mathbf{p} \in \mathcal{M}_k} \frac{1}{2} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}(Q_k\mathbf{p})\|^2. \quad (2.21)$$

where each \mathcal{M}_k contains m independent samples of the uniform distribution over \mathcal{P} and the elements of \mathcal{M}_k and \mathcal{M}_j are independent for $j \neq k$. Note that this allows us to drop the $\beta \log \pi(\mathbf{p})$ term that appears in equation (2.4). We denote the approximations of \mathbf{p}_k by the same symbol \mathbf{p}_k for simplicity. Here $|\mathcal{M}_k| = 100$, and $Q_k = P_k(P_k^T P_k)^{-1} P_k^T$ where

$$P_k = [\mathbf{p}_1 \cdots \mathbf{p}_k].$$

To quantify how well the reduced model approximation $\mathbf{y}(Q_k\mathbf{p})$ can recover the full model $\mathbf{y}(\mathbf{p})$ for arbitrary \mathbf{p} , we use the relative L_2 norm

$$\frac{\sqrt{\int_{\mathcal{P}} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}(Q_k\mathbf{p})\|^2 d\omega}}{\sqrt{\int_{\mathcal{P}} \|\mathbf{y}(\mathbf{p})\|^2 d\omega}},$$

where ω is a probability measure over \mathcal{P} . We approximate this relative root mean squared error by Monte Carlo quadrature as

$$\rho_k = \frac{\sqrt{\frac{1}{m} \sum_{\mathbf{p} \in \mathcal{M}_{k+1}} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}(Q_k\mathbf{p})\|^2}}{\sqrt{\frac{1}{m} \sum_{\mathbf{p} \in \mathcal{M}_{k+1}} \|\mathbf{y}(\mathbf{p})\|^2}}. \quad (2.22)$$

Reduced bases were constructed for full basis dimensions $N = 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100$. Figure ?? shows ρ_k vs k for each N , calculated using (2.22), and figure ?? shows

the number of reduced basis vectors required to achieve $\rho_k < 0.05$. This shows that for more complex problems (a higher N), more reduced basis functions are required to achieve the error threshold, but the relative number of basis functions required decreases. In the case of $N = 100$, a reduced model error of $\rho_k < 0.05$ is achieved with 48 basis vectors, a reduction in the parameter space dimension of more than half.

Inverse Experiment

Next we investigate how well the reduced basis model we have constructed is able to reproduce the initial conditions on the wire. Simulated data $\mathbf{y}_d = \mathbf{y}(\mathbf{p}_0)$ is produced using a sample parameter value \mathbf{p}_0 from the uniform distribution over the parameter space \mathcal{P} . The problem in the full model case is to find the parameter \mathbf{p} that satisfies

$$\mathbf{p} = \arg \min_{\mathbf{p}' \in \mathcal{P}} \|\mathbf{y}_d - \mathbf{y}(\mathbf{p}')\|^2. \quad (2.23)$$

In the reduced model this becomes finding the reduced parameter \mathbf{p}_r that satisfies

$$\mathbf{p}_r = \arg \min_{\mathbf{p}'_r \in \mathcal{P}_r} \|\mathbf{y}_d - \mathbf{y}(\mathbf{p}'_r)\|^2. \quad (2.24)$$

For each of the full basis dimensions $N = 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100$, ten simulated data sets were constructed. Equations (2.23) and (2.24) were solved using Nelder-Mead minimisation with an initial guess of $\mathbf{p}_0 + \Delta\mathbf{p}_0$ for the full case and $\mathbf{p}_0 + Q_k\Delta\mathbf{p}_0$ in the reduced case, where the error $\Delta\mathbf{p}_0$ is randomly sampled from the uniform distribution over \mathcal{P} and in each case normalised such that $\|\Delta\mathbf{p}_0\| = 0.05$ and $\|Q_k\Delta\mathbf{p}_0\| = 0.05$ respectively.

The number of iterations and function calls required to solve (2.23) and (2.24), averaged over the ten simulations, are shown in Figures ?? and ?? respectively. These plots show an exponential increase in iterations and function calls required to solve the optimisation problem in the full basis, and a sub-exponential relationship between problem dimension and the number of iterations and function calls in the reduced basis space. Further work is required to determine the nature of this relationship.

2.4 Conclusions and further work

These results show the potential for improving the computational efficiency in solving inverse problems using reduced basis methods. Further work is required to investigate the sensitivity of this approach to errors in the initial guess in the optimisation algorithm and noise in the data. The greedy method of constructing the reduced basis should be compared to alternative methods such as proper orthogonal decomposition.