Surrogate gravitational waveform models

Scott Field

Department of Physics, University of Maryland

Theoretical astrophysics seminar

Cornell University, November 30, 2013

Previous works

- Field, Galley, Herrmann, Hesthaven, Ochsner, Tiglio (Phys. Rev. Lett. 2011)

- Caudill, Field, Galley, Herrmann, Tiglio (Class. Quant. Grav. 2012)

- Field, Galley, Ochsner (Phys. Rev. D 2012)

- Jason Kaye (Dissertation, Brown University, 2012)

- Antil, Field, Herrmann, Nochetto, Tiglio (Journal Sci. Comp. 2013)

This talk (arXiv:1304.0462, arXiv:1308.3565)

Priscilla Canizares (Cambridge), Jonathan Gair (Cambridge), Chad Galley (Caltech), Jan Hesthaven (Brown), Jason Kaye (Brown), Manuel Tiglio (UMD)

Outline

Outline

Motivation

Modeling of gravitational waves from compact binary coalescences and/or analysis of data represents a high dimensional challenge...

- Large number of intrinsic/extrinsic *parametric* dimensions
- Waveform's *physical* dimension
    - Long durations with large number of cycles

Motivation

Modeling of gravitational waves from compact binary coalescences and/or analysis of data represents a high dimensional challenge...

- Large number of intrinsic/extrinsic *parametric* dimensions
- Waveform's *physical* dimension
  - Long durations with large number of cycles

The high dimensionality of the problem is a bottleneck for most tasks...

- Waveform generation through solving ODEs or PDEs
- Parameter estimation using effective models
- Template based detection algorithms

Even easy problems are hard

Consider the simple TaylorF2 frequency-domain inspiral waveform

$$h(f;\mu) = \mathcal{A}(\mu) f^{-7/6} \mathrm{e}^{\mathrm{i}\Psi(f;\mu)}$$

where μ labels the parameters

Timings

- Evaluation at a single parameter and frequency value takes $\sim 10^{-7}$s
- Typical BNS waveform starting at 40Hz $\sim 5 \times 10^{-3}$s
- LIGO parameter estimation study with a BNS signal $\sim$ days
- ALIGO parameter estimation study at 10Hz $\sim$ weeks

Strategy for parameterized problems

Parameterized problems can be split into two phases

Offline

- Before study/analysis begins – data unknown
- Extra computational and human resources

Train a fast to evaluate *surrogate* GW model

Online

- Data is known
- Evaluate the model at many (data dependent) parameter values
- Surrogate must be accurate and fast to justify offline efforts

Outline

Surrogate/Reduced order models have been employed in other fields such as optimization of airplane design, Radar detection, blood flow

Active area of research. Annual SIAM meeting (2013) featured 22 minisymposia with over 80 talks

First GR workshop on this topic: `www.tapir.caltech.edu/~rom-gr`

Surrogate and reduced order models. What is it? When could it work?

What is a surrogate model? (I)

Surrogate (Merriam-webster)
: one that serves as a substitute

Surrogate (This talk)
: Easy-to-compute model that mimics behavior of the full, underlying model for a fixed range of the parameter/physical variable

What is a surrogate model? (II)

Features

- *NOT* reduced physics, but reduced representations of underlying model
- Surrogate will converge to underlying model as representation is improved
- Only reproduces GW, not other quantities such as objects' motion

Decisions

- Where to sample the underlying model?
- How to tie together these samples?

Examples

- Fits/interpolation
- Machine learning
- Reduced order modeling

What is a reduced order model?

- Seek a representation of the gravitational wave model

$$h_\mu(t) \qquad \text{or} \qquad h_\mu(f)$$

where μ labels the parameterization, such that

$$h_\mu \approx \sum_{i=1}^{m} c_i(\mu) e_i$$

for as small an m as possible

- Leverage representation to accelerate a computation of interest

Whats special about e_i

- Application-specific basis
- Numerical problem's *degrees of freedom* $=$ # of basis
  - Fewer basis $\rightarrow$ faster computations

Reduced basis (RB)-greedy algorithm

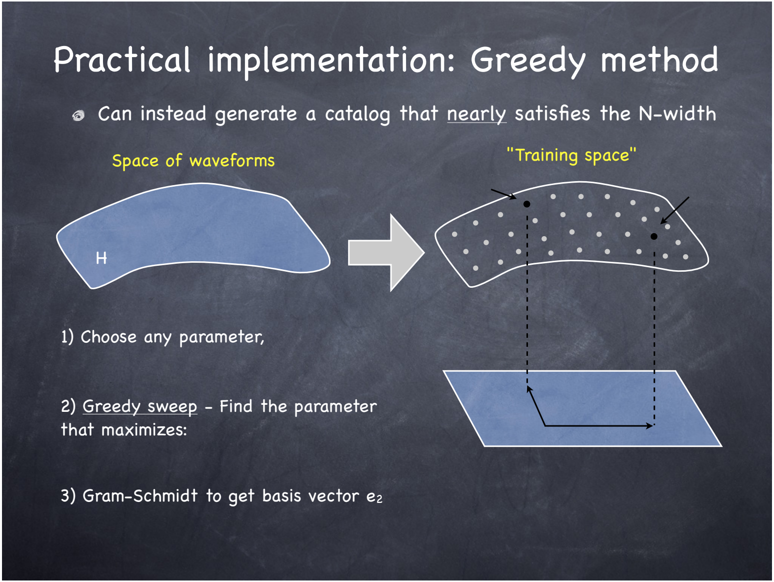
Algorithm to generate the e_i

Definitions

- *Kolmogorov n-width problem*: From all possible basis find the minimum number to achieve accurate approximations $h_\mu \approx \sum_{i=1}^{m} c_i(\mu) e_i$ for all μ
- *RB-Greedy algorithm*: Approximately solve n-width problem
    - Other approaches for basis generation are possible (SVD)

Key features

- Near-optimal basis selection (Binev 2011, DeVore 2012)
- Basis elements are evaluations of the physical model
- Parameter and basis selections carried out simultaneously



Slide courtesy of Chad Galley

Example: Parameterized Heaviside (toy IMR model)

Continuum:

$$H(\mu - x)$$

$$x \in [-1, 1]$$

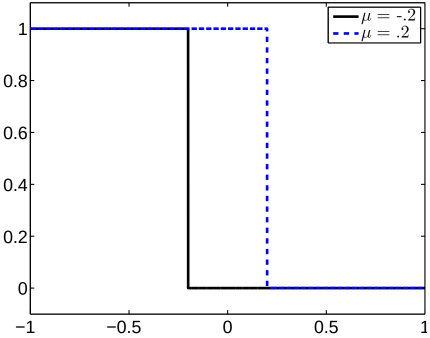
$$\mu \in [-.2, .2]$$

Training set:

$$\{H(\mu_i - x)\}$$

$$\mu_i = -.2 + \frac{.4}{4000} i$$

$$i \in [0, \ldots, 4000]$$

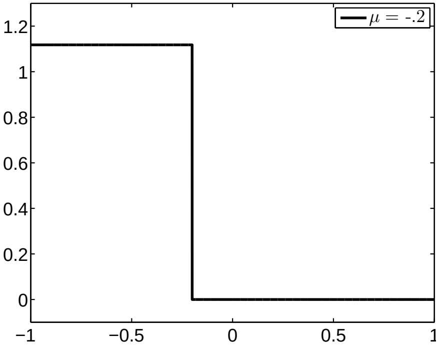


Two representative functions

Example: Parameterized Heaviside (toy IMR model)

1. Select first basis (seed):

$$H(-.2 - x)$$



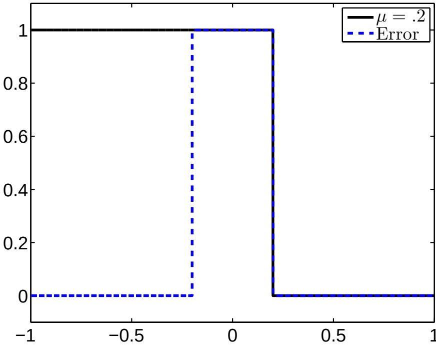
Example: Parameterized Heaviside (toy IMR model)

1. Select first basis (seed):

$$H(-.2 - x)$$

2. Find worst approximation:

$$\mathrm{Err}_i = H(\mu_i - x) - cH(-.2 - x)$$



Example: Parameterized Heaviside (toy IMR model)

1. Select first basis (seed):

$$H(-.2 - x)$$

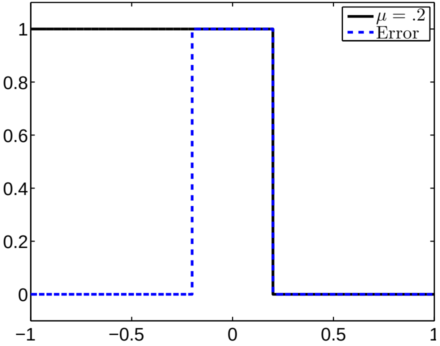
2. Find worst approximation:

$$\mathrm{Err}_i =$$

$$H(\mu_i - x) - c H(-.2 - x)$$

3. Second basis:

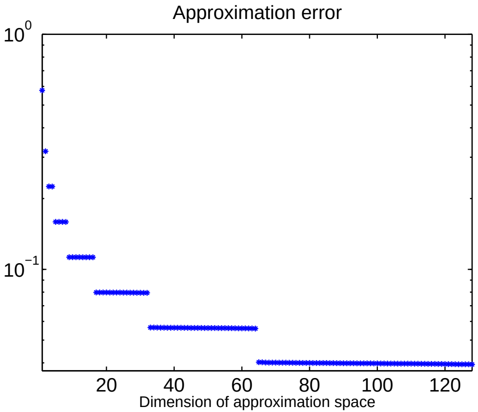
$$\mu = .2 \quad \rightarrow \quad H(.2 - x)$$



Repeat steps 2 & 3 until an approximation threshold is achieved

Greedy output (basis):

$$\mu^{\mathrm{RB}} = \{-0.2, 0.2, 0.0203, -0.0844, 0.1147, \ldots\}$$



- Greedy algorithm "fails". Non-smooth w.r.t. parameter variations.
- If we let $y(\mu) = \mu - x$ then only 1 basis function $H(y)$ needed

Assumption: For reduced/surrogate models to be successful there must be (smooth) structure of the solution with respect to parameter variations...

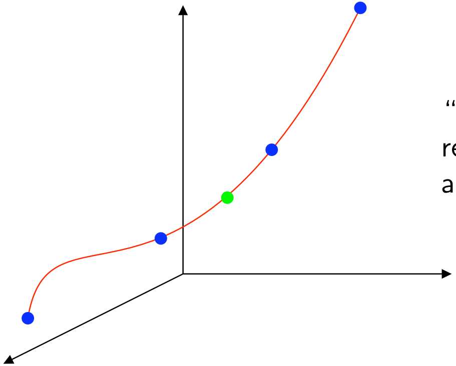


Figure courtesy of Jan Hesthaven

"Well chosen" samples should be representative and lead to accurate approximations

Checking the assumption: EOB results

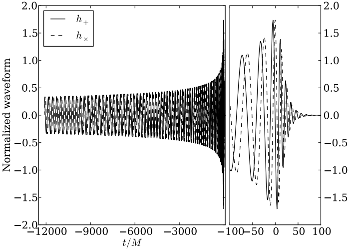
Example with EOBNRv2 waveforms

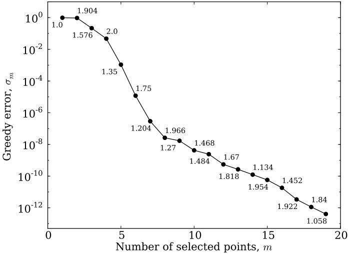
- (2,2) mode for $q \in [1,2]$, duration $\approx 12,000M$ and cycles $\approx$ 65-70
- *Must* align at peak (Heaviside example)
- Fast decay of approximation error

$$\|h_\mu - \sum_{i=1}^{m} c_i(\mu) e_i\|$$

Other evidence

- Observed across models, regimes
- Observed by groups using POD/SVD
  - Ex: Cannon et al (arXiv:1005.0012)

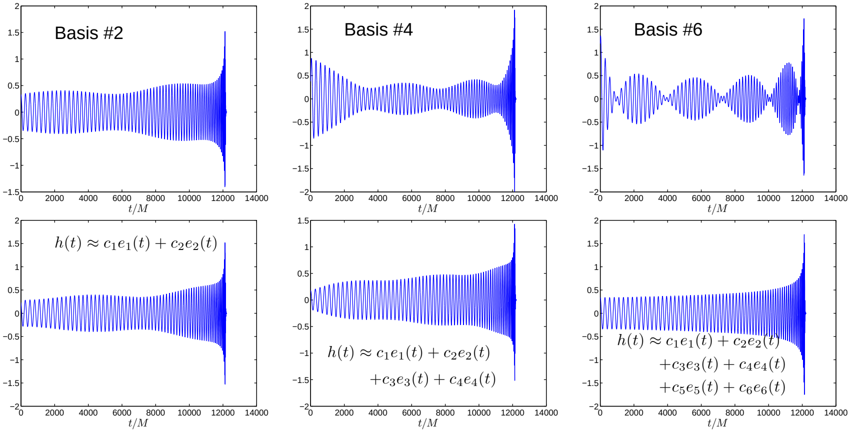




Waveform compression application (ex: $q \sim 1.2040$)

Ortho. Basis

Approx:



(a) 2 term, err ~ 1 (b) 4 term, err $\sim 10^{-1}$ (c) 6 term, err $\sim 10^{-6}$

Now what?

Need a fast way to compute the coefficients $c_i(\mu)$ for *any* parameter μ

$$h_\mu(t) \approx \sum_{i=1}^{m} c_i(\mu) e_i(t)$$

First we look for a convenient expression for $c_i(\mu)$...

Interpolation in time

In principle we can find the approximation

$$h_\mu(t) \approx \mathcal{I}_m[h] = \sum_{i=1}^{m} c_i(\mu) e_i(t)$$

by solving an interpolation problem

$$\sum_{i=1}^{m} c_i(\mu) e_i(T_j) = h_\mu(T_j), \qquad j = 1, \ldots, m$$

- Provided we know m “good” times to sample $h_\mu(t)$
- Naively selected points do not guarantee a solution or accuracy
- For application-specific basis good points are not known a-priori

Empirical interpolation method[1]

- Input: m basis $\{e_i(t)\}_{i=1}^m$
- Output: Nearly optimal selection of m times $\{T_i\}_{i=1}^m$
- These times are adapted to the problem/basis - unlike Chebyshev nodes

Algorithm

- Sequential selection of points: $\{T_1\} \to \{T_1, T_2\} \to \dots$
- Set of points $\{T_j\}_{j=1}^{i-1}$ for interpolation with the first $i-1$ basis
- Extend set $\{T_j\}_{j=1}^{i-1} \to \{T_j\}_{j=1}^{i}$ to minimize the approximation error. Equivalent to selecting

$$T_i = \mathrm{argmax}_t |e_i(t) - \mathcal{I}_{i-1}[e_i](t)|$$

[1] Barrault 2004, Maday 2009, Chaturantabut 2009, Sorensen 2009

Example: Points for polynomial interpolation

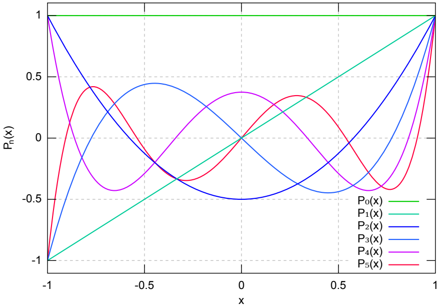
Basis are normalized Legendre polynomials defined on $[-1, 1]$

$$P_0(x) = \frac{1}{\sqrt{2}}$$

$$P_1(x) = \sqrt{\frac{3}{2}}x$$

$$P_2(x) = \sqrt{\frac{5}{8}}\left(3x^2 - 1\right)$$

$$\vdots$$



Q: What are the EIM points?

Example: Points for polynomial interpolation

Basis:

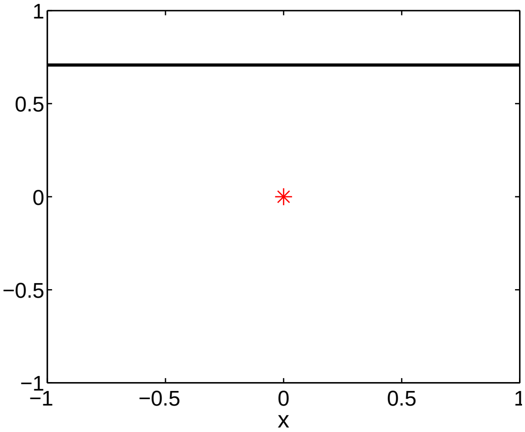
$$P_0(x) = \frac{1}{\sqrt{2}}$$

Residual:

$$P_0(x) - 0 = \frac{1}{\sqrt{2}}$$

Point selection (no preference):

$$x = 0$$



Example: Points for polynomial interpolation

Basis:

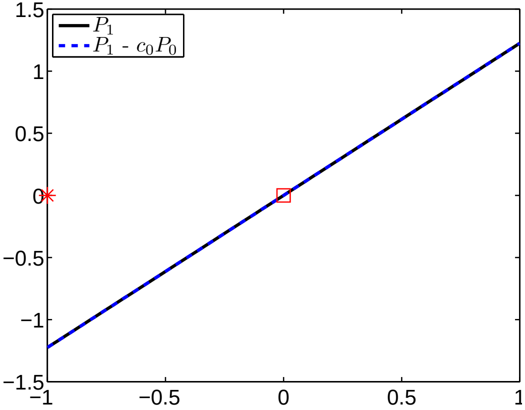
$$P_1(x) = \sqrt{\frac{3}{2}}x$$

Residual:

$$P_1(x) - c_0 P_0 = \sqrt{\frac{3}{2}}x$$

Point selection (either ± 1):

$$x = -1$$



Example: Points for polynomial interpolation

Basis:

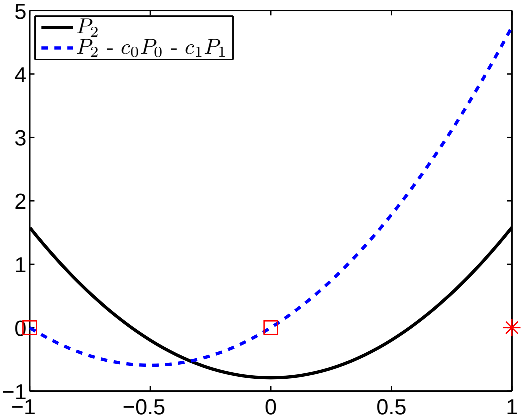
$$P_2(x) = \sqrt{\frac{5}{8}}\left(3x^2 - 1\right)$$

Residual:

$$P_2(x) - (c_0 P_0 + c_1 P_1)$$

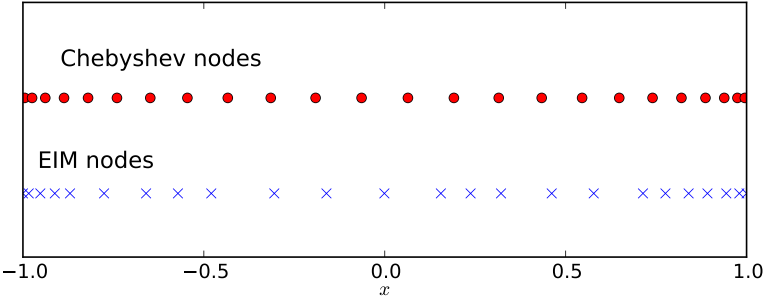
Point selection:

$$x = 1$$



Example: Points for polynomial interpolation

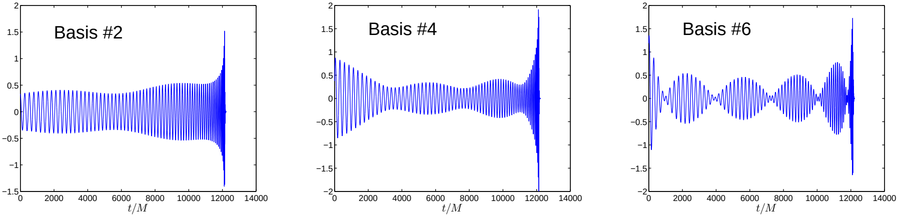
Continue the process until # points = # basis



Distribution and approximation error properties similar to Chebyshev nodes

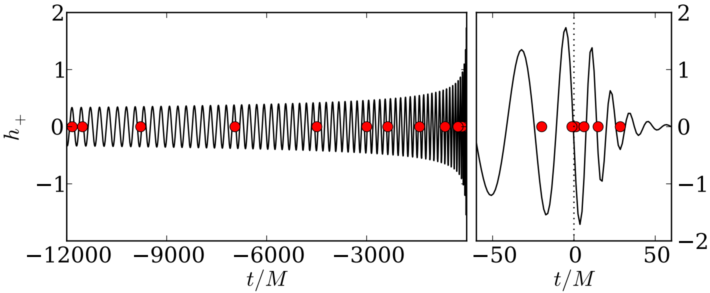
Interpolation points for EOB waveforms

What are the best temporal interpolation points for an EOB-basis?



- We do not know and, for example, Chebyshev nodes won't work.
- Identify these points by empirical interpolation method

Model: non-spinning EOB, $q \in [1, 2]$, 65-70 wave cycles (previous example)



- Any waveform in the above range can be recovered through its evaluation at these 5 (error $\sim 10^{-4}$) to 19 (error $\sim 10^{-12}$) empirical time nodes

$$h_\mu(t) \approx \mathcal{I}[h_\mu](t) = \sum_i^m c_i(\mu) e_i(t) = \sum_i^m B_i(t) h_\mu(T_i)$$

where c_i solved the interpolation problem

Outline

Review

Engines of surrogate and reduced order model generation

- RB-greedy algorithm will select points in parameter space which are most representative of the waveform family and comprises an accurate basis
- EIM algorithm will select most representative time samples which lead to an accurate (temporal) interpolation scheme

GW parameter estimation

- Time series of N data samples $s(t_i) = h_\lambda(t_i) + n(t_i)$, λ true parameter
  - Ex: $32s$ at 4096Hz suggests $N \approx 130,000$ samples
- Bayesian parameter estimation to compute a posterior probability
- Computational cost dominated by likelihood evaluations

$$p\left(s|\mu, H\right) \propto \exp\left(-\chi^2/2\right), \quad \chi^2 = \langle s - h_\mu, s - h_\mu \rangle = \Delta f \sum_{i=1}^{N} \frac{\left|s(f_i) - h_\mu\left(f_i\right)\right|^2}{S_n(f_i)}$$

where $S_n(f)$ is the detector's noise curve

- Each particular μ_0 requires N evaluations of $h_{\mu_0}(t)$
- Overall cost scales with N

Notice

$$\chi^2 = \langle s, s \rangle + \langle h_\mu, h_\mu \rangle - 2\Re\langle s, h_\mu \rangle$$

- $\langle s, s \rangle$ computed once
- $\langle h_\mu, h_\mu \rangle$ often has simple expression (e.g. closed form)

Cost dominated by $\langle s, h_\mu \rangle$, let us focus on this piece...

Plan of attack

- Design a custom quadrature rule tailored to our model $h_\mu(f)$
- Once built the rule is reused *online*, e.g. when new data is available

Reduced order quadrature approximation

Given data $\{s_i\}_{i=1}^N$ and the empirical interpolation representation of $h_\mu(f)$

$$\langle s, h_\mu \rangle = \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) h_\mu(f_i)}{S_n(f_i)} \approx \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) \mathcal{I}_n[h_\mu](f_i)}{S_n(f_i)} = \sum_{i=1}^{m} \omega_i h_\mu(F_i)$$

where the data-specific weights

$$\vec{\omega}^{\,T} = \vec{E}^{\,T} V^{-1} \qquad E_j := \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) e_j(f_i)}{S_n(f_i)}$$

comprise the startup cost. Here V is the interpolation matrix.

Reduced order quadrature approximation

Given data $\{s_i\}_{i=1}^N$ and the empirical interpolation representation of $h_\mu(f)$

$$\langle s, h_\mu \rangle = \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) h_\mu(f_i)}{S_n(f_i)} \approx \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) \mathcal{I}_n[h_\mu](f_i)}{S_n(f_i)} = \sum_{i=1}^{m} \omega_i h_\mu(F_i)$$

where the data-specific weights

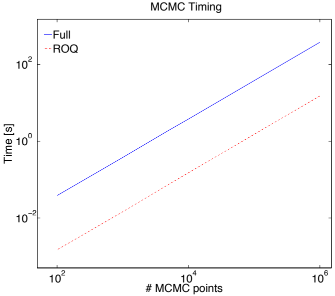
$$\vec{\omega}^{\,T} = \vec{E}^{\,T} V^{-1} \qquad E_j := \Delta f \sum_{i=1}^{N} \frac{s^*(f_i) e_j(f_i)}{S_n(f_i)}$$

comprise the startup cost. Here V is the interpolation matrix.

Properties

- N is a property of the experiment whereas m is a property of the model
- Model's approximation properties are *independent* of data, $m \ll N$

Mock data: 4-dimensional burst signal in Gaussian noise



- About 25 times faster
- Anticipated speedup is (N = data samples)/(m = basis)

- Mean values accurately recovered

SNR	Method	Recovered values f_0	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

Roadmap to a surrogate model

Goal: Fast and accurate evaluation of a parameterized *underlying* GW model

Problem setup

1. Given a model: It is not the job of the surrogate to propose a model
2. Given a range of parameters: Surrogate will only work in this range
3. Given a ODE/PDE solver: Surrogate will not solve equations for you

Attributes

1. *Non-intrusive*: Existing codes are complicated, we don't want to edit them

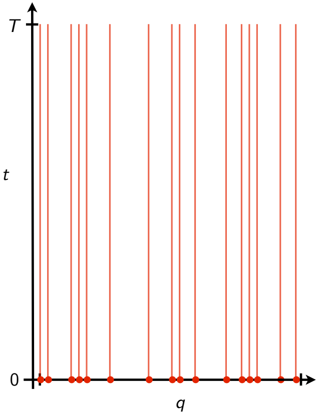
Test Case

Model: Effective One Body (EOB) for inspiral-merger-ringdown of non-spinning binary black holes based on Pan et al. 2011 (arXiv:1106.1021)

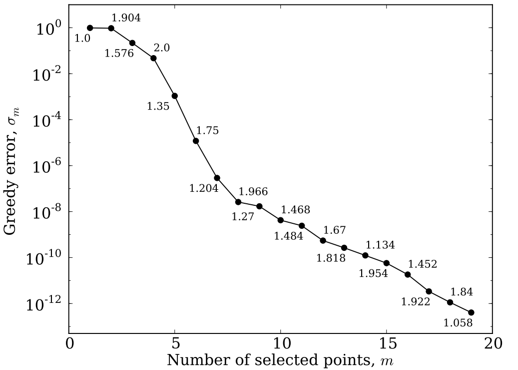
Parameter/physical ranges: Mass ratio in the 1:2 and 9:10 ranges, for 65-80 wave cycles before merger. Results will be shown for 1-2 mass ratio case

Solver: Model implemented in the routine EOBNRv2 as part of the publicly available LIGO Analysis Library (LAL)

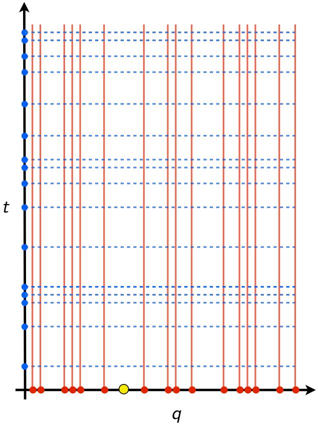
Building our surrogate model in 3 steps



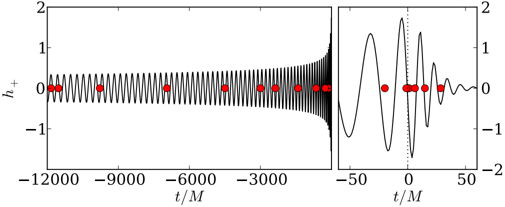
1. Greedy algorithm. Build the basis, the greedy parameter selections are here shown as red points in the horizontal axis. The rb waveforms are solved for at those parameter values. **Offline**



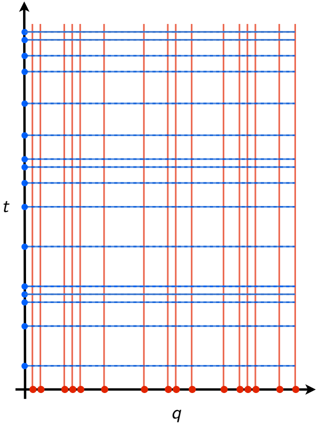
Building our surrogate model in 3 steps



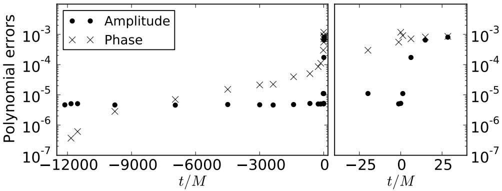
1. Reduced basis waveforms $h_i^{\mathrm{RB}}(t)$, greedy parameters q_i
2. Find a set of empirical time samples (blue points in the vertical axis) for accurate temporal interpolation with rb waveforms. **Offline**



Building our surrogate model in 3 steps

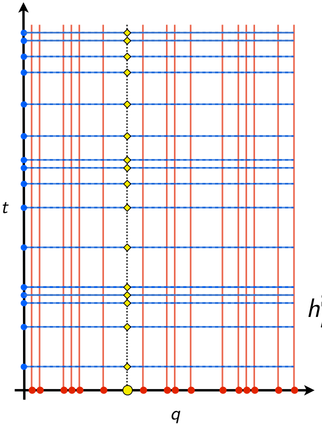


1. Reduced basis waveforms $h_i^{\mathrm{RB}}(t)$, greedy parameters q_i
2. Empirical time samples T_i
3. At each empirical time build a fit for the waveform's parametric dependence *only using waveforms evaluated at* q_i. **Offline**



Fit errors for the amplitude (relative) and phase (absolute) for the mass ratio value for which these errors are **largest**

Building our surrogate model in 3 steps

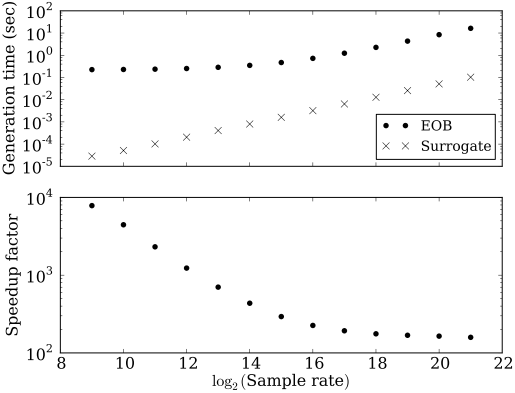


1. Reduced basis waveforms $h_i^{\mathrm{RB}}(t)$, greedy parameters q_i
2. Empirical time samples T_i
3. Fits $h_\mu^{\mathrm{FIT}}(T_i)$ at each empirical time T_i
4. **Online:** Evaluate the surrogate at *any* parameter value (yellow points) by i) evaluating the fits at each empirical time which ii) specifies the full waveform via the empirical interpolant.

$$h_\mu^{\mathrm{S}}(t) \equiv \sum_{i=1}^{m} B_i(t) h_\mu^{\mathrm{FIT}}(T_i) \, , \quad B_j(t) \equiv \sum_{i=1}^{m} h_i^{\mathrm{RB}}(t) \left(V^{-1}\right)_{ij}$$

where V is the interpolation matrix and $\{B_i\}$ are precomputed offline

Evaluation speedup

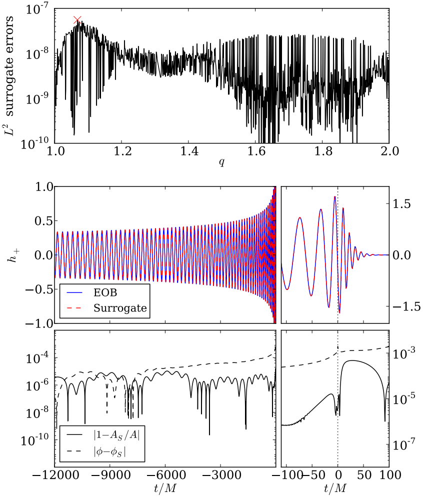


The figure of interest is a sampling rates around $2^{11} \approx 2048$Hz

Speedup of more than three orders of magnitude compared to solving the original EOB equations

Surrogate's evaluation cost is independent of computational costs of the ODE/PDE solver

Putting the pieces together. Full surrogate model



Top: overlap error of the surrogate model compared to solving the original EOB equations

Center: waveform for the mass ratio for which the overlap error is the largest

Bottom: amplitude and phase errors for the same waveform as center plot.

These errors are smaller than the errors of the model itself and those of numerical relativity simulations. Equivalent to underlying model.

What if we cannot build a dense training set?

Cost of full Einstein solver prohibits a proper training set....

Algorithm wish list

1. Make the most of limited number of physical model solves
2. Surrogate model should converge to physical model as solves $\to \infty$
3. Guidance for where to solve physical model

What if we cannot build a dense training set?

Cost of full Einstein solver prohibits a proper training set....

Algorithm wish list

1. Make the most of limited number of physical model solves
2. Surrogate model should converge to physical model as solves $\rightarrow \infty$
3. Guidance for where to solve physical model

Status

- Run solver at greedy points selected for EOB model
- Distribution of greedy points likely more important than actual values
- Jonathan Blackman and Bela Szilagyi are performing runs – we should know soon

Remarks

- Surrogate and reduced order modeling offers an exciting new approach to overcome a variety of computationally challenging problems of GW physics
- Proposed specific surrogate based on three offline steps. Online evaluation fast and accurate
- Reduced order quadratures directly offset cost of overlap inner products

Future outlook

- Multi-mode surrogates for larger mass ratio, more cycles and with spin
- Potentially applicable to full numerical relativity waveforms
- PE accelerated by surrogates and/or reduced order quadratures
- Making these tools publicly available
- More broadly, continue to adapt tools from engineering and applied math communities for GWs