Fast waveform extraction from gravitational perturbations

Alex Benedict[1] Scott Field[2] Stephen Lau[1]

[1]University of New Mexico, Mathematics and Statistics
[2]University of Maryland, Department of Physics

June 16, 2012

15th Capra Meeting

Outline

Motivation

- Recover the asymptotic signal reaching future null infinity using only knowledge of a signal recorded at an arbitrarily close location
- Theoretical and practical interest – gravitational waveform modeling

Approach

- Will show convolution with extraction kernels yields asymptotic signal
- Closely related to exact radiation boundary kernels
  - On a spatially finite computational domain we require boundary conditions
  - Will show extraction kernels are given as an integral over boundary kernels

Old area of study, some especially relevant techniques...

- Geometric approach using hyperboloidal-layers (Zenginoglu, Diener)
- Gravitational multipoles for general relativity linearized about flat spacetime (Abrahams and Evans)

Outline

3+1 wave equation

We wish to solve...

$$(-\partial_t^2 + \partial_x^2 + \partial_y^2 + \partial_z^2)\psi = 0$$

Problem posed on spatially unbounded domain and with compactly supported initial data.

3+1 wave equation

We wish to solve...

$$(-\partial_t^2 + \partial_x^2 + \partial_y^2 + \partial_z^2)\psi = 0$$

Problem posed on spatially unbounded domain and with compactly supported initial data.

We actually solve...

- For computational reasons the problem is solved on a spatially finite domain
- Outer *computational* boundary is a sphere located at $r = r_b$

GOAL: mimic open space problem by i) supplying correct non-reflecting boundary conditions and ii) recovering solution which escapes to infinity.

4 step roadmap

Working in the Laplace frequency domain...

1. What are the outgoing solutions?
2. What are the outgoing boundary conditions?
3. What is asymptotic solution? (preview: related to boundary conditions)
4. Finally, inverse Laplace transform to get time-domain information.

Flatspace and RWZ wave equations follow similar approach. However, we may carry out 4 steps analytically for flatspace, while relying more heavily on numerical results for the RWZ equations.

Laplace transformed wave equation

- Flatspace wave equation for spherical harmonic modes:

$$\psi = \sum_{\ell m} \frac{1}{r} \Psi_{\ell m}(t,r) Y_{\ell m}(\theta,\phi) \rightarrow \qquad \left[\frac{\partial^2}{\partial t^2} - \frac{\partial^2}{\partial r^2} + \frac{\ell(\ell+1)}{r^2} \right] \Psi_{\ell m} = 0$$

Laplace transformed wave equation

- Flatspace wave equation for spherical harmonic modes:

$$\psi = \sum_{\ell m} \frac{1}{r} \Psi_{\ell m}(t,r) Y_{\ell m}(\theta,\phi) \to \qquad \left[\frac{\partial^2}{\partial t^2} - \frac{\partial^2}{\partial r^2} + \frac{\ell(\ell+1)}{r^2} \right] \Psi_{\ell m} = 0$$

- Laplace transformed solution $\hat{\Psi}_{\ell m}(s,r) = \int_0^\infty \Psi_{\ell m}(t,r) \mathrm{e}^{-st} dt$ solves

$$\left[s^2 - \frac{\partial^2}{\partial r^2} + \frac{\ell(\ell+1)}{r^2} \right] \hat{\Psi}_{\ell m} = \frac{\partial \Psi_{\ell m}}{\partial t}(0,r) + s\Psi_{\ell m}(0,r)$$

This equation serves as starting point for our analysis

Outgoing solutions

$$\left[s^2 - \frac{\partial^2}{\partial r^2} + \frac{\ell(\ell+1)}{r^2}\right]\hat{\Psi}_\ell = 0$$

- Ordinary differential equation
- A modified Bessel equation – solutions well studies

- General outgoing solution: $\boxed{\widehat{\Psi}_\ell(s,r) = a(s)s^\ell e^{-sr} W_\ell(sr)}$
  Where $a(s)$ some known function encoding the initial data
- Key point: Kernels are built from W_ℓ and its derivative
- $W_\ell(sr) = (sr)^{-\ell}\sum_{k=0}^{\ell} c_{\ell k}(sr)^k$
- Coefficients $c_{\ell k}$ known (e.g. Jackson)
  - Example $W_2(sr) = (sr)^{-2}\left[3 + 3sr + (sr)^2\right]$

Frequency domain boundary conditions for outgoing modes

- We supply 1 piece of information: $(\partial_t + \partial_r)\,\Psi_\ell = ???$
- Compute $s + \partial_r$ for an outgoing solution $\widehat{\Psi}_\ell(s,r) = a(s)s^\ell e^{-sr} W_\ell(sr)$

$$s\widehat{\Psi}_\ell(s,r) + \partial_r \widehat{\Psi}_\ell(s,r) = \frac{1}{r}\left[sr\frac{W_\ell'(sr)}{W_\ell(sr)}\right]\widehat{\Psi}_\ell(s,r)$$

$$= \frac{1}{r}\left[\sum_{k=1}^{\ell} \frac{b_{\ell,k}/r}{s - b_{\ell,k}/r}\right]\widehat{\Psi}_\ell(s,r) \equiv \frac{1}{r}\widehat{\Omega}_\ell(s,r)\widehat{\Psi}_\ell(s,r)$$

- $b_{\ell,k}$ are zeros of $W_\ell(b_{\ell,k}) = 0$
- $\widehat{\Omega}_\ell(s,r)$ is the boundary kernel – evidently a sum-of-poles

Time domain boundary conditions for outgoing modes

Using well known properties of inverse Laplace transforms...

$$\boxed{\partial_t \Psi_\ell + \partial_r \Psi_\ell = \frac{1}{r} \int_0^t \Omega_\ell(t - t', r) \Psi_\ell(t', r) dt'}$$

where $\Omega_\ell(t, r) = \sum_{k=1}^{\ell} \frac{b_{\ell,k}}{r} \exp\left(\frac{b_{\ell,k} t}{r}\right)$.

Observations

- Exact outgoing boundary condition in time domain at any r_b
- Numerical solution computed with boundary at r_b and ∞ are *identical*

Flatspace teleportation/extraction for outgoing modes

- Using knowledge of the outgoing solution $\widehat{\Psi}_\ell(s,r) = a(s)s^\ell e^{-sr} W_\ell(sr)$[1]

$$\widehat{\Psi}_\ell(s,r_2) = \mathrm{e}^{s(r_1-r_2)}\left[\frac{W_\ell(sr_2)}{W_\ell(sr_1)}\right]\widehat{\Psi}_\ell(s,r_1) \equiv \mathrm{e}^{s(r_1-r_2)}\widehat{\Phi}_\ell(s,r_1,r_2)\widehat{\Psi}_\ell(s,r_1)$$

- $\widehat{\Phi}_\ell(s,r_1,r_2)$ is the teleportation kernel
- When $r_2=\infty$, $\widehat{\Phi}_\ell(s,r_1,\infty)$ is the extraction kernel

[1] Disclaimer: must define $\widehat{\Phi}_\ell(s,r_1,r_2) = W_\ell(sr_2)/W_\ell(sr_1) - 1$ so that $\widehat{\Phi}_\ell \to 0$ along path of inverse Laplace transform. This amounts to offsetting by $\widehat{\Psi}_\ell(s,r_1)$

Connection with boundary kernels

Straightforward to show

$$\widehat{\Phi}_\ell(s, r_1, r_2) = \frac{W_\ell(sr_2)}{W_\ell(sr_1)} = \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_\ell(s,\eta)}{\eta} d\eta\right]$$

Teleportation kernel is an integral over boundary kernels

- $\widehat{\Phi}_\ell(s, r_1, r_2)$ is numerically generated and **NOT** a sum of poles. How should we invert to the time domain?

Need for rational approximation

Propose that we can write

$$\widehat{\Phi}_\ell(s, r_1, r_2) \approx \frac{\text{degree } d-1 \text{ polynomial}}{\text{degree } d \text{ polynomial}} = \sum_{i=1}^{d} \frac{\gamma_i}{s-\beta_i}$$

- Clearly rational approximation won't be accurate for all $s \in \mathbb{C}$
  - In fact infinitely bad
- If rational approximation is accurate for only $s \in \mathrm{i}\mathbb{R}$ we can *analytically* perform the inversion!
- Carried out via a non-linear least squares fitting procedure. Details of this "black box" unimportant: Approximation known and accurate
  - Typical pointwise relative error of 10^{-12} achieved

Schematic for practical extraction kernel generation

Used for for flatspace wave and RWZ equations

1. Working in the frequency domain, identify the boundary kernel $\widehat{\Omega}_\ell(s, r)$

Schematic for practical extraction kernel generation

Used for for flatspace wave and RWZ equations

1. Working in the frequency domain, identify the boundary kernel $\widehat{\Omega}_\ell(s, r)$
2. Numerically integrate to generate the relevant teleportation/extraction kernel $\widehat{\Phi}_\ell(s, r_1, r_2) = \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_\ell(s,\eta)}{\eta} d\eta\right]$

Schematic for practical extraction kernel generation

Used for for flatspace wave and RWZ equations

1. Working in the frequency domain, identify the boundary kernel $\widehat{\Omega}_\ell(s, r)$
2. Numerically integrate to generate the relevant teleportation/extraction kernel $\widehat{\Phi}_\ell(s, r_1, r_2) = \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_\ell(s,\eta)}{\eta} d\eta\right]$
3. Evaluate this objects along inversion path $s \in \mathrm{i}\mathbb{R}$

Schematic for practical extraction kernel generation

Used for for flatspace wave and RWZ equations

1. Working in the frequency domain, identify the boundary kernel $\widehat{\Omega}_\ell(s, r)$
2. Numerically integrate to generate the relevant teleportation/extraction kernel $\widehat{\Phi}_\ell(s, r_1, r_2) = \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_\ell(s,\eta)}{\eta} d\eta\right]$
3. Evaluate this objects along inversion path $s \in \mathrm{i}\mathbb{R}$
4. Rational approximation to good agreement on $s \in \mathrm{i}\mathbb{R}$

Schematic for practical extraction kernel generation

Used for for flatspace wave and RWZ equations

1. Working in the frequency domain, identify the boundary kernel $\widehat{\Omega}_\ell(s,r)$
2. Numerically integrate to generate the relevant teleportation/extraction kernel $\widehat{\Phi}_\ell(s,r_1,r_2) = \exp\left[\int_{r_1}^{r_2} \frac{\widehat{\Omega}_\ell(s,\eta)}{\eta} d\eta\right]$
3. Evaluate this objects along inversion path $s \in \mathrm{i}\mathbb{R}$
4. Rational approximation to good agreement on $s \in \mathrm{i}\mathbb{R}$
5. Analytically perform an inverse Laplace transform using the rationally approximated kernel

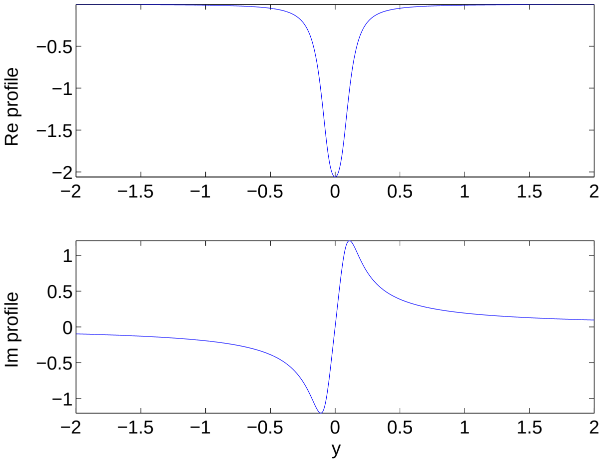
Regge–Wheeler equation (Zerilli follows similar approach)

$$-\partial_t^2\Psi_{\ell m} + \partial_x^2\Psi_{\ell m} - \frac{f(r)}{r^2}\left[\ell(\ell+1) - \frac{6M}{r}\right]\Psi_{\ell m} = 0$$

$x = r + 2M\log(\frac{1}{2}r/M - 1)$ is tortoise coordinate and $f(r) = 1 - 2M/r$

- One can find (and numerically integrate) an ODE in the Laplace frequency domain for the RWZ boundary kernel (Lau 2004)

$\ell = 2$, $r_b = 30M$ boundary kernel evaluated along $s = iy$



Profiles shown on the left

$\Downarrow$

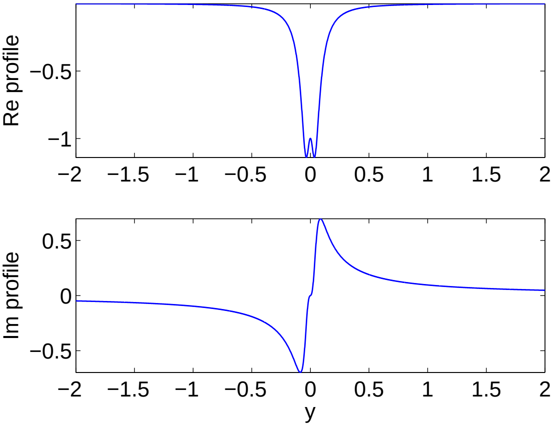
Rational approximation

$\Downarrow$

Poles and strengths

$\ell = 2,\ r_1 = 30M,\ r_2 = \infty$ extraction kernel along $s = iy$

Numerically compute $\widehat{\Phi}_2(s) = \exp\left[\int_{30M}^{\infty} \frac{\widehat{\Omega}_2(s,\eta)}{\eta} d\eta\right]$



Profiles shown on the left

$\Downarrow$

Rational approximation

$\Downarrow$

Poles and strengths

Pole #	Gamma strengths	Beta locations
1	-1.7576263057e-08 + 0i	-5.4146529341e-01 + 0i
2	-6.4180514293e-08 + 0i	-4.1310954989e-01 + 0i
3	-6.2732971050e-06 + 0i	-3.1911338482e-01 + 0i
4	-6.9363117988e-05 + 0i	-2.4711219871e-01 + 0i
5	-5.7180637750e-04 + 0i	-1.9108163722e-01 + 0i
6	-2.7884247577e-03 + 0i	-1.4749601558e-01 + 0i
7	-5.8836792033e-03 + 0i	-1.1366299945e-01 + 0i
8	-3.6549136132e-03 + 0i	-8.6476935381e-02 + 0i
9	-1.0498746767e-03 + 0i	-6.4512065175e-02 + 0i
10	-2.4204781878e-04 + 0i	-4.7332374442e-02 + 0i
11	-5.5724464176e-05 + 0i	-3.4115775484e-02 + 0i
12	-1.2157296793e-05 + 0i	-2.4048935704e-02 + 0i
13	-2.6651813247e-06 + 0i	-1.6468632919e-02 + 0i
14	-4.8661708981e-07 + 0i	-1.0845690423e-02 + 0i
15	-8.6183677612e-08 + 0i	-6.7552918597e-03 + 0i
16	-9.3735071189e-09 + 0i	-3.8525630196e-03 + 0i
17	-8.7881787023e-10 + 0i	-1.8481215040e-03 + 0i
18	-9.1164536027e-02 -5.3953709155e-02i	-9.4779490815e-02 +5.9927979877e-02i
19	-9.1164536027e-02 +5.3953709155e-02i	9.4779490815e-02 -5.9927979877e-02i

For $s \in \mathrm{i}\mathbb{R}$, $\widehat{\Phi}_2(s) \approx \sum_{i=1}^{19} \frac{\gamma_i}{s-\beta_i} \to \Phi_2(t) \approx \sum_{i=1}^{19} \gamma_i \exp\left(\beta_i t\right)$

Key features of extraction technique

- With a time-series at *ANY* radial location can *EXACTLY* extract signal to any other radial value
  - Exact for RWZ equation
- Extraction as a post-processing step on existing data
- Non-intrusive to existing code

Outline

Kernel tables

For someone wishing to use boundary and extraction kernels the previous part of the talk is largely irrelevant

- Kernels computed using MPI and quad precision
- Once a table has been generated very easy to use
- All kernels are (or will be) available online

www.dam.brown.edu/people/sfield/KernelsRWZ
www.math.unm.edu/~lau/KernelsRWZ

Implementation: extraction of RW masterfunction

- Suppose we have evolved the RW equation, recording a (discrete) time-series $\Psi^n = \Psi(t_n, x_b)$ at the outer computational boundary
- Discrete times from the numerical scheme are $t^n = 0 + n\Delta t$
- From $\Psi(t_n, x_b)$ we want to compute $\Psi(t_n + \infty, x_b + \infty)$

$$\Psi(t+\infty, b+\infty) \simeq \sum_{q=1}^{d} \gamma_q \int_0^t e^{\beta_q (t-t')} \Psi(t', b) dt' + \Psi(t, b)$$

$$\Psi(t+\infty, b+\infty) \simeq \sum_{q=1}^{d} \gamma_q \int_0^t e^{\beta_q(t-t')}\Psi(t', b)dt' + \Psi(t, b)$$

1) Download and import a table

Pole #	Alpha strengths	Beta locations
1	-6.2237645749568241E-008	-2.2294005169277857E-001
2	-3.9539987058586121E-006	-1.5508374693643587E-001
	18 More Entires Here	

$$\Psi(t+\infty, b+\infty) \simeq \sum_{q=1}^{d} \gamma_q \int_0^t e^{\beta_q (t-t')} \Psi(t', b) dt' + \Psi(t, b)$$

1) Download and import a table

```
Pole #      Alpha strengths                                   Beta locations
   1     -6.2237645749568241E-008                       -2.2294005169277857E-001
   2     -3.9539987058586121E-006                       -1.5508374693643587E-001
                             18 More Entires Here
```

2) Read these values into the code

$(\alpha_1, \beta_1) = \left(-6.2237645749568241 \times 10^{-8}, -2.2294005169277857 \times 10^{-1}\right)/(2M)$

$(\alpha_2, \beta_2) = \dots$

$$\Psi(t+\infty, b+\infty) \simeq \sum_{q=1}^{d} \gamma_q \int_0^t e^{\beta_q(t-t')}\Psi(t', b)dt' + \Psi(t, b)$$

1) Download and import a table

```
Pole #     Alpha strengths                                Beta locations
   1     -6.2237645749568241E-008                    -2.2294005169277857E-001
   2     -3.9539987058586121E-006                    -1.5508374693643587E-001
                              18 More Entires Here
```

2) Read these values into the code

$(\alpha_1, \beta_1) = \left(-6.2237645749568241 \times 10^{-8}, -2.2294005169277857 \times 10^{-1}\right)/(2M)$

$(\alpha_2, \beta_2) = \dots$

3) Integrate a few ODEs (Forward Euler implementation)

For $q = 1$ **to** 20

$(\Xi_q * \Psi)^{n+1} = (\Xi_q * \Psi)^n + \Delta t\left[\beta_q(\Xi_q * \Psi)^n + \Psi^n\right]$

EndFor

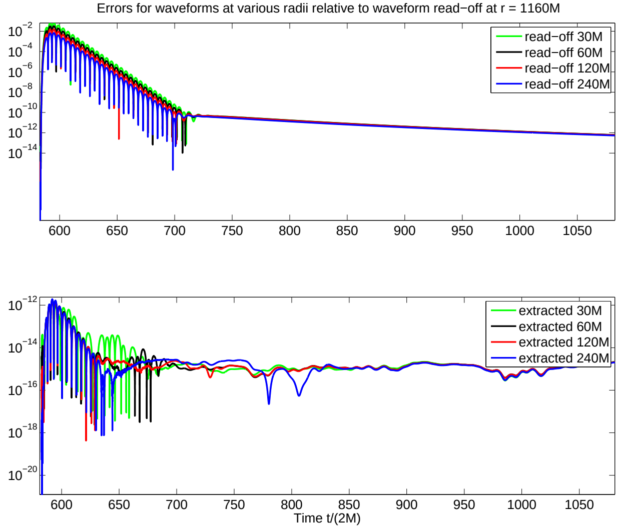
$\Psi_\infty^{n+1} = \sum_{q=1}^{20} \gamma_q(\Xi_q * \Psi)^{n+1} + \Psi^{n+1}$

Teleportation: finite radius extraction

Experiment setup

- Finite boundary to finite boundary location (RW potential)
- "bump" function with support $-10M < r_* < 3M$
- Record $\Psi(t, r_2)$ as a time-series at some location r_2
- Record $\Psi(t, r_1)$ as a time-series at some location $r_1 < r_2$
- Find $\Psi(t, r_2)$ by convolving $\Psi(t, r_1)$ with a teleportation kernel

Extraction $r_1 \to r_2 = 1160M$



Test particle orbiting a Schwarzschild blackhole (Setup)

Orbital parameters	
$e = 0.764124$	$p = 8.75455$

- Perturbations $h_{\mu\nu}$ determined by *master functions*: $\Psi^{\mathrm{ZM}}_{\ell m}$ and $\Psi^{\mathrm{CPM}}_{\ell m}$.
- Each obeys a forced wave equation:

$$-\partial_t^2\Psi_{\ell m} + \partial_x^2\Psi_{\ell m} - V_\ell(r)\Psi_{\ell m}$$
$$= f(r)\big[G_{\ell m}(t,r)\delta(r - r_p(t)) + F_{\ell m}(t,r)\delta'(r - r_p(t))\big].$$

- $V_\ell(r)$ either the Regge–Wheeler or Zerilli potential for which we construct boundary and extraction kernels
- Spectrally accurate time-domain discontinuous Galerkin code

Luminousity results from measurements at $r_b = 60M$

Compare with accurate frequency domain results

S. Hopper and C. R. Evans, Phys. Rev. D82, 084010 (2010).

$$\dot{E}^{\infty}_{\ell m} = \frac{1}{64\pi}\frac{(\ell+2)!}{(\ell-2)!}\left\langle |\dot{\Psi}^{\mathrm{Z}}_{\ell m}|^2 + |\dot{\Psi}^{\mathrm{RW}}_{\ell m}|^2 \right\rangle$$

$$\dot{L}^{\infty}_{\ell m} = \frac{im}{64\pi}\frac{(\ell+2)!}{(\ell-2)!}\left\langle \dot{\Psi}^{\mathrm{Z}}_{\ell m}\bar{\Psi}^{\mathrm{Z}}_{\ell m} + \dot{\Psi}^{\mathrm{RW}}_{\ell m}\bar{\Psi}^{\mathrm{RW}}_{\ell m} \right\rangle$$

m	Alg.	$\dot{E}^{\infty}_{2m}$	$\dot{L}^{\infty}_{2m}$
0	FR	$1.27486196317 \times 10^{-8}$	0
	WE	$1.27486196187 \times 10^{-8}$	0
1	FR	$1.15338054092 \times 10^{-6}$	$1.44066000650 \times 10^{-5}$
	WE	$1.15338054091 \times 10^{-6}$	$1.44066000619 \times 10^{-5}$
2	FR	$1.55967717209 \times 10^{-4}$	$2.07778922470 \times 10^{-3}$
	WE	$1.55967717211 \times 10^{-4}$	$2.07778922439 \times 10^{-3}$

Outline

Introduction

Theory: boundary and extraction kernels

Practice: implementation with tables and results

Final remarks

Final remarks

- Theoretical development for flatspce case
- RWZ case treated similarly but relies more heavily on numerics
- Rational approximation provides for an accurate sum-of-poles representation
- While computing a kernel table is hard, implementing it takes a few minutes
- Extraction is an easy post-processing step (existing data, no changes to code)
- Extraction and boundary kernels are (will) be made available for variety of boundary locations and up to high ℓ

QUESTIONS?