UCD RELATIVITY GROUP

UCD DUBLIN

IRISH RESEARCH COUNCIL
An Chomhairle um Thaighde in Éirinn
www.research.ie

Time Domain Method for the Green Function in Schwarzschild Spacetime

Conor O'Toole
Adrian Ottewill, Barry Wardell

Green Functions in Black Hole Spacetimes

- The 1st Order self-force can be decomposed as

$$F^{\alpha}_{\text{self}}(z) = F^{\alpha}_{\text{inst}} + F^{\alpha}_{\text{hist}}$$

- The historical part relates to the effects from the part of the worldline *inside* the past light cone

$$h \sim (\text{local terms}) + \lim_{\epsilon \to 0} \int_{-\infty}^{\tau^- - \epsilon} G_{\alpha\beta\alpha'\beta'}(z(\tau), z(\tau'))u^{\alpha'} u^{\beta'} d\tau'$$

- G is the retarded Green Function

Mino, Sasaki,Tanaka*, 1997, Phys. Rev. D* **55**
Quinn, Wald*, 1997, Phys. Rev. D* **56**
Detweiler, Whiting, *2003, Phys. Rev. D* **67**

Green Functions in Black Hole Spacetimes

- For now, we restrict ourselves to a *scalar* self force, in Schwarzschild

$$(\Box_{(4)} - \zeta R)\Phi(x) = -4\pi\rho(x)$$

- This is just a wave equation in curved spacetime, so

$$\Phi(x) = \int G(x,x')\rho(x')\sqrt{-g}\; d^4x'$$

$$\Rightarrow F^{\alpha}_{\text{hist}}(z) = q^2 g^{\alpha\beta} \lim_{\epsilon\to 0^+} \int_{-\infty}^{\tau-\epsilon} \nabla_\beta G(z(\tau), z(\tau'))d\tau'$$

- So we try to solve for a Green function, which will satisfy:

$$\left(\frac{1}{\sqrt{-g}}\partial_\mu(\sqrt{-g}g^{\mu\nu}\partial_\nu) - \zeta R\right)G(x,x') = -4\pi\frac{\delta_4(x-x')}{\sqrt{-g}}$$

Time Domain Numerical Method

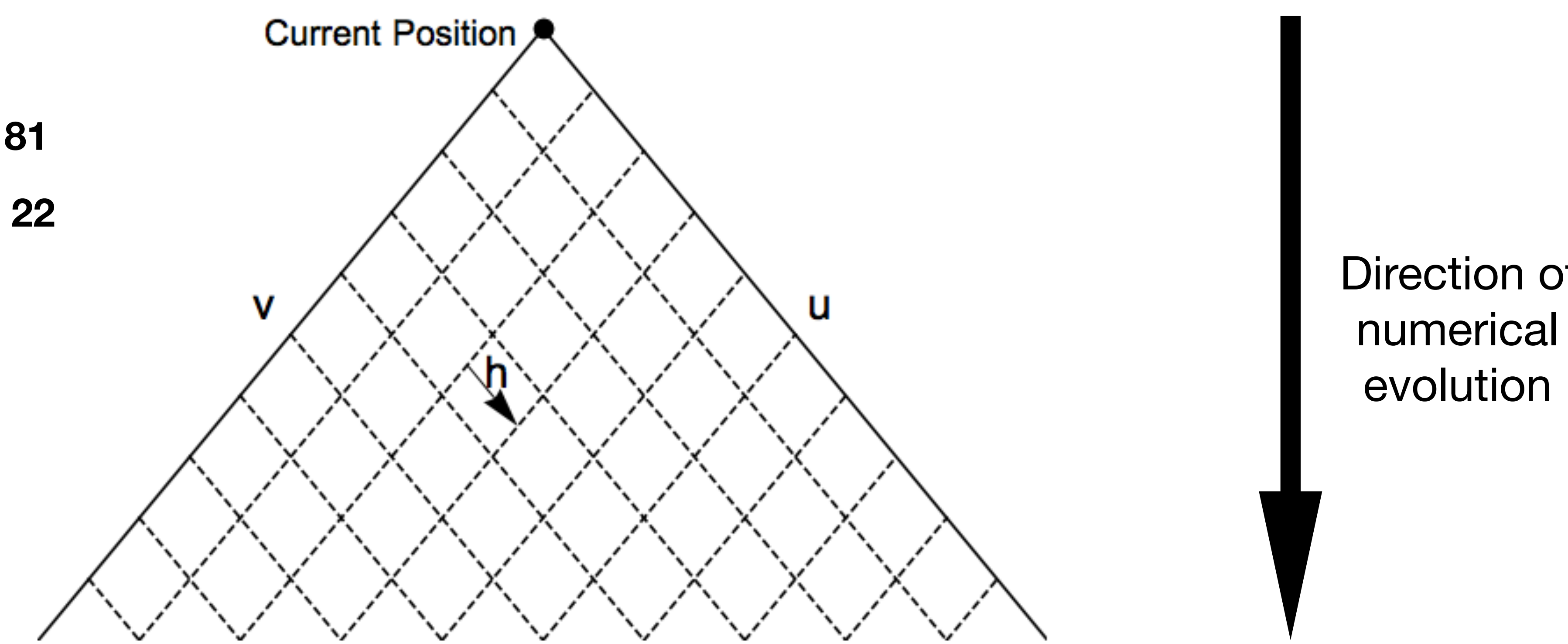
- Inside the past light cone, reduces to a 2D wave equation
- Grid in null coordinates, (u, v)

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$

Barack, Sago, *2010, Phys Rev D*, **81**

Lousto, *2005, Class. Quant. Grav.* **22**

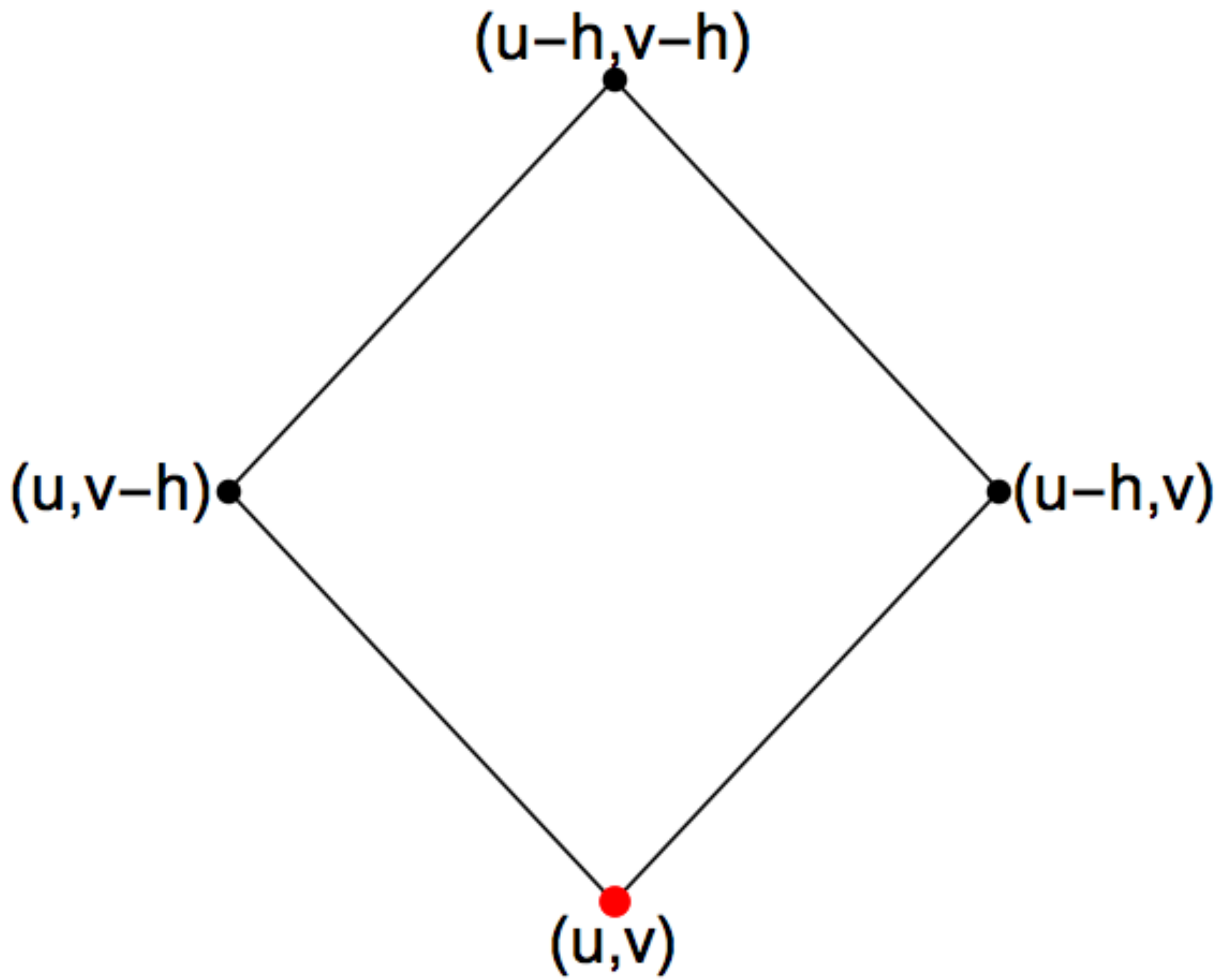
Haas, *2007, Phys. Rev. D* **75**



Past Light Cone

Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$



Points needed for 2nd Order in h

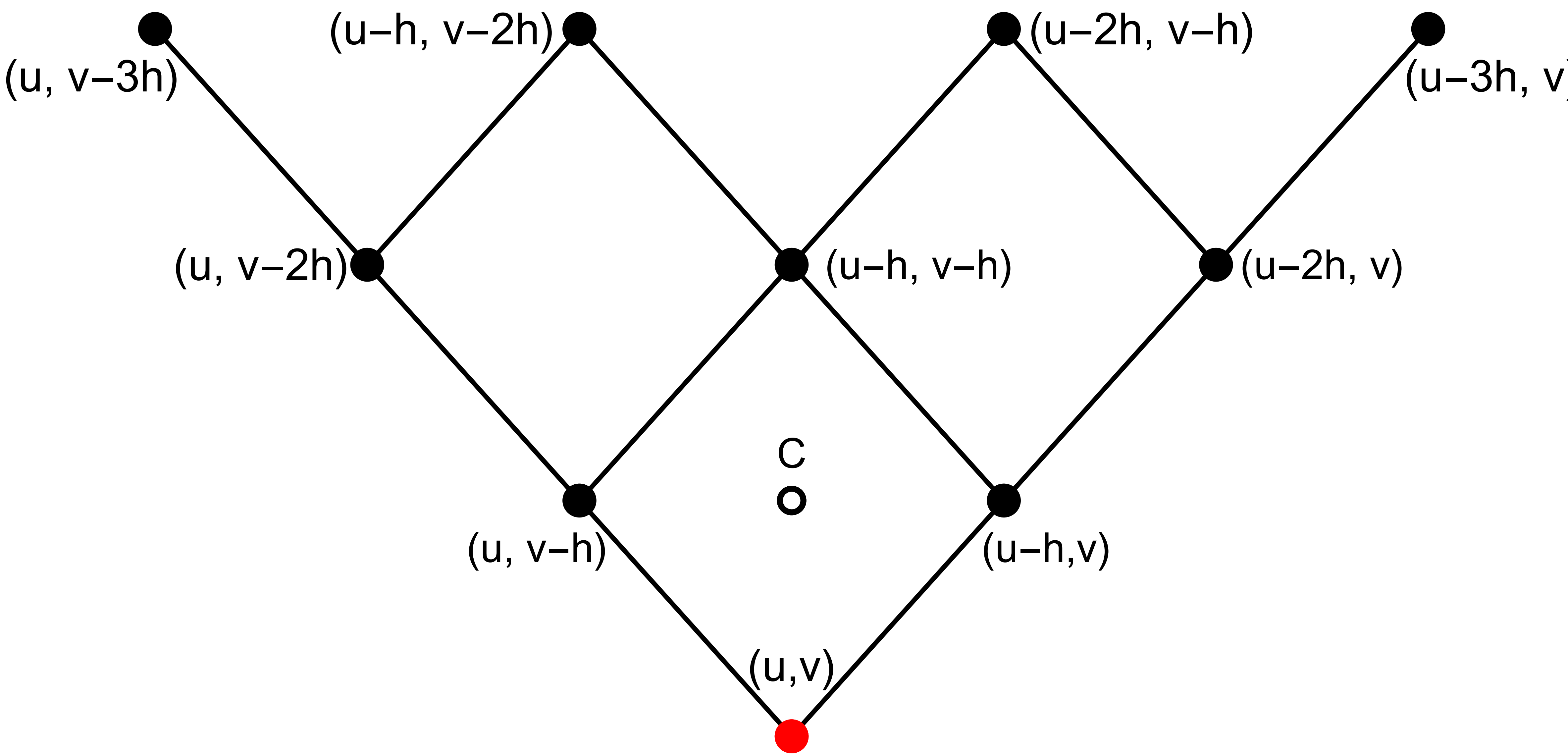
Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$

$$G_l(u,v) = -G_l(u-h, v-h) + [G_l(u-h, v) + G_l(u, v-h)]\left(1 - \frac{h^2}{2} W(u,v)\right)$$

Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$



Points needed for 4th Order in h

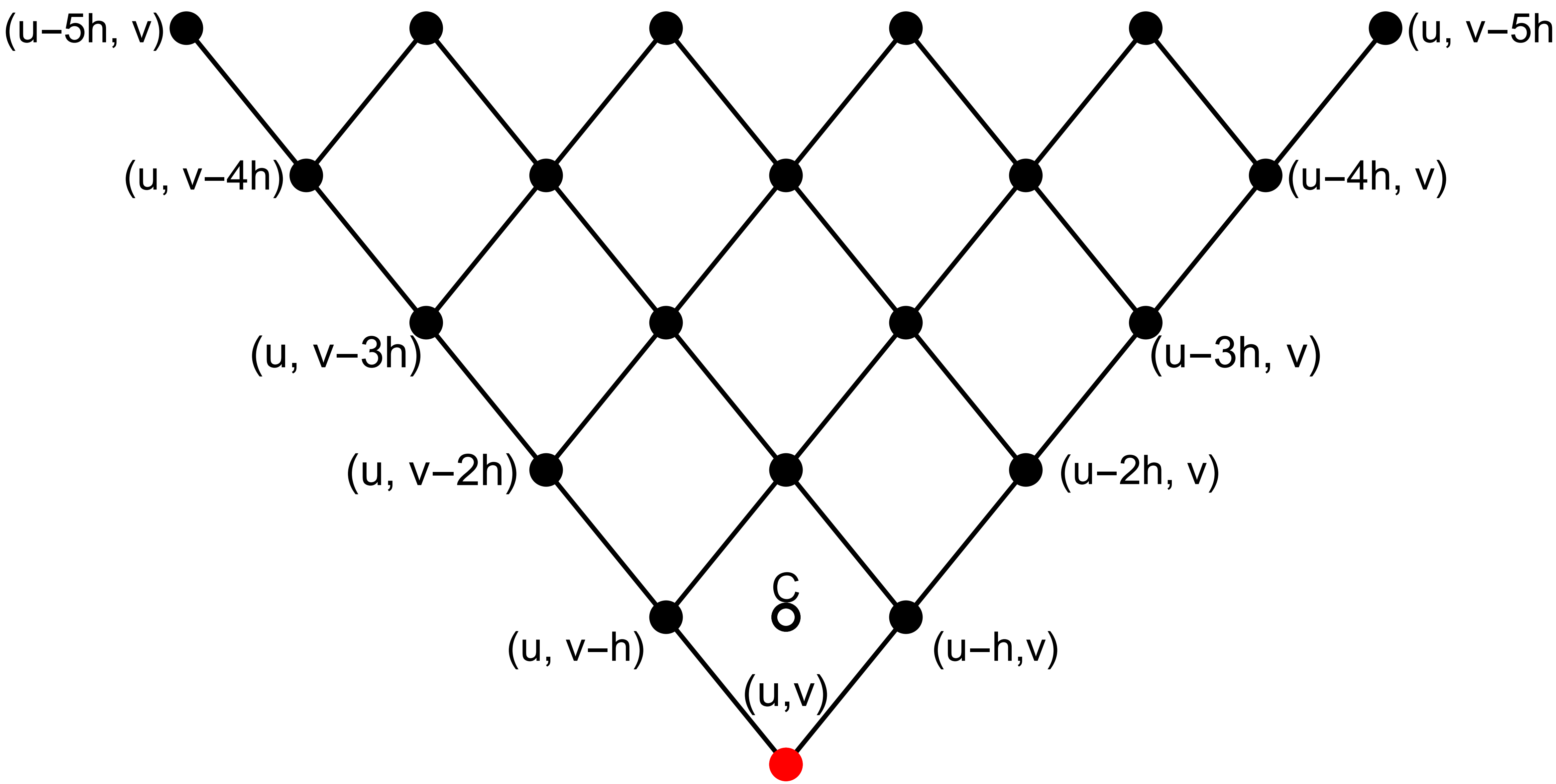
Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$

$$\begin{aligned}
G_l(u,v) \quad = \quad & -G_l(u-h,v-h) + G_l(u,v-h) + G_l(u-h,v) \\
& -\frac{h^2}{24}\left(2W(u,v)\left(-G_l(u-h,v-h) + (G_l(u-h,v) + G_l(u,v-h))\left(1 - \frac{h^2}{2}W(u,v)\right)\right)\right. \\
& +10(W(u-h,v-h)G_l(u-h,v-h) + W(u,v-h)G_l(u,v-h) + W(u-h,v)G_l(u-h,v)) \\
& -4(W(u-2h,v)G_l(u-2h,v) + W(u,v-2h)G_l(u,v-2h)) \\
& +(W(u-3h,v)G_l(u-3h,v) + W(u,v-3h)G_l(u,v-3h) \\
& \left. -W(u-h,v-2h)G_l(u-h,v-2h) - W(u-2h,v-h)G_l(u-2h,v-h))\right)
\end{aligned}$$

Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right) G_l = 0$$



Points needed for 6th Order in h

Time Domain Numerical Method

$$-4\partial^2_{uv}G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M}{r^3}\right)G_l = 0$$

$$\begin{aligned}
G_l(u,v) \quad = \quad & -G_l(u-h,v-h) + G_l(u,v-h) + G_l(u-h,v) \\
& -\frac{h^2}{1440}\Big(108W(u,v)G_l^{(4)}(u,v) + 371(W(u,v-3h)G_l(u,v-3h) + W(u-3h,v)G_l(u-3h,v)) \\
& -154(W(u-4h,v)G_l(u-4h,v) + W(u,v-4h)G_l(u,v-4h)) \\
& +116(W(u-3h,v-h)G_l(u-3h,v-h) + W(u-h,v-3h)G_l(u-h,v-3h)) \\
& +40W(u-2h,v-2h)G_l(u-2h,v-2h) \\
& +27(W(u-5h,v)G_l(u-5h,v) + W(u,v-5h)G_l(u,v-5h)) \\
& -19(W(u-4h,v-h)G_l(u-4h,v-h) + W(u-h,v-4h)G_l(u-h,v-4h)) \\
& -5(W(u-3h,v-2h)G_l(u-3h,v-2h) + W(u-2h,v-3h)G_l(u-2h,v-3h)) \\
& +627(W(u-h,v)G_l(u-h,v) + W(u,v-h)G_l(u,v-h)) \\
& +1032W(u-h,v-h)G_l(u-h,v-h) \\
& -504(W(u-2h,v)G_l(u-2h,v) + W(u,v-2h)G_l(u,v-2h)) \\
& -329(W(u-2h,v-h)G_l(u-2h,v-h) + W(u-h,v-2h)G_l(u-h,v-2h))\Big))
\end{aligned}$$

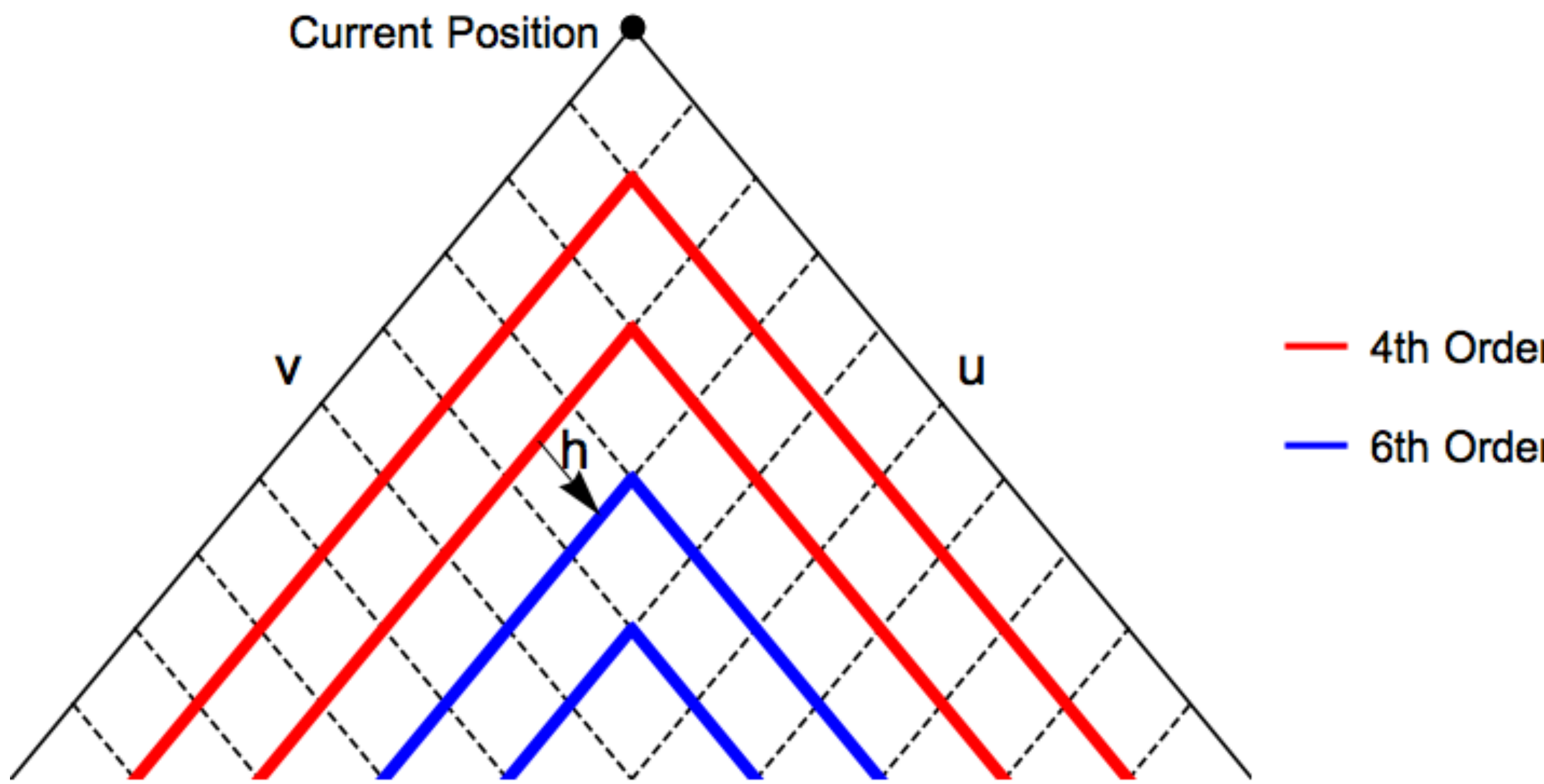
Time Domain Numerical Method

$$-4\partial_{uv}^2 G_l - \left(1 - \frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2} + \frac{2M(1-s^2)}{r^3}\right) G_l = 0$$

$$\begin{aligned}
G_l(u,v) \quad = \quad & -G_l(u-h,v-h) + G_l(u,v-h) + G_l(u-h,v) \\
& -\frac{h^2}{1440}\Big(108W(u,v)G_l^{(4)}(u,v) + 371(W(u,v-3h)G_l(u,v-3h) + W(u-3h,v)G_l(u-3h,v)) \\
& -154(W(u-4h,v)G_l(u-4h,v) + W(u,v-4h)G_l(u,v-4h)) \\
& +116(W(u-3h,v-h)G_l(u-3h,v-h) + W(u-h,v-3h)G_l(u-h,v-3h)) \\
& +40W(u-2h,v-2h)G_l(u-2h,v-2h) \\
& +27(W(u-5h,v)G_l(u-5h,v) + W(u,v-5h)G_l(u,v-5h)) \\
& -19(W(u-4h,v-h)G_l(u-4h,v-h) + W(u-h,v-4h)G_l(u-h,v-4h)) \\
& -5(W(u-3h,v-2h)G_l(u-3h,v-2h) + W(u-2h,v-3h)G_l(u-2h,v-3h)) \\
& +627(W(u-h,v)G_l(u-h,v) + W(u,v-h)G_l(u,v-h)) \\
& +1032W(u-h,v-h)G_l(u-h,v-h) \\
& -504(W(u-2h,v)G_l(u-2h,v) + W(u,v-2h)G_l(u,v-2h)) \\
& -329(W(u-2h,v-h)G_l(u-2h,v-h) + W(u-h,v-2h)G_l(u-h,v-2h))\Big))
\end{aligned}$$

Initial Conditions

- For higher order, we need data on more than one ‘previous’ null surface



- We use a standard ansatz from BHPT, the Hadamard series

$$G_l = \Theta(-\sigma) \sum_{n=0}^{\infty} U_n \sigma^n \qquad \sigma(x, x') = -\frac{1}{2}(\Delta\tau)^2 = -\frac{1}{2}\Delta u \Delta v$$

Décanini, Folacci, *2006, Phys Rev D*, **73**

Initial Conditions

- Can derive transport equations for the U_n's

$$U_0 = \Delta(x, x') = 1$$

$$-fV_l + 2(\Delta u \partial_u U_1 + \Delta v \partial_v U_1) + 2U_1 = 0$$

$$\Box U_n - fV_l U_n + 2(n+1)(\Delta u \partial_u U_{n+1} + \Delta v \partial_v U_{n+1}) + 2(n+1)^2 U_{n+1} = 0$$

- Coefficients independent of t (in Schwarzschild)

- For 4th order, need at least U_5, for 6th order U_7, etc.

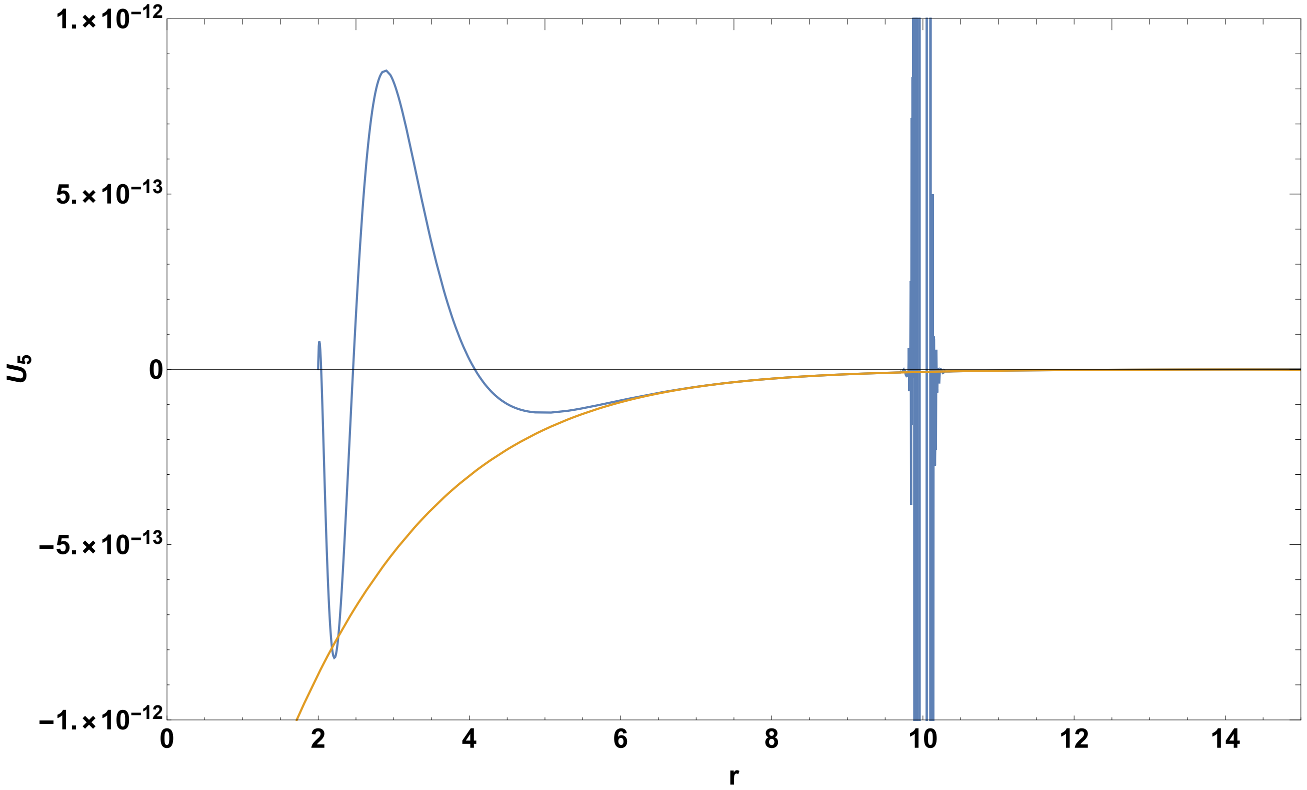
Initial Conditions

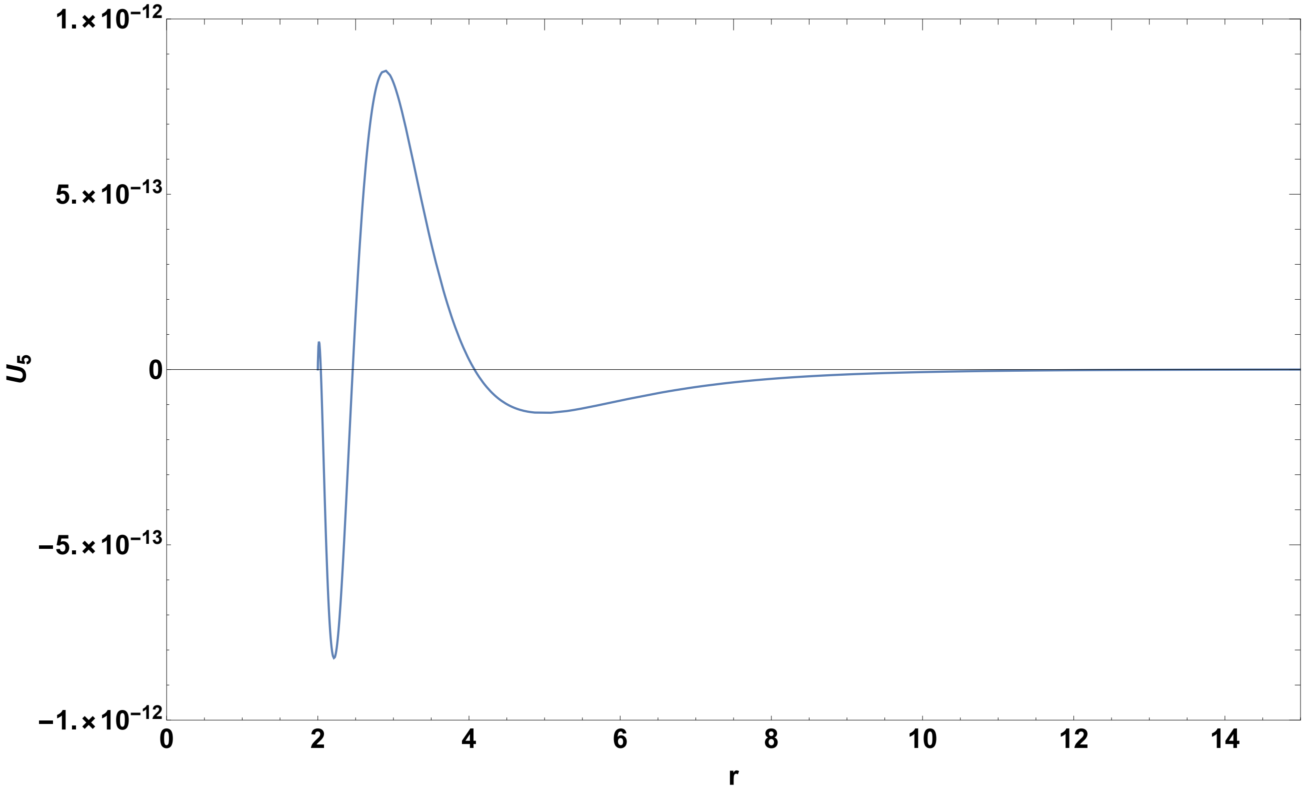
$$U_1(r, r_0) = \frac{(r - r_0)(-M(r + r_0)(s^2 - 1) + r r_0 l(l+1))}{2r^2 r_0^2 (r^* - r_0^*)}$$

$$\begin{aligned} U_2(r, r_0) \quad &= \quad \frac{(r - r_0)^2(-M(r + r_0)(s^2 - 1) + r r_0 l(l+1))^2}{16 r^4 r_0^4 (r^* - r_0^*)^2} \\ &+ \frac{(r - r_0)(-M(r + r_0)(s^2 - 1) + l(l+1) r r_0)}{4r^2 r_0^2 (r^* - r_0^*)^3} \\ &- \frac{(-2M + r0)(l(l+1) r_0 - 2M(s^2 - 1))}{4 r_0^4 (r^* - r_0^*)^2} \\ &- \frac{1}{8(r^* - r_0^*)^2} \left(1 - \frac{2M}{r}\right) \left(\frac{l(l+1)}{r^2} - \frac{2M(s^2 - 1)}{r^3}\right) \\ &+ \frac{1}{8(r^* - r_0^*)^2} \left(1 - \frac{2M}{r_0}\right) \left(\frac{l(l+1)}{r_0^2} - \frac{2M(s^2 - 1)}{r_0^3}\right) \end{aligned}$$

Initial Conditions

- We must treat the value of the initial conditions carefully near coincidence





Green Function Derivatives

- For the self-force, also require the derivatives of the Green Function: $\partial_u G$, $\partial_v G$
- For example, for a scalar charge:

$$F_\mu = q \int_{-\infty}^{\tau^-} \nabla_\mu G(z(\tau), z(\tau'))d\tau'$$

- Can use finite differencing or compute directly
- Latter requires values of both derivatives and Green function on previous points in the domain

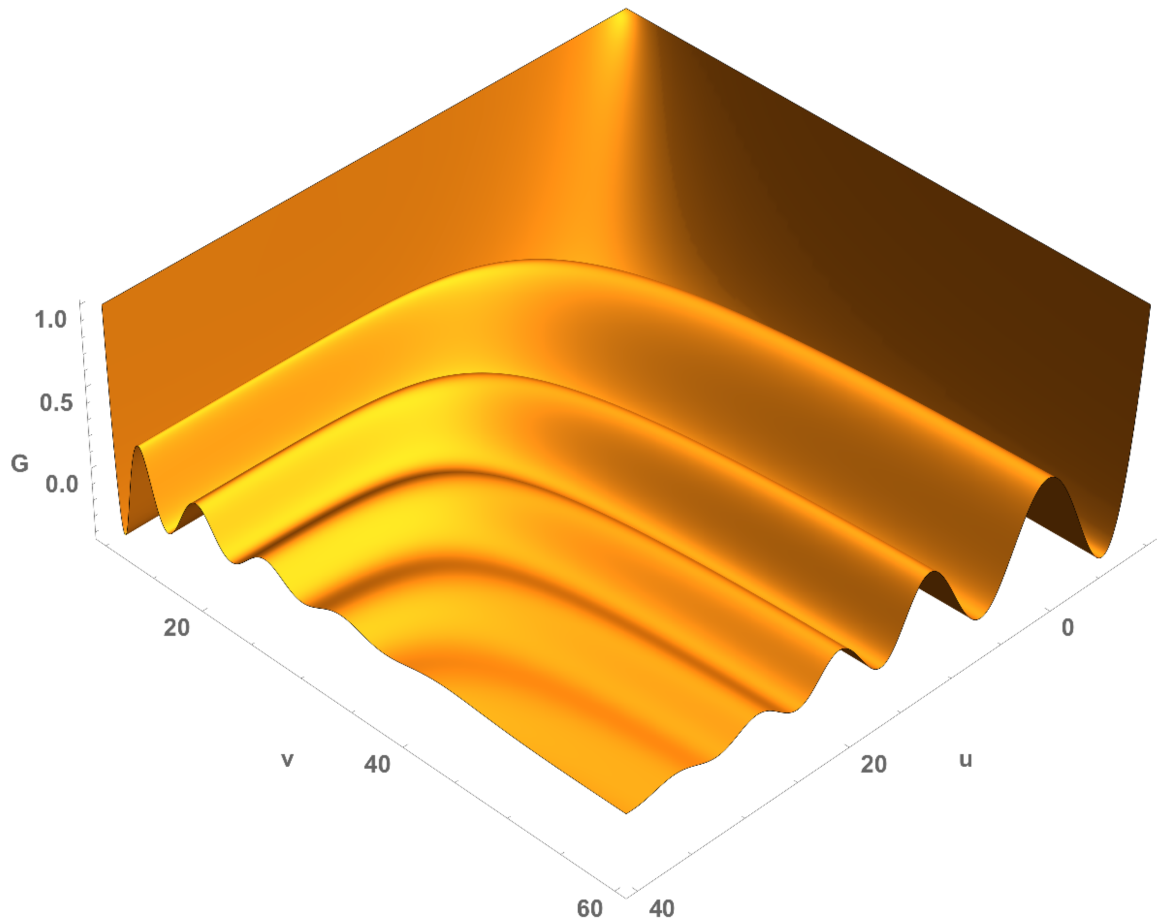
Green Function Derivatives

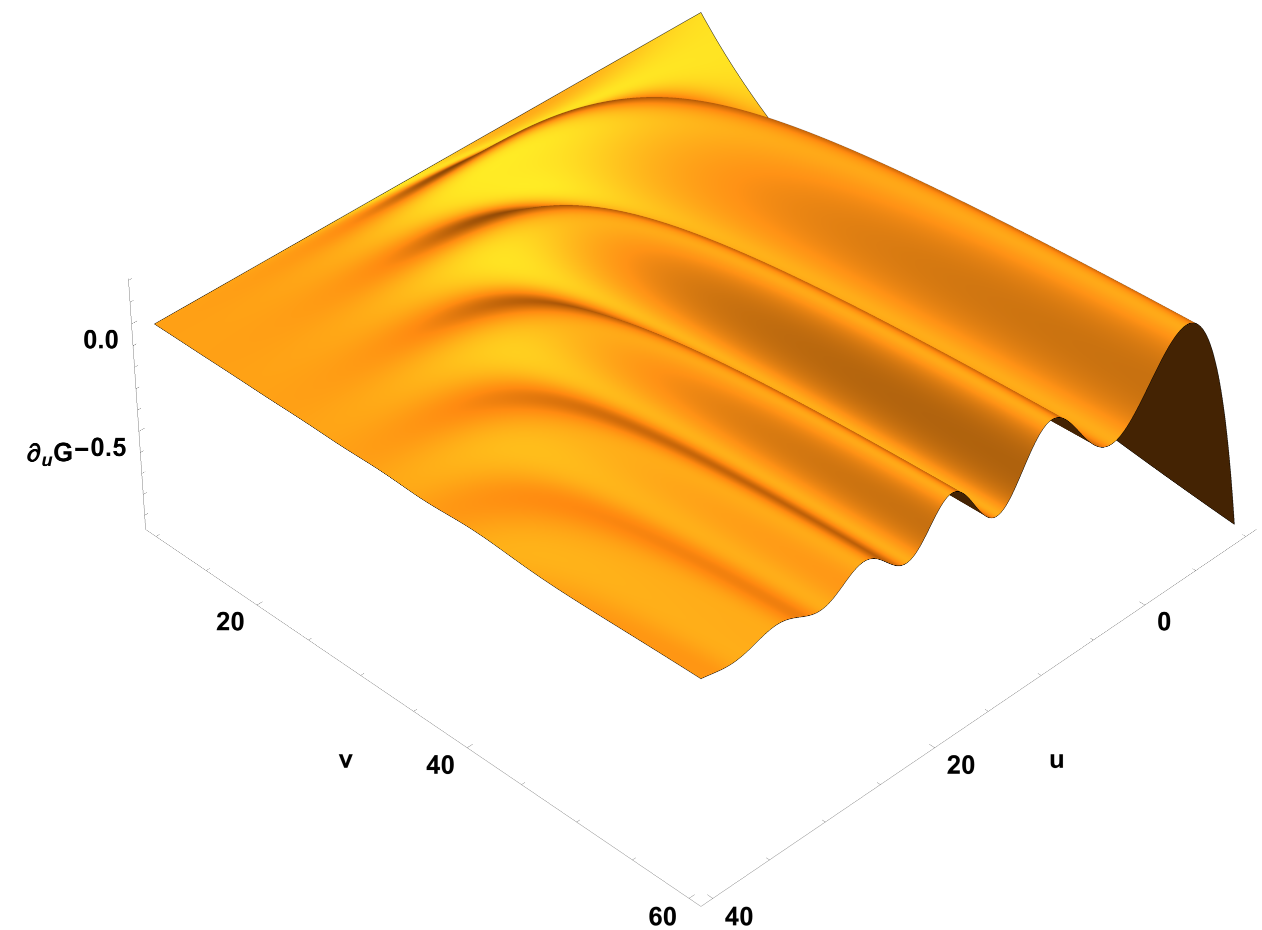
- For direct computation, start by differentiating the wave equation

$$-4\partial^2_{uv}(\partial_u G_l) - \partial_u\left(\left(1-\frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2}+\frac{2M(1-s^2)}{r^3}\right)\right)G_l$$
$$-\left(1-\frac{2M}{r}\right)\left(\frac{l(l+1)}{r^2}+\frac{2M(1-s^2)}{r^3}\right)\partial_u G_l = 0$$

- The first term is the same as before, the second and third are both of familiar form
- We then differentiate the Hadamard series to obtain the necessary initial data
- This requires finding derivatives of the U_n's, which we have expressions for
- Note that now 2nd order will require U_1, 4th order U_6, 6th order U_8, etc.

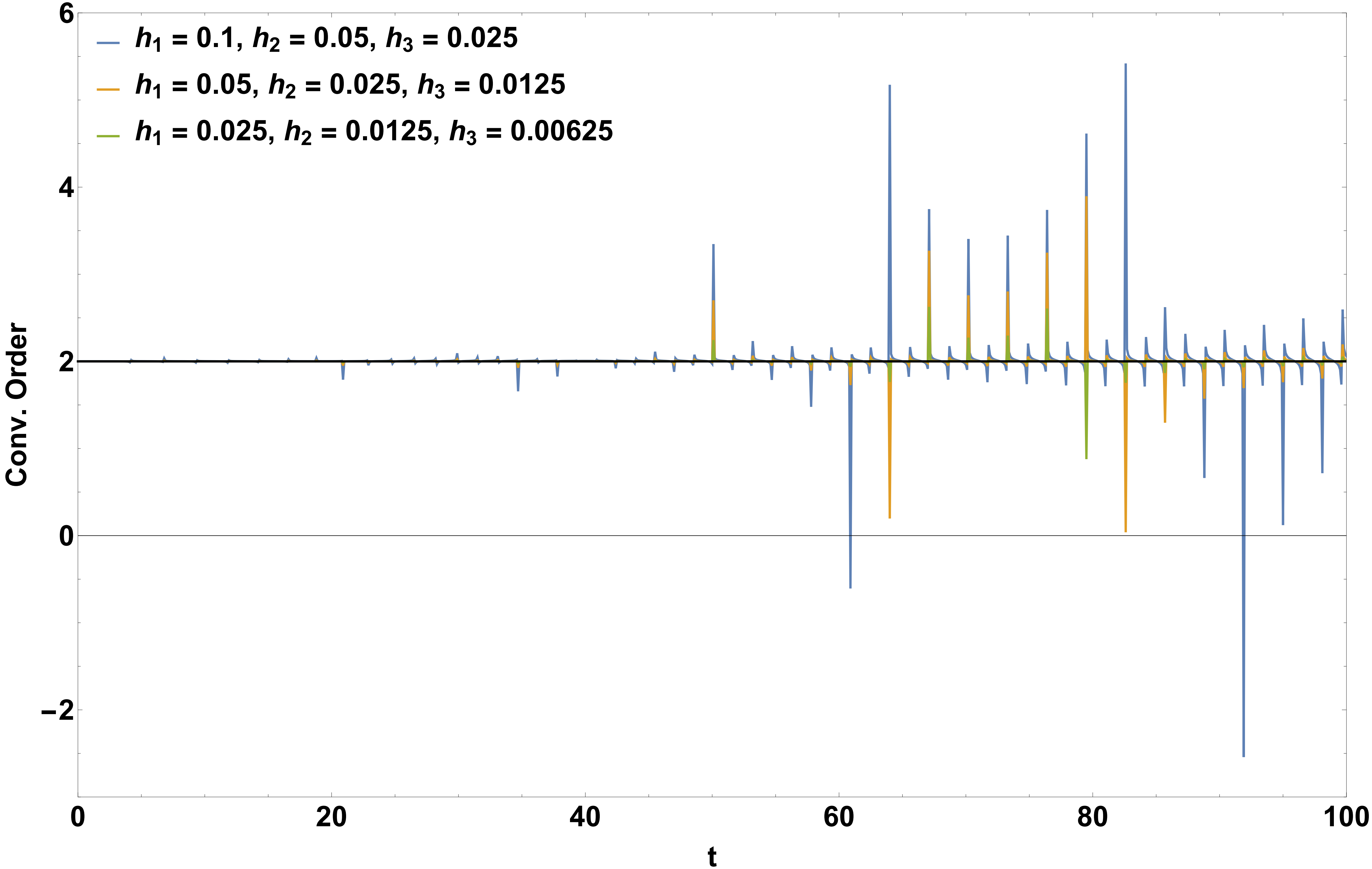
Results





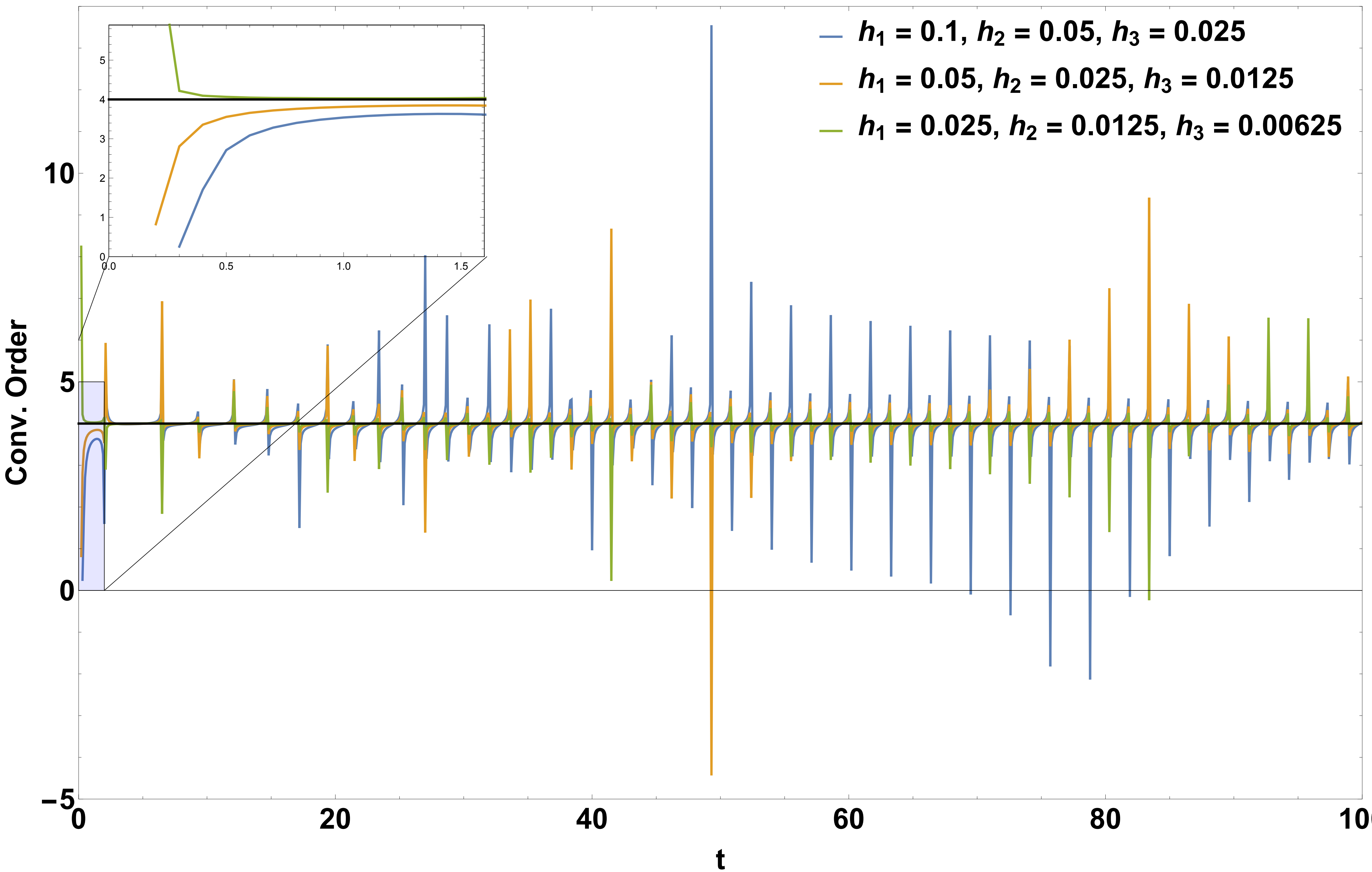
- Mathematica implementation of 2nd and 4th order algorithms for G and both derivatives
- Algorithms valid for scalar, electromagnetic and gravitational perturbations to background
- Extension to higher orders straightforward (6th started)
- C code implementation of the above

Results



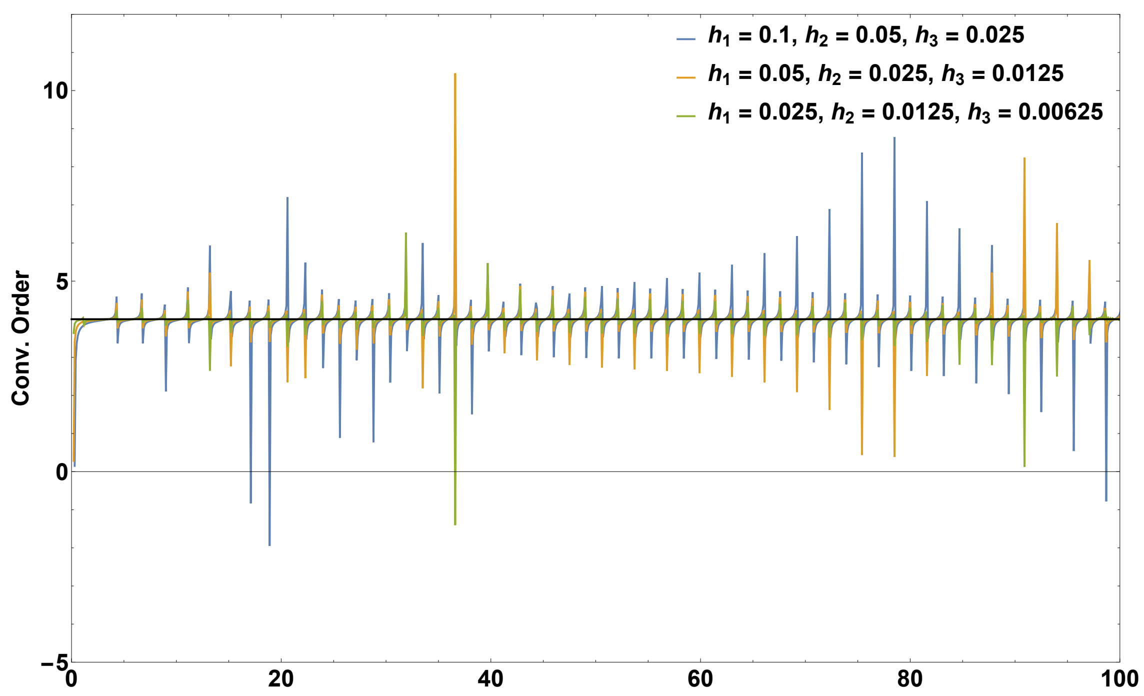
h_1 = 0.1, h_2 = 0.05, h_3 = 0.025
h_1 = 0.05, h_2 = 0.025, h_3 = 0.0125
h_1 = 0.025, h_2 = 0.0125, h_3 = 0.00625
Conv. Order
t
6
4
2
0
−2
0
20
40
60
80
100

Results

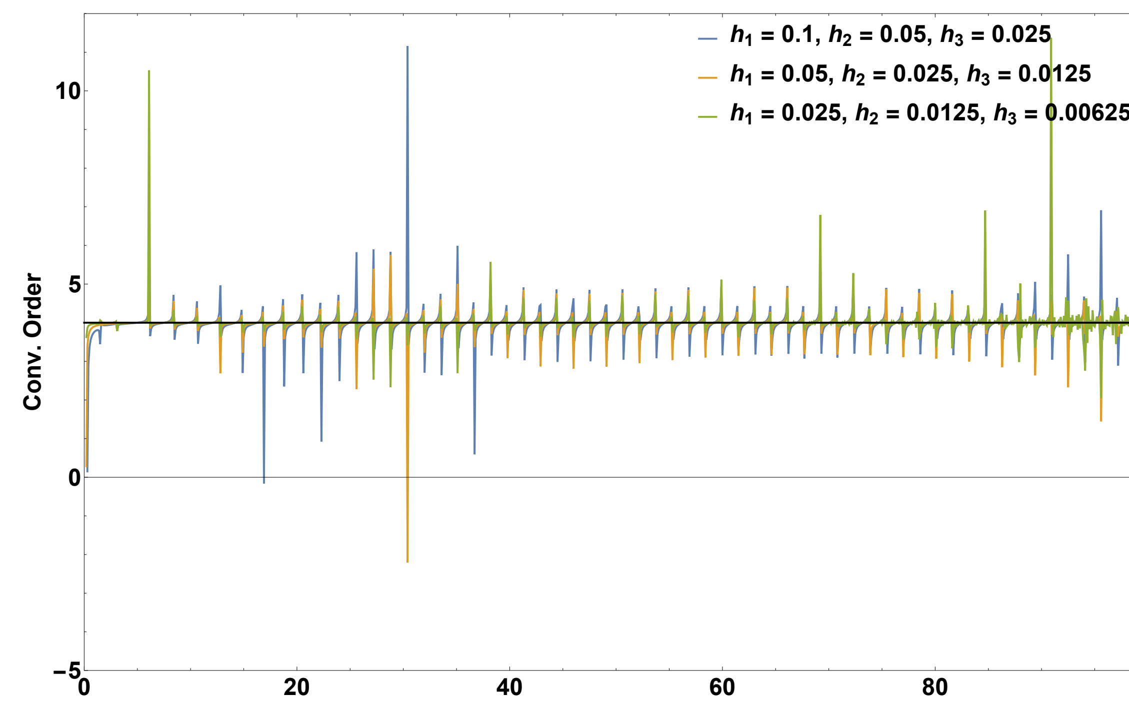


h_1 = 0.1, h_2 = 0.05, h_3 = 0.025
h_1 = 0.05, h_2 = 0.025, h_3 = 0.0125
h_1 = 0.025, h_2 = 0.0125, h_3 = 0.00625
Conv. Order
t

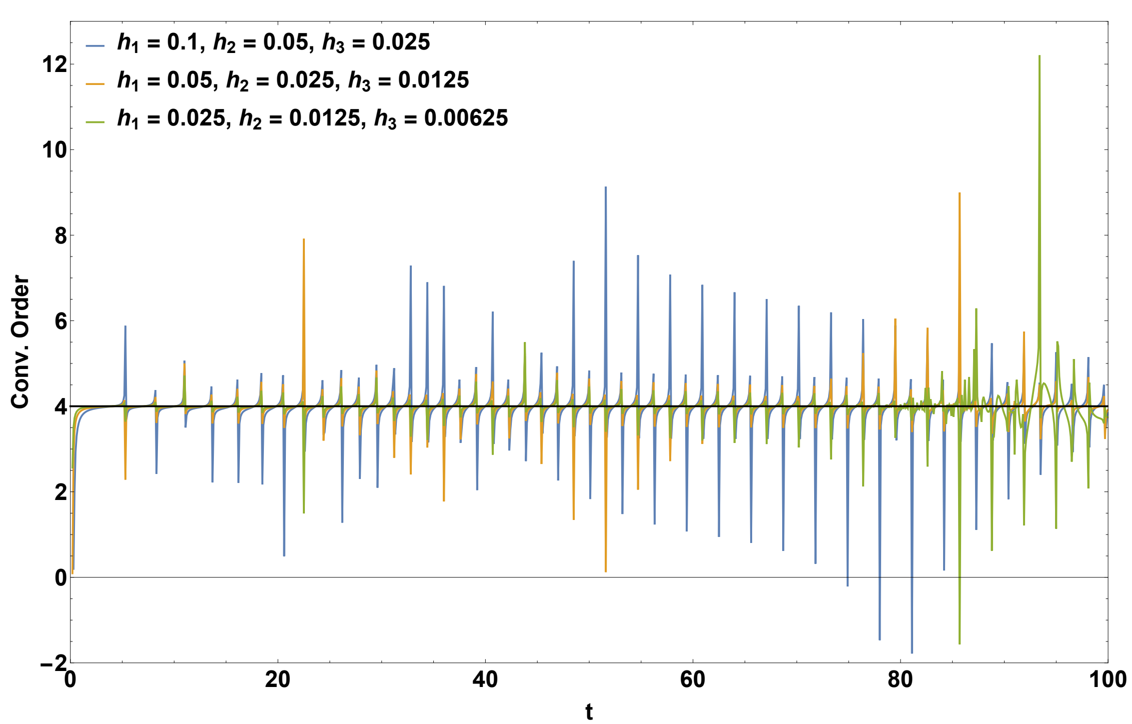
Results



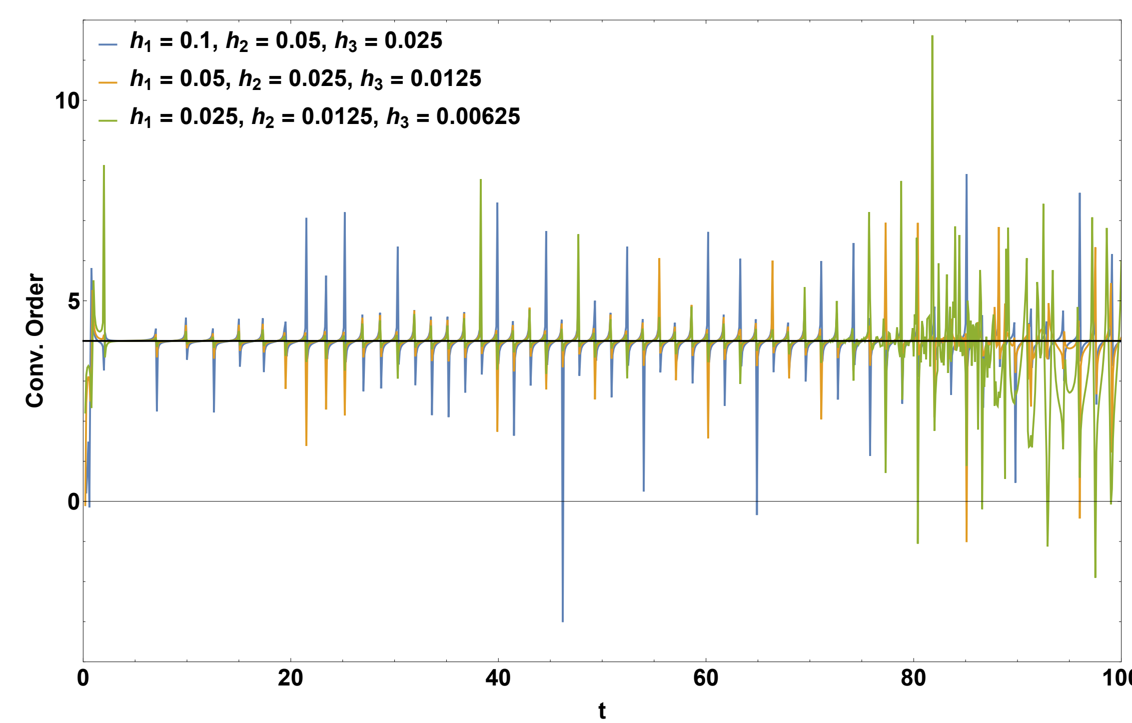
$h_1 = 0.1$, $h_2 = 0.05$, $h_3 = 0.025$
$h_1 = 0.05$, $h_2 = 0.025$, $h_3 = 0.0125$
$h_1 = 0.025$, $h_2 = 0.0125$, $h_3 = 0.00625$
Conv. Order
10
5
0
−5
0
20
40
60
80
100



$h_1 = 0.1$, $h_2 = 0.05$, $h_3 = 0.025$
$h_1 = 0.05$, $h_2 = 0.025$, $h_3 = 0.0125$
$h_1 = 0.025$, $h_2 = 0.0125$, $h_3 = 0.00625$
Conv. Order
10
5
0
−5
0
20
40
60
80
100

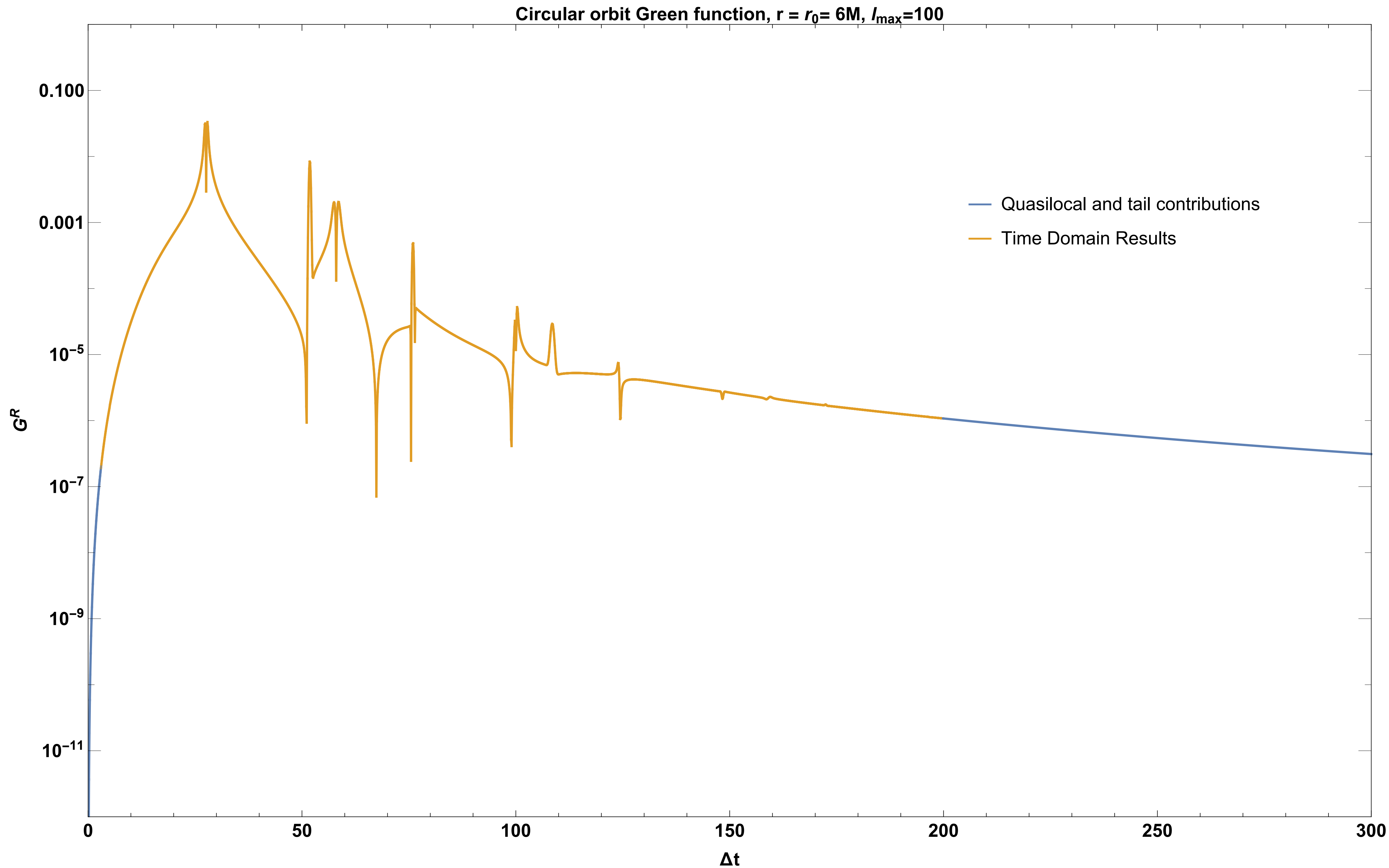


$h_1 = 0.1$, $h_2 = 0.05$, $h_3 = 0.025$
$h_1 = 0.05$, $h_2 = 0.025$, $h_3 = 0.0125$
$h_1 = 0.025$, $h_2 = 0.0125$, $h_3 = 0.00625$
Conv. Order
12
10
8
6
4
2
0
−2
0
20
40
60
80
100
t



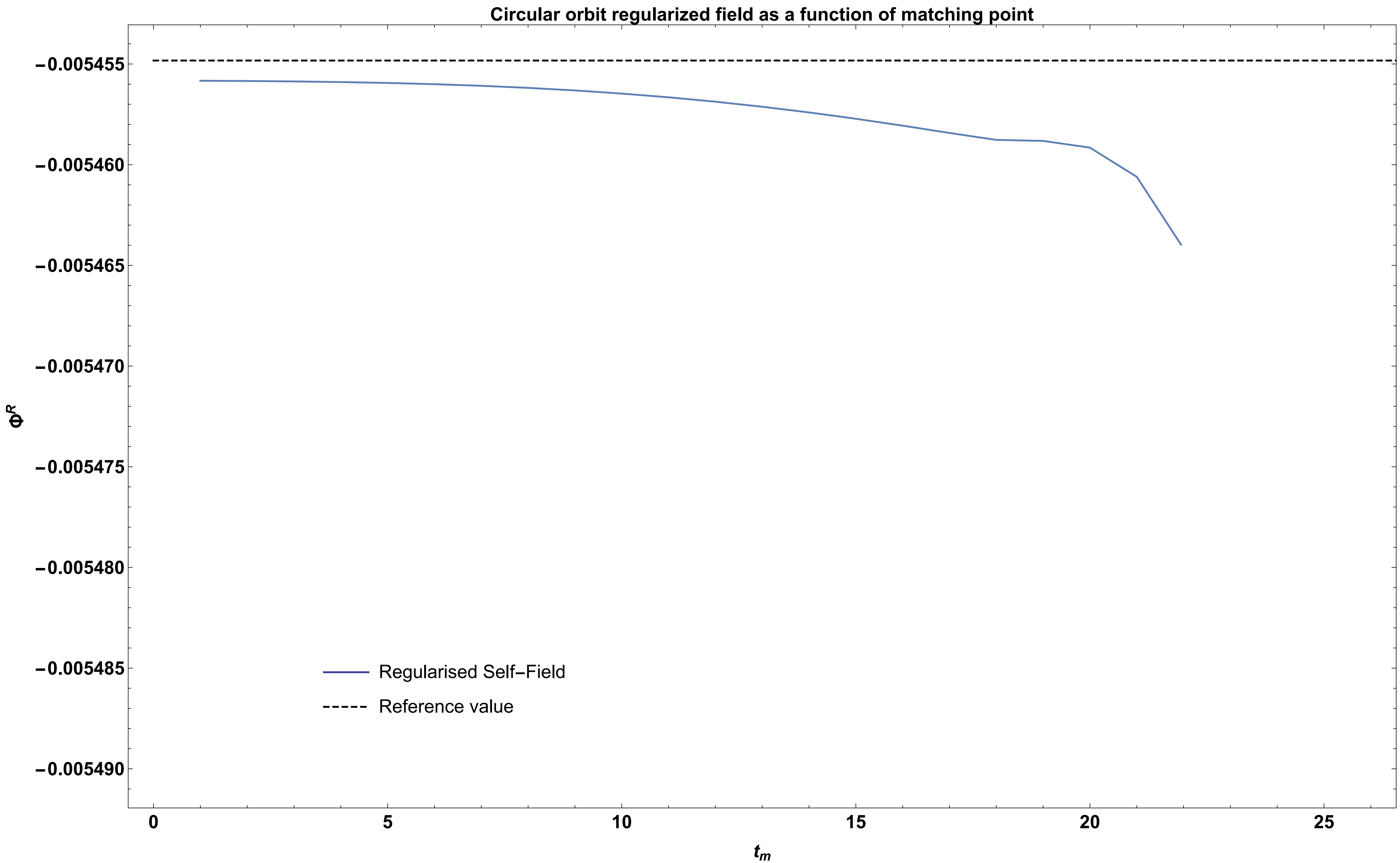
$h_1 = 0.1$, $h_2 = 0.05$, $h_3 = 0.025$
$h_1 = 0.05$, $h_2 = 0.025$, $h_3 = 0.0125$
$h_1 = 0.025$, $h_2 = 0.0125$, $h_3 = 0.00625$
Conv. Order
10
5
0
0
20
40
60
80
100
t

Results



Casals, Nolan, Ottewill, Wardell, *In prep.*

Results



Casals, Nolan, Ottewill, Wardell, *In prep.*
Diaz-Rivera, Messaritaki, Whiting, Detweiler, *2004, Phys. Rev. D* **70**

Future Work

- Scalar self-force
- Gravitational self-force
- Higher order schemes (in progress)
- Variety of orbits, bound and unbound
- Mixed second derivatives - Derivatives with respect to both the base point and field point (in progress)
- Input into Surrogate Model(s)
- Extension to Kerr

To name a few things…

Thank you
Questions?

Bonus Slide 1
From 4D to 2D

- Simplification to 2D is done through a number of straightforward steps

1. Tortoise Coordinate

$$r_* = r + 2M \log\left(\frac{r}{2M} - 1\right) \qquad \frac{dr}{dr_*} = 1 - \frac{2M}{r} = f$$

$$(*) \to -\frac{\partial^2 G}{\partial t^2} + \frac{\partial^2 G}{\partial r_*^2} + \frac{2f}{r}\frac{\partial G}{\partial r_*} + \frac{f}{r^2}\nabla^2_{\Omega_2} G = \frac{-4\pi}{r^2}\delta_2(x^A - x^{A'})\delta_{\Omega_2}(x^a, x^{a'})$$

2. Conformal Transformation

$$\hat{g}_{\mu\nu}(t, r_*, \theta, \phi) = \frac{g_{\mu\nu}(t, r_*, \theta, \phi)}{r^2} \qquad G = \frac{1}{rr'}\hat{G}$$

$$(*) \to -\frac{\partial^2 \hat{G}}{\partial t^2} + \frac{\partial^2 \hat{G}}{\partial r_*^2} + \frac{f}{r^2}\left(\nabla^2_{\Omega_2} - \frac{2M}{r}\right)\hat{G} = -4\pi\delta_2(x^A - x^{A'})\delta_{\Omega_2}(x^a, x^{a'})$$

3. Spherical Harmonic Decomposition

$$\hat{G}(x, x') = \frac{1}{4\pi}\sum_{l=0}^{\infty}(2l+1)G_l(x^A, x^{A'})P_l(\cos\gamma)$$

$$(*) \to -\frac{\partial^2 G_l}{\partial t^2} + \frac{\partial^2 G_l}{\partial r_*^2} - f\left(\frac{l(l+1)}{r^2} + \frac{2M(1-s^2)}{r^3}\right)G_l = -4\pi\delta_2(x^A - x^{A'})$$

4. Null Coordinates

$$u = t - r_* \qquad v = t + r_* \qquad g_{\mu\nu}(u, v) = -\frac{1}{2}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$(*) \to -4\frac{\partial^2 G_l}{\partial u \partial v} - f\left(\frac{l(l+1)}{r^2} + \frac{2M(1-s^2)}{r^3}\right)G_l = -4\pi\delta_2(x^A - x^{A'})$$

Bonus Slide 2
Kerr

- A number of significant issues posed by transition to Kerr:
  - Cannot fully separate angular coordinates
  - Initial conditions more complex
  - Reformulation of the problem

- But essentially the same steps as for Schwarzschild

- First of the above has been solved previously for similar problems

- Third may require exploring a number of coordinate systems

- Starting with Bardeen-Press-Teukolsky, then moving on to Teukolsky

Bonus Slide 3
Other To-Dos

- Mixed second derivatives - Derivatives with respect to both the base point and field point
  - Necessary for computing stress-energy
  - Also possibly useful for 2nd-Order Osculating Elements
- Experiment with hyperboloidal coordinates
  - Possibly obtain the global Green function
- Hyperbolic scatterings
  - Potential comparisons with Post-Minkowskian methods
- Zoom-whirls
  - Building on discussions from Monday
- Arbitrary precision using GSL
  - Improves C codes robustness to numerical roundoff errors
- Applications to QFT in curved spacetime
- Self-consistent evolution