Self-force on a scalar particle on a generic orbit in Kerr spacetime

Jonathan Thornburg

in collaboration with

Barry Wardell

Department of Astronomy
Indiana University
Bloomington, Indiana, USA

Numerical Relativity Group
Albert Einstein Institute
Potsdam, Germany

Outline

The need for high accuracy

Outline

The need for high accuracy ⟦already discussed by Evans⟧

Outline

The need for high accuracy ⟦already discussed by Evans⟧

Goals

Outline

The need for high accuracy ⟦already discussed by Evans⟧

Goals

Overall plan

Outline

The need for high accuracy [[already discussed by Evans]]

Goals

Overall plan

Some details of puncture-function computations:

- time-domain, AMR (vs. frequency-domain)
- *m*-mode decomposition and 2-D evolution (vs. 3-D evolution)
- worldtube scheme (vs. window function)
- computing the puncture function and effective source
  - efficiently computing the ϕ integrals
  - efficiently evaluating Wardell's 4th-order puncture
- finite differencing near the puncture

Outline

The need for high accuracy ⟦already discussed by Evans⟧

Goals

Overall plan

Some details of puncture-function computations:

- time-domain, AMR (vs. frequency-domain)
- m-mode decomposition and 2-D evolution (vs. 3-D evolution)
- worldtube scheme (vs. window function)
- computing the puncture function and effective source
  - efficiently computing the ϕ integrals
  - efficiently evaluating Wardell's 4th-order puncture
- finite differencing near the puncture

(Very) Preliminary results

Conclusions, Directions for Further Research

Goals

- Kerr
- scalar field for now, gravitational field in the future
- compute self-force very accurately
- equatorial circular orbit $\rightarrow$ equatorial eccentric orbit $\rightarrow$ generic orbit
- as efficient as possible (orbital evolution in the future)

This is work in progress!

Overall Plan

Puncture-function regularization ⟦as per Vega, Wardell, Dolan talks⟧

- use Wardell's 4th order puncture function and effective source
- scalar field for now, gravitational field in the future

Overall Plan

Puncture-function regularization ⟦as per Vega, Wardell, Dolan talks⟧

- use Wardell's 4th order puncture function and effective source
- scalar field for now, gravitational field in the future

Time domain

- can handle any orbit, including high eccentricity
- Cauchy evolution (well-understood, could even use Cactus)

Overall Plan

Puncture-function regularization ⟦as per Vega, Wardell, Dolan talks⟧

- use Wardell's 4th order puncture function and effective source
- scalar field for now, gravitational field in the future

Time domain

- can handle any orbit, including high eccentricity
- Cauchy evolution (well-understood, could even use Cactus)
- AMR ($\Rightarrow$ accurate, efficient)
  - hopefully competetive with frequency-domain at much smaller e

Overall Plan

Puncture-function regularization ⟦as per Vega, Wardell, Dolan talks⟧

- use Wardell's 4th order puncture function and effective source
- scalar field for now, gravitational field in the future

Time domain

- can handle any orbit, including high eccentricity
- Cauchy evolution (well-understood, could even use Cactus)
- AMR ($\Rightarrow$ accurate, efficient)
  - hopefully competetive with frequency-domain at much smaller e
- spatial boundaries far enough away to be causally disconnected (with AMR this isn't that expensive)
- try higher order finite differencing for improved accuracy/efficiency

Overall Plan

Puncture-function regularization [[as per Vega, Wardell, Dolan talks]]

- use Wardell's 4th order puncture function and effective source
- scalar field for now, gravitational field in the future

Time domain

- can handle any orbit, including high eccentricity
- Cauchy evolution (well-understood, could even use Cactus)
- AMR ($\Rightarrow$ accurate, efficient)
  - hopefully competetive with frequency-domain at much smaller e
- spatial boundaries far enough away to be causally disconnected (with AMR this isn't that expensive)
- try higher order finite differencing for improved accuracy/efficiency

m-mode decomposition $\Rightarrow$ 2+1D numerical evolution for each m

Worldtube scheme to treat far-from-the-particle region

m-mode Decomposition and 2-D Evolution

Vega, Detweiler, Diener, *et al.* numerically solve for the residual field φ_r in 3+1D,

$$\Box\varphi_\mathrm{r} = S_\mathrm{eff}$$

We (Barack, Dolan, Wardell, Thornburg) prefer to Fourier-decompose in ϕ [actually in $\tilde{\phi} := \phi + f(r)$ to avoid infinite-twisting at horizon], then numerically solve for each individual Fourier mode of the residual field $\varphi_\mathrm{r}^m := (\varphi - \varphi_\mathrm{punct})^m$ in 2+1D,

$$\Box^m\varphi_\mathrm{r}^m = S_\mathrm{eff}^m$$

where the 2-D puncture function and effective source are given by

$$\begin{aligned}
\varphi_\mathrm{punct}^m(t,r,\theta) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} \varphi_\mathrm{punct}(t,r,\theta,\phi) e^{-im\tilde{\phi}}\, d\tilde{\phi} \\
S_\mathrm{eff}^m(t,r,\theta) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} S_\mathrm{eff}(t,r,\theta,\phi) e^{-im\tilde{\phi}}\, d\tilde{\phi}
\end{aligned}$$

The Worldtube (Idea)

Problem: puncture function and effective source aren't well-defined far from the particle.

Solutions:

- window function ⟦Vega, Detweiler, Diener, *et al.*⟧

The Worldtube (Idea)

Problem: puncture function and effective source aren't well-defined far from the particle.

Solutions:

- window function ⟦Vega, Detweiler, Diener, *et al.*⟧
- world tube

World tube: Introduce 2-D world tube (of macroscopic size) containing particle. Then the "numerical field"

$$\varphi^m_{\text{num}} := \begin{cases} (\varphi - \varphi_{\text{punct}})^m & \text{inside worldtube} \\ \varphi^m & \text{outside worldtube} \end{cases}$$

satisfies

$$\Box_m \varphi^m_{\text{num}} = \begin{cases} S^m_{\text{eff}} & \text{inside worldtube} \\ 0 & \text{outside worldtube} \end{cases}$$

The Worldtube (Implementation)

The worldtube is very easy to implement at the finite-differencing level:

- store φ^m_{num} as a grid function in the usual manner
- for each finite difference operation at each grid point
    - if the finite difference molecule crosses the worldtube boundary, then copy all the input data for the molecule to a small scratch array, adjusting by $\pm\varphi^m_{\mathsf{punct}}$ as appropriate, and apply the standard finite difference molecule to the scratch array
    - if the finite difference molecule doesn't cross the worldtube boundary, then use the standard finite difference operation

Only a small fraction of grid points are within a molecule radius of the worldtube boundary & hence need the [slow] adjustment; most grid points can use the [fast] ordinary finite difference operation. Thus this scheme is quite fast on average.

Efficiently Computing the ϕ Integrals (1)

For the 4th order puncture we haven't been able to do these Fourier integrals analytically, so we have to do them numerically. Since S_{eff}^m appears as the source term in the 2+1D evolution equation, we have to compute the S_{eff}^m Fourier integral at each 2+1D grid point (actually, at each 2+1D grid point in the worldtube) at each time step.

That's a lot of integrals!

[For an equatorial-circular orbit we can just compute the Fourier integrals at each 2+1D grid point in the worldtube at the first time step, then apply a phase factor to get S_{eff}^m cheaply at each later time step. But this "rotation trick" doesn't work for more general particle orbits.]

Efficiently Computing the ϕ Integrals (1)

For the 4th order puncture we haven't been able to do these Fourier integrals analytically, so we have to do them numerically. Since S_{eff}^m appears as the source term in the 2+1D evolution equation, we have to compute the S_{eff}^m Fourier integral at each 2+1D grid point (actually, at each 2+1D grid point in the worldtube) at each time step.

That's a lot of integrals!

[For an equatorial-circular orbit we can just compute the Fourier integrals at each 2+1D grid point in the worldtube at the first time step, then apply a phase factor to get S_{eff}^m cheaply at each later time step. But this "rotation trick" doesn't work for more general particle orbits.]

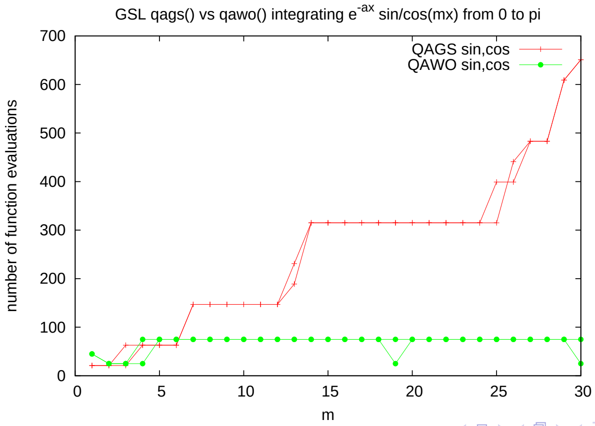
$\Rightarrow$ We don't yet know what fraction of the total computational cost will be consumed by these integrals.

Efficiently Computing the ϕ Integrals (2)

To actually compute the Fourier integrals, we use an adaptive numerical quadrature routine (GSL's `QAWO`) designed for oscillatory integrands.

$\Rightarrow$ cost of computing a Fourier integral is $\approx$ independent of m

(with a general-purpose numerical quadrature routine, cost $\propto m$)



The 3-D Puncture Function and Effective Source

Fix a $t =$ constant slice. Supose the particle is at x_p^i and we want to evaluate the 3-D puncture fn φ_{punct} at the position $x^i := x_p^i + \delta x^i$. Then

$$\varphi_{\text{punct}}(\delta r, \delta\theta, \delta\phi) = \frac{\sum_{ijk} N_{ijk} (\delta r)^i (\delta\theta)^j (\delta\phi)^{(k)}}{\left(\sum_{ijk} D_{ijk} (\delta r)^i (\delta\theta)^j (\delta\phi)^{(k)}\right)^{3/2}}$$

where the 18 N_{ijk} and 18 D_{ijk} coefficients depend on M, a, and x_p^i, but do **not** depend on δx^i, $k \in \{0, 2, 4\}$, and where

$$\begin{aligned}
\delta\phi^{(0)} &= 1 \\
\delta\phi^{(2)} &= \frac{5}{2} - \frac{8}{3}\cos\delta\phi + \frac{1}{6}\cos 2\delta\phi \quad \sim \quad (\delta\phi)^2 \ \text{ for } |\delta\phi| \ll 1 \\
\delta\phi^{(4)} &= 6 - 8\cos\delta\phi + 2\cos 2\delta\phi \quad \sim \quad (\delta\phi)^4 \ \text{ for } |\delta\phi| \ll 1
\end{aligned}$$

The 3-D effective source $S_{\text{eff}}(\delta r, \delta\theta, \delta\phi)$ is similar, but messier.

Efficiently Computing the 2-D Puncture Fn & Effective Src

To compute the 2-D effective source S^m_{eff} at a given $(\delta r, \delta\theta)$ (corresponding to a given 2-D (r, θ) grid point), the numerical integration of the Fourier integral requires computing $S_{\text{eff}}(\delta r, \delta\theta, \delta\phi)$ for many (~ 100) different $\delta\phi$ values.

Efficiently Computing the 2-D Puncture Fn & Effective Src

To compute the 2-D effective source S^m_{eff} at a given $(\delta r, \delta\theta)$ (corresponding to a given 2-D (r, θ) grid point), the numerical integration of the Fourier integral requires computing $S_{\text{eff}}(\delta r, \delta\theta, \delta\phi)$ for many ($\sim$100) different $\delta\phi$ values.

Thus, we define the coefficients

$$N_k(\delta r, \delta\theta) := \sum_{ij} N_{ijk}(\delta r)^i (\delta\theta)^j \qquad D_k(\delta r, \delta\theta) := \sum_{ij} D_{ijk}(\delta r)^i (\delta\theta)^j$$

so that

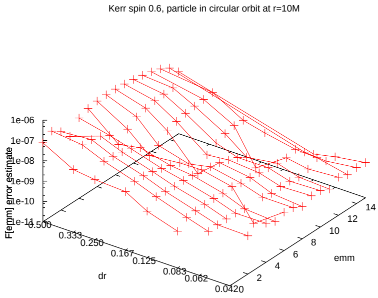
$$\varphi_{\text{punct}}(\delta r, \delta\theta, \delta\phi) = \frac{\sum_k N_k(\delta r, \delta\theta) \delta\phi^{(k)}}{\left(\sum_k D_k(\delta r, \delta\theta) \delta\phi^{(k)}\right)^{3/2}}$$

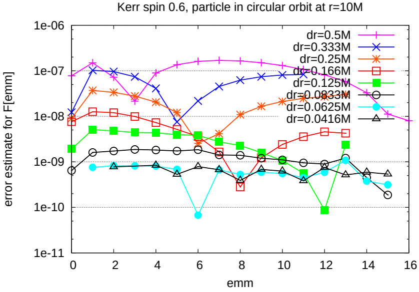
The coefficients N_{ijk} and D_{ijk} are computed once per time step (for the equatorial circular orbit case they're actually time-independent). The coefficients N_k and D_k are computed once per (worldtube) grid point per time step. Given N_k and D_k, $\varphi_{\text{punct}}(\delta r, \delta\theta, \delta\phi)$ is fairly cheap to compute.

(Very) Preliminary Results (1)

- Kerr BH (spin 0.6), particle in equatorial circular orbit at $r = 10M$
- unigrid, highest resolution $\Delta r_* = \frac{M}{16}$, $\Delta\theta = \frac{\pi}{640}$ radians ($\Rightarrow r\,\Delta\theta \approx \Delta r$ at particle)
- 4th order finite differencing (ignore non-smoothness at the particle)
- evolutions for $0 \le m \le 15$

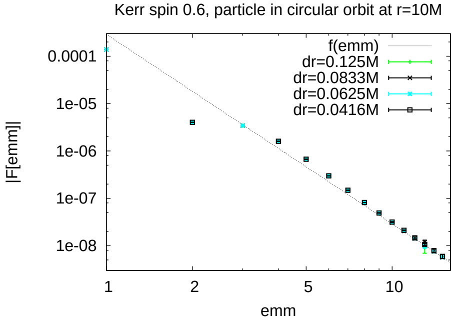
$\Rightarrow$ Between 2nd and 3rd order convergence with resolution:

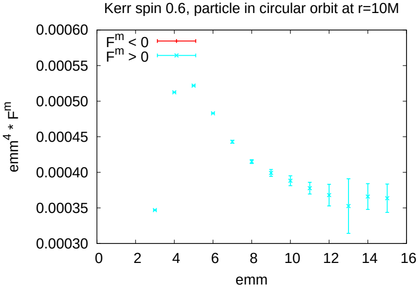




(Very) Preliminary Results (2)

Nice $\sim m^{-4}$ convergence of F^m at large m:

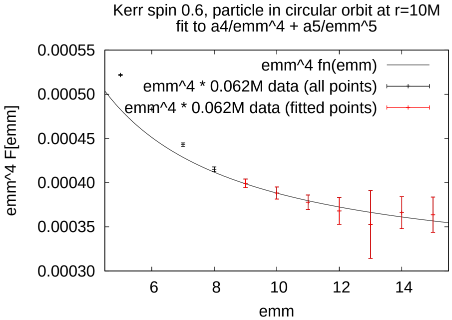




(Very) Preliminary Results (3)

Self-force in units of $10^{-6} q^2/M^2$:

$$
\begin{aligned}
\sum_{m=0}^{15} F^m &= -7.529 \pm 0.004 \\
\text{fitted } m^{-4} \text{ tail sum} &= 0.026 \pm 0.0008 \\
\text{fitted } m^{-5} \text{ tail sum} &= 0.004 \pm 0.0004 \\
\text{sum} &= -7.499 \pm 0.004 \quad (0.05\%) \\[4pt]
\text{true } F_{\text{self}} \text{ (Niels Warburton)} &= -7.491205 \\
\text{actual error} &= -0.008 \quad (0.11\%)
\end{aligned}
$$



Conclusions

- puncture-function regularization is very nice!
- *m*-mode decomposition & 2-D evolution are very nice!

Conclusions

- puncture-function regularization is very nice!
- *m*-mode decomposition & 2-D evolution are very nice!
- worldtube is quite easy to implement at the finite differencing level
  - don't yet know whether cost of this quantitatively

Conclusions

- puncture-function regularization is very nice!
- m-mode decomposition & 2-D evolution are very nice!
- worldtube is quite easy to implement at the finite differencing level
  - don't yet know whether cost of this quantitatively
- evaluating Wardell's 4th order puncture fn & effective src:
  - precompute coefficients at each 2-D grid point
  - use adaptive quadrature routine designed for oscillatory integrands
  - don't yet know whether evaluation cost will still be a problem

Conclusions

- puncture-function regularization is very nice!
- m-mode decomposition & 2-D evolution are very nice!
- worldtube is quite easy to implement at the finite differencing level
  - don't yet know whether cost of this quantitatively
- evaluating Wardell's 4th order puncture fn & effective src:
  - precompute coefficients at each 2-D grid point
  - use adaptive quadrature routine designed for oscillatory integrands
  - don't yet know whether evaluation cost will still be a problem
- (very) preliminary results:
  - naive 4th order finite differencing across particle gets $\approx$ 2.75th order
  - nice m^{-4} convergence of F^m at large m
  - $\Rightarrow \delta F_{\text{self}} \sim 10^{-3}$

Conclusions

- puncture-function regularization is very nice!
- m-mode decomposition & 2-D evolution are very nice!
- worldtube is quite easy to implement at the finite differencing level
  - don't yet know whether cost of this quantitatively
- evaluating Wardell's 4th order puncture fn & effective src:
  - precompute coefficients at each 2-D grid point
  - use adaptive quadrature routine designed for oscillatory integrands
  - don't yet know whether evaluation cost will still be a problem
- (very) preliminary results:
  - naive 4th order finite differencing across particle gets $\approx$ 2.75th order
  - nice m^{-4} convergence of F^m at large m
  - $\Rightarrow \delta F_{\text{self}} \sim 10^{-3}$
  - AMR should greatly improve this
  - don't yet know equal-efficiency point with frequency-domain

Directions for Further Research

Near-Term:

- don't integrate outside result domain of dependence
- angular boundary condition at equator
- better finite-differencing error estimates
- non-naive formula for $\delta\phi^{(2)}$ and $\delta\phi^{(4)}$
- interpolate singular field very close to particle to avoid cancellations in naive formula there
- more generic orbits (equatorial eccentric is in progress now)
- fancier finite differencing at particle (try to get $\geq$ 4th order)
- phase-align grid with particle?
- auto-adjust domain size

Medium-Term:

- AMR
- high-accuracy runs