Self Force Calculations using Field Regularization and a 3D Finite Difference Code

Peter Diener

Center for Computation & Technology
and
Department of Physics & Astronomy
Louisiana State University

In collaboration with Ian Vega, Steve Detweiler and Wolfgang Tichy

$\mathbf{12}^{\mathrm{th}}$ Capra Meeting on Radiation Reaction,

Indiana University, Bloomington,

June 18, 2009

Outline

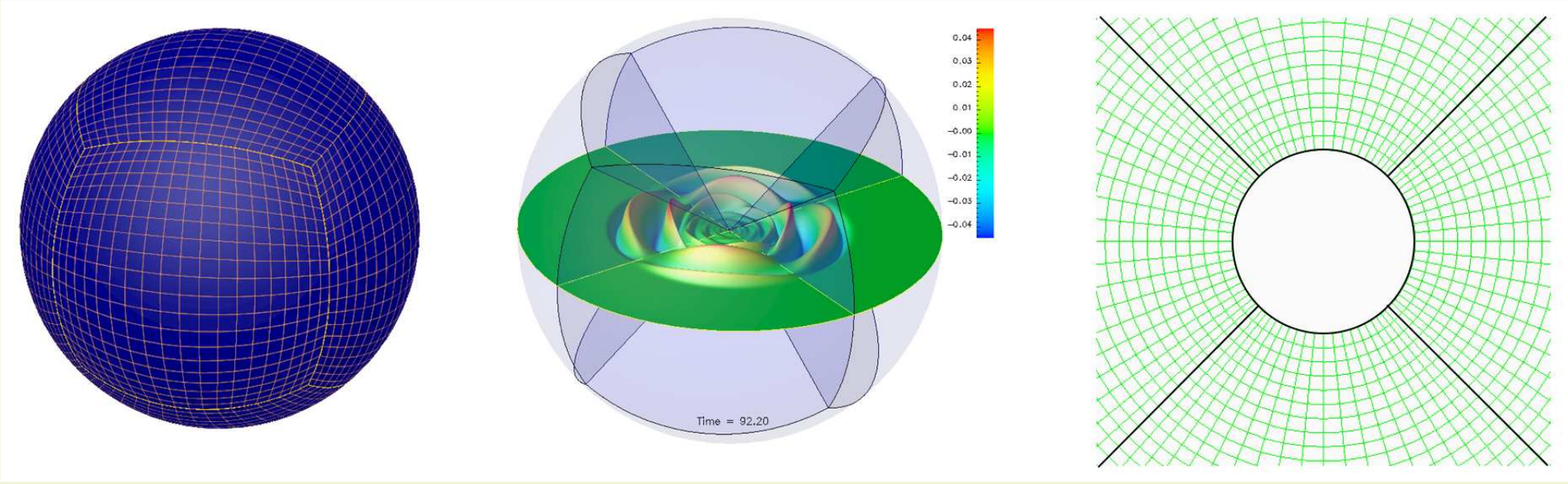
- Multi-block considerations.
- Summation by parts finite differencing.
- The penalty method.
- Self-force results.
- Concluding remarks.

Multi-block considerations.

The main reason for going to the trouble of implementing a multiple block code is that it allows for smooth (spherical) outer and/or excision boundaries without introducing coordinate singularities.

Additional advantages are:

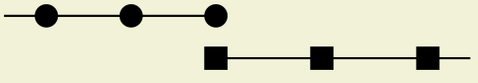
- Unnecessary angular resolution at large radii is avoided.
- It is possible to vary the radial resolution without introducing coordinate distortions.



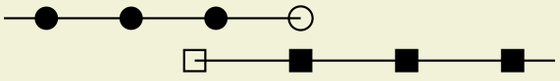
Multi-block considerations.

Choices to make:

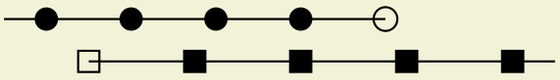
- Touching or overlapping patches.
  - ⋆ Touching patches (blocks) are mathematically well defined for linear equations in the sense that energy estimates can be obtained.
  - ⋆ Overlapping patches require interpolation.
- Global or local coordinate basis.
  - ⋆ Local basis is more natural, however it makes the inter-patch boundary conditions more complicated.
  - ⋆ Global basis, normally require some dissipation for stability. Inter-patch boundary condition is simpler. Visualization is simpler.



(a) Touching patches, also known as blocks.



(b) Touching patches with additional boundary points.



(c) Overlapping patches with boundary points.

Based on the experiences from an implementation by Lehner, Reula & Tiglio, Class. Quantum Grav. 22, 5283, 2005 we decided on a multi-block code. The scalar wave equation is implemented in a local coordinate basis. Implementation in Cactus using Carpet as the driver (E. Schnetter, PD, E. N. Dorband and M. Tiglio, Class. Quantum Grav. 23, S553, 2006).

Summation by parts finite differencing

Consider a 1D domain $[a, b]$ that is discretised with points $i = 0 \ldots N$. A difference operator D is said to satisfy summation by parts (SBP) if there exist a scalar product H defined by its coefficients σ_{ij}

$$\langle u, v \rangle = h \sum_{i,j=0}^{N} u_i v_j \sigma_{ij}$$

such that the property

$$\langle u, Dv \rangle + \langle Du, v \rangle = [uv]_a^b$$

holds for all grid-functions u, v. This is clearly just a discrete version of integration by parts.

The characteristics of the operators depend on the properties of the norm.

- Diagonal norm: The order of operator near the boundary is half the interior (2-1, 4-2, 6-3, 8-4, 10-5).
- Restricted full norm: The order of the operator near the boundary is one lower than the interior (4-3, 6-5).

SBP operators become more and more one sided near the boundary.

Summation by parts finite differencing

Diagonal norm (here interior 4th order and boundary 2nd order shown).

$$H = \begin{pmatrix} x & & & & & & \\ & x & & & & & \\ & & x & & & & \\ & & & x & & & \\ & & & & 1 & & \\ & & & & & \ddots & \end{pmatrix} \qquad D = \begin{pmatrix} x & x & x & x & & & & \\ x & x & x & x & & & & \\ x & x & x & x & x & & & \\ x & x & x & x & x & x & & \\ & & x & x & 0 & x & x & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}$$

2-1 and 4-2 operators are unique, 6-3 has one free parameter, 8-4 has three free parameters.

Full restricted norm (here interior 4th order and boundary 3rd order shown).

$$H = \begin{pmatrix} x & & & & & & \\ & x & x & x & x & & \\ & x & x & x & x & & \\ & x & x & x & x & & \\ & x & x & x & x & & \\ & & & & & 1 & \\ & & & & & & \ddots \end{pmatrix} \qquad D = \begin{pmatrix} x & x & x & x & x & & & & \\ x & x & x & x & x & x & x & & \\ x & x & x & x & x & x & x & & \\ x & x & x & x & x & x & x & & \\ x & x & x & x & x & x & x & & \\ & & & x & x & 0 & x & x & \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix}$$

4-3 has three free parameters, 6-5 has four free parameter.

The penalty method

Consider the advection equation $u_t = \lambda u_x$ on two domains $(-\infty, 0]$ and $[0, \infty)$, that touches at $x = 0$.
Discretize the left and the right domains with grid-spacings h^l and h^r and finite difference operators D^l and D^r that satisfies SBP with respect to their scalar products σ^l and σ^r and add a penalty term on each side:

$$\begin{aligned} \dot{u}_i^l &= \lambda D^l u_i^l + \frac{\delta_{i,0} S^l}{h^l \sigma_{00}^l}(u_0^r - u_0^l) \\ \dot{u}_i^r &= \lambda D^r u_i^r + \frac{\delta_{i,0} S^r}{h^r \sigma_{00}^r}(u_0^l - u_0^r). \end{aligned}$$

Defining the energy to be

$$E = \langle u^l, u^l \rangle + \langle u^r, u^r \rangle$$

its time derivative becomes

$$\dot{E} = \lambda (u_0^l)^2 + 2S^l(u_0^r - u_0^l)u_0^l - \lambda (u_0^r)^2 + 2S^r(u_0^l - u_0^r)u_0^r.$$

If we choose $S^l - S^r = \lambda$, this becomes

$$\dot{E} = -\left(S^l + S^r\right)\left(u_0^l - u_0^r\right)^2$$

The penalty method

Summarizing: S^l and S^r has to be chosen so that $\lambda + S^r - S^l = 0$ and $S^l + S^r \geq 0$. The following cases can occur:

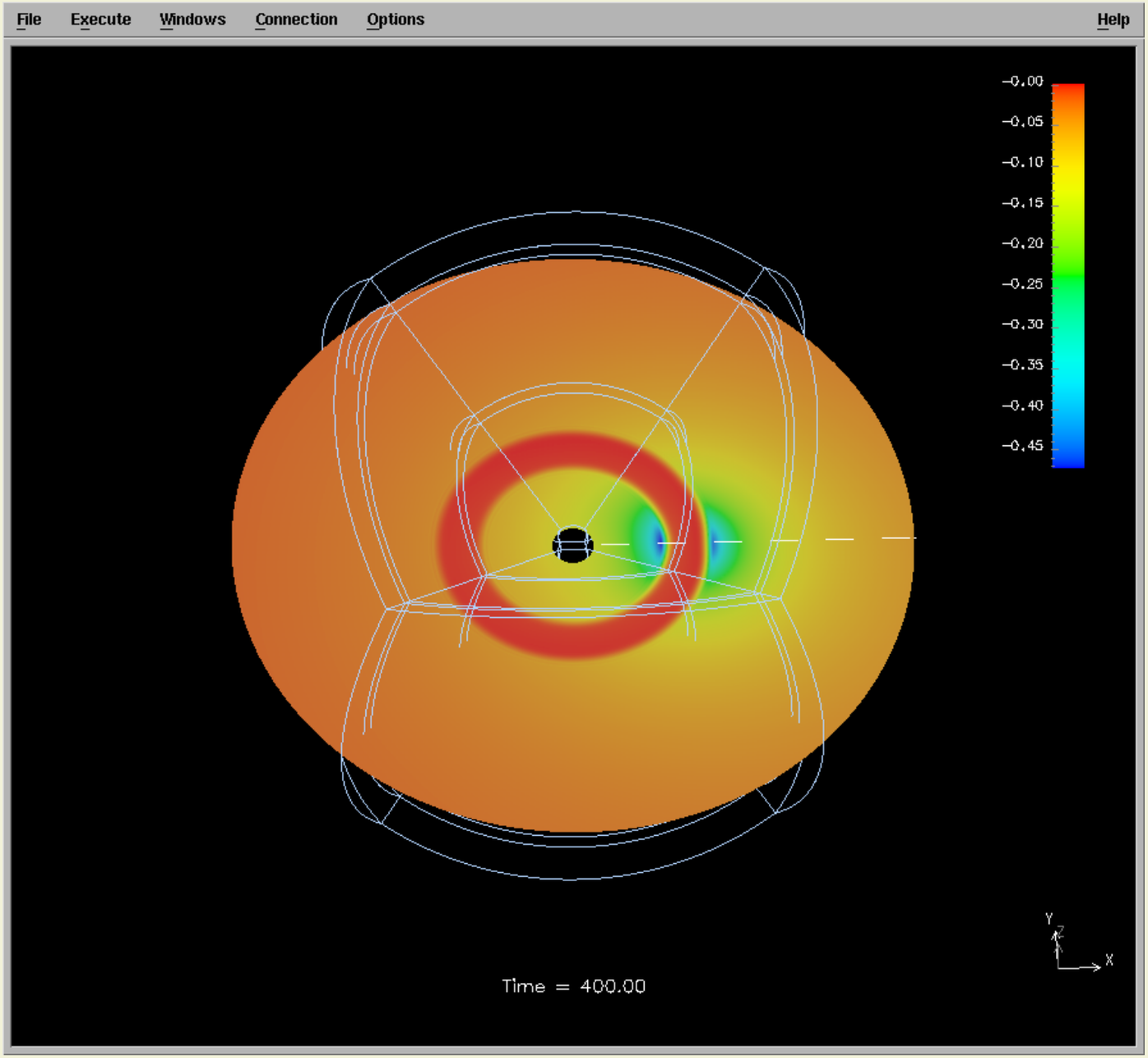
- $\lambda > 0$: $\quad S^l = \lambda + \delta, \quad S^r = \delta, \quad \delta \geq -\frac{\lambda}{2}.$
- $\lambda < 0$: $\quad S^r = -\lambda + \delta, \quad S^l = \delta, \quad \delta \geq \frac{\lambda}{2}.$
- $\lambda = 0$: A limiting case of either of the above cases.

For linear hyperbolic systems in first order form (both space and time) the penalty method is applied to the characteristic variables with the characteristic speeds taking the place of λ.

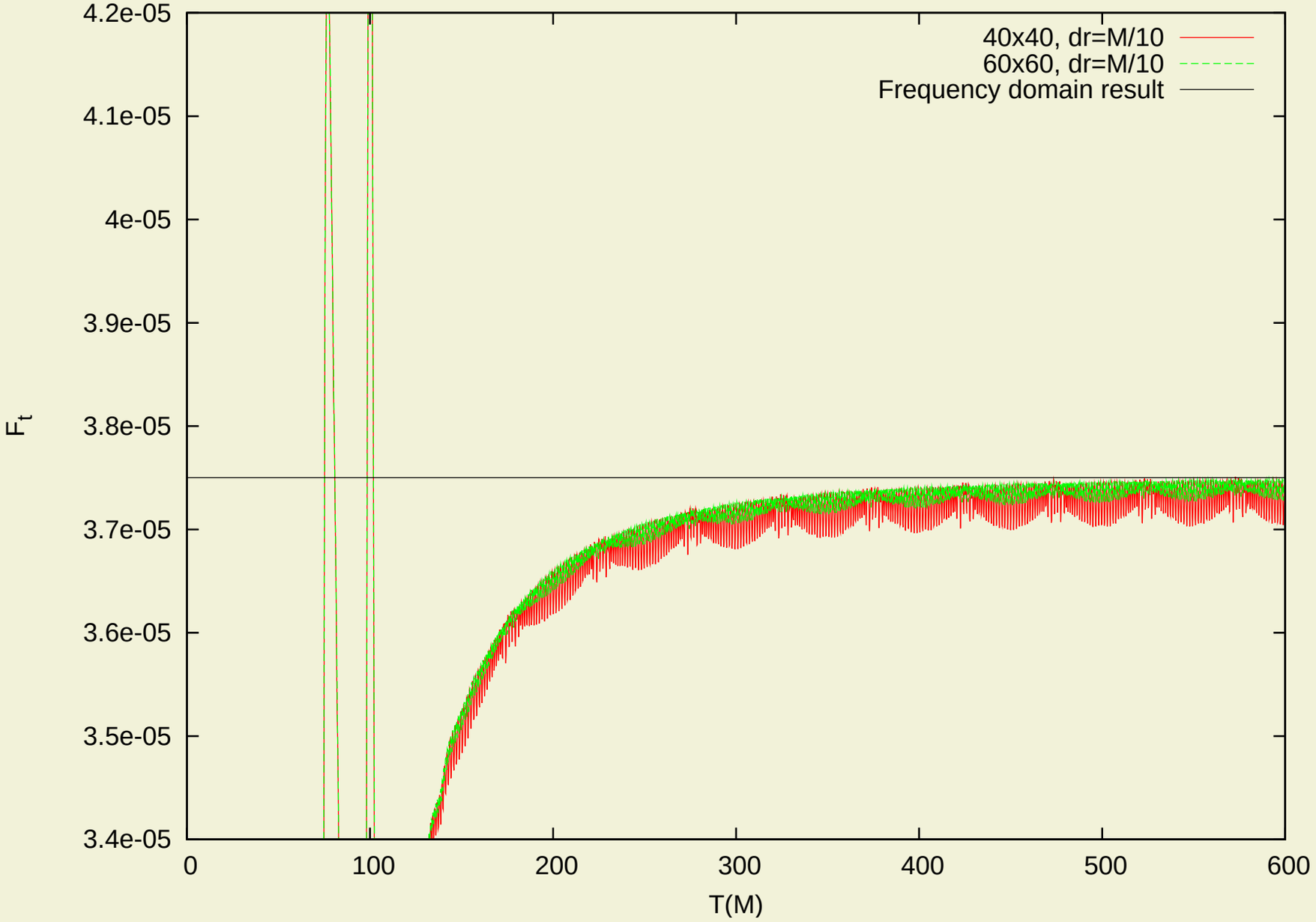
For nonlinear systems, it is generally not possible to obtain an energy estimate. It is usually necessary to stabilize the numerical method by adding artificial dissipation. The artificial dissipation operators need to be compatible with the SBP derivative operators.

Even for linear systems in 3D and/or with non-constants coefficients stability is not guaranteed for restricted full norm operators. Artificial dissipation may stabilize the system.

Self-force results

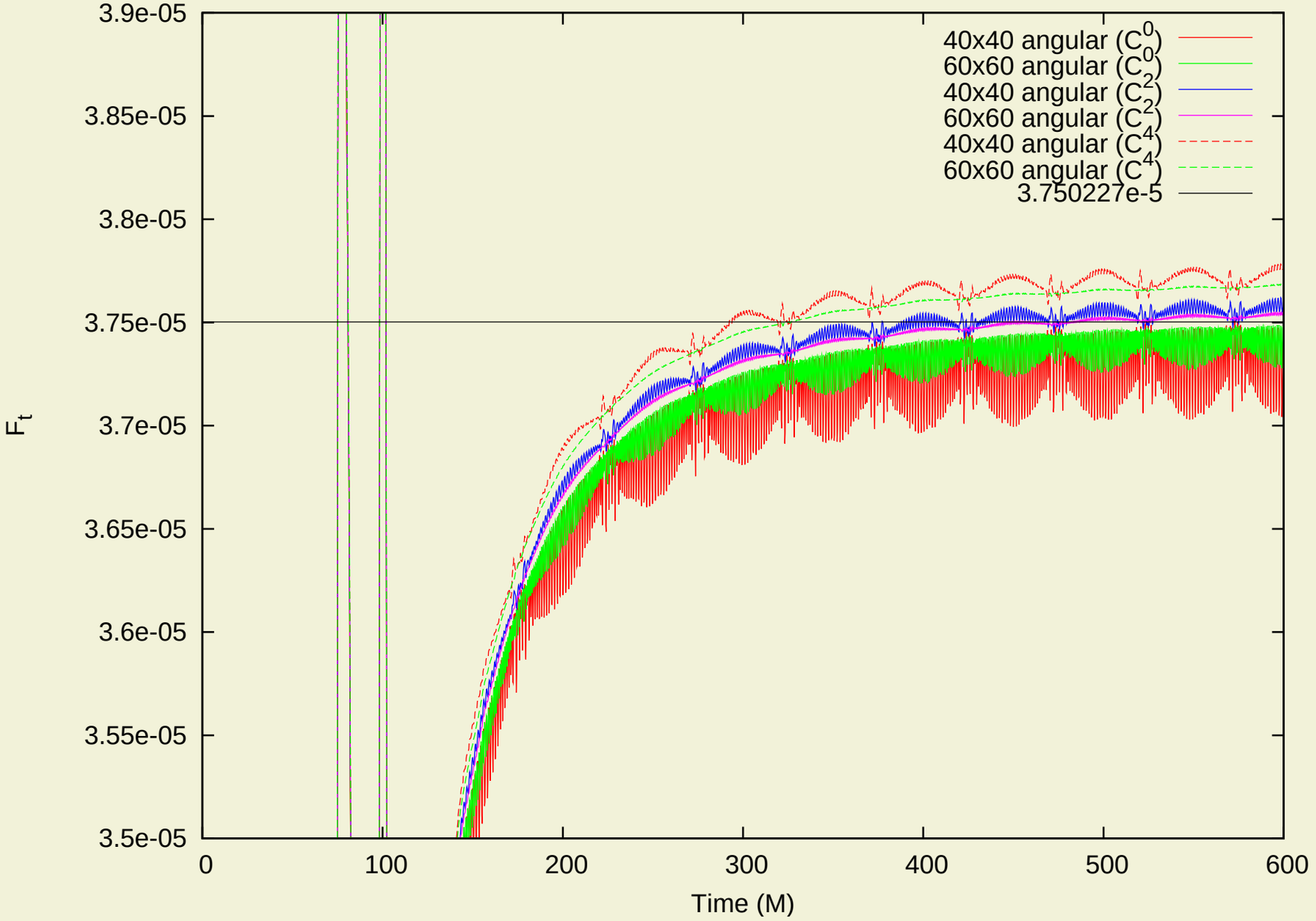


Self-force results

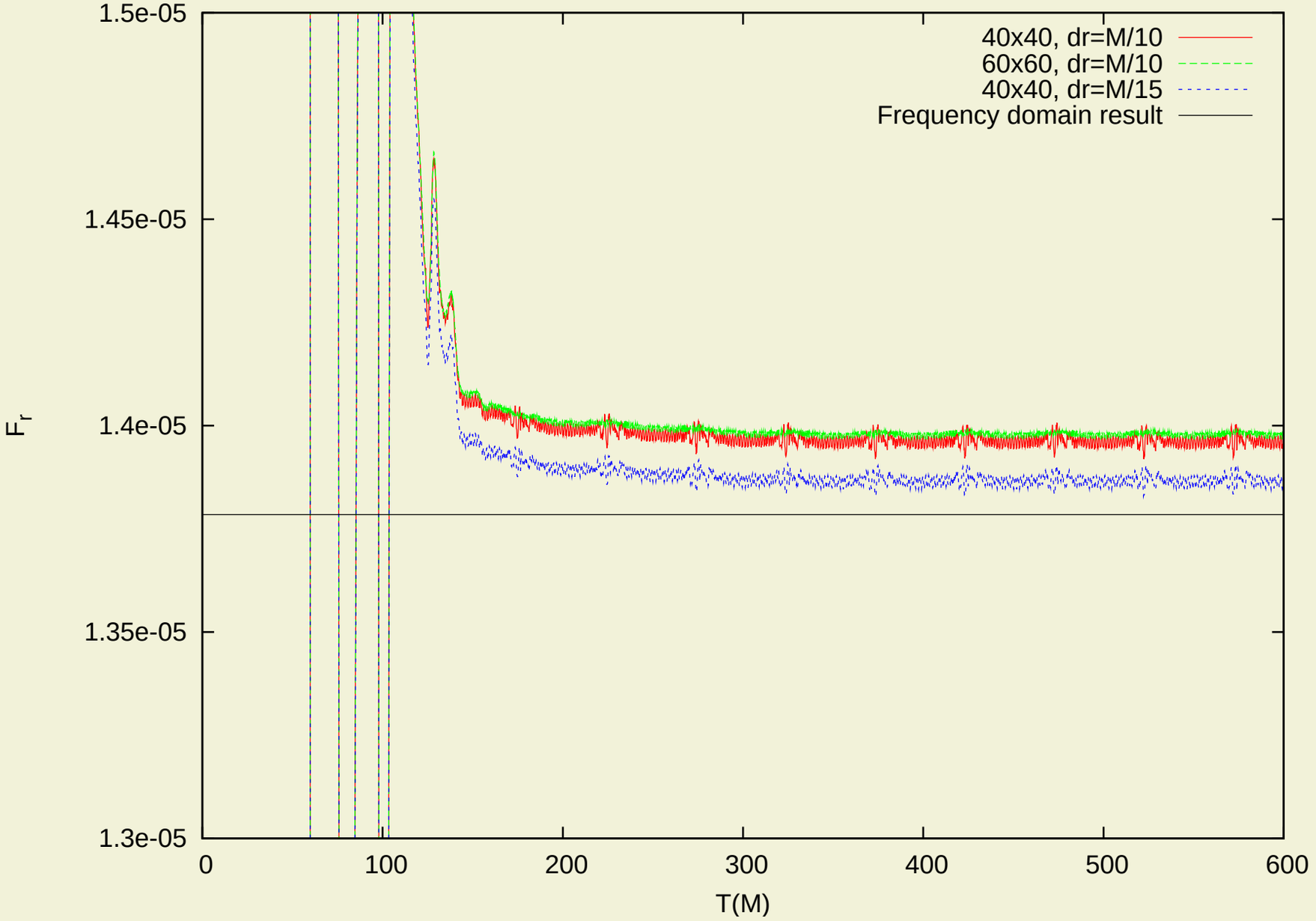


The noise is at the 1% level.

Self-force results

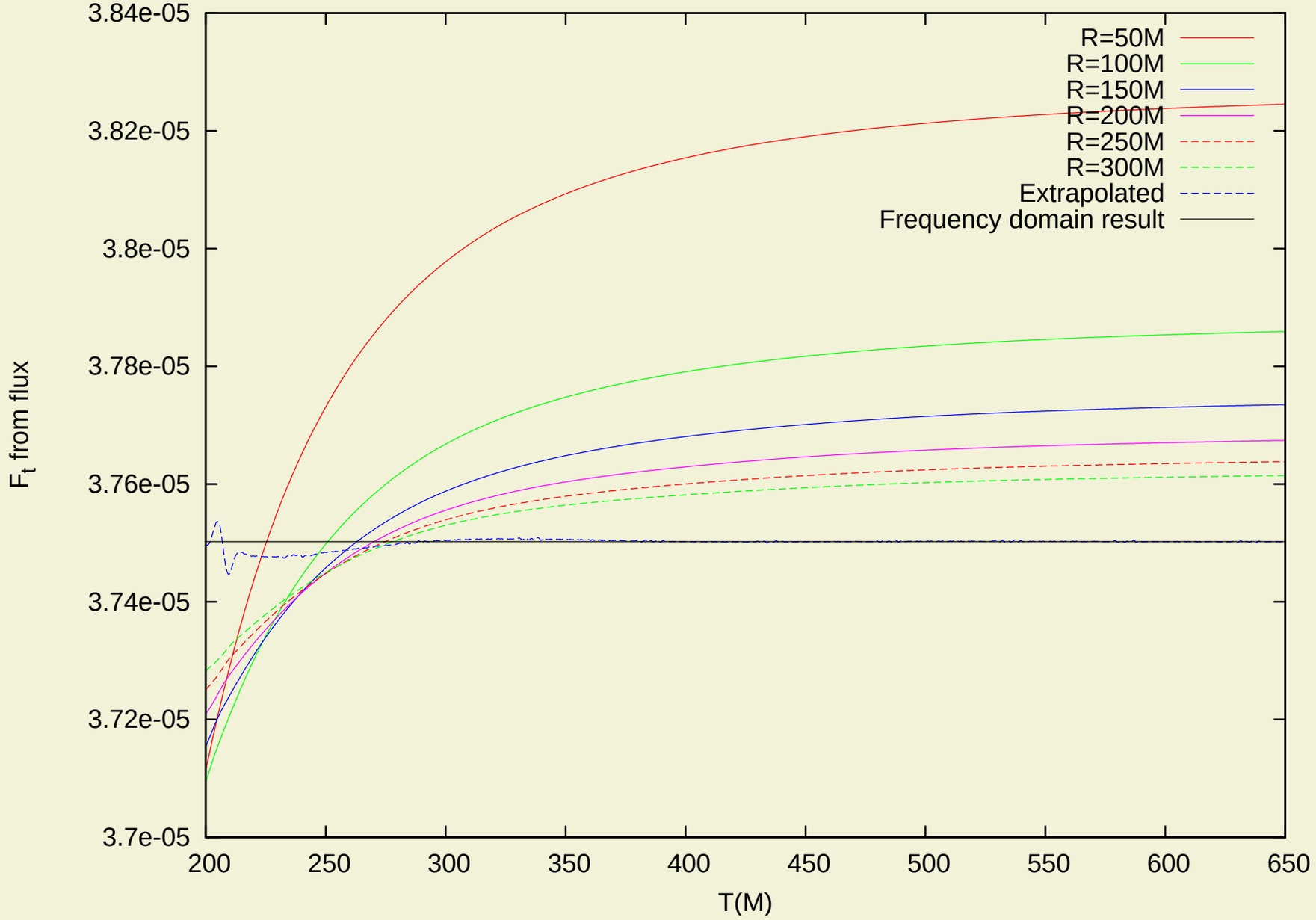


Self-force results

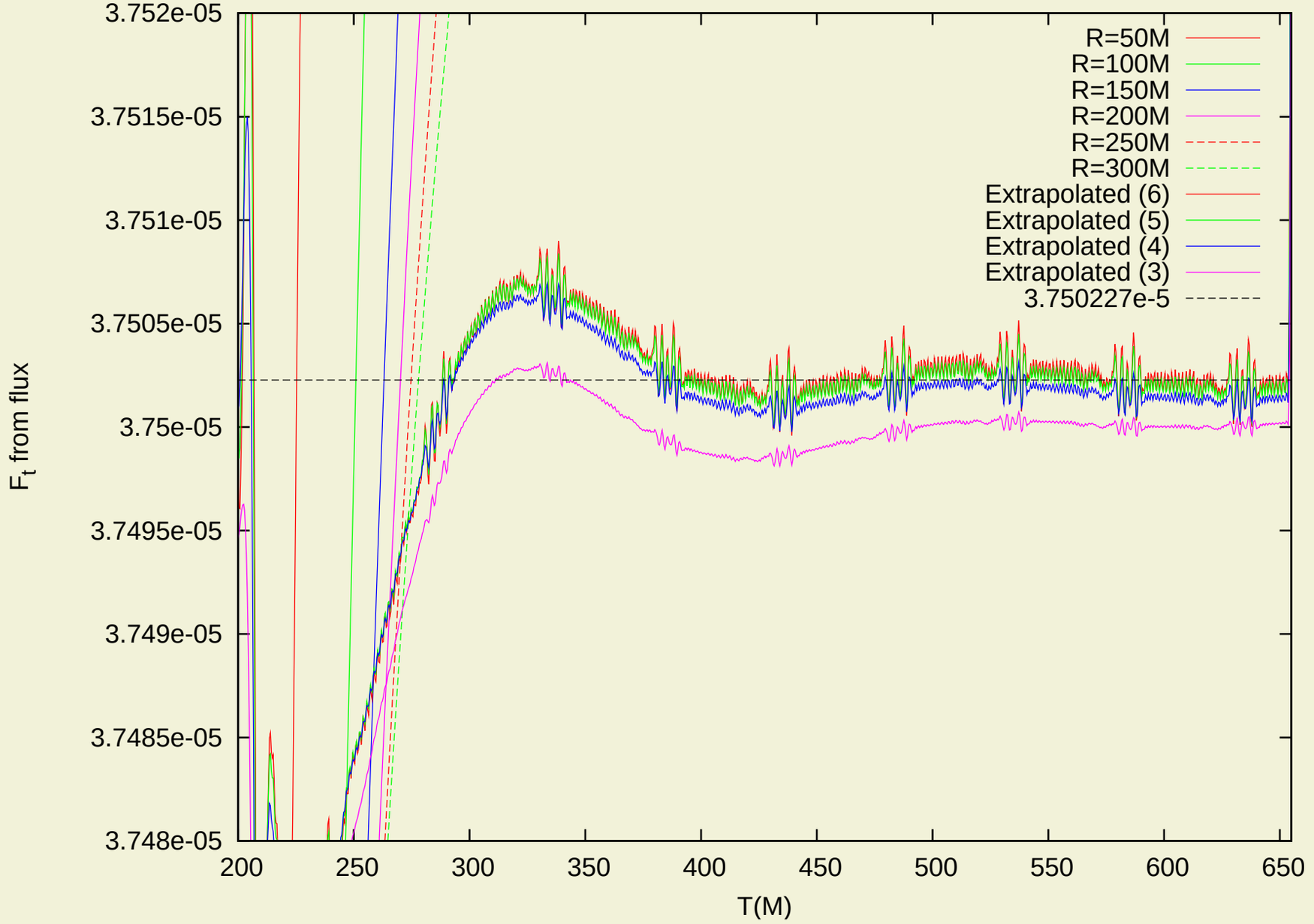


The error in the high resolution result is about 1%.

Self-force results



Self-force results



The extrapolated flux varies within $\mathbf{0.007\%}$ of the exact value.

Concluding remarks

- The choice of window function does not affect the local determination of the self-force. It only affects the initial transient.
- The local determination of the self-force is currently limited to 2nd order convergence due to the C^0 nature of the source.
- The flux calculation is very accurate (after extrapolation to infinity) considering this is a 3+1 code.
- I'm hopeful that by changing to a new multi-block infrastructure and with a more clever finite differencing scheme near the particle the noise can be decreased and the accuracy increased dramatically.
- Better boundary conditions or non-uniform radial resolution should result in decreased computational resource requirements.
- Once the noise in the self-force is reduced, it should be straightforward to correct the orbit every time-step.