

Subcycling in CarpetX

Liwei Ji

Collaborator: Manuela Campanelli, Roland Haas, Yosef Zlochower

North American ETK School 2024



RIT

College of Science

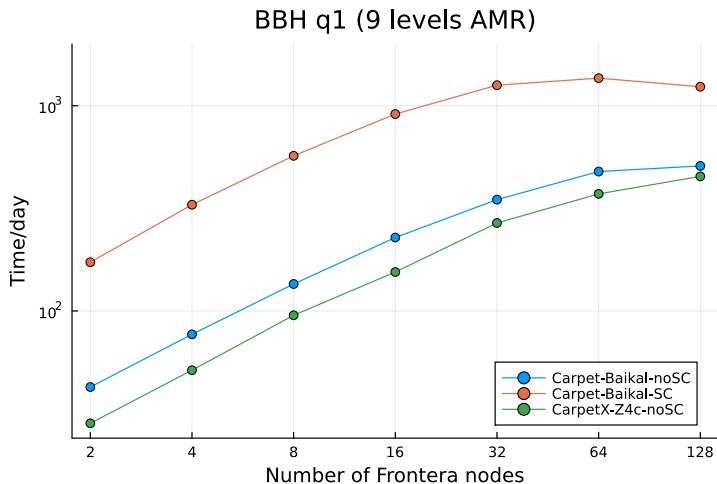
Center for Computational Relativity and Gravitation

CARPETX: a new driver for CACTUS

- CARPET $\Rightarrow$ CARPETX (based on AMReX)
 - SPACETIMEX <https://github.com/EinsteinToolkit/SpacetimeX>
 - ASTERX <https://github.com/jaykalinani/AsterX>
- Support for cell-centered, face-centered, and nodal data
- Adaptive mesh refinement
- Performance portability: parallelization via MPI, OpenMP, hybrid MPI/OpenMP, hybrid MPI/(**CUDA** or **HIP/ROCm** or **SYCL**)
- Efficient I/O (openPMD, Silo) or ASCII
- Correctness checks (prevent undefined grid points)
- Public available at:
<https://github.com/EinsteinToolkit/CarpetX.git>



Motivation



Runge-Kutta Method (RK4)

- To integrate the ODE

$$\frac{dy}{dt} = f(t, y), \quad y(t_0) = y_0,$$

we can pick a step-size $h > 0$

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$k_1 = f(t_n, y_n) \qquad \qquad \qquad =: f(t_n, Y_1),$$

$$k_2 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_1}{2}\right) \qquad \qquad \qquad =: f\left(t_n + \frac{h}{2}, Y_2\right),$$

$$k_3 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_2}{2}\right) \qquad \qquad \qquad =: f\left(t_n + \frac{h}{2}, Y_3\right),$$

$$k_4 = f(t_n + h, y_n + hk_3) \qquad \qquad \qquad =: f(t_n + h, Y_4).$$

Subcycling in Carpet



$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$



$$t = t_0$$

Subcycling in Carpet



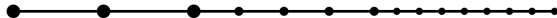
$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$



$$t = t_0$$

Subcycling in Carpet



$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$



$$t = t_0$$

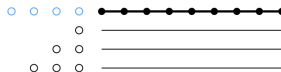
Subcycling in Carpet



$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$



$$t = t_0$$

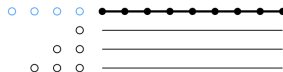
Subcycling in Carpet



$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$



$$t = t_0$$

Runge-Kutta Method (RK4)

- RK4

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$k_1 = f(t_n, y_n)$$

$$=: f(t_n, Y_1),$$

$$k_2 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_1}{2}\right)$$

$$=: f\left(t_n + \frac{h}{2}, Y_2\right),$$

$$k_3 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_2}{2}\right)$$

$$=: f\left(t_n + \frac{h}{2}, Y_3\right),$$

$$k_4 = f(t_n + h, y_n + hk_3)$$

$$=: f(t_n + h, Y_4).$$

Runge-Kutta Method (RK4)

- RK4

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

$$k_1 = f(t_n, y_n) \qquad \qquad \qquad =: f(t_n, Y_1),$$

$$k_2 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_1}{2}\right) \qquad \qquad \qquad =: f\left(t_n + \frac{h}{2}, Y_2\right),$$

$$k_3 = f\left(t_n + \frac{h}{2}, y_n + h\frac{k_2}{2}\right) \qquad \qquad \qquad =: f\left(t_n + \frac{h}{2}, Y_3\right),$$

$$k_4 = f(t_n + h, y_n + hk_3) \qquad \qquad \qquad =: f(t_n + h, Y_4).$$

- The Taylor expansion of k_i around t_n are (MONGWANE+2015, McCORQUODALE+2011)

$$k_1 = y',$$

$$k_2 = y' + \frac{h}{2}y'' + \frac{h^2}{8}(y''' - f_y y''),$$

$$k_3 = y' + \frac{h}{2}y'' + \frac{h^2}{8}(y''' + f_y y'').$$

Runge-Kutta Method (RK4)

- The Taylor expansion of Y_i around t_n are

$$Y_1 = y_n,$$

$$Y_2 = y_n + \frac{h}{2}y',$$

$$Y_3 = y_n + \frac{h}{2}y' + \frac{h^2}{4}y'' + \frac{h^3}{16}(y''' - f_y y''),$$

$$Y_4 = y_n + hy' + \frac{h^2}{2}y'' + \frac{h^3}{16}(y''' + f_y y'').$$

where $f_y y'' \equiv (y'')^2 / y' = \frac{4(k_3^{(c)} - k_2^{(c)})}{h_{(c)}^2}$, y' , y'' and y''' can be represented with $k_i^{(c)}$ of coarse grid using the dense output formula.

Dense Output

- Consider formulas of the form for RK4 method (ALEXANDER+1990)

$$u(\theta) = y_n + h \sum_{i=1}^4 b_i(\theta) k_i,$$

where $b_i(\theta)$ are polynomials to be determined such that $u(\theta) - y(t_n + \theta h) = \mathcal{O}(h^4)$. It produce a unique solution

$$b_1(\theta) = \theta - \frac{3}{2}\theta^2 + \frac{2}{3}\theta^3,$$

$$b_2(\theta) = b_3(\theta) = \theta^2 - \frac{2}{3}\theta^3,$$

$$b_4(\theta) = -\frac{1}{2}\theta^2 + \frac{2}{3}\theta^3.$$

Dense Output

- Consider formulas of the form for RK4 method (ALEXANDER+1990)

$$u(\theta) = y_n + h \sum_{i=1}^4 b_i(\theta) k_i,$$

where $b_i(\theta)$ are polynomials to be determined such that $u(\theta) - y(t_n + \theta h) = \mathcal{O}(h^4)$. It produce a unique solution

$$b_1(\theta) = \theta - \frac{3}{2}\theta^2 + \frac{2}{3}\theta^3,$$

$$b_2(\theta) = b_3(\theta) = \theta^2 - \frac{2}{3}\theta^3,$$

$$b_4(\theta) = -\frac{1}{2}\theta^2 + \frac{2}{3}\theta^3.$$

- The dense output formula can be summarized as

$$y(t_n + \theta h) = y_n + h \sum_{i=1}^4 b_i(\theta) k_i + \mathcal{O}(h^4),$$

$$\frac{d^{(m)}}{dt^{(m)}} y(t_n + \theta h) = \frac{1}{h^{(m-1)}} \sum_{i=1}^s k_i \frac{d^{(m)}}{d\theta^{(m)}} b_i(\theta) + \mathcal{O}(h^{4-m}).$$

Runge-Kutta Method (RK4)

- The Taylor expansion of Y_i around t_n are

$$Y_1 = y_n,$$

$$Y_2 = y_n + \frac{h}{2}y',$$

$$Y_3 = y_n + \frac{h}{2}y' + \frac{h^2}{4}y'' + \frac{h^3}{16}(y''' - f_y y''),$$

$$Y_4 = y_n + hy' + \frac{h^2}{2}y'' + \frac{h^3}{16}(y''' + f_y y'').$$

where y' , y'' and y''' can be represented with $k_i^{(c)}$ of coarse grid using the dense output formula.

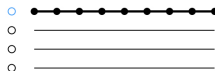
Subcycling in CarpetX



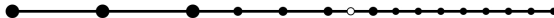
$$t = t_0 + \Delta t$$



$$t = t_0 + \Delta t/2$$



$$t = t_0 + \Delta t/4$$

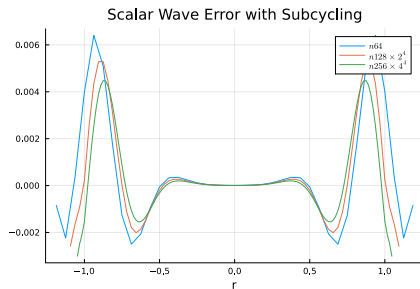
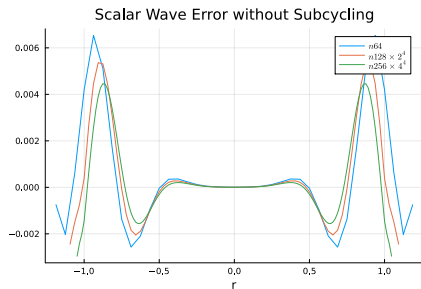


$$t = t_0$$

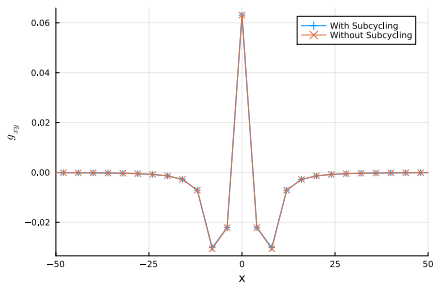
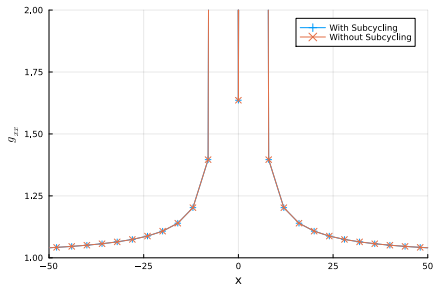
Subcycling in CarpetX

- Better scaling performance
- 4-th order convergence

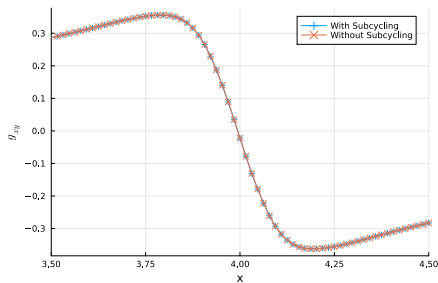
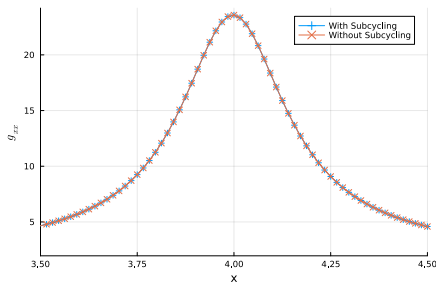
Scalar Wave



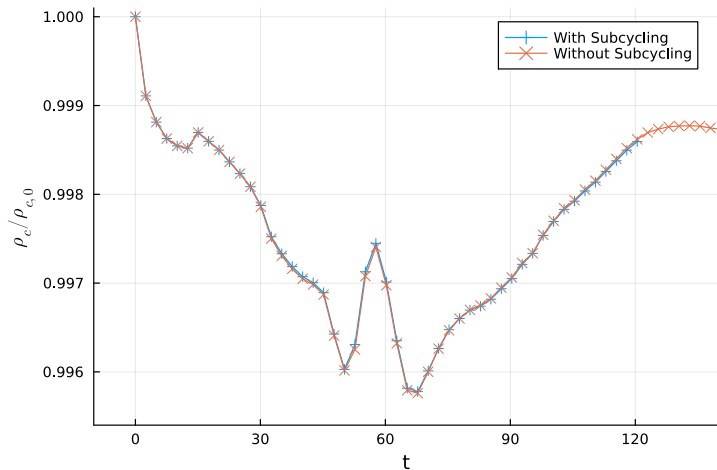
Binary Black Hole (q1)



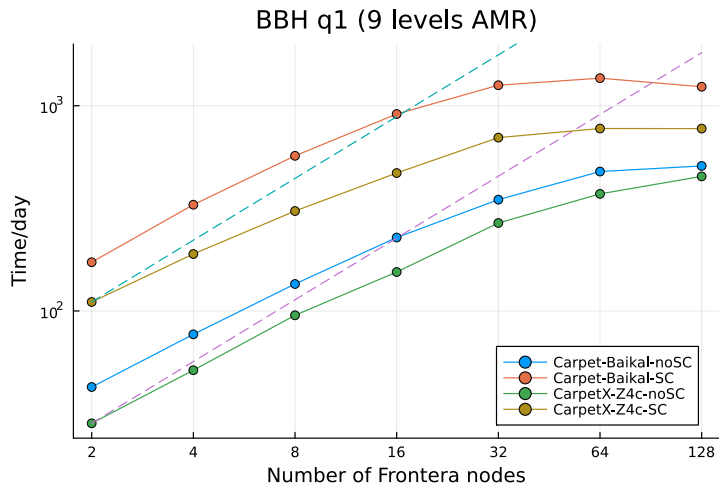
Binary Black Hole (q1)



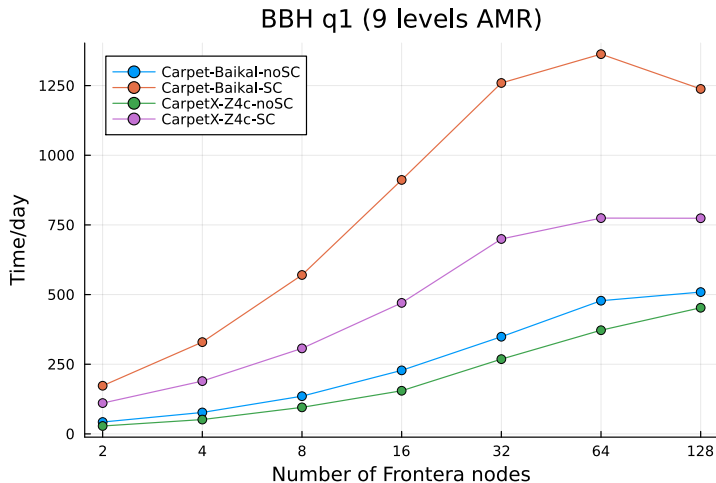
Magnetised TOV Star



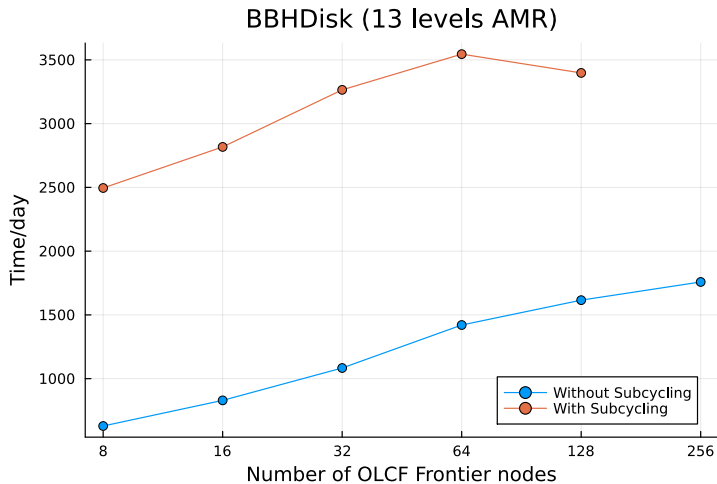
Scaling on Frontera (CPU)



Scaling on Frontera (CPU)



Scaling on Frontier (GPU)



Next Step

- Optimize the implementation
 - special data structure for the mesh refinement boundary points
 - special syncs
- Refluxing

Thanks for your attention!

