

Practical Tutorial

Why is Simulating Jets Hard?

- Typical AGN jet has $\gamma \sim 10$
- How do you get something to move that fast?
- You need energy!
$$\frac{B^2}{8\pi} = \epsilon = \gamma \rho c^2$$
- Set $\frac{B^2}{8\pi} = 10$, then $\rho c^2 = 1$
- 10% error in B^2 means $\Delta\epsilon = 1$, or 100% error in ρc^2 !
- This is a *stiff* problem
- We need to minimize errors. How?

Equations of Motion in Conservative Form

Non-relativistic:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v_x)}{\partial x} = S$$

Gen. relativistic
(GR):

$$\frac{\partial(\sqrt{-g}\rho\gamma)}{\partial t} + \frac{\partial(\sqrt{-g}\rho\gamma v^x)}{\partial x} = S$$

(g is the determinant of the metric)

Conservation
law:

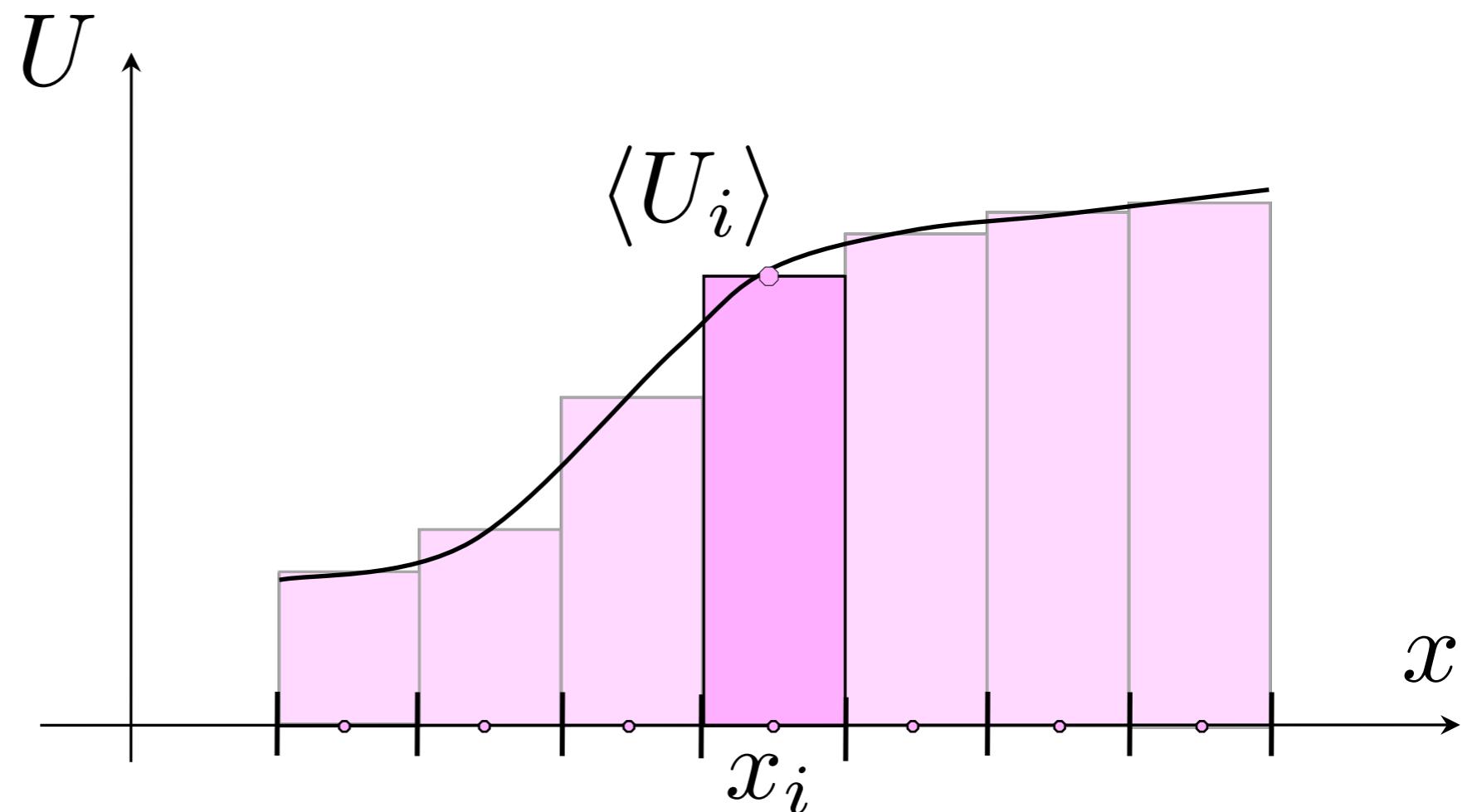
$$\frac{\partial U}{\partial t} + \frac{\partial F^x}{\partial x} = S$$

The rest of equations of motion reduce to this form of conservation law as well.

How Do We Put Equations on the Grid?

Conservation law:

$$\frac{\partial U}{\partial t} + \frac{\partial F^x}{\partial x} = S$$



How Do We Put Equations on the Grid?

Conservation law:

$$\frac{\partial U}{\partial t} + \frac{\partial F^x}{\partial x} = S$$

Integrate over the volume of a grid cell:

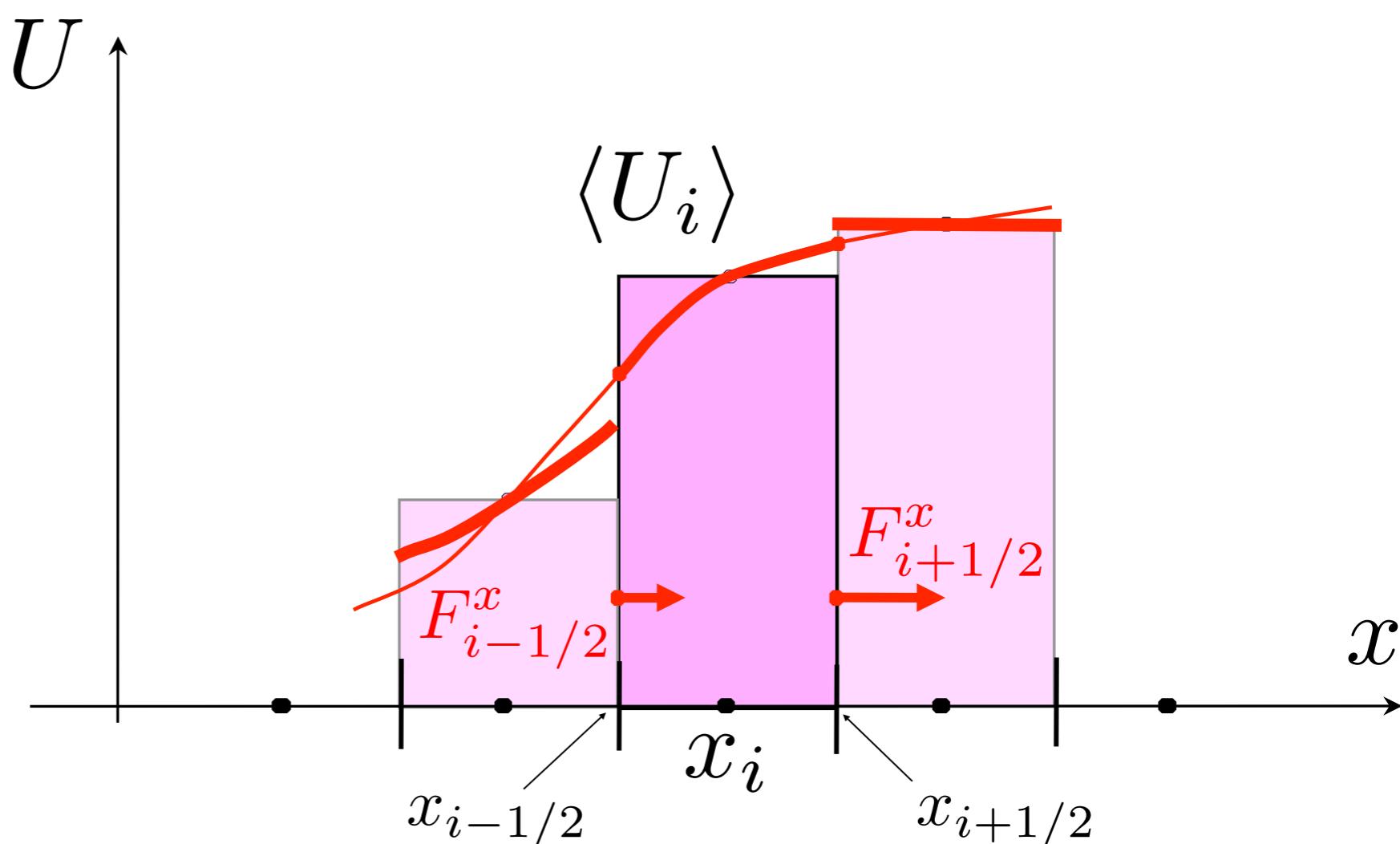
$$\frac{\partial \langle U_i \rangle}{\partial t} + \frac{F_{i+1/2}^x - F_{i-1/2}^x}{\Delta x} = \langle S_i \rangle$$

Fit a high-order non-oscillatory polynomial

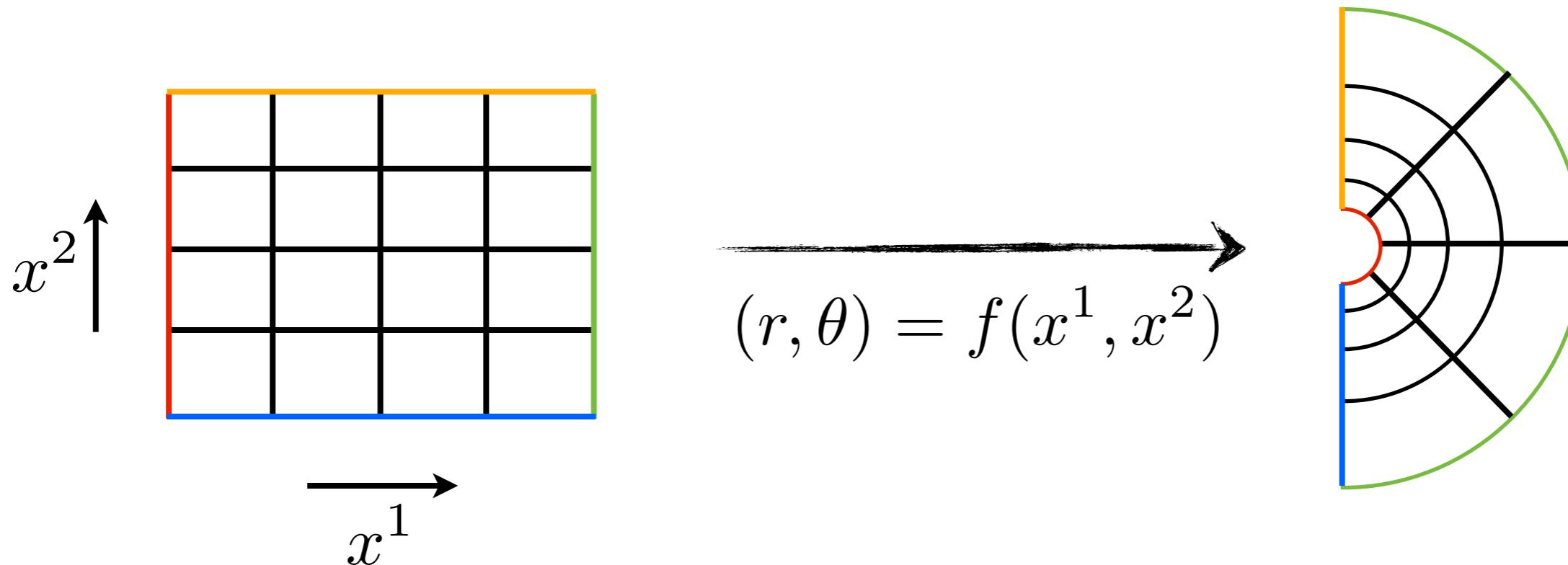
(e.g., high order schemes possible in GR, e.g., AT+2007)

Compute fluxes

Compute $\Delta \langle U_i \rangle$



Flexible Grid: Concentrate Resolution where Needed

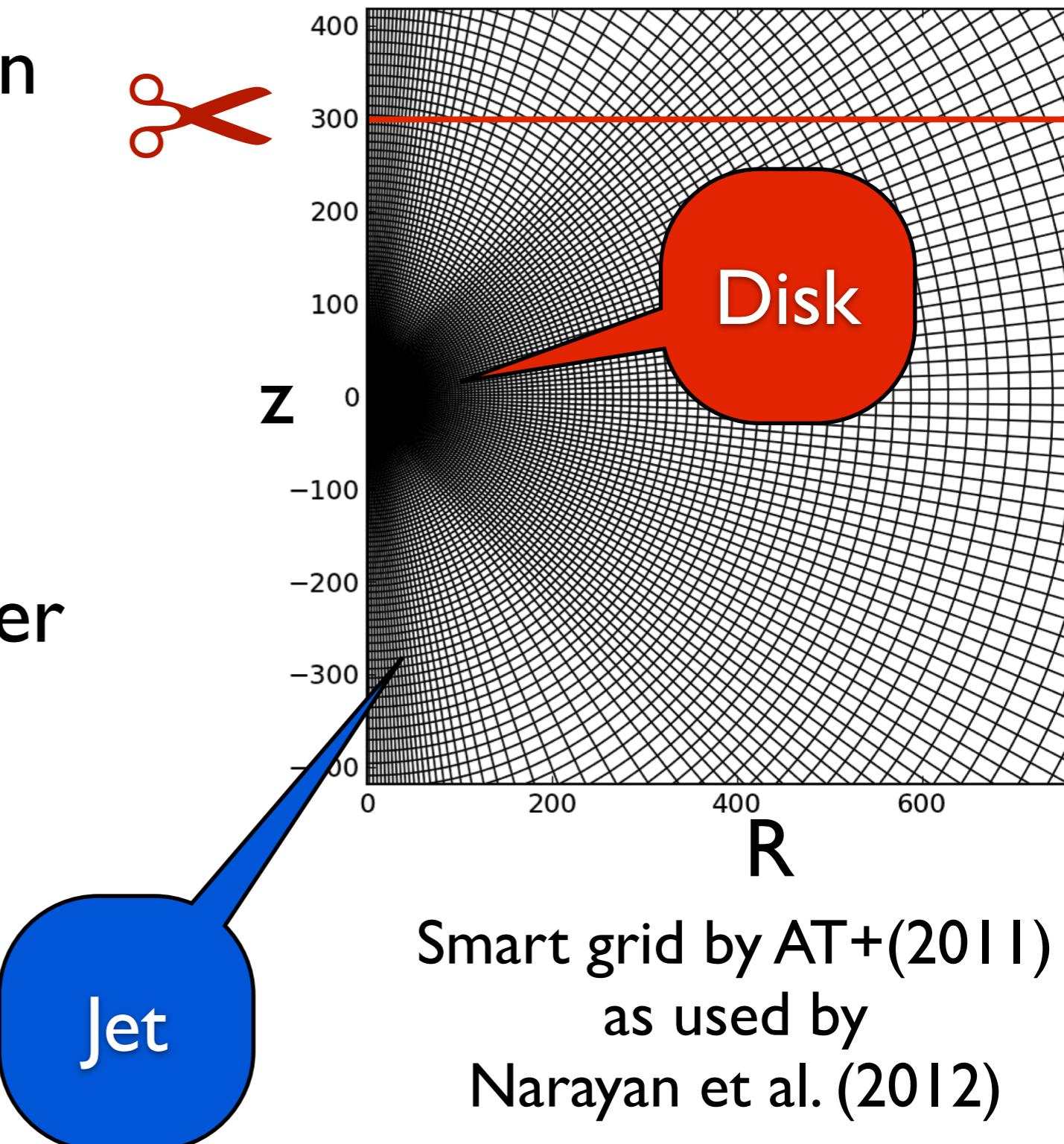


We can concentrate resolution in regions of interest by choosing an appropriate mapping f .

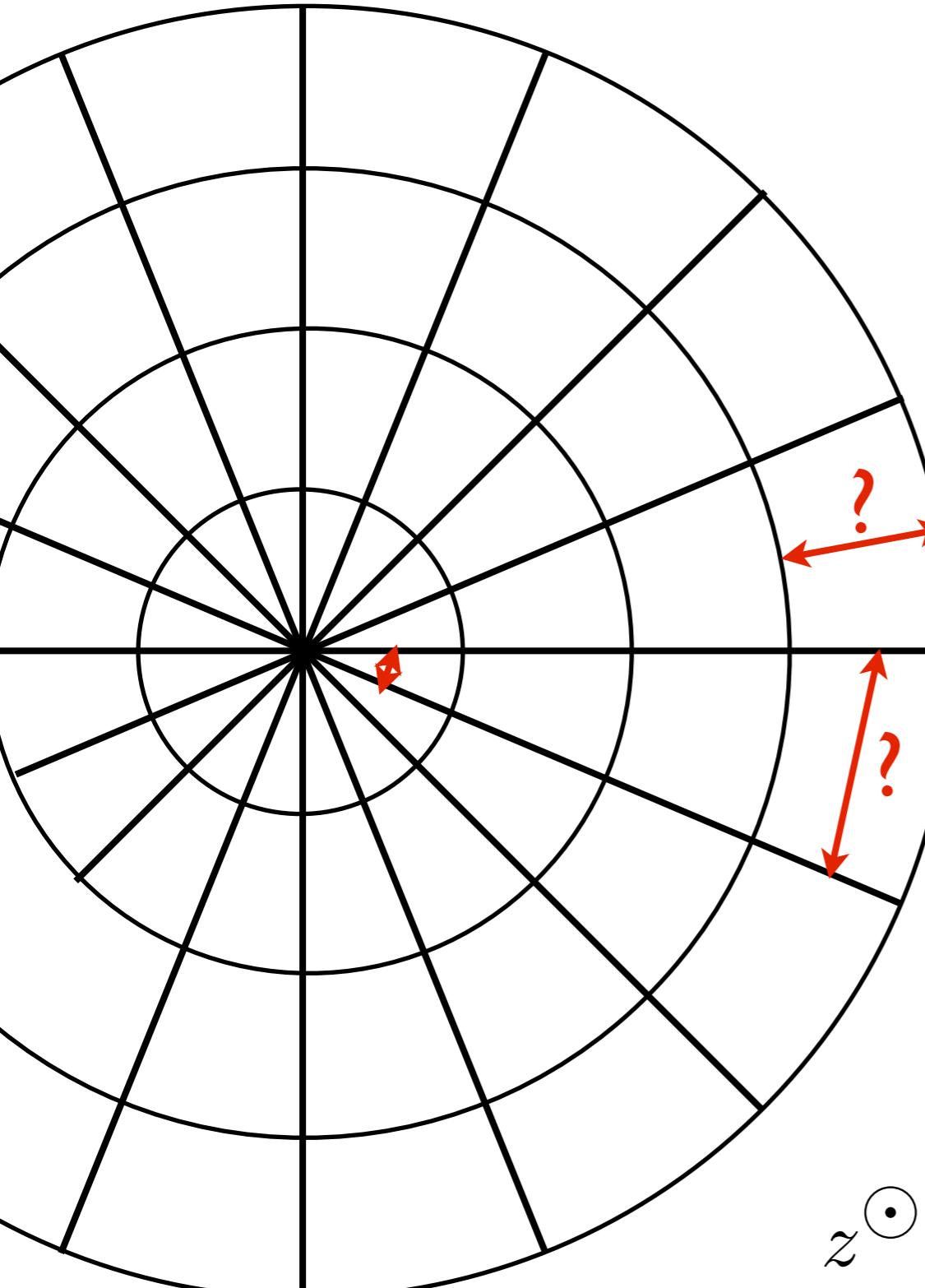
Literally flexible: grid can be curved or non-uniform, to conform to the shape of the boundary or geometry of the problem.

Smart Grid that Focuses Resolution on Disk and Jet

- Concentrates resolution where needed: **jet** and **disk**
(AT et al. 2011)
- Follows collimating jet
- Allows the use of smaller resolutions
- *Question:* isn't the time step too small in 3D?



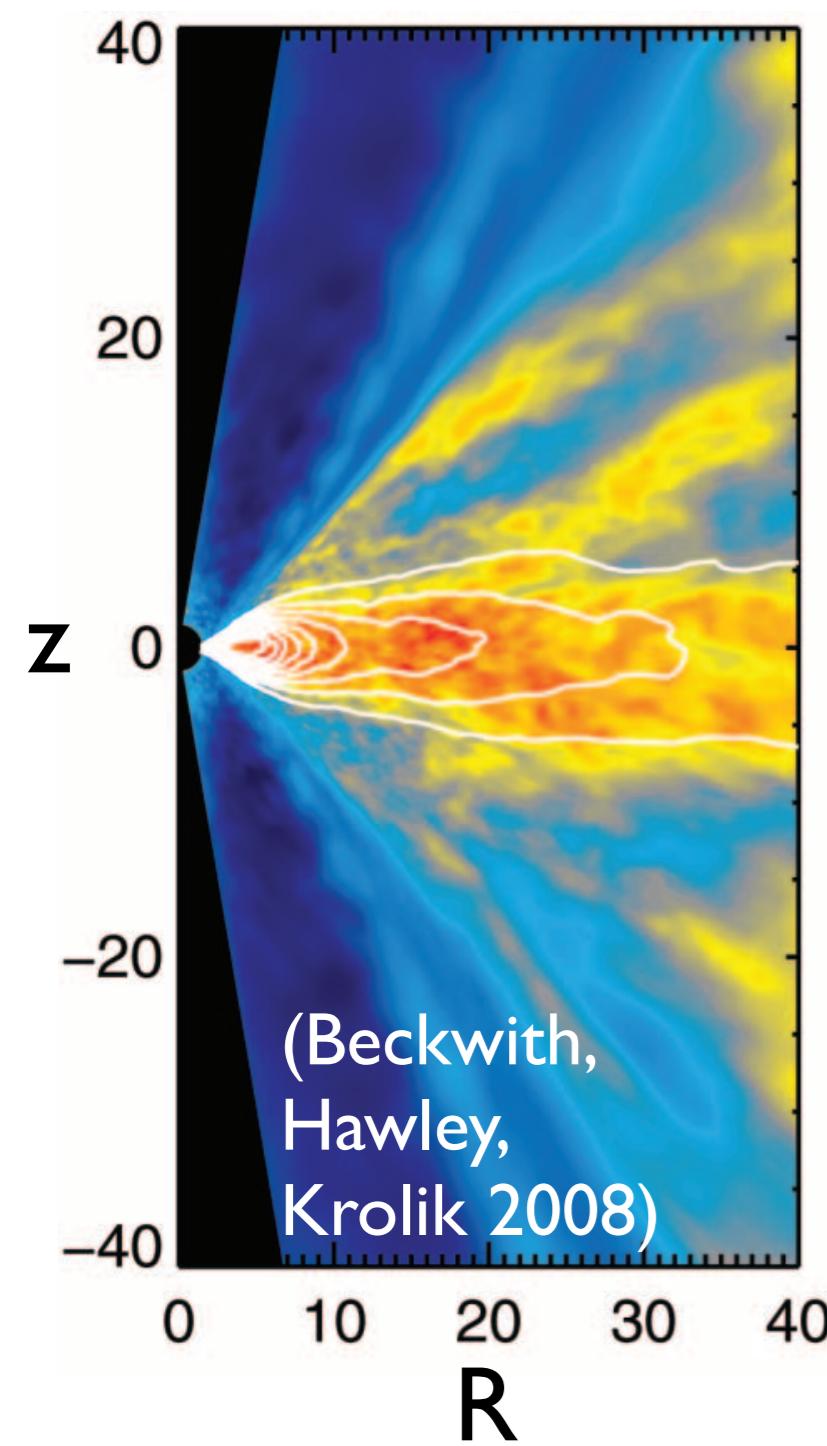
Advanced Approach to Polar Axis



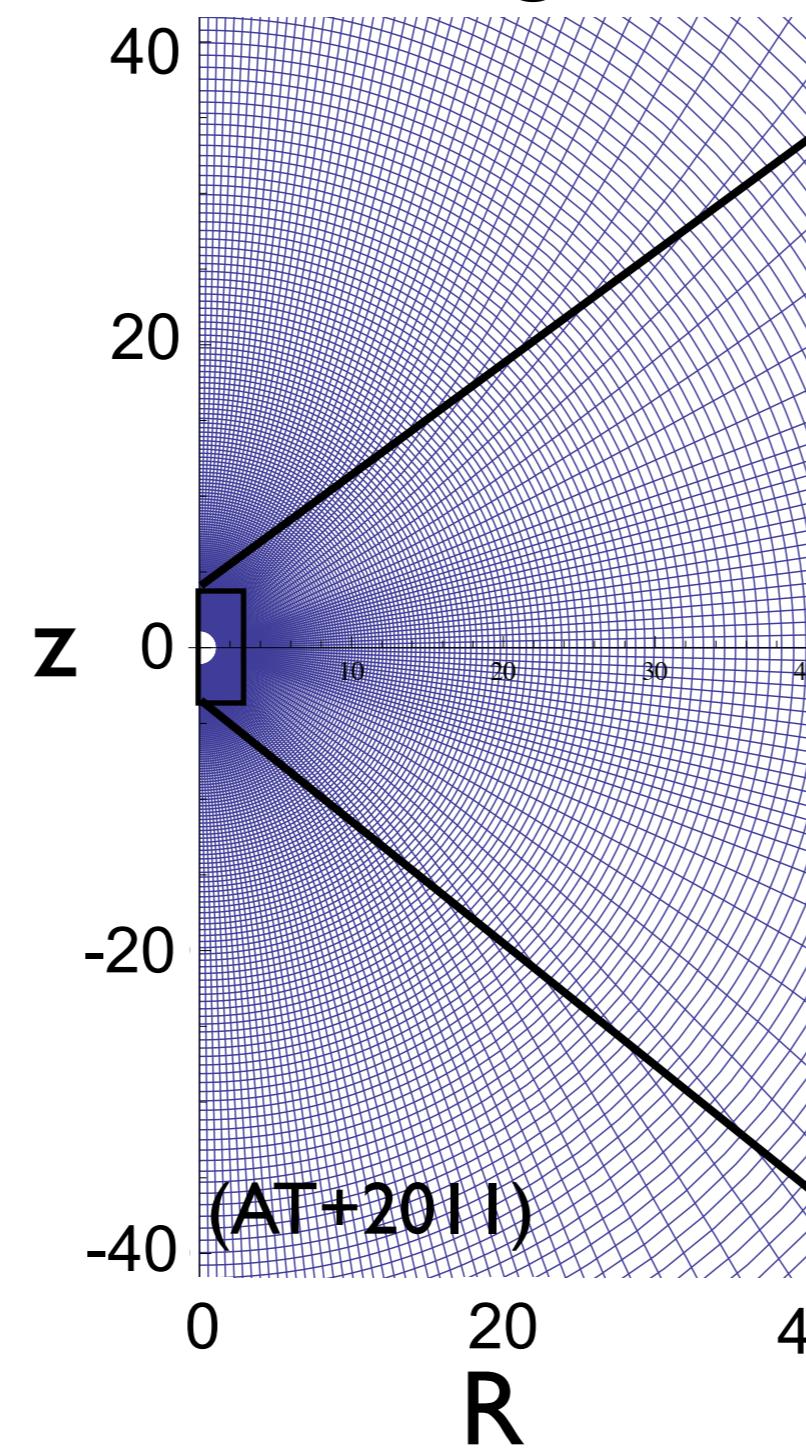
- Time step is the *smallest* light crossing time among all cells
- Small cell azimuthal extent can slow down a run by 10x
- This issue is of great importance for 3D performance

Advanced Approach to Polar Axis

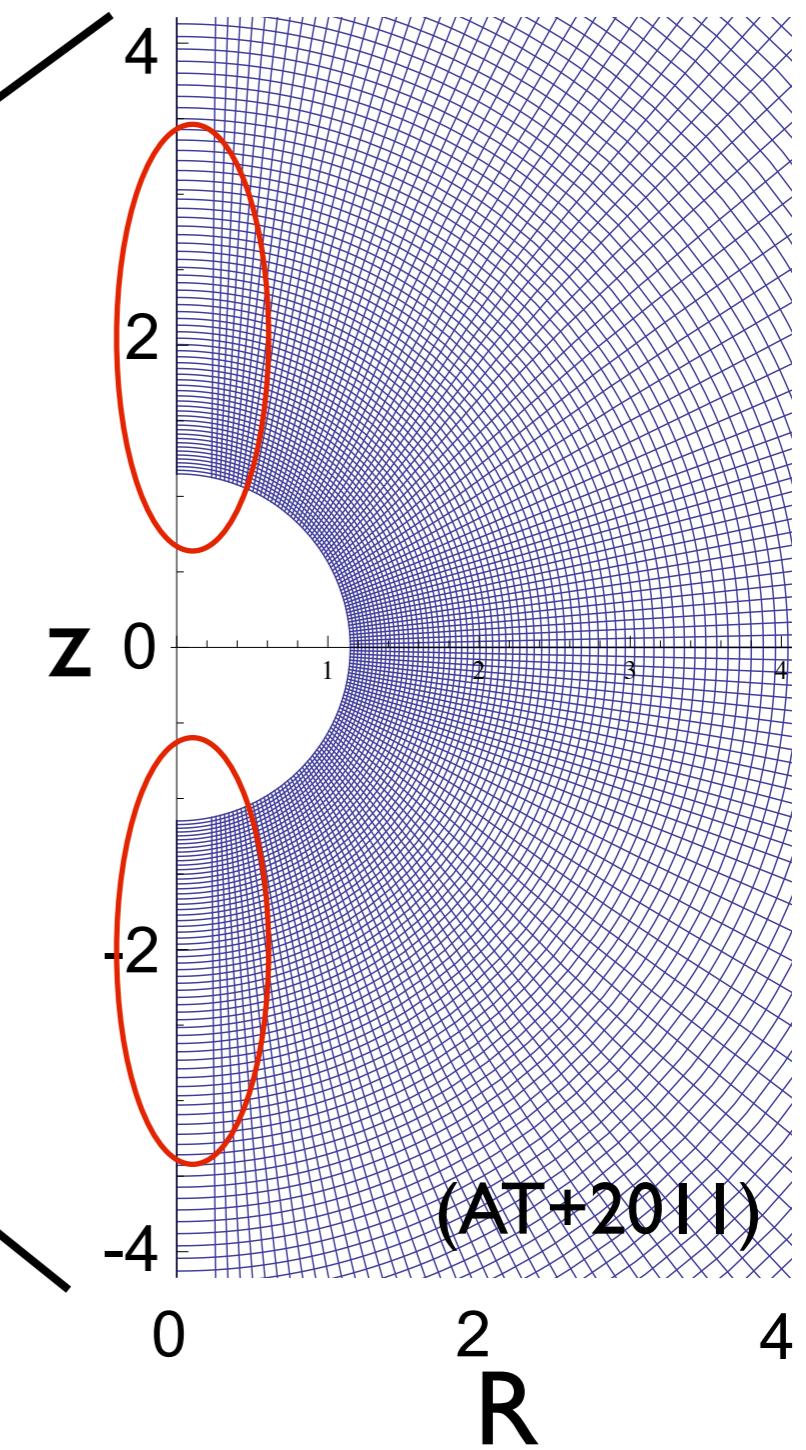
I) Remove offending cells



2) Deform offending cells

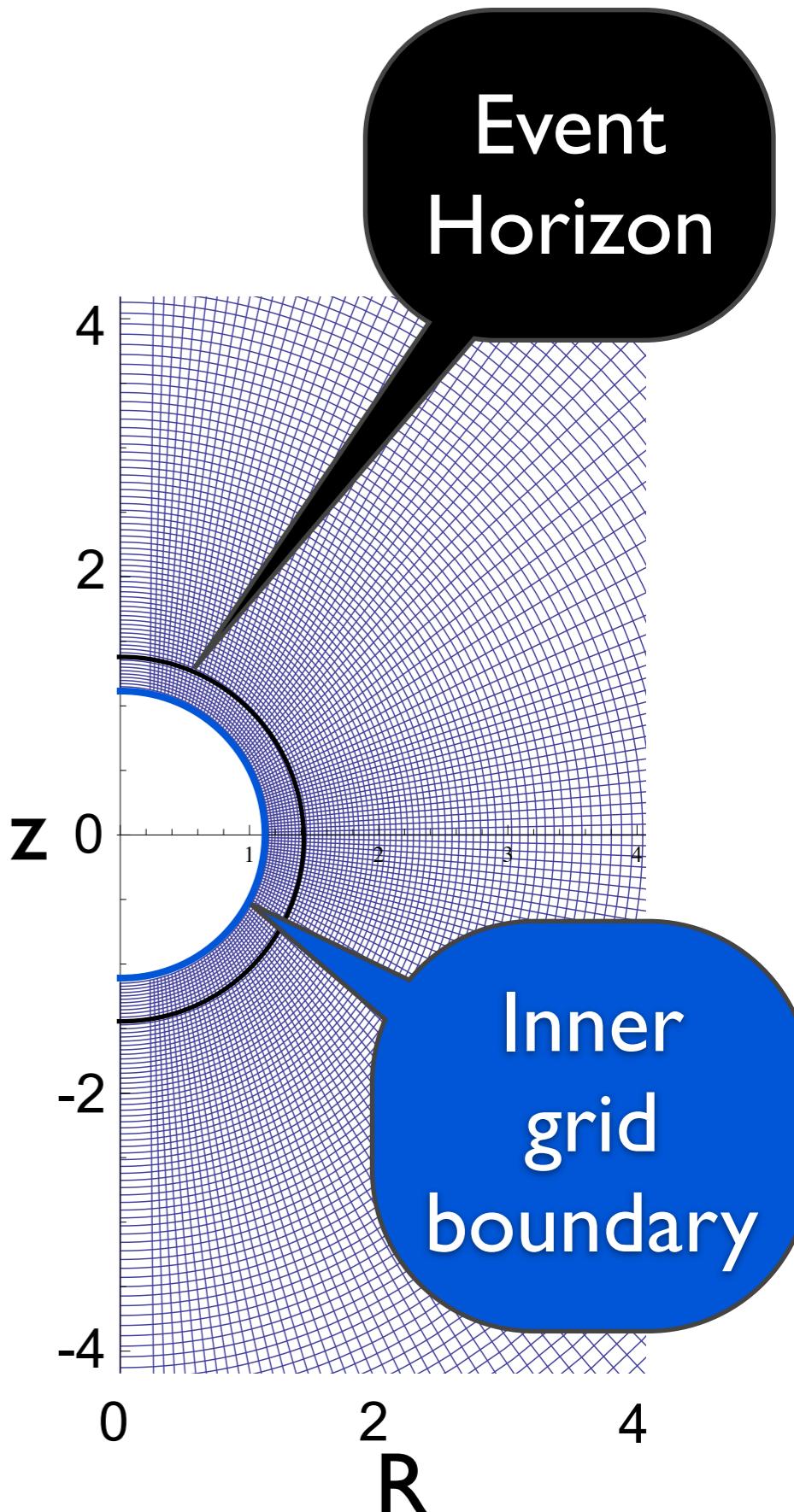


Speedup of
10x!



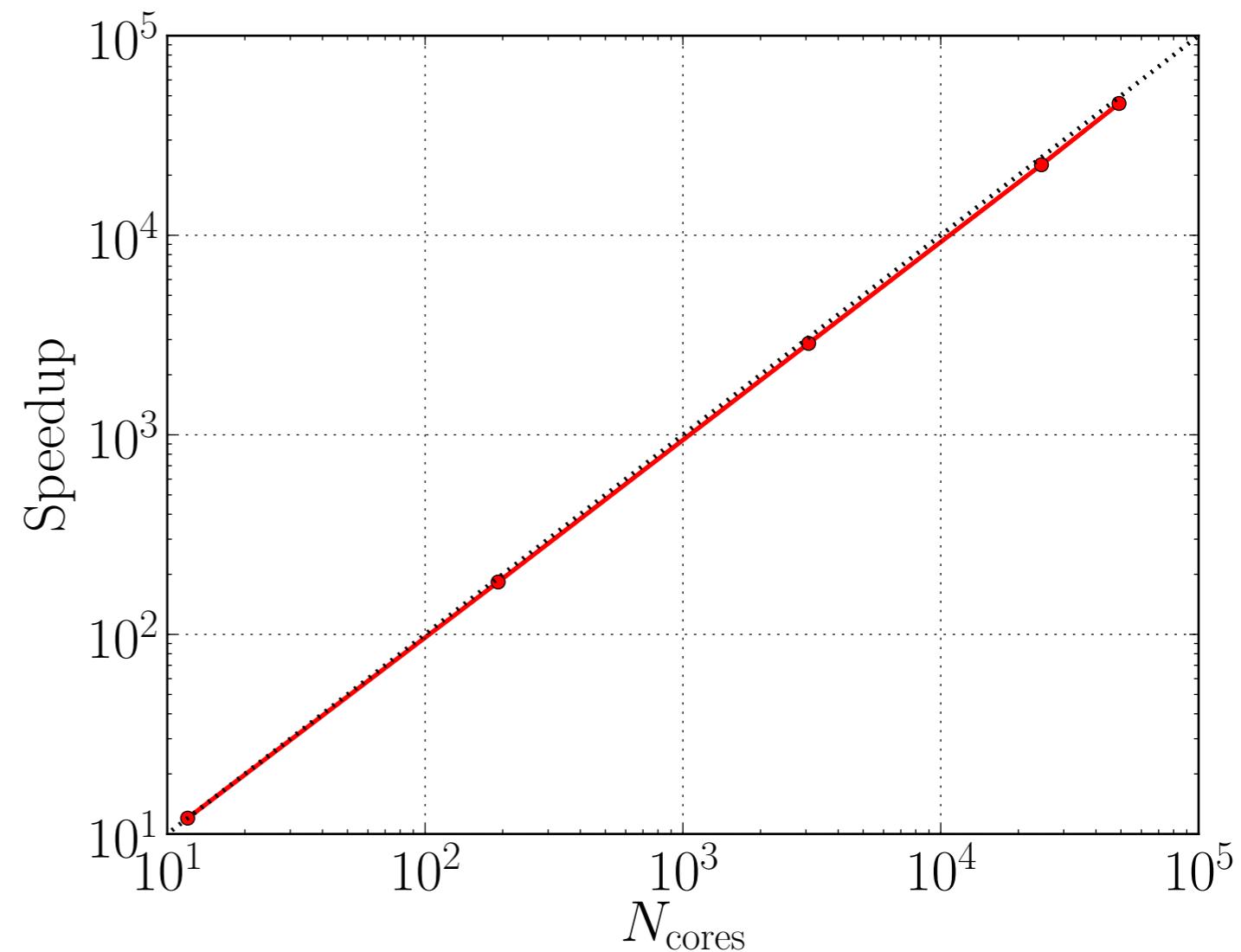
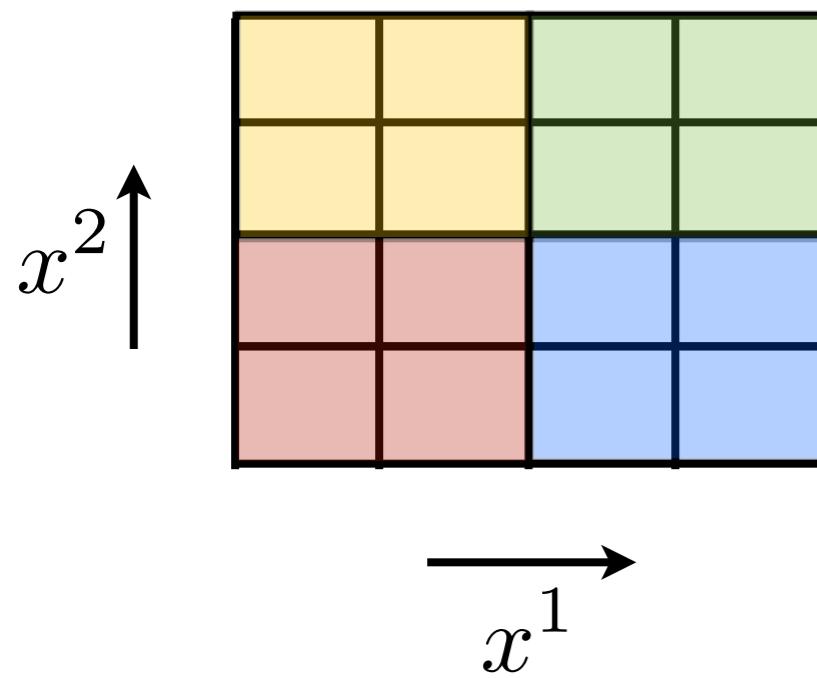
“Event Horizon” Boundary Condition

- Need to ensure that *numerically* no signals escape from the inner grid boundary
- We use horizon-penetrating (Kerr-Schild) coordinates
- Grid extends to the *interior* of the event horizon
- This ensures causal disconnect of the inner radial boundary from the rest of the grid



Code Parallelization and Scaling

- Both HARM and Athena Fully parallelized via domain decomposition (hybrid MPI+OpenMP)
- Near-ideal scaling up to 50,000 cores (weak scaling, 32^3 tile, NICS Kraken, Cray XT5)



New Code HARMPI

- General Relativistic MHD code
- Based on serial, 2D code
HARM2D (Gammie et al. 2003)
- Parallelized it via MPI
- Extended to 3D
- Kept it simple (graduate student startup time = hours)
- Made it open-source:
github.com/atchekho/harmpi
- Added extra physics

 <https://github.com/atchekho/harmpi/blob/master/tutorial.md>

HARMPI Tutorial by Sasha Tchekhovskoy

Please also see useful [exercises](#) that give you an idea of scientific applications.

How to set up HARMPI: choose the problem

- To install the code, you do:

```
git clone git@github.com:atchekho/harmpi.git
cd harmpi
make clean
make
```

 <https://github.com/atchekho/harmpi/blob/master/exercises.md>

HARMPI exercises by Sasha Tchekhovskoy

Please also see the [tutorial](#) that explains basic code use.

Hydro problems

To run problems with HARMPI and to analyze the results, please follow the steps below.

1D hydro problems

- Bondi accretion

Set `WHICHPROBLEM` to `BONDI_PROBLEM_1D` in [decs.h](#). Note: a good

- Plot the profiles of density at a few times in a simulation.

Hint: look at where `v1p` variable changes sign. Ordinarily there is a fast wave (in this problem there is no magnetic field, so fast wave is just adiabatic compression). The flow barely falls inward, so it will be > 0 but at small radial distance it will be < 0 .

How to get the code

- login onto Cartesius:
`ssh -Y <user>@cartesius.surfsara.nl`
- get the code:
`cp -a /scratch-shared/ata2019/codes/harmpi .`
- load modules:
`module load mpi/openmpi/2.0.2 icc python`
- compile the code:
`make clean && make`
- compile the code:
`make clean && make`

How to run the code

- copy the executable to scratch directory

```
mkdir $TMPDIR/myrun  
cd $TMPDIR/myrun  
cp ~/harmpi/harm .
```

- choose a problem of choice in decs.h

```
#define WHICHPROBLEM TORUS_PROBLEM
```

- get an interactive job:

```
srun -t 30 -N 1 -n 16 --pty bash -il
```

- get the interactive job:

```
mpirun ./harm 4 4 1
```

Analyzing the data

- start interactive python session:

```
ipython --pylab
```

- check if graphical windows work

```
plt.figure()
```

- look at the simulation results!

```
%run -i ~/harmpi/harm_script.py  
rg("gdump")  
rd("dump000")  
plco(np.log10(rho),xy=1,xmax=100,ymax=50)
```

-