Adding New Simulations in FLASH: Unfinished by me but will be finished by you!

Dongwook Lee

Flash Center for Computational Science
The University of Chicago

FLASH Tutorial
Rutherford Appleton Laboratory, UK, May 31 – Jun 1, 2012

Collaborators and…

- Don Lamb, Klaus Weide, Anthony Scopatz, Anshu Dubey

- And Anti Collaborators:
  - Raoul, Robbie, Tom, John, Milad, Petros

Setting Up Your Own Simulations

- **Guideline is available:**
  - http://flash.uchicago.edu/site/flashcode/user_support/flash4b_ug/node6.html
  - http://flash.uchicago.edu/site/flashcode/user_support/flash4b_ug/node33.html

- **Where to add:**
  - Hydro: `/source/Simulation/SimulationMain/yourSim`
  - MHD: `/source/Simulation/SimulationMain/magnetoHD/yourSim`

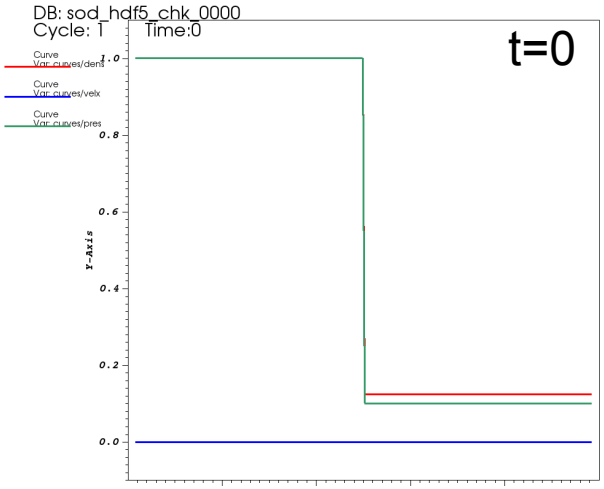
- **What you need:**
  - `Config, Makefile`
  - `Simulation_data.F90`
  - `Simulaiton_init.F90`
  - `Simulation_initBlock.F90`
  - `flash.par`
  - Extras `(e.g., Grid_bcApplyToRegionSpecialized.F90, Simulation_adjustEvolution.F90, Simulation_initSpecies.F90, hy_uhd_unsplit.F90, etc.)`

Files included in `/yourSim`

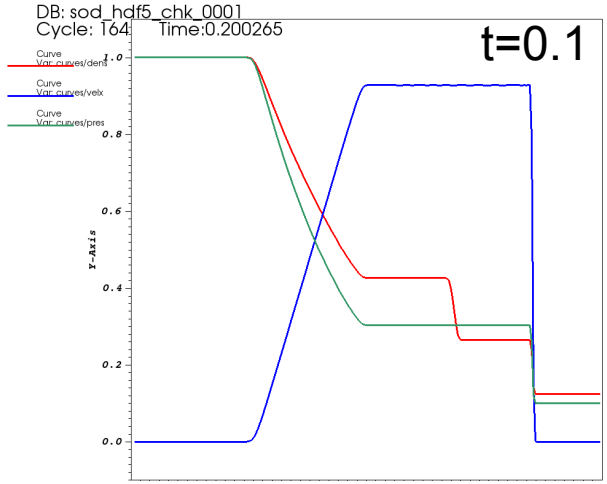
`Config`	Lists the units and variables required for the problem, defines runtime parameters and initializes them with default values.
`Makefile`	The `make` include file for the Simulation.
`Simulation_data.F90`	Fortran module which stores data and parameters specific to the Simulation.
`Simulation_init.F90`	Fortran routine which reads the runtime parameters, and performs other necessary initializations.
`Simulation_initBlock.F90`	Fortran routine for setting initial conditions in a single block.
`Simulation_initSpecies.F90`	Optional Fortran routine for initializing species properties if multiple species are being used.
`flash.par`	A text file that specifies values for the runtime parameters. The values in flash.par override the defaults from Config files.

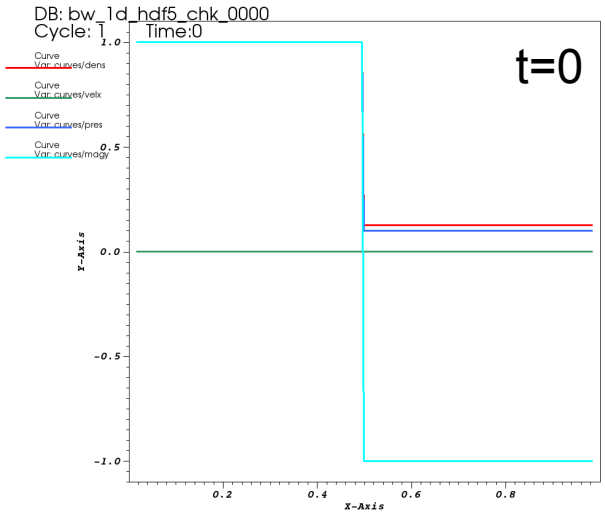
Magnetize Sod Shock Tube

- **1D Brio-Wu MHD shock tube is a magnetized 1D hydro Sod**

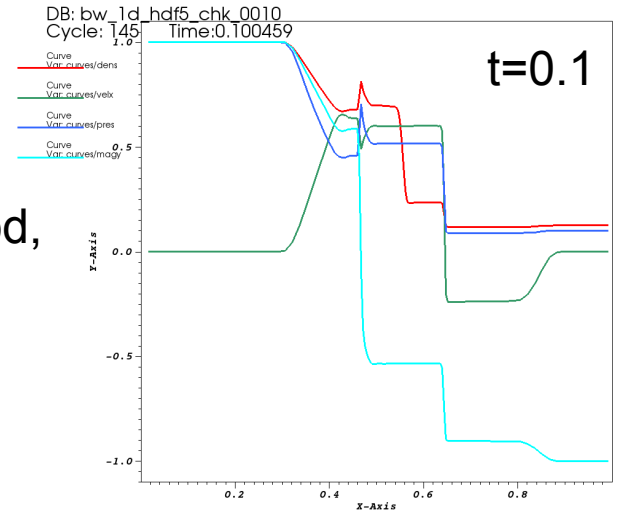


Sod





Magnetized Sod,
or Brio-Wu



Sod's initial condition

$$\gamma = 1.4$$

Table 25.1: Runtime parameters used with the `Sod` test problem.

Variable	Type	Default	Description
`sim_rhoLeft`	real	1	Initial density to the left of the interface (ρ_L)
`sim_rhoRight`	real	0.125	Initial density to the right (ρ_R)
`sim_pLeft`	real	1	Initial pressure to the left (p_L)
`sim_pRight`	real	0.1	Initial pressure to the right (p_R)
`sim_uLeft`	real	0	Initial velocity (perpendicular to interface) to the left (u_L)
`sim_uRight`	real	0	Initial velocity (perpendicular to interface) to the right (u_R)
`sim_xangle`	real	0	Angle made by interface normal with the x -axis (degrees)
`sim_yangle`	real	90	Angle made by interface normal with the y -axis (degrees)
`sim_posn`	real	0.5	Point of intersection between the interface plane and the x -axis

Sod's initial condition

$$\gamma = 1.4$$

Table 25.1: Runtime parameters used with the `Sod` test problem.

Variable	Type	Default	Description
`sim_rhoLeft`	real	1	Initial density to the left of the interface (ρ_L)
`sim_rhoRight`	real	0.125	Initial density to the right (ρ_R)
`sim_pLeft`	real	1	Initial pressure to the left (p_L)
`sim_pRight`	real	0.1	Initial pressure to the right (p_R)
`sim_uLeft`	real	0	Initial velocity (perpendicular to interface) to the left (u_L)
`sim_uRight`	real	0	Initial velocity (perpendicular to interface) to the right (u_R)
`sim_xangle`	real	0	Angle made by interface normal with the x -axis (degrees)
`sim_yangle`	real	90	Angle made by interface normal with the y -axis (degrees)
`sim_posn`	real	0.5	Point of intersection between the interface plane and the x -axis

Additional changes required for IC for magnetized Sod (or Brio-Wu):

$$\gamma = 2$$

$$B_x = 0.75$$

$$B_y = \begin{cases} 1, & \text{if } x < 0.5 \\ -1 & \text{if } x > 0.5 \end{cases}$$

$$B_z = 0$$

$$B_p = \frac{1}{2}\left(B_x^2 + B_y^2 + B_z^2\right)$$

Simulaiton_initBlock.F90

- **Let's magnetize a simple 1D hydro setup!**
  - `cd -r source/Simulation/SimulationMain/`
  - `cp -r Sod ./magnetoHD/MagSod`
  - `cd ./magnetoHD/MagSod`

- **Use your favorite text editor, open** `Simulation_initBlock.F90:`
  - **Add local real variables:** `magxZone, magyZone, magzZone, magpZone`
  - **Initialize magnetic fields using "**`sim_*`**" parameters specific to** `MagSod`

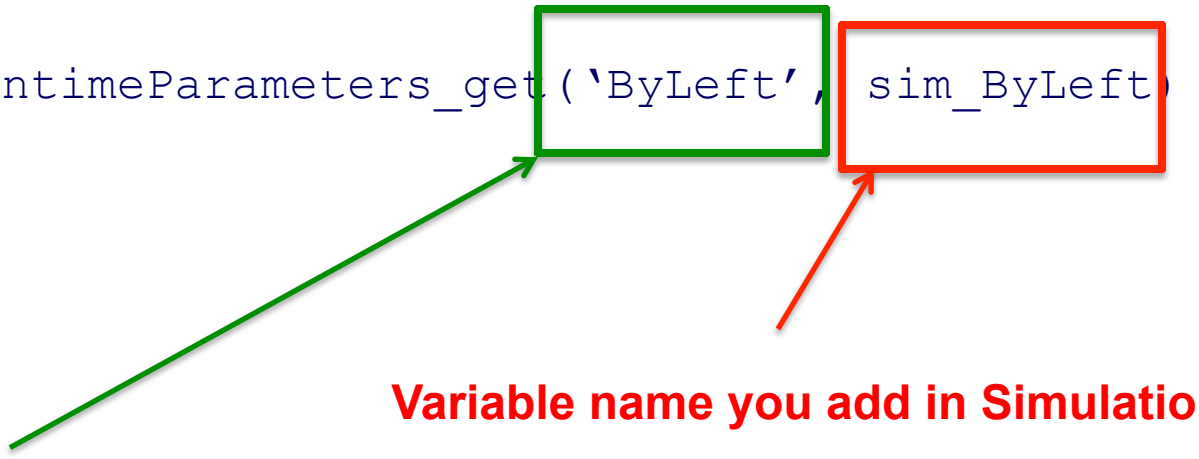
Simulaiton_Data.F90 & Simulation_init.F90

- **Open** `Simulation_Data.F90`:
  - **Declare data module variables:**

```
sim_Bx, sim_ByLeft, sim_ByRight, sim_BzLeft, sim_BzRight
```

- **Open** `Simulation_init.F90`:
  - **Add lines to read-in user defined** `sim_` **values from** `Config` **&** `flash.par`**:**

e.g. `call RuntimeParameters_get('ByLeft', sim_ByLeft)`



Variable name you add in Simulation_*.F90

Variable name you add in Config and flash.par

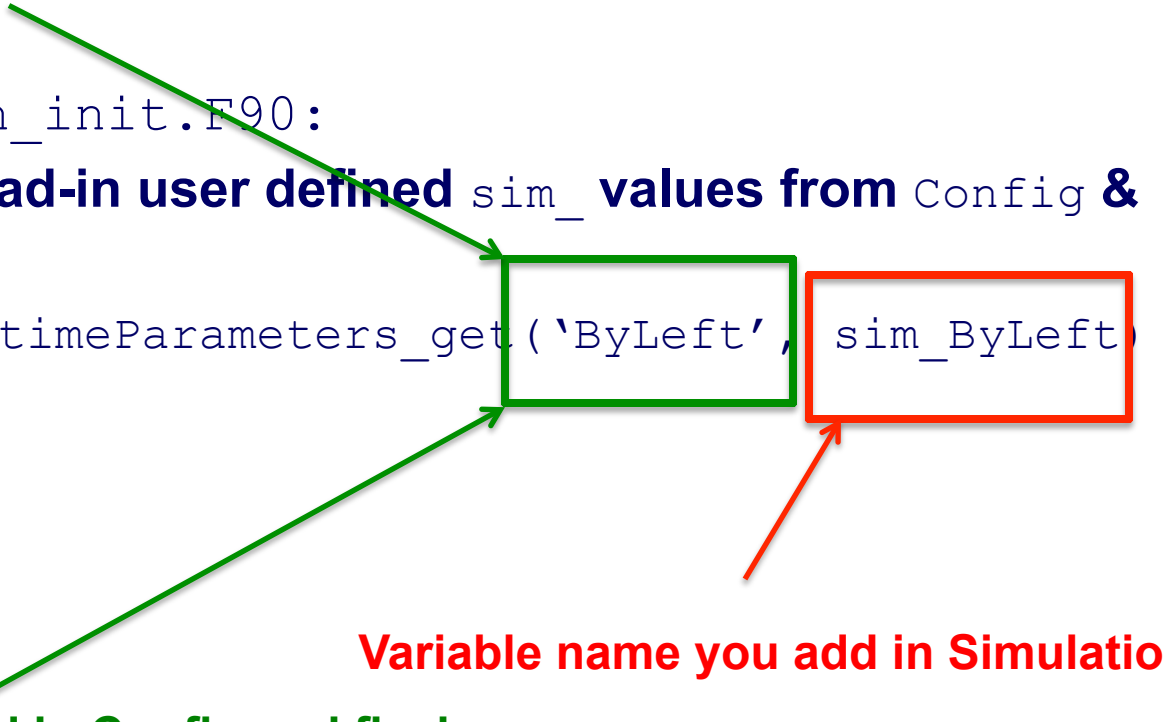
Config

- **Open** `Config`:
  - **Specify default values:**

    `Bx, ByLeft, ByRight, BzLeft, BzRight`

- **Open** `Simulation_init.F90`:
  - **Add lines to read-in user defined** `sim_` **values from** `Config` **&** `flash.par`:

    e.g. `call RuntimeParameters_get('ByLeft', sim_ByLeft)`



Variable name you add in Simulation_*.F90

Variable name you add in Config and flash.par

Lastly, ./setup

- **Let's setup!**
  - `./setup magnetoHD/MagSod -1d -auto +usm -objdir=MagSod`

Runtime Parameters


```
# Runtime Parameters
# (1) Interpolation, reconstruction, slope limiter:
PARAMETER order                 INTEGER  2               # Order of scheme: 1st/2nd/3rd/5th order
PARAMETER transOrder            INTEGER  1               # Order of transverse flux: 1st order. 3rd order is experimental.
PARAMETER slopeLimiter          STRING   "vanLeer"       # Slope limiter for Riemann state
PARAMETER charLimiting          BOOLEAN  TRUE            # Turn on/off characteristic/primitive limiting
PARAMETER LimitedSlopeBeta      REAL     1.0             # Any real value specific for the Limited Slope
                                                         # limiter (e.g., 1.0 for minmod, 2.0 for superbee)
PARAMETER use_steepening        BOOLEAN  FALSE           # Turn on/off PPM contact steepening
PARAMETER use_flattening        BOOLEAN  FALSE           # Turn on/off flattening
PARAMETER use_avisc             BOOLEAN  FALSE           # Turn on/off artificial viscosity
PARAMETER cvisc                 REAL     0.1             # artificial viscosity constant
PARAMETER use_upwindTVD         BOOLEAN  FALSE           # Turn on/off upwinding TVD slopes

# (2) For 3D CTU
PARAMETER use_3dFullCTU         BOOLEAN  TRUE            # FALSE will give the simpler CTU without corner upwind coupling
                                                         # and will only provide CFL < 1/2
# (3) Riemann solvers
PARAMETER RiemannSolver         STRING   "Roe"           # Approximate Riemann solver:
                                                         # Roe (default), HLL, HLLC, Marquina, MarquinaMod, Hybrid
                                                         # or local Lax-Friedrichs, plus HLLD for MHD
PARAMETER entropy               BOOLEAN  FALSE           # Turn on/off an entropy fix routine
PARAMETER entropyFixMethod      STRING   "HARTENHYMAN"   # Entropy fix method for the Roe Riemann solver:
                                                         # Harten or HartenHyman
PARAMETER shockDetect           BOOLEAN  FALSE           # Turn on/off a shock detecting switch
PARAMETER EOSforRiemann         BOOLEAN  FALSE           # Turn on/off EOS calls for the Riemann states
PARAMETER addThermalFlux        BOOLEAN  TRUE            # Add/don't add thermal fluxes to hydro fluxes

# (4) Gravity updates
PARAMETER use_gravHalfUpdate    BOOLEAN  FALSE           # Include gravitational accelerations to hydro coupling at n+1/2
PARAMETER use_gravConsv         BOOLEAN  FALSE           # Use conservative variables for gravity coupling at n+1/2
PARAMETER use_GravPotUpdate     BOOLEAN  FALSE           # Parameter for half timestep update of gravitational potential
```

Runtime Parameters


```
# Runtime Parameters for unsplit USM-MHD solver
PARAMETER killdivb              BOOLEAN  TRUE            # Turn on/off  DivB cleaning
PARAMETER E_modification        BOOLEAN  TRUE            # Turn on/off electric field modification
PARAMETER E_upwind              BOOLEAN  FALSE           # Turn on/off upwind update for induction equations
PARAMETER energyFix             BOOLEAN  FALSE           # Turn on/off an energy correction for CT scheme
PARAMETER ForceHydroLimit       BOOLEAN  FALSE           # Turn on/off a hydro limiting switch
PARAMETER facevar2ndOrder       BOOLEAN  TRUE            # Turn on/off a 2nd order facevar update
PARAMETER use_Biermann          BOOLEAN   FALSE          # Biermann Battery Term
PARAMETER use_Biermann1T        BOOLEAN   FALSE          # 1T Biermann Battery Term
PARAMETER hy_biermannSource     BOOLEAN   FALSE          # enable Battery Source (vs. flux)
PARAMETER hy_bier1TZ            REAL     -1.0            # Zbar value for 1T Biermann Battery Term
PARAMETER hy_bier1TA            REAL     -1.0            # Abar value for 1T Biermann Battery Term
PARAMETER prolMethod            STRING   "INJECTION_PROL" # Prolongation method: injection_prol/Balsara_prol
PARAMETER hy_biermannCoef       REAL     1.0             # Coefficient of Biermann Battery flux
```

```
# Number of guard cells at each boundary
USESETUPVARS SupportWeno, SupportPpmUpwind
IF SupportWeno
   GUARDCELLS 6 # the Unsplit Hydro/MHD solver requires 6 guard cells to support WENO!
ELSEIF SupportPpmUpwind
   GUARDCELLS 6 # the Unsplit Hydro/MHD solver requires 6 guard cells to support PPM Upwind!
ELSE
   GUARDCELLS 4 # the Unsplit Hydro/MHD solver requires 4 guard cell layers!
ENDIF
```


Cres
cunt
Vita
crescendo
latur

Questions?


Office of
Science
U.S. DEPARTMENT OF ENERGY
NNSA
National Nuclear Security Administration