FLASH, HEDP, and Future Needs

July 30th, 2012, CScADS, Snowbird, UT

Anthony Scopatz
The FLASH Center
The University of Chicago
scopatz@flash.uchicago.edu

What is FLASH?

FLASH code is a modular, parallel multiphysics simulation code for modeling terrestrial and astrophysical plasmas. Features include:

What is FLASH?

FLASH code is a modular, parallel multiphysics simulation code for modeling terrestrial and astrophysical plasmas. Features include:

- Grid: Uniform Grid, AMR

What is FLASH?

FLASH code is a modular, parallel multiphysics simulation code for modeling terrestrial and astrophysical plasmas. Features include:

- Grid: Uniform Grid, AMR
- Equation of State: Ideal gas, Multimaterial

What is FLASH?

FLASH code is a modular, parallel multiphysics simulation code for modeling terrestrial and astrophysical plasmas. Features include:

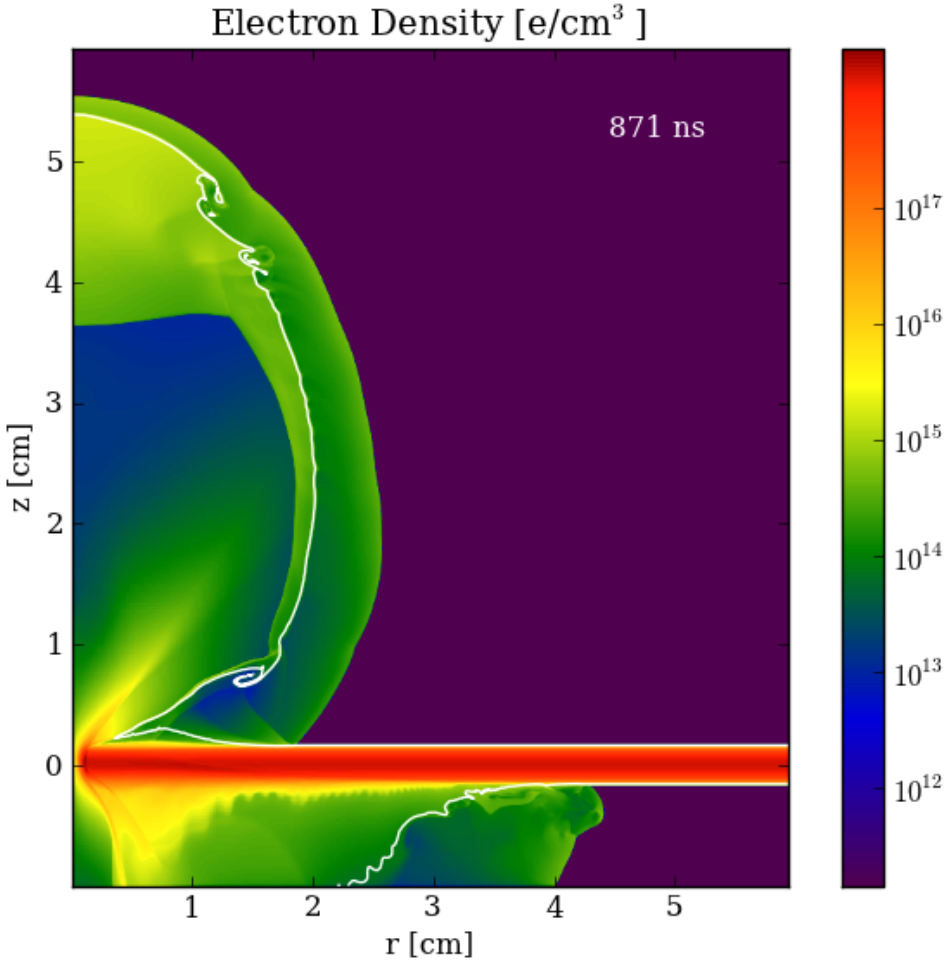
- Grid: Uniform Grid, AMR
- Equation of State: Ideal gas, Multimaterial
- Laser ray trace package

What is FLASH?

FLASH code is a modular, parallel multiphysics simulation code for modeling terrestrial and astrophysical plasmas. Features include:

- Grid: Uniform Grid, AMR
- Equation of State: Ideal gas, Multimaterial
- Laser ray trace package
- Nuclear Burning

What is FLASH?



FLASH Architecture

FLASH compiles simulation-specific binaries from user defined code Units.

FLASH Architecture

FLASH compiles simulation-specific binaries from user defined code Units.

Code Units are mix of Fortran, C, C++, and Python and have an OO-esque inheritance model via posix directories.

FLASH Architecture

FLASH compiles simulation-specific binaries from user defined code Units.

Code Units are mix of Fortran, C, C++, and Python and have an OO-esque inheritance model via posix directories.

To execute a FLASH simulation you must:

- setup (pure python),

FLASH Architecture

FLASH compiles simulation-specific binaries from user defined code Units.

Code Units are mix of Fortran, C, C++, and Python and have an OO-esque inheritance model via posix directories.

To execute a FLASH simulation you must:

- setup (pure python),
- build (make),

FLASH Architecture

FLASH compiles simulation-specific binaries from user defined code Units.

Code Units are mix of Fortran, C, C++, and Python and have an OO-esque inheritance model via posix directories.

To execute a FLASH simulation you must:

- setup (pure python),
- build (make),
- and run (flash binary).

What is HEDP?

(**H**)igh (**E**)nergy (**D**)ensity (**P**)hysics is the study of hot, dense plasmas where temp >10^6 C & pres >10^8 ATM.

What is HEDP?

(**H**)igh (**E**)nergy (**D**)ensity (**P**)hysics is the study of hot, dense plasmas where temp >10^6 C & pres >10^8 ATM.

Many features have been added to FLASH enabling HEDP simulations in the past couple of years.

What is HEDP?

(**H**)igh (**E**)nergy (**D**)ensity (**P**)hysics is the study of hot, dense plasmas where temp >10^6 C & pres >10^8 ATM.

Many features have been added to FLASH enabling HEDP simulations in the past couple of years.

Specifically, we are interested in the related field of Laboratory Astrophysics.

What is HEDP?

(**H**)igh (**E**)nergy (**D**)ensity (**P**)hysics is the study of hot, dense plasmas where temp >10^6 C & pres >10^8 ATM.

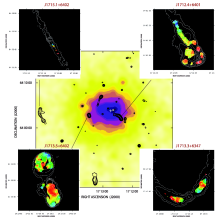
Many features have been added to FLASH enabling HEDP simulations in the past couple of years.

Specifically, we are interested in the related field of Laboratory Astrophysics.

- Many of the fundamental laws of plasma physics have no implicit scale.

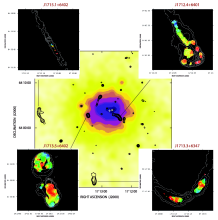
Problem Statement

The is a small, but measurable intergalactic magnetic field (femto- to pico-gauss) [*origin unexplained*].



Problem Statement

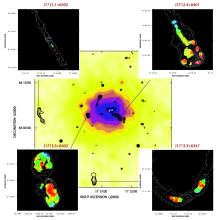
The is a small, but measurable intergalactic magnetic field (femto- to pico-gauss) [*origin unexplained*].



One theory is that they are born from asymmetric shocks caused when galaxies or galaxy clusters collide.

Problem Statement

The is a small, but measurable intergalactic magnetic field (femto- to pico-gauss) [*origin unexplained*].



One theory is that they are born from asymmetric shocks caused when galaxies or galaxy clusters collide.

Generation of magnetic fields from a neutral plasma has recently been demonstrated in the lab [1].

1. G. Gregori, et al., *Generation of scaled protogalactic seed magnetic fields in laser-produced shock waves*, Nature 481 (2012) 480-483.

Biermann Battery Mechanism

The generalized Ohm's law sets the strength of the electric field in the MHD approximation. Only the Battery term can produce magnetic fields from an initially unmagnetized plasma:

$$\mathbf{E} = \mathbf{u} \times \mathbf{B} + \eta \mathbf{j} + \frac{1}{n_e e} \mathbf{j} \times \mathbf{B} - \frac{\nabla P_e}{n_e e}$$

Biermann Battery Mechanism

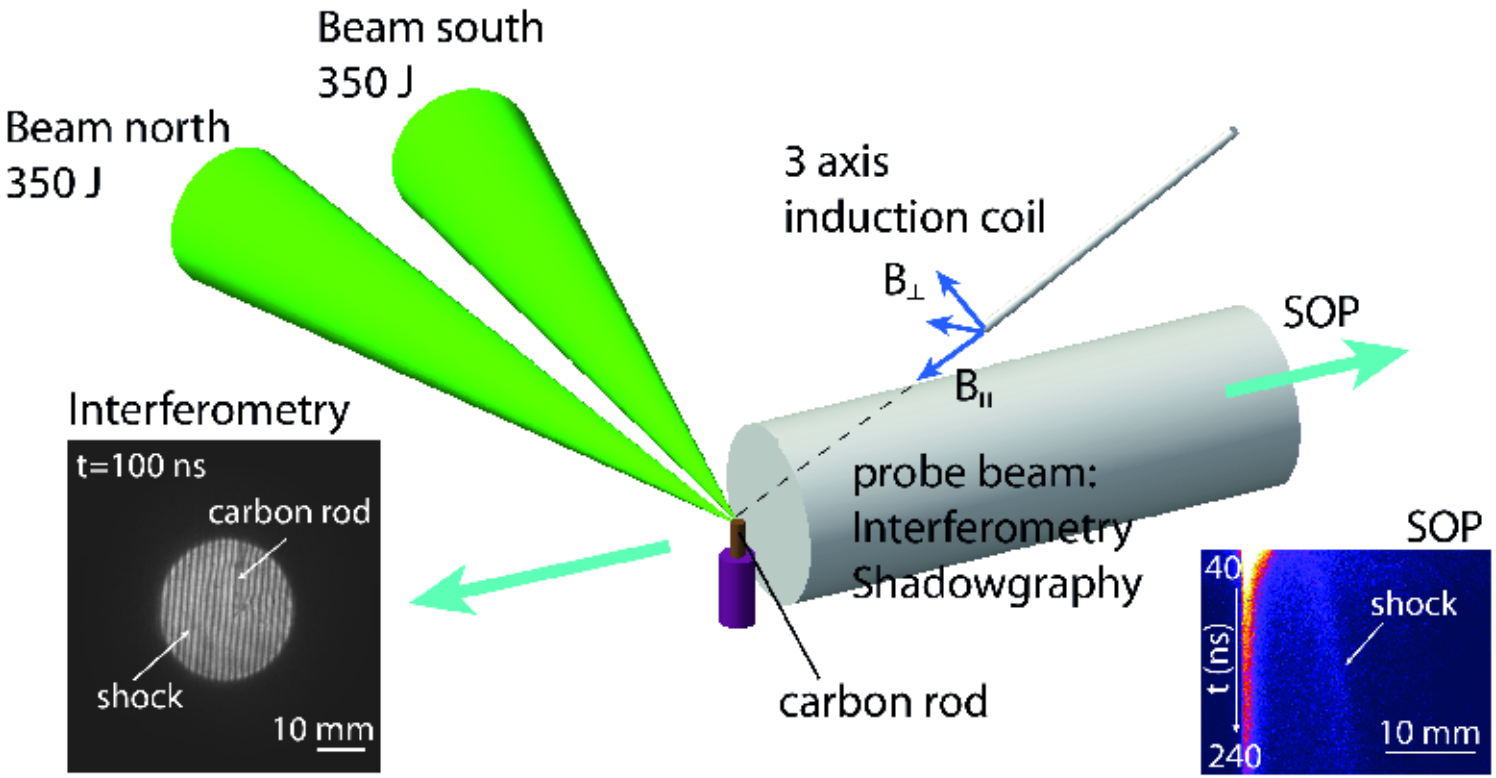
The generalized Ohm's law sets the strength of the electric field in the MHD approximation. Only the Battery term can produce magnetic fields from an initially unmagnetized plasma:

$$\mathbf{E} = \mathbf{u} \times \mathbf{B} + \eta \mathbf{j} + \frac{1}{n_e e} \mathbf{j} \times \mathbf{B} - \frac{\nabla P_e}{n_e e}$$

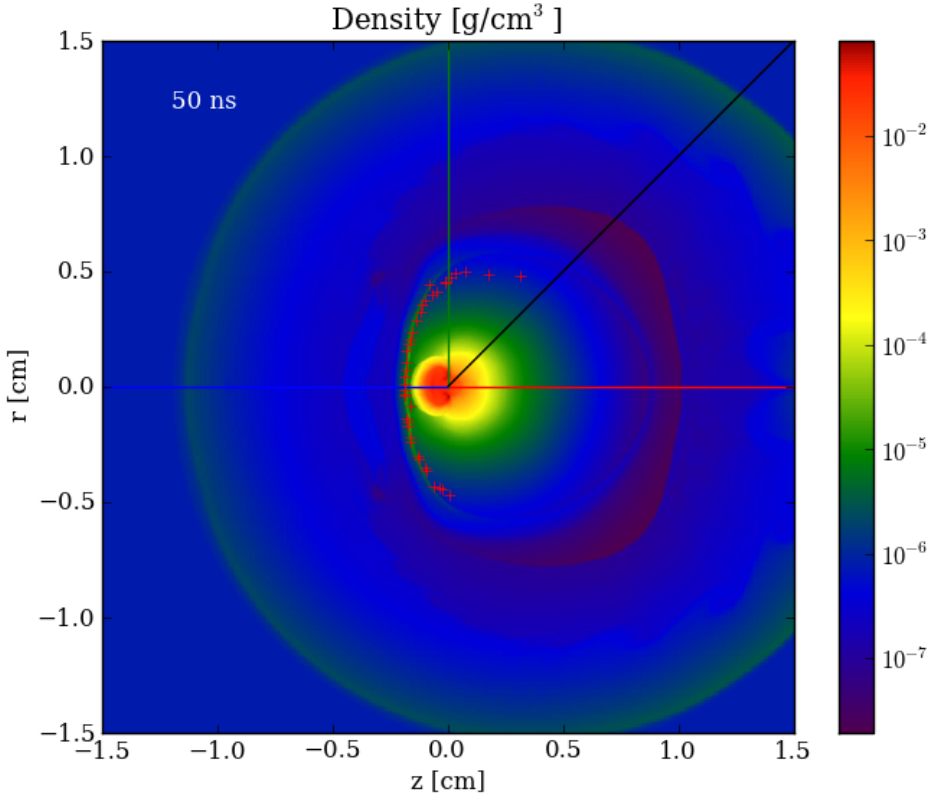
Faraday's law relates the electric field to the rate of change of the magnetic field:

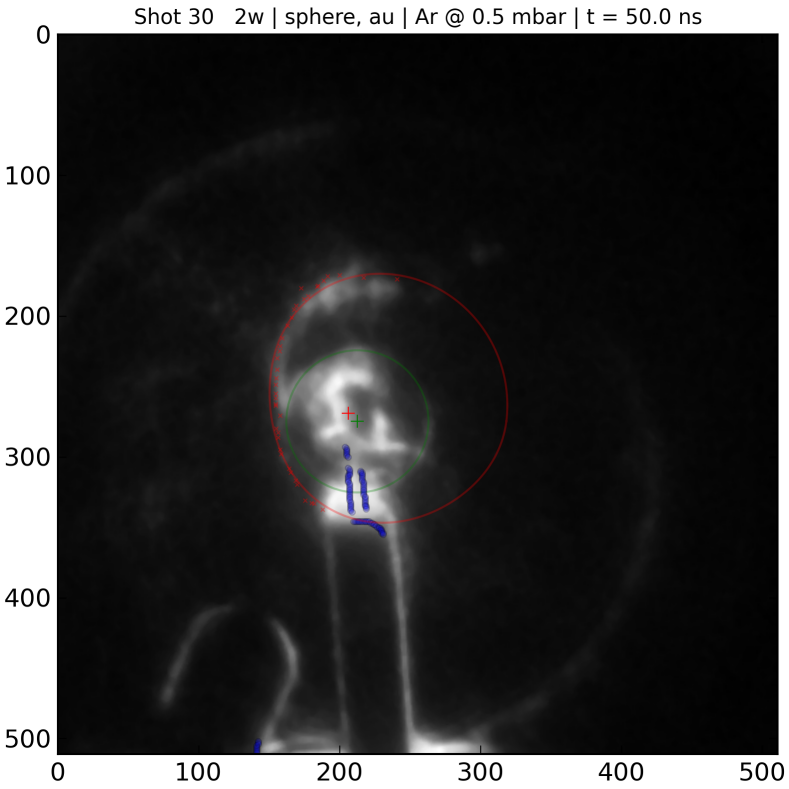
$$\left(\frac{\partial \mathbf{B}}{\partial t}\right)_{Biermann} = c \nabla \times \left(\frac{\nabla P_e}{n_e e}\right) = c \frac{\nabla P_e \times \nabla n_e}{e n_e^2}$$

Experimental Setup



Shadowgraphy: Simulation & Data





Shadowgraphy

Computationally, shadowgraphy is a little annoying. The intensity in the image is proportional to:

$$I \propto \int_{n_1}^{n_2} \left(\frac{\partial^2 n_e}{\partial x^2} + \frac{\partial^2 n_e}{\partial y^2} \right) dn$$

Shadowgraphy

Computationally, shadowgraphy is a little annoying. The intensity in the image is proportional to:

$$I \propto \int_{n_1}^{n_2} \left(\frac{\partial^2 n_e}{\partial x^2} + \frac{\partial^2 n_e}{\partial y^2} \right) dn$$

Some vector calculus later:

$$I \propto \int_{n_1}^{n_2} \left(\nabla^2 n_e - \nabla (\nabla n_e \cdot \hat{n}) \cdot \hat{n} \right) dn$$

Shadowgraphy

Computationally, shadowgraphy is a little annoying. The intensity in the image is proportional to:

$$I \propto \int_{n_1}^{n_2} \left(\frac{\partial^2 n_e}{\partial x^2} + \frac{\partial^2 n_e}{\partial y^2} \right) dn$$

Some vector calculus later:

$$I \propto \int_{n_1}^{n_2} \left(\nabla^2 n_e - \nabla (\nabla n_e \cdot \hat{n}) \cdot \hat{n} \right) dn$$

This is not quite volumetric ray tracing...

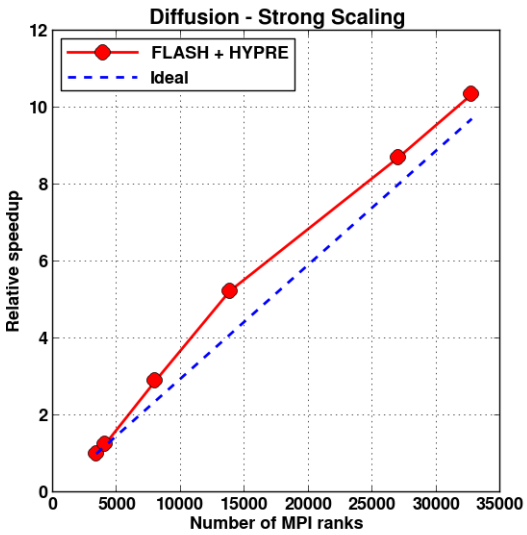
FLASH on BG/P

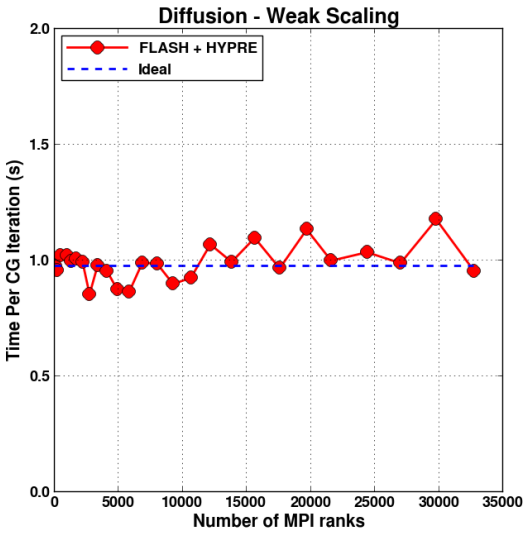
- Intrepid BG/P: 4 cores/node, 2 GB/node, 40,960 nodes
  - FLASH has run on Intrepid for the last several years
    - Scales to the whole machine
    - MPI-only is sufficient
    - Run in VN mode (4 MPI ranks/node)

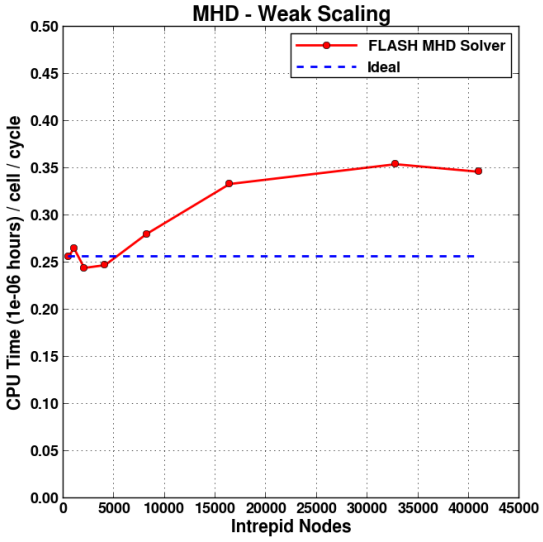
FLASH on BG/Q

- Mira BG/Q 4 hw threads/core, 16 cores/node, 16 GB/node, 49,152 nodes
  - MPI-only approach not suitable for BG/Q
  - OpenMP directives have been added to FLASH to take advantage of the additional intra-node parallelism

Scaling


Diffusion - Strong Scaling
FLASH + HYPRE
Ideal
Relative speedup
Number of MPI ranks
0
2
4
6
8
10
12
0
5000
10000
15000
20000
25000
30000
35000


Diffusion - Weak Scaling
FLASH + HYPRE
Ideal
Time Per CG Iteration (s)
Number of MPI ranks
0.0
0.5
1.0
1.5
2.0
0
5000
10000
15000
20000
25000
30000
35000


MHD - Weak Scaling
FLASH MHD Solver
Ideal
CPU Time (1e-06 hours) / cell / cycle
Intrepid Nodes
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0
5000
10000
15000
20000
25000
30000
35000
40000
45000

Resource Usage

- We foresee needing to use 20% of Mira (or 1 Intrepid) due to the increased number of physics models included.

Resource Usage

- We foresee needing to use 20% of Mira (or 1 Intrepid) due to the increased number of physics models included.

- During a full 3D run, the checkpoint files will be ~68 GB and the plot files will be ~6.8 GB. We anticipate needing 36 checkpoints and 144 plots.

Resource Usage

- We foresee needing to use 20% of Mira (or 1 Intrepid) due to the increased number of physics models included.

- During a full 3D run, the checkpoint files will be ~68 GB and the plot files will be ~6.8 GB. We anticipate needing 36 checkpoints and 144 plots.

- By performing *in situ* simulated diagnostics, huge data reductions are possible.

Rough Plan

Over the next 6 months, I will be attempting to:

- Add capabilities to yt to support simulated diagnostic calculations,

Rough Plan

Over the next 6 months, I will be attempting to:

- Add capabilities to yt to support simulated diagnostic calculations,
- Extend yt and/or h5py to a GLEAN backend,

Rough Plan

Over the next 6 months, I will be attempting to:

- Add capabilities to yt to support simulated diagnostic calculations,
- Extend yt and/or h5py to a GLEAN backend,
- Create infrastructure for co- or post-processing of FLASH data for initial diagnostics (Schlieren, shadowgraphy, and magnetic field probes).

Acknowledgements

- **FLASH:** Milad Fatenejad, Chris Daley, Don Lamb, Anshu Dubey
- **The University of Oxford:** Gianluca Gregori, Jena Meineke
- **yt:** Matt Turk, John ZuHone
- **ANL:** Venkat Vishwanath

Questions