



The Center for Astrophysical Thermonuclear Flashes

Performance bottlenecks in FLASH

Chris Daley

27th July



An Advanced Simulation & Computing (ASC)
Academic Strategic Alliances Program (ASAP) Center
at The University of Chicago





FLASH and introduction to main bottleneck

- ❑ FLASH is a multiphysics, adaptive mesh refinement (AMR) code used to simulate problems occurring in e.g. astrophysics, plasma physics and cosmology.
- ❑ The center will use FLASH in 2009/2010 to:
 - study turbulent nuclear combustion.
 - perform a systematic validation of current models of type Ia supernova.
- ❑ Some of the supernova simulations carried out in early 2009 spent up to 1/3 total runtime in I/O!



Breakdown of bottleneck

Worst case I/O performance:

(from a 12-hour experiment run on 2048 cores of Intrepid)

Type	Approx size (GB)	Number of times written	Total runtime impact
Checkpoint file	8.0	10	3%
Plot file	2.5	104	9%
Particle file	0.1	417	22%

- ❑ Checkpoint file: Complete state of simulation in double precision.
- ❑ Plot file: Subset of grid variables in single precision.
- ❑ Particle file: All particle attributes in double precision.



Bottleneck discussion

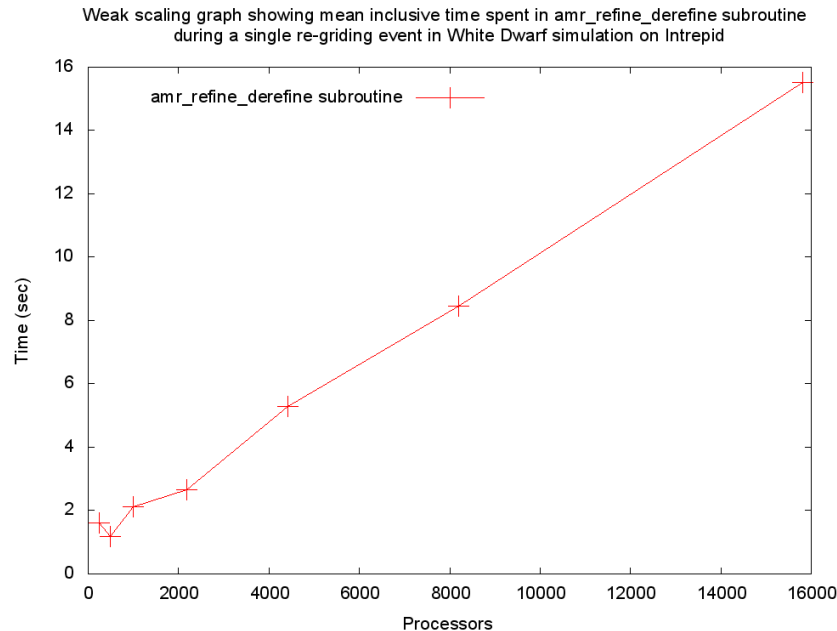
- This is a lot of data and a high write frequency:
 - FLASH I/O is not particularly inefficient.
 - Already makes use of parallel I/O through the HDF-5 and Parallel-netCDF libraries.

- We recently started work with staff at Argonne National Laboratory to further improve FLASH I/O.
 - We are going to investigate:
 - using a different FLASH data file layout to maximize data throughput.
 - buffering particle data for a few steps to decrease the frequency of writes.



Other significant FLASH bottleneck

- There is very poor scaling during AMR re-gridding events with the PARAMESH package.



- Only significantly impacts those simulations that use large numbers of processors and frequently refine / derefine.

- Side note: PARAMESH no longer actively developed.
 - Any modifications need to be made by Flash team.
 - We may use other mesh packages in future.