

LA-UR-24-23469

Approved for public release; distribution is unlimited.

Title: MCNP6(R) New Features and Improvements for Reactor Physics Applications

Author(s): Rising, Michael Evan
Armstrong, Jerawan Chudoung
Maldonado, Alexis
Vaquer, Pablo Andres
Weaver, Colin Andrew

Intended for: International Conference on Physics of Reactors (PHYSOR 2024),
2024-04-21/2024-04-24 (San Francisco, California, United States)

Issued: 2024-07-23 (rev.1)



Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

MCNP6® New Features and Improvements for Reactor Physics Applications

¹Michael E. Rising, ¹Jerawan Armstrong, ²Alexis Maldonado, ¹Pablo Vaquer, and ¹Colin Weaver

¹XCP-3 and ²NEN-5, LANL

International Conference on Physics of Reactors 2024 (PHYSOR 2024)

Sunday, April 21, 2024

LA-UR-24-23469

Acknowledgements

This work is supported by the Department of Energy through Los Alamos National Laboratory (LANL) operated by Triad National Security, LLC, for the National Nuclear Security Administration (NNSA) under Contract No. 89233218CNA000001.

This work is supported by the LANL MCNP Site Support Project, the LANL LDRD Program, and the DOE Nuclear Criticality Safety Program.

Thanks to the PHYSOR 2024 Topical meeting organizers for including us here.

Thanks to those who provided content to this workshop, including Jennifer Alwin, Simon Bolding, Forrest Brown, Jeffrey Bull, Alexander Clark, Micky Dzur, Colin Josey, Joel Kulesza, Scott Mosher, Josh Spencer, Kristin Stolte, and Sriram Swaminarayan.

Thanks to Jerawan Armstrong and Joel Kulesza for reviewing the content of this workshop.

MCNP[®] Trademark

MCNP[®] and Monte Carlo N-Particle[®] are registered trademarks owned by Triad National Security, LLC, manager and operator of Los Alamos National Laboratory. Any third party use of such registered marks should be properly attributed to Triad National Security, LLC, including the use of the [®] designation as appropriate.

- ▶ Please note that trademarks are adjectives and should not be pluralized or used as a noun or a verb in any context for any reason.
- ▶ Any questions regarding licensing, proper use, and/or proper attribution of Triad National Security, LLC marks should be directed to trademarks@lanl.gov.

Outline

MCNP6.3 Overview

Deimos Project

Unstructured Mesh and HDF5

- Unstructured Mesh Overview

- Aside: HDF5

- Demo UM to HDF5 and Visualization in Paraview

Fission Matrix for Criticality Calculations

- Fission Matrix Overview

- Fission Matrix Interrogation

- Demo Fission Matrix Calculation and Visualization

New Tally Features

- FMESH Upgrades and HDF5+XDMF

- Special Tally Treatments for Multigroup Cross Sections

- Demo FMESH and Multigroup Cross Section Calculations

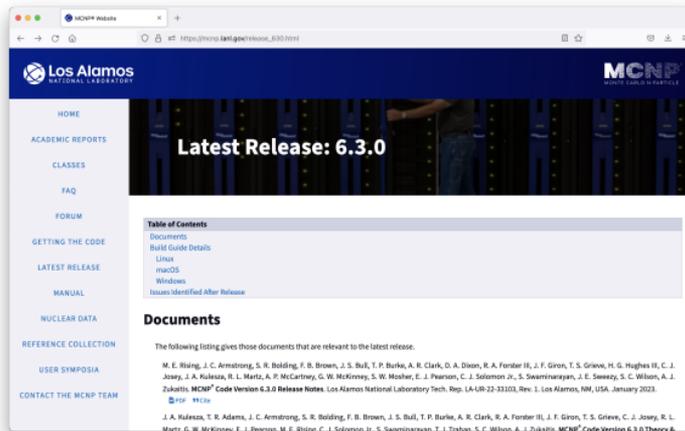
MCNP Exercise for Nuclear Criticality Safety

Future Work and Q&A

MCNP6.3 Overview

MCNP6.3 Release

- ▶ For the most up-to-date details on the latest release, see the Latest Release (https://mcnp.lanl.gov/release_630.html) tab on the MCNP website.



- ▶ The primary MCNP6.3 Release Notes [1], Theory & User Manual [2], Build Guide [3], and Verification and Validation Testing [4] documents can be found on the website and in the software distribution.

New Features

- ▶ **Fission-matrix-based convergence testing and acceleration**
- ▶ ***Multigroup cross section tallies for reactor analysis***
- ▶ Doppler broadening resonance correction
- ▶ Stochastic $S(\alpha, \beta)$ temperature mixing
- ▶ Mixed-material treatment for structured meshes
- ▶ ***New FMESH tally backend options***
- ▶ *Parallel PTRAC support*
- ▶ CMake build system
- ▶ ***HDF5-formatted output files and XDMF support***

Covered or partially discussed in this workshop

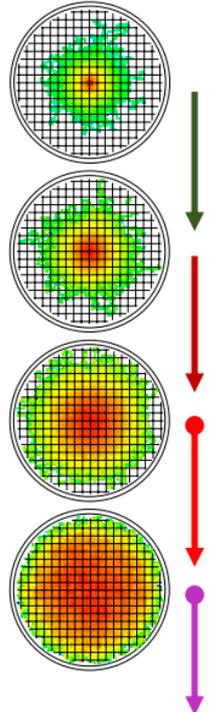
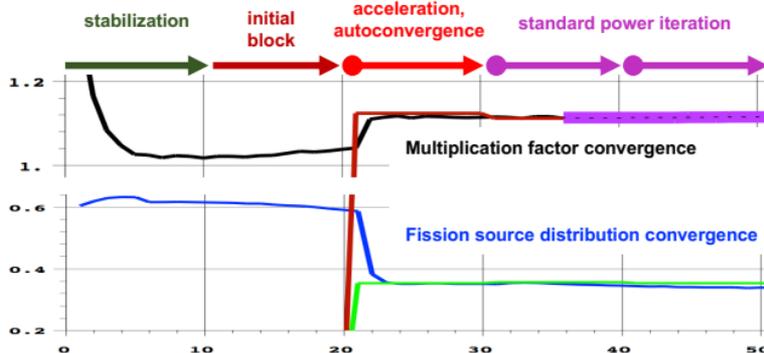
Improvements

- ▶ ***New Qt-based plotter preview available as technology preview***
- ▶ All of the MCNP code meets Fortran 2008 and C++ 17 standard requirements
- ▶ Preliminary support for upcoming ENDF/B-VIII.1 data changes
 - ▶ For new $S(\alpha, \beta)$ format options
 - ▶ For photonuclear physics
- ▶ *Upgrade to use open-source version of CGMF 1.1.1*
- ▶ ***Improved unstructured mesh input file processing***
- ▶ ***Added unstructured mesh quality metrics and reporting***
- ▶ Decrease of delayed-gamma-line memory usage
- ▶ *Various deprecated features with plans for future removal*

Covered or partially discussed in this workshop

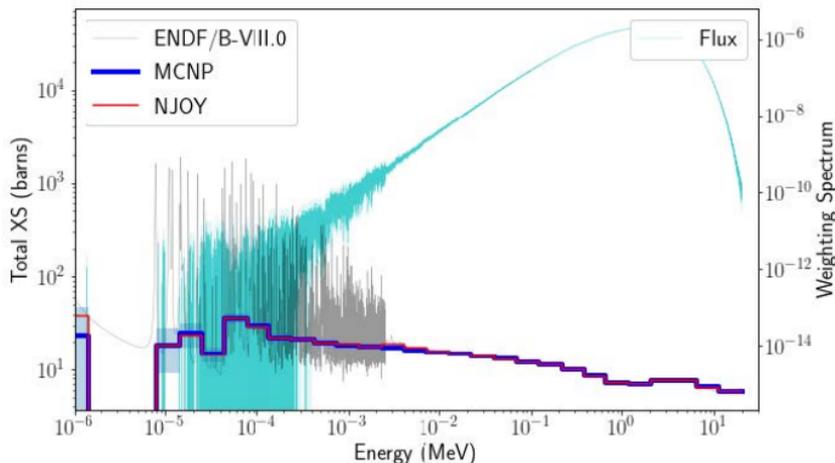
Fission-matrix-based Convergence Testing and Acceleration

- ▶ Automated acceleration of the convergence of the fission source distribution
 - ▶ Eliminates user intervention and trial-and-error testing
 - ▶ Saves computational cost
- ▶ More robust statistical convergence and population testing
 - ▶ Ensuring the simulation has converged



Features for Advanced Reactor Analysis

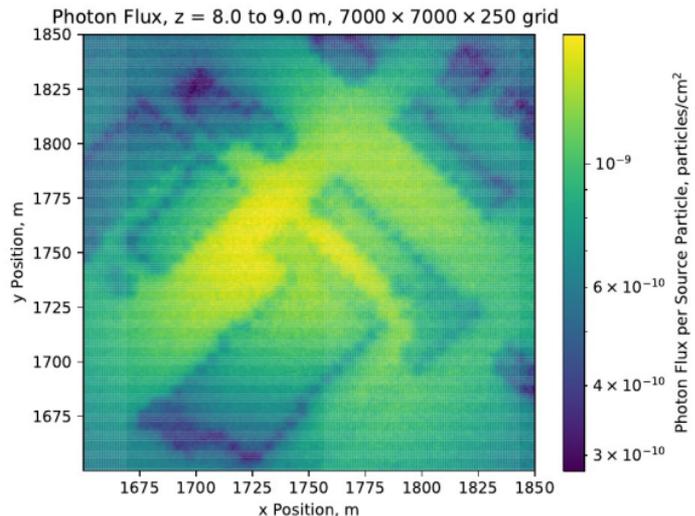
- ▶ Unstructured mesh developments for high-fidelity reactor simulations
 - ▶ MCNP material properties based on element sets
 - ▶ Developed MCNP and ABAQUS based Reactor Multiphysics (MARM) framework for high temperature reactor analysis [5]
- ▶ Multigroup cross section tallies computed for coupling to deterministic, time-dependent reactor feedback codes [6]



Mesh Tally Backend Improvements Enabling Extreme-scale, High-fidelity Simulations [7]

- ▶ History and batch statistics
- ▶ Advanced MPI parallelism options
- ▶ Better performance on current problems
- ▶ Scaling to extreme scale for higher resolution future problems
 - ▶ E.g., full core reactor depletion

12.2 billion tally region problem run with new FMESH capability



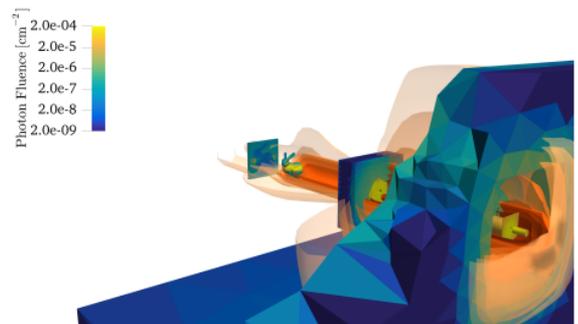
Introduction of the New File Formats (1)

- ▶ Introduced a flexible, Hierarchical Data Format (HDF5) to replace legacy binary and ASCII files
 - ▶ Permits natural organization
 - ▶ Parallel input/output can provide substantial performance
 - ▶ Parallel data transfer and distribution provides extensive mesh tally scalability
 - ▶ Intrinsic data-compression capability can provide 10–100× file size savings
 - ▶ Accessible via C, C++, Fortran, Python, Matlab, etc.
- ▶ The restart file (runtpe) now an HDF5 file in MCNP6.3.
- ▶ The ptrac file and unstructured mesh model / elemental edit files all now have an HDF5 option in MCNP6.3.

Introduction of the New File Formats (2)

- ▶ XDMF visualization through openly available software (e.g., ParaView, VisIt)
 - ▶ XML-formatted ASCII XDMF file can be easily interrogated via standard text editor
 - ▶ Permits many geometric representations: structured, unstructured, polyhedral, etc.
 - ▶ Low-overhead roadmap into the aforementioned HDF5 data file(s)
- ▶ HDF5+XDMF: directly accessible data files providing immediate visualization

Mesh Tally Results Visualized via ParaView



Simple Godiva Example (1)

Include last line if running with MCNP6.3, otherwise exclude.

Listing 1: godiva.mcnpinp

```
1 Godiva Solid Bare HEU sphere HEU-MET-FAST-001
2 1 100 4.7984e-02 -10 imp:n=1
3 2 0 10 imp:n=0
4
5 10 so 8.7407
6
7 kcode 5000 1.0 50 250
8 ksrc 0 0 0
9 c
10 m100 92234.00c 4.9184e-04
11 92235.00c 4.4994e-02
12 92238.00c 2.4984e-03
13 c
14 fmesh4:n geom=xyz origin=-10 -10 -10
15 imesh=10 iints=200
16 jmesh=10 jints=200
17 kmesh=10 kints=200
18 out=xdmf $ New out option in MCNP6.3
```

Simple Godiva Example (2)

MCNP6.2

- ▶ Execution command:

```
mcnp6 i=godiva.inp n=godiva. tasks 8
```

- ▶ Directory listing:

```
/
|- godiva.inp
|- godiva.o
|- godiva.r
|- godiva.s
|- godiva.msht
```

- ▶ ~ 78 M Histories / hr

MCNP6.3

- ▶ Execution command:

```
mcnp6 i=godiva.inp n=godiva. tasks 8
```

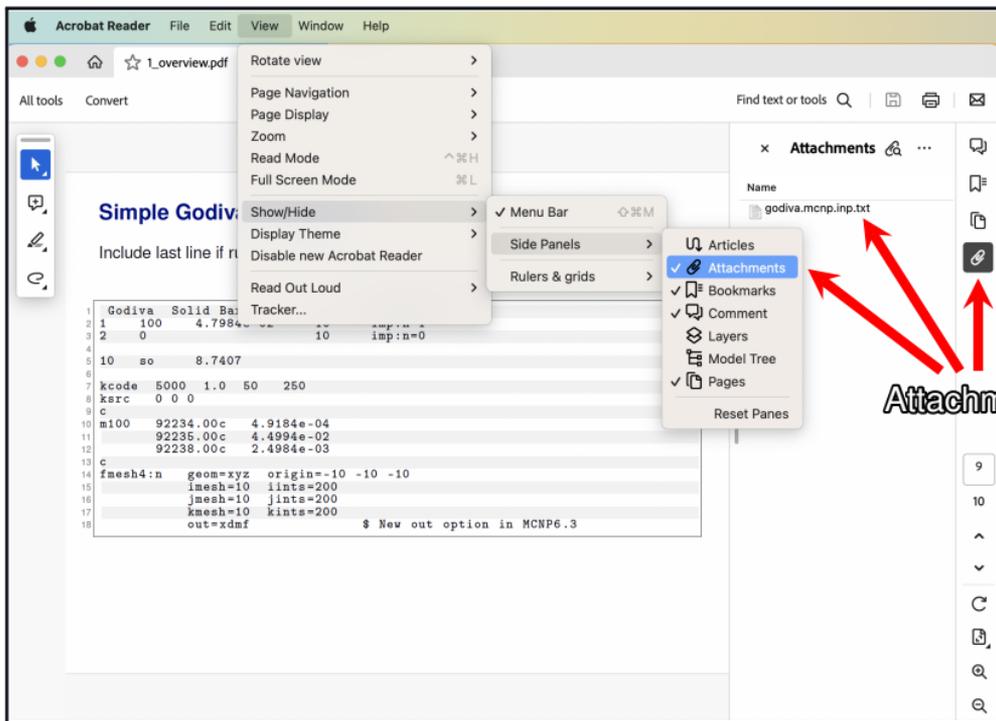
- ▶ Directory listing:

```
/
|- godiva.inp
|- godiva.o
|- godiva.r.h5
|- godiva.s
|- godiva.msht.xdmf
```

- ▶ **.h5** file extension for new HDF5 files
- ▶ **.xdmf** file extension for new XDMF files
- ▶ ~ 710 M Histories / hr

Extracting Inputs and Scripts from Workshop PDF

- ▶ Recommend downloading attachments and removing .txt file extension



Deimos Project

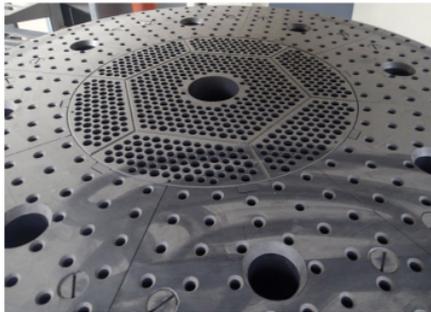
LANL R&D Project



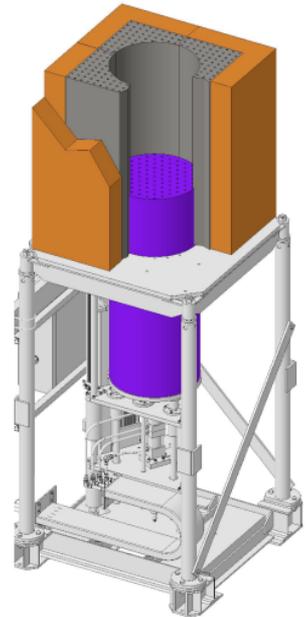
- ▶ Deimos is an ongoing LANL Laboratory Directed Research & Development (LDRD) project
- ▶ Motivation
 - ▶ Nuclear power reactor design, measurement, and validation for deep space missions
 - ▶ Small reactors need high assay low enriched uranium (HALEU) and/or particulate fuel
- ▶ Goals
 - ▶ Critical experiment design and measurement at Nuclear Criticality Experiments Research Center (NCERC)
 - ▶ Schedule to be executed in 2024
 - ▶ Materials research for optimal coatings and moderator materials
 - ▶ Modeling and simulation work
 - ▶ Development of new MCNP features (e.g. Delta tracking capability)
 - ▶ Multi-physics calculations of Deimos

Deimos Experiment

- ▶ Compact Nuclear Power Source (CNPS) hexagonal graphite matrix and HALEU fuel from a critical experiment in the 1980s
- ▶ The HALEU TRISO fuel used in CNPS is used in the Deimos experiment
- ▶ Due to the Comet assembly weight restriction at NCERC, a beryllium reflector is used in parts where some of the original graphite was located in CNPS design



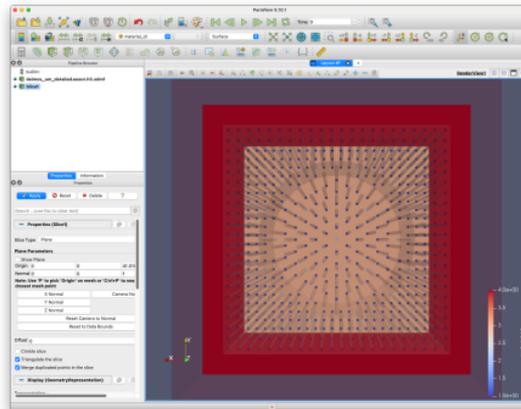
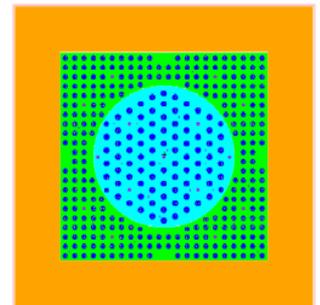
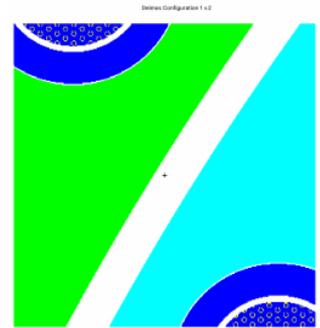
CNPS Graphite



MCNP Deimos Models

- ▶ CSG TRISO Particulate Model
 - ▶ Most accurate / most computationally expensive
- ▶ CSG Homogenized Model (not shown)
 - ▶ Can be accurate / least computationally expensive
- ▶ Unstructured Mesh Models
 - ▶ Accuracy depends on how well model preserves volume/mass and shape

MCNP CSG TRISO
Plotter Visualization



ParaView UM Visualization

Multi-physics Coupling



Coupling MCNP6.3 to Griffin

- ▶ MCNP6.3 calculates
 - ▶ Multigroup cross sections for Deimos reactor
 - ▶ Adjoint-weighted kinetics parameters
- ▶ Griffin, part of the NEAMS tools, calculates
 - ▶ Reactor dynamics, transient analysis
 - ▶ Heat conduction simulation
 - ▶ Includes deterministic (SN) transport
- ▶ For more discussion on multiphysics analysis techniques related to Deimos, see “Transient Modeling and Simulation for Deimos, an Advanced Reactor Experiment” in the Tuesday, 3:30-5:15 PM Experimental Reactor Physics technical session here at PHYSOR 2024.

MCNP[®]
MONTE CARLO N-PARTICLE



NEAMS



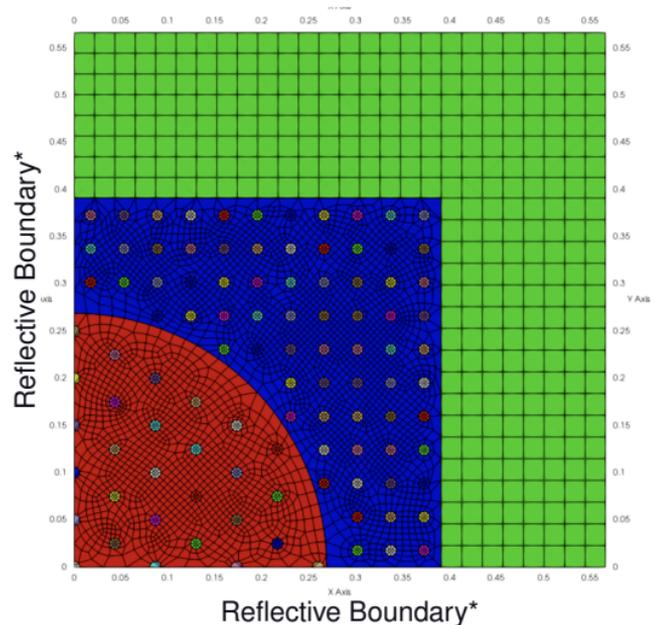
Griffin

Deimos Model Used in Workshop Exercises



- ▶ Using a quarter core (with proper reflecting boundary conditions) unstructured mesh model of a simplified Deimos reactor model
 - ▶ Includes both the Abaqus and MCNP input files
 - ▶ Abaqus mesh model generated using CUBIT
- ▶ Both Abaqus and MCNP inputs electronically attached to this PDF
 - ▶ `deimos_um_simple.abaq.inp`
 - ▶ `deimos_um_simple.mcnp.inp`

Representative Deimos Quarter-core UM Model
(different from workshop model)



Aside: MCNP UM Input File Construction

- ▶ Before venturing into a variety of new MCNP6.3 features
 - ▶ Open LA-UR-24-23422_mcnp_um_input_files.pdf document
 - ▶ “Constructing Input Files for MCNP Unstructured Mesh Simulations”

```
MCNP6_PHYSOR_2024_Workshop/  
  MCNP6_PHYSOR_2024_Workshop.pdf  
  outputs/  
  supplemental/  
    LA-UR-24-23422/  
      LA-UR-24-23422_mcnp_um_input_files.pdf  
    simple_deimos/  
    simple_thermal/
```

- ▶ Additional exercises and scripts provided with this supplemental workshop material that allow for easier MCNP UM input file pre-processing and setup

Unstructured Mesh and HDF5

Unstructured Mesh Overview

Overview of Unstructured Mesh in MCNP6.3

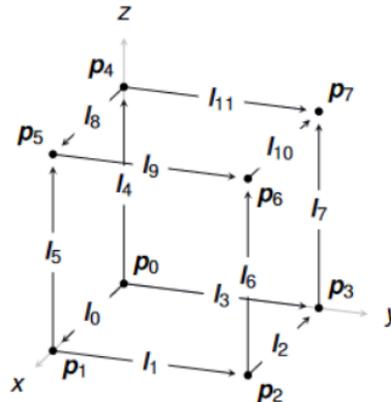
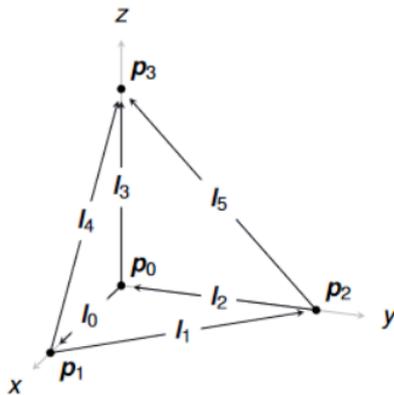
- ▶ Significant refactoring and modernization work done to clean-up existing and unused code paths
- ▶ Added mesh element quality assessment to the MCNP6.3 code
- ▶ Additional regression, verification, and validation testing with additional comment, warning, and fatal error messages
- ▶ New HDF5 input format option and HDF5+XDMF elemental edit output format option
 - ▶ The Abaqus input format can be used to produce the new HDF5 input format with the MCNP input (`mcnp6 i . . .`) processing option
 - ▶ The elemental edit HDF5+XDMF output builds on top of the input HDF5 format such that it can be used as input for an initial calculation or for a restart calculation
- ▶ Reduced mesh input processing times and memory consumption
- ▶ Deprecated legacy UM Fortran utilities in favor of new Python-based scripts

Mesh Elemental Quality Assessment

- ▶ Recent review of MCNP UM quality assessment and reporting work [8]
- ▶ Detailed discussion on calculating element quality [9, 10]
- ▶ Mesh-element quality is a thoroughly studied topic, e.g., [9, 11–13]
 - ▶ Also: International Meshing Roundtable [<https://imr.sandia.gov>]
 - ▶ Origins in mesh generation: calculate metrics and remesh toward optima
- ▶ The MCNP `um_pre_op` utility [14] has an “element checker”
 - ▶ Evaluates elemental Jacobian matrix determinant: go/no-go criterion
 - ▶ Must be run/inspected manually by a user; not implicitly in workflow
 - ▶ Invoked as, e.g., `um_pre_op -ec -o inp1034.ec.out um1034.abaq`
 - ▶ Looking for final line: “Total number of failed elements: 0”
- ▶ MCNP6.3 advances in mesh quality assessment
 - ▶ Quality metrics implemented for MCNP UM element types
 - ▶ Reproduced `um_pre_op` capability
 - ▶ Most-commonly used elements emphasized
 - ▶ Default enabled, opt-out available via embed card `elementchk` toggle

Approach to UM Quality Assessment (1)

- ▶ Start by focusing on linear tetrahedra and hexahedra
 - ▶ Most commonly used elements by MCNP UM practitioners
 - ▶ Most thoroughly studied elements by mesh-quality community
 - ▶ Identify metrics that can be characterized versus recommended ranges
 - ▶ All metrics exist on continua (i.e., no discrete and/or Boolean metrics)
 - ▶ Recommended ranges usually support FEA mesh-quality needs
 - ▶ This may lead to overly conservative recommended min/max values
 - ▶ Most consolidated source identified: Verdict library



Approach to UM Quality Assessment (2)

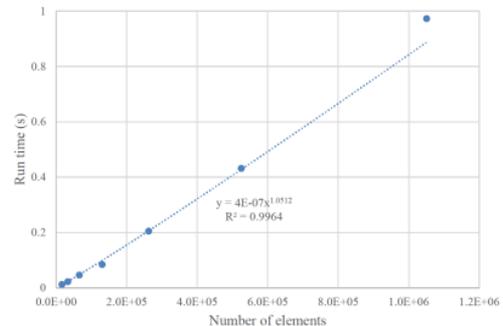
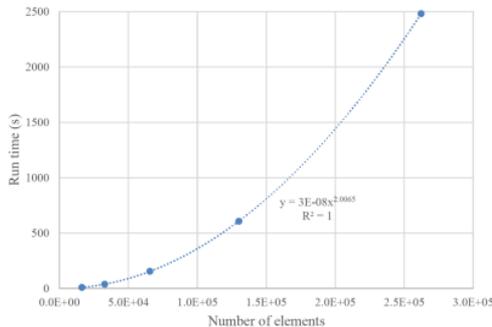
- ▶ Provide enough information to know if more information is needed
 - ▶ Provide information implicitly; don't require more workflow tasks
 - ▶ Ability to opt out in case mesh is "known to be good"
 - ▶ Elemental quality metric distribution information (a balancing act)
 - ▶ Number of elements inside / outside recommended ranges
 - ▶ Ample warning that more attention may be needed
 - ▶ Conservatism, until better guidance is available, is good
 - ▶ Education on how to learn more about mesh quality
- ▶ Warnings and Output Tables
 - ▶ Functional approach: warnings (below) and output tables
 - ▶ Warnings in standard output prompt user to inspect output tables
 - ▶ Output tables inform whether additional, external, study is necessary
 - ▶ (Conservative) warnings sent to standard output and output file...

```
warning. at least one element Maximum Edge Ratio outside recommended range.  
warning. at least one element Oddy outside recommended range.  
warning. at least one element Scaled Jacobian outside recommended range.  
warning. at least one element Shear outside recommended range.  
warning. at least one element Skew outside recommended range.  
warning. at least one element Stretch outside recommended range.  
warning. at least one element Taper outside recommended range.
```

Reduction in UM Initialization Computational Time

- ▶ For efficient particle tracking, the UM input processing includes a step to store neighbor-to-neighbor element information
- ▶ In MCNP6.2 this step is $O(N^2)$ and in MCNP6.3 this step is $O(N)$
- ▶ For large UM models, this is a significant improvement
 - ▶ For example, a city-sized UM model with 700k+ linear hexahedral element model had a 20x speedup in input processing time

UM Input Processing Times for MCNP6.2 (left) vs. MCNP6.3 (right)



Reduction in UM Memory Consumption

- ▶ In MCNP6.2, UM models with many parts that have a large variance in the number of elements can lead to excessive amounts of unnecessary memory allocated
- ▶ This can be especially important when running MPI parallel calculations with a large number of ranks
- ▶ The example below is using a UM model of the ORNL PCA Problem

UM Memory Usage for MCNP6.2 (Devel) vs. MCNP6.3 (Mem)

| Branch | Case | Mesh Init Time | Memory reported by top | RSS reported by top | Actual Savings | Expected Savings |
|--------|-------|----------------|------------------------|---------------------|----------------|------------------|
| Devel | 1x1 | 49s | 3.3 GB | 2.8 GB | 1.7 GB | 1.4 GB |
| Mem | 1x1 | 49s | 1.6 GB | 1.2 GB | | |
| Devel | 1x36 | 22s | 6.0 GB | 3.1 GB | 1.3 GB | 1.4GB |
| Mem | 1x36 | 22s | 4.7 GB | 1.9 GB | | |
| Devel | 144x1 | 130s | 416 GB | 332 GB | 192 GB | 201 GB |
| Mem | 144x1 | 37s | 224 GB | 144 GB | | |
| Devel | 72x1 | 70s | 199 GB | 161 GB | 94 GB | 101 GB |
| Mem | 72x1 | 31s | 105 GB | 70 GB | | |
| Devel | 72x2 | 70s | 193 GB | 162 GB | 119 GB | 101 GB |
| Mem | 72x2 | 28s | 74. GB | 27 GB | | |

MCNP Unstructured Mesh Geometry

- ▶ The MCNP unstructured mesh (UM) feature was implemented to allow MCNP to read in an Abaqus mesh file, generated by Abaqus, a multiphysics and finite element analysis code.
- ▶ The use of a mesh in MCNP allows a multiphysics code such as Abaqus to be used in tandem to perform neutronics analysis with MCNP and other analyses such as heat transfer with a separate code.
- ▶ Mesh quality is important
 - ▶ Mass/volume may not be preserved
 - ▶ Especially important for criticality calculations
- ▶ It is possible to generate a mesh, which reflects the geometry adequately for most purposes, and yet does not properly preserve mass and/or volume to the degree necessary for correct validation efforts using criticality safety benchmarks and shielding benchmarks.

Criticality Calculations using UM Geometry

Recent ICSBEP benchmarks modeled in MCNP6 UM [15].

HEU-MET-FAST-001: Godiva- bare, fast, spherical assembly of 94% ²³⁵U metal.

HEU-MET-FAST-007-037: HEU metal slabs moderated with polyethylene and reflected with polyethylene.

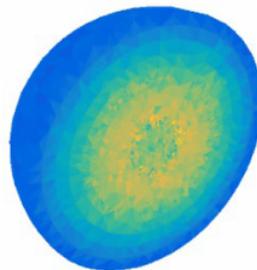
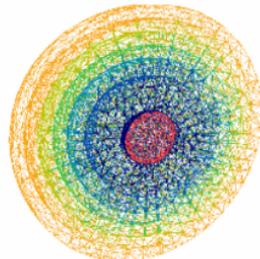
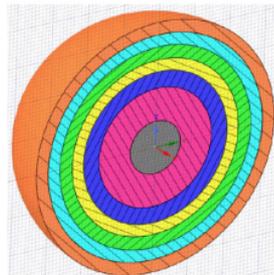
IEU-MET-FAST-007: Big Ten, a large, mixed-uranium cylindrical core with 10% average ²³⁵U enrichment, surrounded by a thick ²³⁸U reflector.

PU-MET-FAST-022: A bare, fast, spherical assembly of delta-phase plutonium metal, 98% ²³⁹Pu.

PU-SOL-THERM-001-001: A water-reflected 11.5-inch diameter sphere of plutonium nitrate solution.

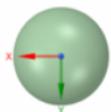
Goals:

1. Determine engineering best practices for mesh parameters
2. Compare CSG, UM and experiment results

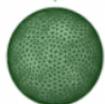


HEU-MET-FAST-001 Godiva

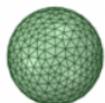
- UM meshing strategy: Element size control and mass correction



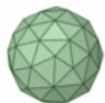
True Sphere (Radius = 8.741 cm)
 Volume = 2797.4 cm³ | Area=960.1 cm²
 SA/Vol = 0.343



Meshed Sphere (Max Edge Length = 1 cm)
 Volume = 2785.6 cm³ | Area = 957.6 cm²
 SA/Vol = 0.344



Meshed Sphere (Max Edge Length = 2 cm)
 Volume = 2746.6 cm³ | Area = 950.3 cm²
 SA/Vol = 0.346



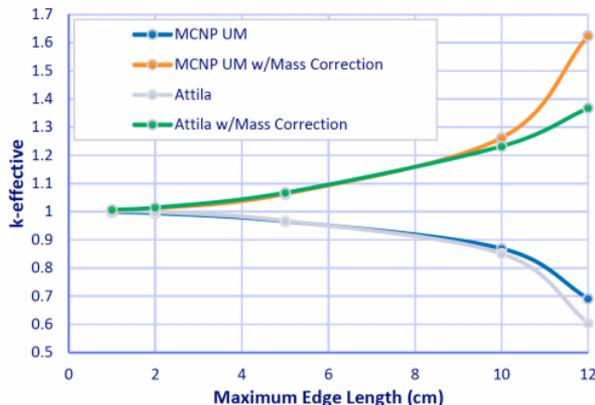
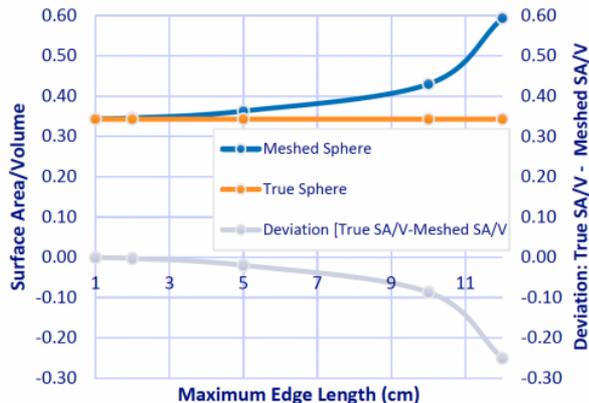
Meshed Sphere (Max Edge Length = 5 cm)
 Volume=2480.9 cm³ | Area=899.2 cm²
 SA/Vol = 0.363



Meshed Sphere (Max Edge Length = 10 cm)
 Volume = 1748.3 cm³ | Area = 752.2 cm²
 SA/Vol = 0.430

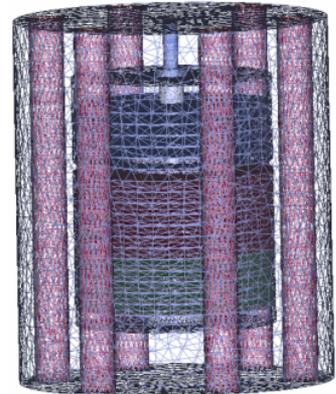
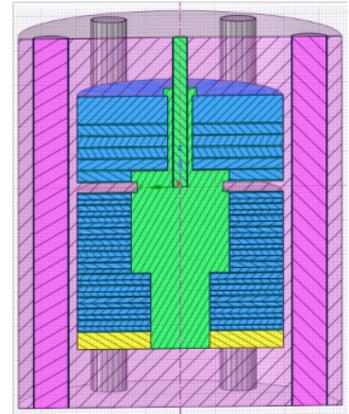
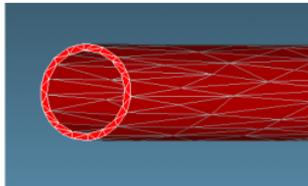
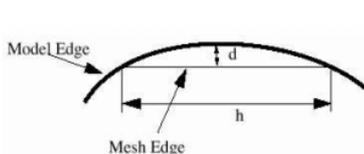


Meshed Sphere (Max Edge Length ≥ 12 cm)
 Volume = 890.5 cm³ | Area = 529.2 cm²
 SA/Vol = 0.594



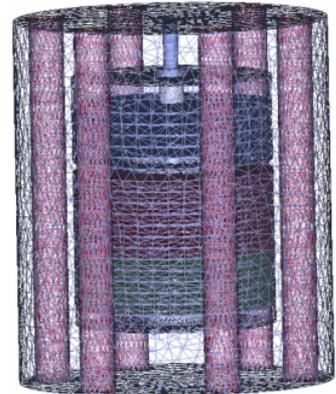
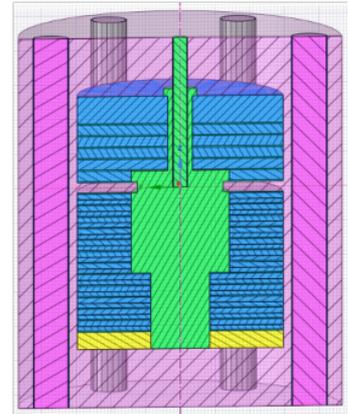
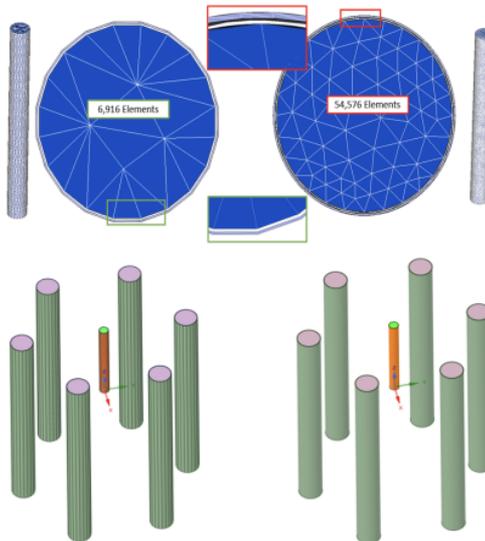
HEU-MET-FAST-007 Big Ten

- ▶ UM meshing strategy: Curvature Refinement
 - ▶ Allows the mesh to be automatically refined to match the curvature of the entities in the geometric model.
 - ▶ Mesh size is selected such that the distance of the model edge curve from the mesh edge (d) over the mesh edge length (h), $d/h < 0.5$
 - ▶ Useful values for d/h are typically in the range of 0.01 to 0.4 (smaller value=more refinement)



HEU-MET-FAST-007 Big Ten

- ▶ UM meshing strategy: Pre-faceting
 - ▶ 20-sided polygon vs. true cylinder
 - ▶ Radial tolerances strictly preserved
 - ▶ Mesh size substantially reduced



Critical Benchmark Validation Results

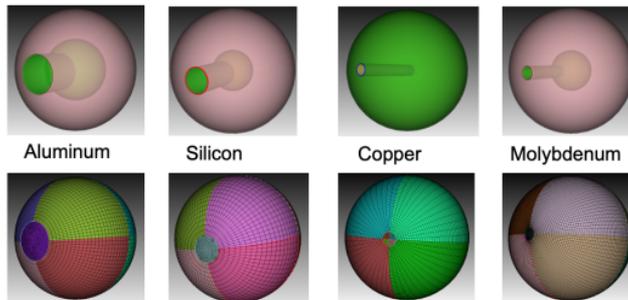
- ▶ Critical benchmarks with various materials and geometry have been studied for use with MCNP6 UM
- ▶ k-effective results that are within $\approx 1/2\%$ experimental values when due care is applied to mesh quality, in preserving both the mass and shape.
- ▶ Mesh within mass and/or volume tolerances 1-2%
 - ▶ Volume within 2%, SA/V within 1%

| Benchmark | Calc/Exp k -effective CSG | Calc/Exp k -effective UM |
|-----------------------------|-----------------------------|----------------------------|
| <i>HEU-MET-FAST-001</i> | 1.0000 | 0.9984 |
| <i>HEU-MET-FAST-007-037</i> | 1.0030 | 1.0029 |
| <i>IEU-MET-FAST-007</i> | 0.9999 | 0.9995 |
| <i>PU-MET-FAST-022</i> | 0.9983 | 0.9960 |
| <i>PU-SOL-THERM-001-001</i> | 1.0058 | 1.0030 |

UM Verification and Validation Summary

- ▶ In the past couple of years, a dedicated UM verification and validation effort has been ongoing
- ▶ Focused on CSG vs. UM comparisons to ultimately understand the mesh quality requirements to obtain comparable results to a proper CSG benchmark model
- ▶ Both criticality safety benchmarks and shielding benchmarks have been studied
- ▶ UM testing will continue and ultimately become part of our standard V&V efforts in the future

Oktaavian Shielding Benchmarks from SINBAD [16]



Aside: HDF5

Motivation

- ▶ MCNP output exists in many forms with various uses
 - ▶ Output file (`outp`)
 - ▶ Human-readable collection of relevant results
 - ▶ Data files (`EEOOUT`, `mctal`, `meshtal`, `runtpe`, `ptrac`, etc.)
 - ▶ Uniquely formatted to support post-processing (plots, tables, etc.)
- ▶ Several of the data files can be written as binary
 - ▶ Fast, efficient storage
 - ▶ Non-standard formats; hard to parse (needs custom applications)
- ▶ MCNP data files are migrating toward HDF5 to eliminate this downside
 - ▶ Mesh tally (`fmesh`), `ptrac`, `runtape`, and UM output are implemented
 - ▶ Other files undergoing development and implementation

Introduction to HDF5

- ▶ HDF: Hierarchical Data Format
- ▶ Developed by The HDF Group – Non-profit organization
 - ▶ Spun off from Nat'l Center for Supercomp. Appl. at Univ. of Illinois
 - ▶ Central authority to ensure quality and prevent fragmentation
- ▶ BSD-like license, freely available, portable, numerous APIs
 - ▶ Official APIs: C, C++, Fortran, Java
 - ▶ Unofficial APIs: Julia, Matlab, Mathematica, Perl, Python, R
- ▶ Developed with speed and scalability in mind
- ▶ Binary format, which requires a program or API to read the data
- ▶ Three major objects: groups, datasets, and attributes
 - Groups** Containers for datasets or other groups (like a file system)
 - Datasets** Homogeneous n-dimensional arrays
 - ▶ Can contain complex objects, e.g., images
 - Attributes** Can be added to either groups or datasets

History of HDF5

- ▶ 1987: work to develop all-encompassing hierarchical object-oriented file
- ▶ 1990 and 1992: NSF grants provided crucial funding
 - ▶ NSF wanted to harmonize netCDF and HDF formats
 - ▶ Drove improved V&V basis
 - ▶ NASA selected HDF as its standard data and information system
- ▶ 1996: major redesign: went to current group & dataset approach
- ▶ More information in videos at: <https://www.hdfgroup.org/about-us/>

- ▶ HDF4 is older but actively supported
- ▶ HDF5 is current (and actively supported)
 - ▶ Attempts to address some HDF4 limitations

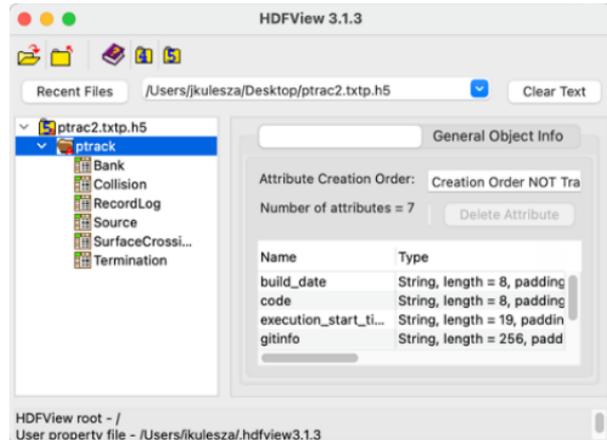
HDF5 is Binary, but Easily Interrogated

- ▶ Each group (e.g., /ptrack/) is like a filesystem directory
- ▶ Each dataset (e.g., Bank) is just an array of data that can be processed

h5ls and h5dump for terminal usage

```
h5ls -r ptrac2.txtp.h5
/                               Group
/ptrack                         Group
/ptrack/Bank                    Dataset {0/Inf}
/ptrack/Collision               Dataset {12920729/Inf}
/ptrack/RecordLog               Dataset {12923729/Inf}
/ptrack/Source                  Dataset {3000/Inf}
/ptrack/SurfaceCrossing         Dataset {0/Inf}
/ptrack/Termination             Dataset {0/Inf}
```

HDFView for graphical exploration



The screenshot shows the HDFView 3.1.3 application window. The left pane displays a tree view of the file structure, with the 'ptrack' group selected. The right pane shows the 'General Object Info' for the selected object, displaying the attribute creation order and a table of attributes.

| Name | Type |
|-----------------------|-----------------------------|
| build_date | String, length = 8, padding |
| code | String, length = 8, padding |
| execution_start_ti... | String, length = 19, paddin |
| gitinfo | String, length = 256, padd |

GUI Options: HDFView, HDF Compass, and ViTables

The image displays three overlapping GUI windows. The top-left window is HDFView 3.1.3, showing a file browser for 'pseudo_godiva_ter_hdf5.p.h5' with a tree view containing 'ptrack', 'Bank', 'Collision', 'RecordLog', 'Source', 'SurfaceCrossing', and 'Termination'. The top-right window is ViTables 3.0.2, showing a table of data from the 'RecordLog' dataset. The bottom window is HDF Compass, showing a tree view of the same file structure and a 'Query results' section.

| ngs | node | type | event_array_index |
|-----|------|------|-------------------|
| 0 | 1 | 6 | 5000 |
| 1 | 1 | 8 | 5000 |
| 2 | 1 | 10 | 5000 |
| 3 | 1 | 10 | 5000 |
| 4 | 1 | 14 | 5000 |
| 5 | 1 | 16 | 5000 |
| 6 | 1 | 18 | 5000 |
| 7 | 1 | 18 | 5000 |
| 8 | 1 | 18 | 5000 |

ViTables 3.0.2
Copyright (c) 2008-2019 Vicent Mas.
All rights reserved.
/Users/jkulesza/RUN/2020_07_30_AS_PTrac_Plot/pseudo_godiva_ter_hdf5.p.h5->/ptrack/RecordLog

Demo UM to HDF5 and Visualization in Paraview

Convert Abaqus Mesh File to HDF5

Step 1: Open MCNP Input File: `deimos_um_simple.mcnp.inp`

Step 2: Ensure `EMBED` card includes `hdf5file` keyword specifying the name of the HDF5-formatted model and elemental edit file:

Listing 2: UM Embed Card Input/Output File Information

```
1 embed1 meshgeo=abaqus
2         mgeoin=deimos_um_simple.abaq.inp
3         hdf5file=deimos_um_simple.um.h5
```

Step 3: Run MCNP in input-processing only mode:

```
> mcnp6 i i=deimos_um_simple.mcnp.inp
```

A new HDF5 (.h5 extension) and XDMF (.h5.xdmf extension) file should now be in the current working directory.

Mesh Information in HDF5 Format

The screenshot displays the HDFView 3.1.3 interface. On the left, a tree view shows the file structure of 'demos_um_simple.eeout.h5', with 'node_point' selected under the 'mesh' folder. The right pane shows 'Object Attribute Info' for the selected dataset, including its path, type, and dimensions. Below this, 'Dataset Dataspace and Datatype' and 'Miscellaneous Dataset Information' are displayed.

Object Attribute Info

| | |
|-------------|--|
| Name: | node_point |
| Path: | /unstructured_mesh/part-fuel0_j0_1P1/mesh/ |
| Type: | HDF5 Dataset |
| Object Ref: | 22920 |

Dataset Dataspace and Datatype

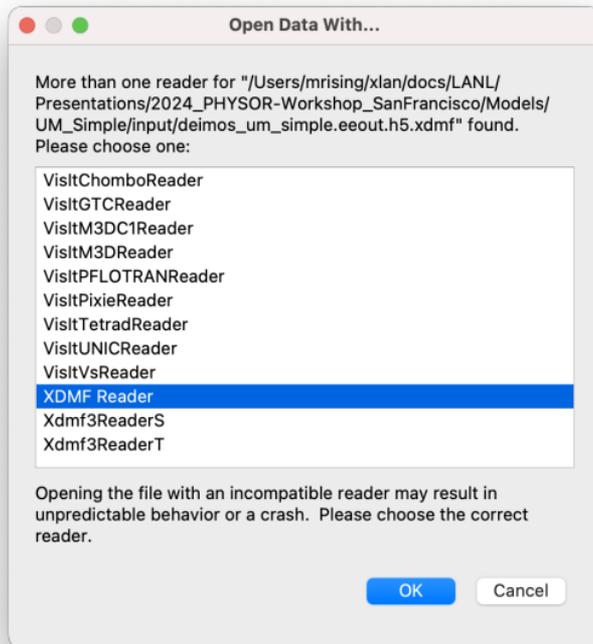
| | |
|------------------------|-----------------------|
| No. of Dimension(s): | 2 |
| Dimension Size(s): | 648 x 3 |
| Max Dimension Size(s): | 648 x 3 |
| Data Type: | 64-bit floating-point |

Miscellaneous Dataset Information

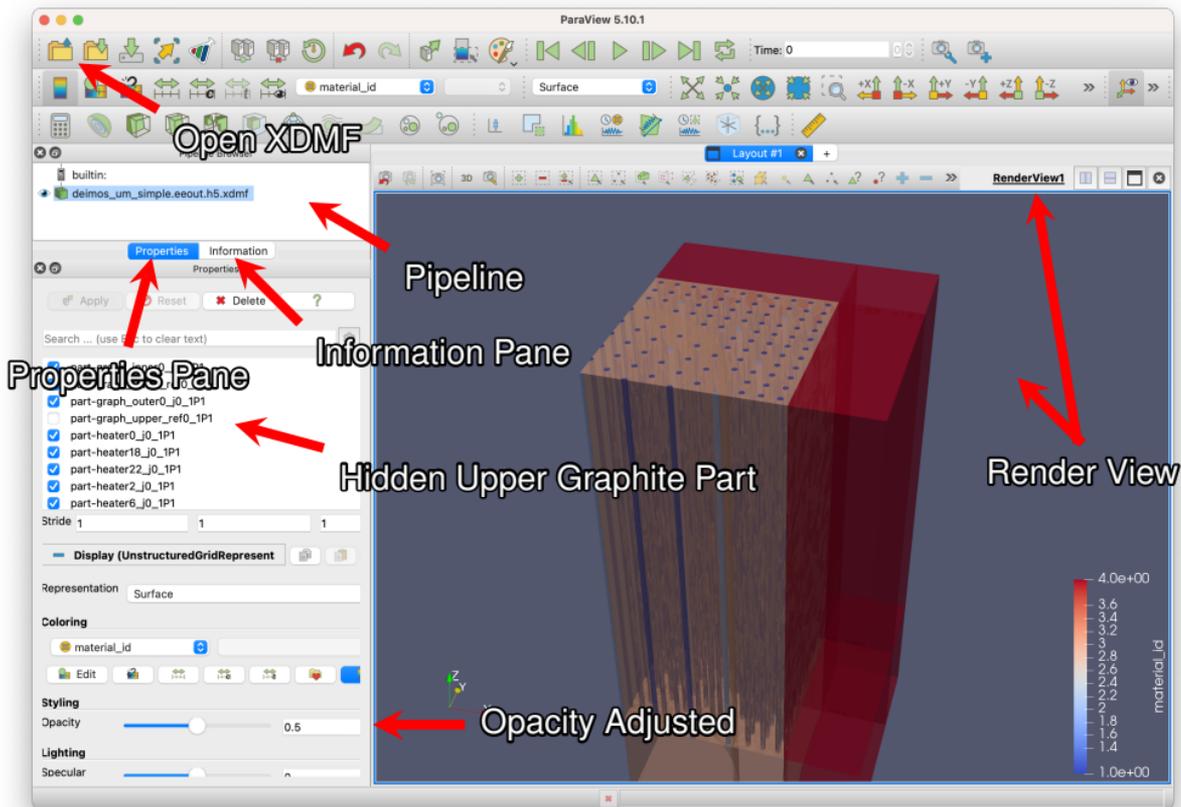
| | |
|-----------------|--|
| Storage Layout: | CHUNKED: 648 X 3 |
| Compression: | 2.496:1GZIP: level = 1 |
| Filters: | SHUFFLE: Nbytes = 8, GZIP |
| Storage: | SIZE: 6231, allocation time: Incremental |
| Fill value: | NONE |

HDFView root - /
User property file - /Users/mrising/hdfview3.1.3
node_id at /unstructured_mesh/part-fuel0_j0_1P1/mesh/ [demos_um_simple.eeout.h5 in /Users/mrising/lan/docs/LANL/Presentations/2024_PHYSOR-Workshop_5
node_point at /unstructured_mesh/part-fuel0_j0_1P1/mesh/ [demos_um_simple.eeout.h5 in /Users/mrising/lan/docs/LANL/Presentations/2024_PHYSOR-Workshop

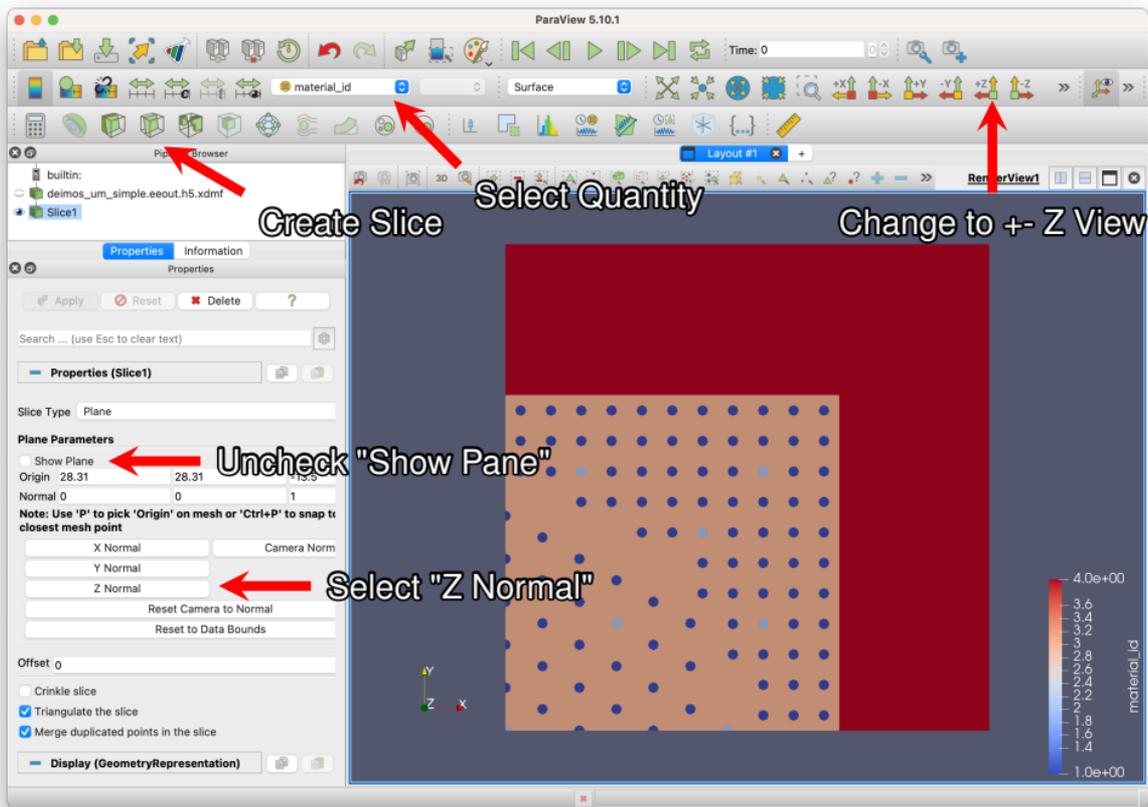
Open XDMF File in ParaView



View Mesh Model in ParaView



Add Slice Through Mesh Model in ParaView



Monte Carlo Simulation and Elemental Edits

Step 1: Open MCNP Input File: `deimos_um_simple.mcnp.inp`

Step 2: Ensure EMBEE card is included corresponding to EMBED card:

Listing 3: UM Embed Elemental Edit Card

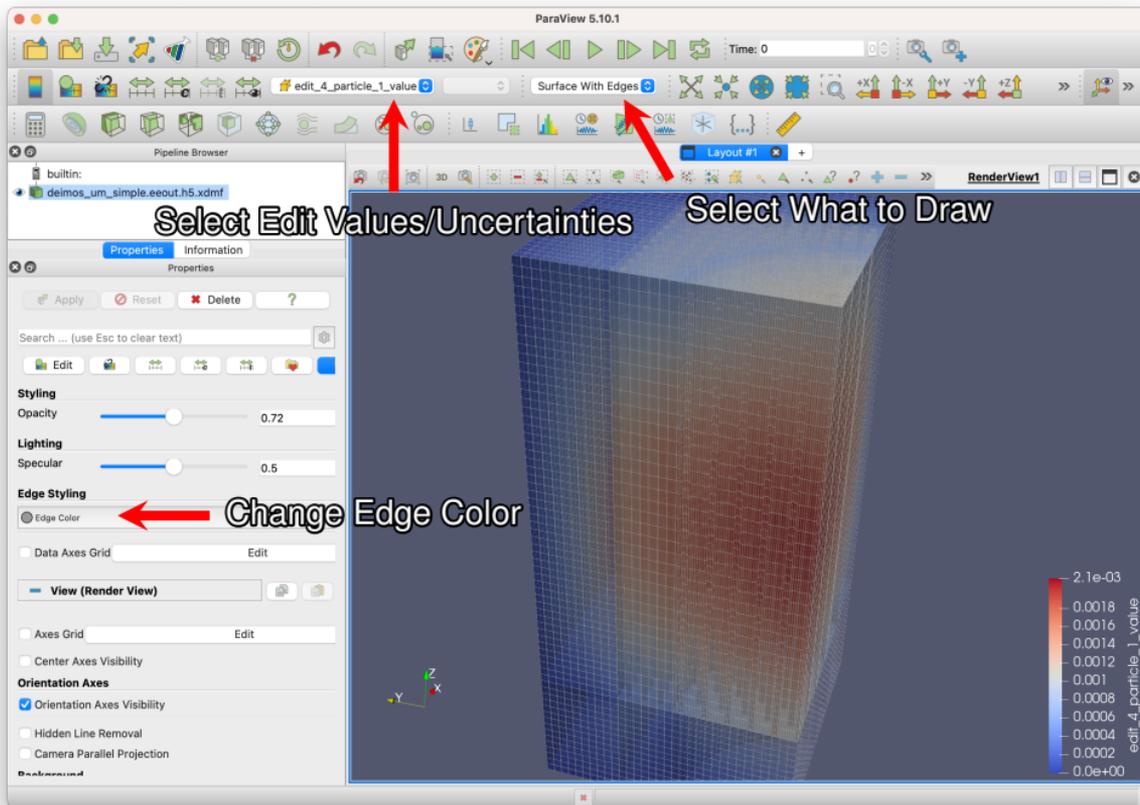
```
1 embee4:n embed=1 errors=yes
```

Step 3: Run MCNP in default input and cross section processing and Monte Carlo mode:

```
> mcnp6 i=deimos_um_simple.mcnp.inp tasks 8
```

The HDF5 (.h5 extension) and XDMF (.h5.xdmf extension) files now include the elemental edits.

View Elemental Edits in ParaView



Fission Matrix for Criticality Calculations

Fission Matrix Overview

Overview of the Integral Transport Equation for Multiplying System

- ▶ Consider Peierl's integral transport equation for a multiplying system:

$$S(\mathbf{p}) = \frac{1}{k} \int \mathcal{F}(\mathbf{p}', \mathbf{p}) S(\mathbf{p}') d\mathbf{p}', \quad \mathbf{p} \equiv (\mathbf{r}, \Omega, E)$$

- ▶ Where

\mathbf{p} is the particle phase space

$\mathcal{F}(\mathbf{p}', \mathbf{p})$ is the transport operator – including leakage, collisions, scattering, and multiplication

k is the effective multiplication factor

$S(\mathbf{p})$ is the fission source

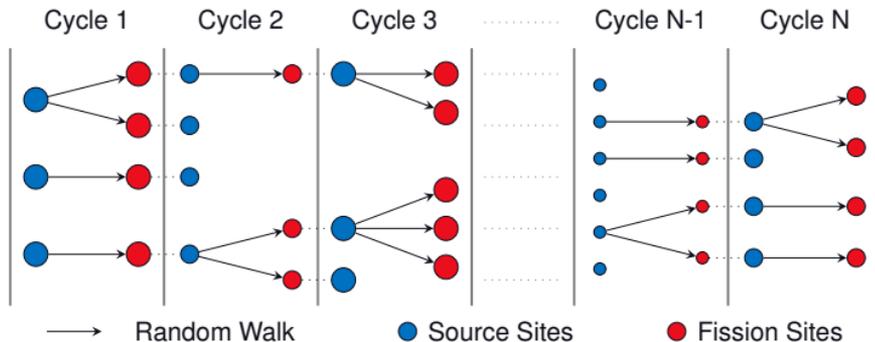
- ▶ Think of $\mathcal{F}(\mathbf{p}', \mathbf{p})$ as the operator that describes the transition of a neutron born from fission to the phase space of the next neutron born from an induced fission reaction

Power Iteration Method

- ▶ In MCNP, the power iteration technique is used to compute the final converged eigenvalue k and eigenfunction S of the system
 - ▶ Where the $k^{(i)}$ and $S^{(i)}$ are given for iteration index i

$$S^{(i+1)} = \frac{1}{k^{(i)}} \mathcal{F} S^{(i)}$$

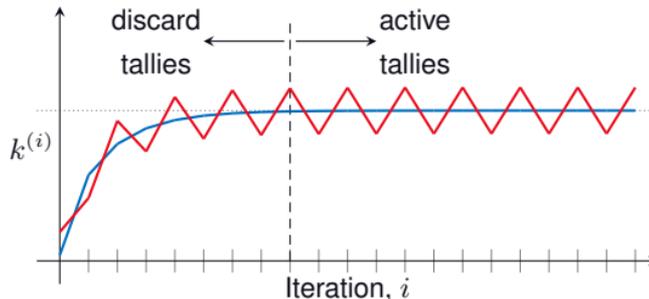
- ▶ The next iteration $k^{(i+1)}$ and $S^{(i+1)}$ are computed during the Monte Carlo random walk (\mathcal{F} operator)
- ▶ Renormalization of the fission source (indicated by resized source sites) at beginning of each cycle



Monte Carlo Criticality Calculations Concerns

For the fundamental eigenvalue, k :

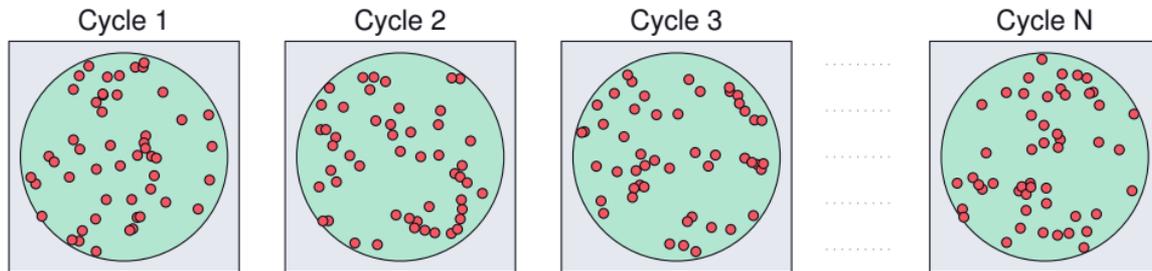
- ▶ Convergence
 - ▶ Assessing trends in k as a function of iteration, i , is straightforward
 - ▶ Use standard statistical testing to confirm converged eigenvalue
- ▶ Renormalization bias
 - ▶ Bias in k is non-conservative
 - ▶ Proportional to $1/N$ (N neutrons/cycle)
- ▶ Undersampling
 - ▶ Typically impacts eigenfunction S more than the eigenvalue k



Monte Carlo Criticality Calculations Concerns

For the fundamental eigenvector, S :

- ▶ Convergence
 - ▶ Local fission source S takes longer to converge than integral k
 - ▶ Bias in source shape related to initial source guess
- ▶ Renormalization bias
 - ▶ Bias in S appears as a tilt
 - ▶ Like k , reduction in bias with more N neutrons/cycle
- ▶ Undersampling
 - ▶ Not enough N neutrons/cycle to cover the physical phase space
 - ▶ Clustering of neutrons may occur



Shannon Entropy of the Fission Source

- ▶ A uniformly spaced cartesian mesh is used to compute the Shannon Entropy
- ▶ Note that in MCNP6.3, the reported Shannon Entropy values have been changed from the absolute Shannon Entropy to the relative Shannon Entropy

- ▶ In MCNP6.2 and before:

$$H(S) = - \sum_i^{N_{xyz}} p_j \ln(p_j),$$

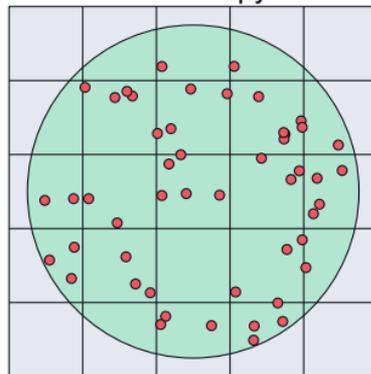
where $0 \leq H(S) \leq \ln(N_{xyz})$

- ▶ In MCNP6.3 and later:

$$\tilde{H}(S) = \frac{- \sum_i^{N_{xyz}} p_j \ln(p_j)}{\ln(N_{xyz})},$$

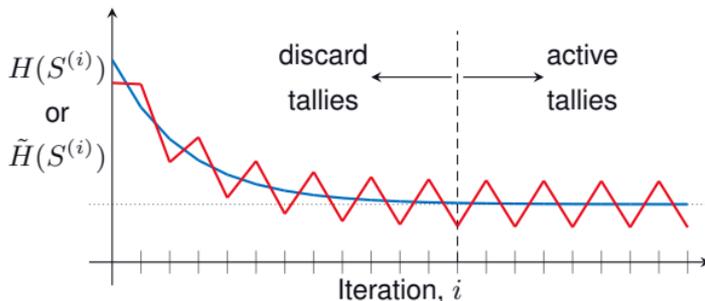
where $0 \leq \tilde{H}(S) \leq 1$

Shannon Entropy Mesh



N_{xyz} = number of bins

p_j = fraction of fissions in bin j



Fission Matrix in Practice

- ▶ An alternative, or supplemental approach to computing the eigenvalue k and eigenfunction S of the system is to determine the $\mathcal{F}(\mathbf{p}', \mathbf{p})$ operator and compute its eigenvalues and eigenfunctions.
- ▶ In practice this can be done by
 - ▶ Discretizing the $\mathcal{F}(\mathbf{p}', \mathbf{p})$ operator such that the matrix $\bar{\mathcal{F}}$ includes elements which describe the next-generation fission neutrons produced in region I for each fission neutron born in region J

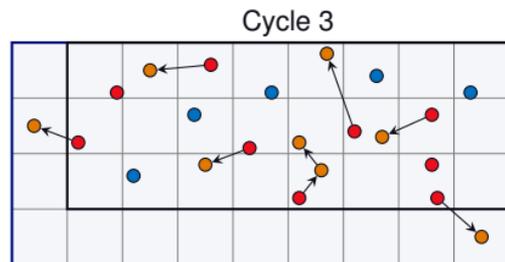
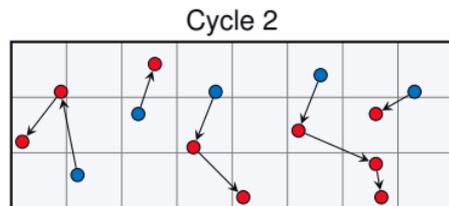
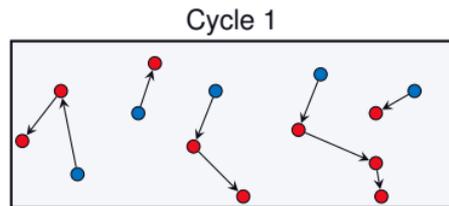
$$\mathcal{F}_{I,J} = \mathcal{F}(\mathbf{r}_J, \boldsymbol{\Omega}_J, E_J \rightarrow \mathbf{r}_I, \boldsymbol{\Omega}_I, E_I)$$

- ▶ Computing the $\bar{\mathcal{F}}$ matrix elements using Monte Carlo
- ▶ Computing the eigenvalues/eigenvectors with standard numerical solvers
- ▶ In MCNP, the fission matrix is discretized in the position \mathbf{r} phase space, integrating over all angle $\boldsymbol{\Omega}$ and energy E transitions, such that

$$\mathcal{F}_{I,J} = \mathcal{F}(\mathbf{r}_J \rightarrow \mathbf{r}_I)$$

Fission Matrix Spatial Meshing

- ▶ The same uniformly spaced cartesian mesh is used for the fission matrix as that used for the Shannon entropy
- ▶ The first cycle of the criticality calculation will start without a pre-defined mesh
- ▶ The code uses a physics-based approach to automatically size the mesh
 - ▶ In the first cycle the RMS distance from birth to next fission location, L_{fiss} , is used for the physical mesh spacing along the x , y , and z axes
- ▶ In later cycles, if fission locations occur outside the mesh boundary the mesh is automatically extended preserving mesh spacing

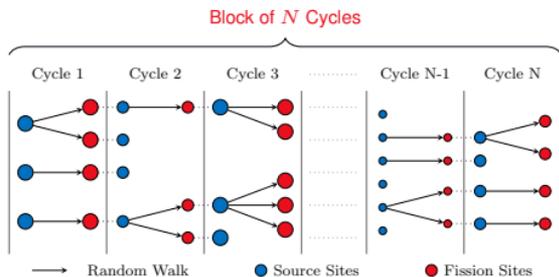


Reference Solution via the Fission Matrix

- ▶ Fission matrix and Shannon entropy
 - ▶ Sparse matrix storage: required for N_{xyz}^2 -sized fission matrix to capture $\mathcal{F}_{I,J} = \mathcal{F}(\mathbf{r}_J \rightarrow \mathbf{r}_I)$ phase space transitions
 - ▶ Shannon entropy on same spatial (N_{xyz} -sized) mesh
- ▶ Power iteration and the fission matrix
 - ▶ Power iteration (PI) used for final eigenvalue k_{PI} and eigenvector S_{PI}
 - ▶ Fission matrix provides **reference solution** for global fission distribution
 - ▶ Compute fission matrix (FM) eigenvalue k_{FM} and eigenvector S_{FM}
 - ▶ Compare to k_{PI} and S_{PI}
 - ▶ Compute and compare Shannon entropy, $H(S_{FM})$ and $H(S_{PI})$
- ▶ Reference solution can be used for improved power iteration convergence testing, assessment, and acceleration

k and S Fission Source Metrics

- ▶ End-of-cycle vs. end-of-block metrics
 - ▶ Information from individual cycle
 - ▶ Combined information from multiple cycles



End-of-cycle metrics from the power iteration solution

- ▶ Track length k_{PI} , collision k_{PI} , and absorption k_{PI} estimators
- ▶ Shannon entropy $H(S_{PI})$, and its' marginal source distribution in x , y , and z , given as $H_x(S_{PI})$, $H_y(S_{PI})$, and $H_z(S_{PI})$, respectively

End-of-block metrics from the power iteration solution

- ▶ Slope of the (above) end-of-cycle metrics across all cycles in block
- ▶ Cumulative source distribution $\sum S_{PI}$, and average Shannon entropy, $\langle H(S_{PI}) \rangle$, across all cycles in block

End-of-block metrics from the fission matrix

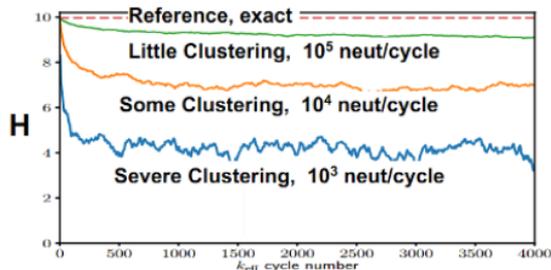
- ▶ k_{FM} , S_{FM} , and $H(S_{FM})$

Statistical Testing for Convergence

- ▶ Slope tests
 - ▶ The power iteration end-of-cycle metrics should not trend in any direction once convergence has been achieved
 - ▶ Compute least-squares slope and uncertainty within block; if the slope is sufficiently small for all metrics, slope tests pass
- ▶ Shannon entropy test
 - ▶ Passes if $\langle H(S_{PI}) \rangle$ within 1% of $H(S_{FM})$
 - ▶ Sensitive to both convergence and population size
- ▶ Source $\sum S_{PI}$ and S_{FM} distribution tests
 - ▶ Kolmogorov-Smirnov test at 95% level
 - ▶ Chi-square 2-point test at 95% level
 - ▶ Relative entropy (Kullback-Liebler discrepancy) test at 95% level
 - ▶ Sensitive to both convergence and population size

Statistical Testing for Population Size

- ▶ Source renormalization leads to:
 - ▶ Bias in source shape
 - ▶ Too low in high-importance regions
 - ▶ Too high in low-importance regions
 - ▶ Undersampling/clustering
- ▶ Population size tests:
 - ▶ Shannon entropy test
 - ▶ Passes if $\langle H(S_{PI}) \rangle$ within 1% of $H(S_{FM})$
 - ▶ Fission source distribution test
 - ▶ Compare $\sum S_{PI}$ and S_{FM}
 - ▶ Relative entropy (Kullback-Liebler discrepancy) test at 95% level
- ▶ If these two tests pass, there is strong evidence that
 - ▶ Source convergence has been achieved
 - ▶ Population size is large enough such that source renormalization bias is negligible



Source Convergence Acceleration

- ▶ At end of each cycle
 - ▶ S_{FM} reference eigenvector is available from end of previous block of cycles
 - ▶ S_{PI} power iteration neutron source distribution at end of current cycle
- ▶ During inactive cycles, can optionally use S_{FM}/S_{PI} for importance sampling of fission source for each mesh region
 - ▶ Pushes neutron distribution toward S_{FM} reference eigenvector
 - ▶ Can reduce inactive cycles needed by a factor of 2–20
- ▶ In MCNP during the inactive cycles,
 - ▶ Iterate until Shannon entropy and fission matrix meshes are stable
 - ▶ Begin first block of cycles to compute reference source distribution from fission matrix
 - ▶ Once convergence has been reached the source importance sampling is disabled during active cycles

Primary Fission Matrix Options

- ▶ For the fission matrix convergence testing and acceleration methods, the KOPTS card includes some added capabilities

FMAT=yes/no Turns on the calculation of the fission matrix and uses this information to compute reference fission distribution metrics (e.g., Shannon entropy) to determine when the actual fission distribution reaches convergence.

FMATCONVRG=yes/no Uses the reference fission-matrix fission distribution to automatically determine how many power iteration cycles must be discarded before active cycles begin. **In this case, the user-specified number of inactive cycles is ignored.**

FMATACCEL=yes/no Uses importance sampling based on the reference fission-matrix distribution to accelerate the convergence of the actual fission distribution. **This can reduce the number of inactive cycles needed in a simulation while the acceleration is applied prior to the active cycles.**

Fission Matrix Interrogation

Fission Matrix Interrogation in the HDF5 Runtime

The screenshot displays the HDFView 3.1.3 application window. The title bar reads "HDFView 3.1.3". The address bar shows the file path: "/Users/jkulesza/GIT/jkulesza_latex_files/2022_06_ANS_NCSD_Workshop/pca/runtpe.h5". The left sidebar shows a tree view of the file structure, with "data" selected under "results/fission_mat...". The main panel, titled "Object Attribute Info", displays the following information:

| | |
|-------------|--------------------------|
| Name: | data |
| Path: | /results/fission_matrix/ |
| Type: | HDF5 Dataset |
| Object Ref: | 52585852 |

Dataset Dataspace and Datatype

| | |
|------------------------|-----------------------|
| No. of Dimension(s): | 1 |
| Dimension Size(s): | 81836 |
| Max Dimension Size(s): | 81836 |
| Data Type: | 64-bit floating-point |

Show Data with Options

Miscellaneous Dataset Information

| | |
|-----------------|--|
| Storage Layout: | CHUNKED: 81836 |
| Compression: | 1,168-1GZIP: level = 1 |
| Filters: | SHUFFLE: Nbytes = 8, GZIP |
| Storage: | SIZE: 560362, allocation time: Incremental |
| Fill value: | NONE |

HDFView root - /
User property file - /Users/jkulesza/.hdfview3.1.3
data at /results/fission_matrix/ [runtpe.h5 in /Users/jkulesza/GIT/jkulesza_latex_files/2022_06_ANS_NCSD_Workshop/pca] [dims0, start0, soon

An Aside on Compressed Sparse-row Storage

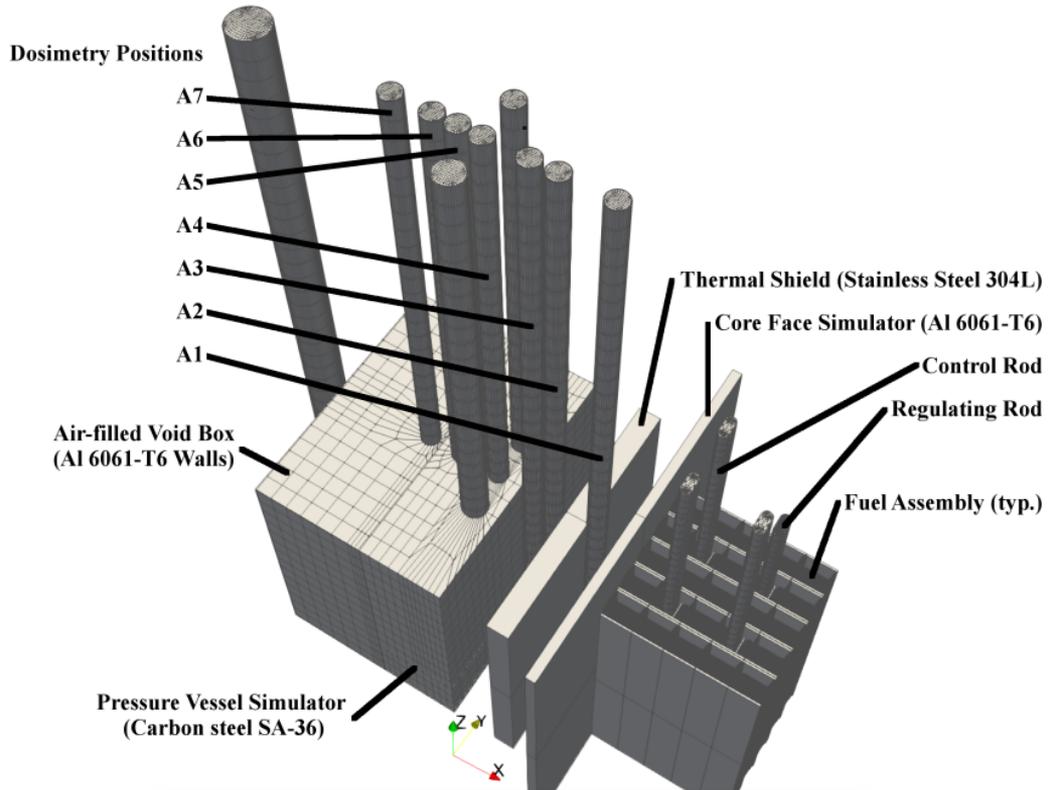
$$A = \begin{pmatrix} 7.5 & 2.9 & 2.8 & 2.7 & 0 & 0 \\ 6.8 & 5.7 & 3.8 & 0 & 0 & 0 \\ 2.4 & 6.2 & 3.2 & 0 & 0 & 0 \\ 9.7 & 0 & 0 & 2.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5.8 & 5.0 \\ 0 & 0 & 0 & 0 & 6.6 & 8.1 \end{pmatrix}$$

rowptr: (0 4 7 10 12 14 16)

colind: (0 1 2 3 0 1 2 0 1 2 0 3 4 5 4 5)

val: (7.5 2.9 2.8 2.7 6.8 5.7 3.8 2.4 6.2 3.2 9.7 2.3 5.8 5.0 6.6 8.1)

Fission Matrix Interrogation Example [17–19]



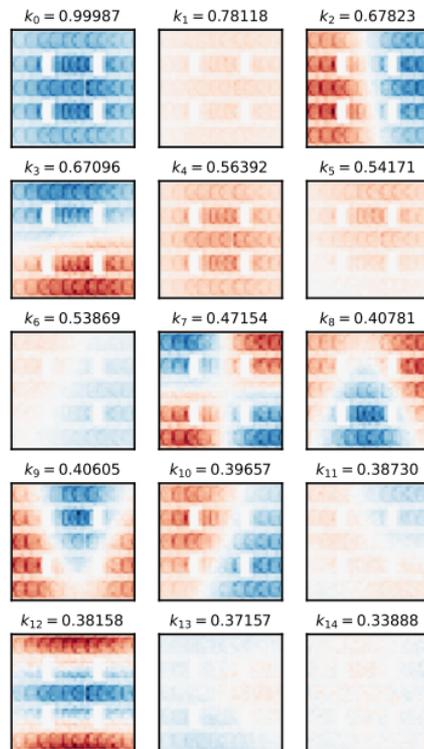
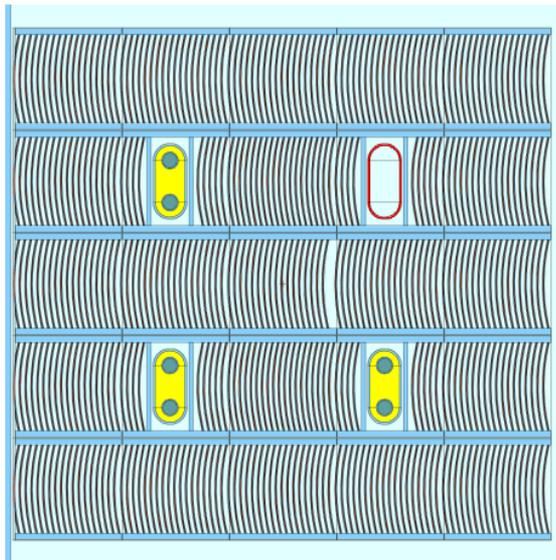
Fission Matrix Input Example

- ▶ Direct control can create a “preferred” fission-matrix mesh
- ▶ ORNL PCA example:

```
1 kcode 250000 1.000 50 250
2 kopts fmat=yes
3 hsrc 40 -20.41 20.25
4 40 -19.275 19.275
5 75 -36.35375 36.35375
```

- ▶ Beware of file size: the HDF5 runtapes can exceed several gigabytes

Fission Matrix Visualization Example



Demo Fission Matrix Calculation and Visualization

Modifications to MCNP Input File

Step 1: Open MCNP Input File: `deimos_um_simple.mcnp.inp`

Step 2: Insert fission matrix options:

Listing 4: Fission Matrix Options

```
1 kopts      fmat=yes  
2           fmatconvrq=yes  
3           fmataccel=yes
```

Step 3: Run MCNP (with threads if possible):

```
> mcnp6 i=deimos_um_simple.mcnp.inp tasks 8
```

Fission Matrix Diagnostic Information

Listing 5: Shannon Entropy and Fission Matrix Mesh Information

```
1 comment.  
2 comment. -----  
3 comment. The MESH (adaptive, axis-aligned, cartesian) to be used for computing  
4 comment. Shannon entropy, fission-matrix tallies (if used), and source  
5 comment. convergence checking is initially defined by:  
6 comment.     max mesh spacing for automesh = 2.8216E+01  
7 comment.  
8 comment.     total mesh cells = 45  
9 comment.  
10 comment.     Xbins=   3     Xmin= 3.4730E-03     Xmax= 3.7887E+01     dx= 1.2628E+01  
11 comment.     Ybins=   3     Ymin= 3.0571E-04     Ymax= 3.7877E+01     dy= 1.2625E+01  
12 comment.     Zbins=   5     Zmin=-4.7496E+01     Zmax= 4.7487E+01     dz= 1.8997E+01  
13 comment.  
14 comment. the mesh will be automatically extended if necessary,  
15 comment. preserving the original mesh cells and spacing.  
16 comment. -----  
17 comment.
```

Fission Matrix Diagnostic Information

Listing 6: Fission Matrix Information

```
1 comment.  
2 comment. -----  
3 comment.  FISSON MATRIX WILL BE COMPUTED to estimate dominance ratio,  
4 comment.    based on fission sites only - not flights or collisions  
5 comment.  
6 comment.  The mesh for the fission matrix is the same as the entropy mesh,  
7 comment.    using 45 mesh bins for tallying fission neutrons  
8 comment.  
9 comment.    Fission matrix mesh will be extended if  
10 comment.      any fission sites are found outside this mesh.  
11 comment.  
12 comment.    Fission matrix tallies will be reset after cycle 1  
13 comment.    Fission matrix eigenfunction will be found every 10 cycles.  
14 comment.  
15 comment.    Fission matrix dimensions: 45 x 45  
16 comment.  
17 comment.    Compressed-row-storage is used for the fission matrix.  
18 comment.    max number of nonzero entries: 2025  
19 comment.
```

Fission Matrix Diagnostic Information

Listing 7: Fission Matrix Convergence and Acceleration Information

```
1 comment.  
2 comment. FMATCONVRG option is being used.  
3 comment. Statistical tests on the neutron & fiss-matrix distributions  
4 comment. will be used to determine convergence & begin active cycles.  
5 comment. The 3rd entry on the KCODE card may be ignored.  
6 comment.  
7 comment. Targets for statistical tests:  
8 comment. h_slope: < 0.95 conf level, or < 0.0001  
9 comment. k_slope: < 0.95 conf level, or < 0.0001  
10 comment. distribs: < 0.95 conf level, h_diff: < 0.01  
11 comment.  
12 comment.  
13 comment. FMATACCEL option is being used.  
14 comment. Fission matrix will be used to ACCELERATE source convergence  
15 comment. of the neutron distribution during inactive cycles.  
16 comment. Importance-factor-limits: min= 0.20, max= 5.00  
17 comment.  
18 comment.  
19 comment. Keff of the fission matrix solution can be plotted using "kcode 4".  
20 comment. Hsrc of the fission matrix solution can be plotted using "kcode 5".  
21 comment. -----  
22 comment.
```

Fission Matrix and Power Iteration Results

- ▶ Both fission matrix and power iteration k results provided
- ▶ Always use the power iteration k for production calculations

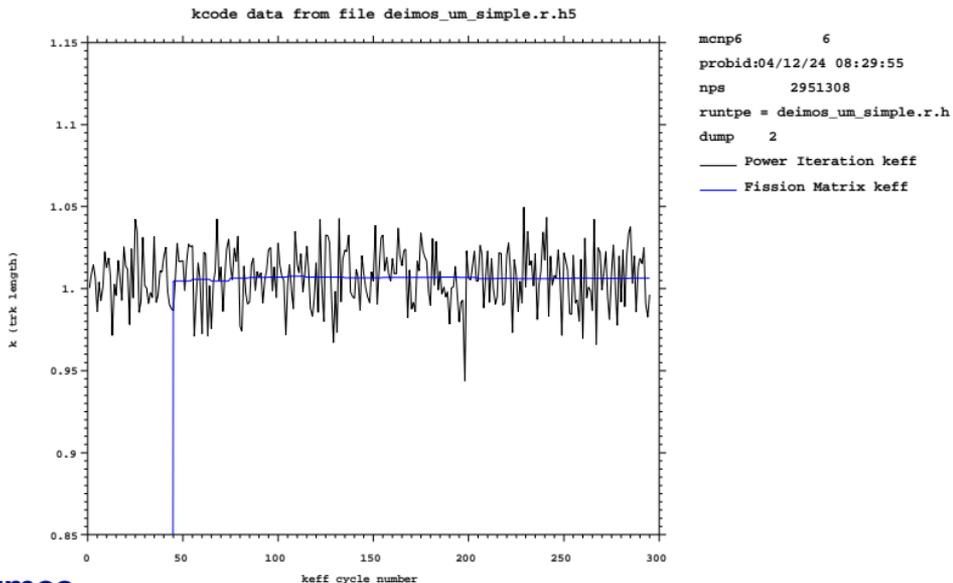
Listing 8: Fission Matrix and Power Iteration Results

```
1 . . . . .
2 Approximate results using the fission-matrix:
3   fission matrix based on last 294 cycles
4   fission matrix size = 175 x 175
5   fission matrix based on fission sites only, not flights or collisions.
6   fission matrix accuracy depends on mesh size.
7
8   k-effective      = 1.00639
9   dominance ratio  = 0.59961
10 . . . . .
11
12
13 final k(col/abs/trk len) = 1.006662      std dev = 0.000665
```

Fission Matrix and Power Iteration Results

- ▶ Plot k -effective as a function of iteration:

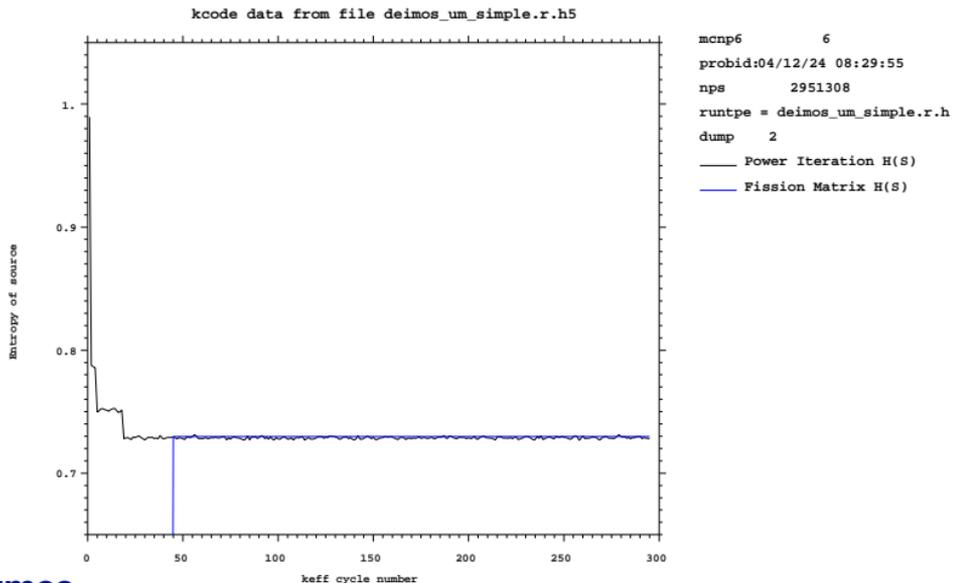
```
> mcnp6 r=deimos_um_simple.mcnpr.h5 z
mcnp6> kcode 3 label "Power Iteration keff" coplot
mcnp6> kcode 4 label "Fission Matrix keff"
```



Fission Matrix and Power Iteration Results

- ▶ Plot $H(S)$ as a function of iteration:

```
> mcnp6 r=deimos_um_simple.mcnp.r.h5 z
mcplot> kcode 6 label "Power Iteration H(S)" coplot
mcplot> kcode 5 label "Fission Matrix H(S)"
```



Extract Fission Matrix, Calculate and Sort Eigenvalues & Eigenvectors, and Plot

- ▶ The fission matrix is accessible through the runtape
- ▶ Python script is available to extract, calculate, and plot eigenvalue and eigenvectors:

Listing 9: Header of fission_matrix_process.py

```
1 #!/usr/bin/env python3
2
3 import h5py
4 import matplotlib.pyplot as plt
5 import numpy as np
6 import scipy.sparse as sparse
7 import scipy.sparse.linalg as sla
8 from matplotlib.cm import get_cmap
9
10 SUPPORTED_RUNTAPE = [1, 0, 0]
```

Extract Mesh Data

Listing 10: Matrix Extraction in fission_matrix_process.py

```
1 def extract_fmat(runtape):
2     """Returns the last saved fission matrix as a scipy.sparse.csr_matrix"""
3     with h5py.File(runtape, "r") as handle:
4         # Check runtape version
5         version_file = handle["config_control"].attrs["version_file"]
6         if any(SUPPORTED_RUNTAPE != version_file):
7             print("Possibly incompatible runtape detected.")
8
9         fmat = handle["results/fission_matrix"]
10
11        n_dim = fmat["n"][(0)]
12        indices = fmat["indices"][:]
13        indptr = fmat["indptr"][:]
14        data = fmat["data"][:]
15
16        n_xyz = fmat["n_xyz"][:]
17        delta_xyz = fmat["delta_xyz"][:]
18        origin = fmat["origin"][:]
19
20    return ( sparse.csr_matrix((data, indices, indptr), shape=(n_dim, n_dim)),
21            n_xyz, delta_xyz, origin )
```

Calculate and Sort Eigenvalues & Eigenvectors

Listing 11: Eigenvalue/Eigenvector Calculation in fission_matrix_process.py

```
1 def plot_eigs(mat, n_xyz, n_tot=6, n_col=2):
2     """Retrieve eigenvalues/vectors, sort, reshape to 3-d object, and plot."""
3     eigenvalues, eigenvectors = sla.eigs(mat, k=n_tot)
4
5     # Clean up and sort the eigenvectors.
6     sorted_eigvals = []
7     sorted_eigvecs = []
8     for i in np.argsort(-np.abs(eigenvalues)):
9         val = eigenvalues[i]
10        val = np.real(val) if np.real(val) == val else val
11        sorted_eigvals.append(val)
12        vec = np.real(eigenvectors[:, i]).reshape(n_xyz[:-1]).transpose()
13        if len(sorted_eigvals) == 1:
14            vec = np.abs(vec)
15        sorted_eigvecs.append(vec)
```

Plot Eigenvalues & Eigenvectors

Listing 12: Eigenvalue/Eigenvector Plotting in fission_matrix_process.py

```
1 # Create example plot grid.
2 cmap = get_cmap("RdBu")
3 fig, ax = plt.subplots(int(n_tot / n_col), n_col, figsize=(3, 3 * 1.75))
4 aspect_ratio_z = delta_xyz[1] / delta_xyz[0]
5 for i in range(int(n_tot / n_col)):
6     for j in range(n_col):
7         k = i * (n_col) + j
8         data = sorted_eigvecs[k]
9         cbar_scaling = max(np.max(data), -np.max(-data))
10        ax[i, j].matshow(
11            data[:, :, int(n_xyz[2] / 2)].transpose(),
12            cmap=cmap, origin="lower", aspect=aspect_ratio_z,
13            vmin=-cbar_scaling, vmax=cbar_scaling,
14        )
15        ax[i, j].set_xticks([]); ax[i, j].set_yticks([])
16        ax[i, j].set_title(
17            f"$k_{{k}}$={{sorted_eigvals[k]:.5f}}$",
18            y=1.0, pad=3, fontsize=6,
19        )
20 plt.savefig("eigenvalues.pdf", bbox_inches="tight")
```

Extracting Example Fission Matrix and Eigenmodes

Step 1: Open the Python script: `fission_matrix_process.py`

Step 2: Update plot and runtape file names at end of script:

Listing 13: Hardcoded Script File Names

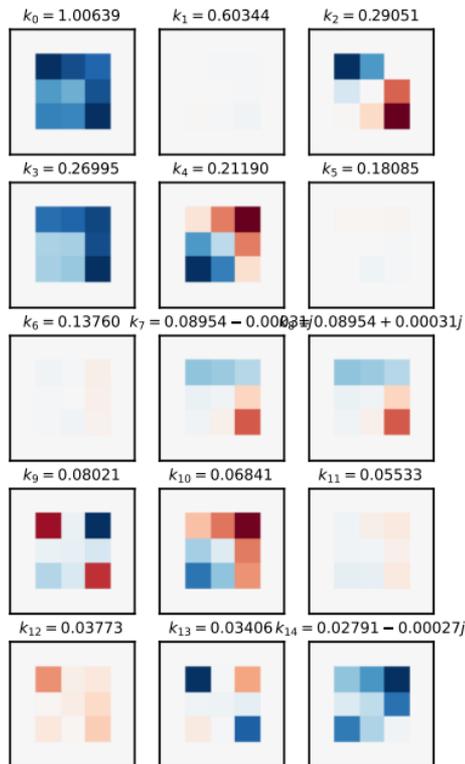
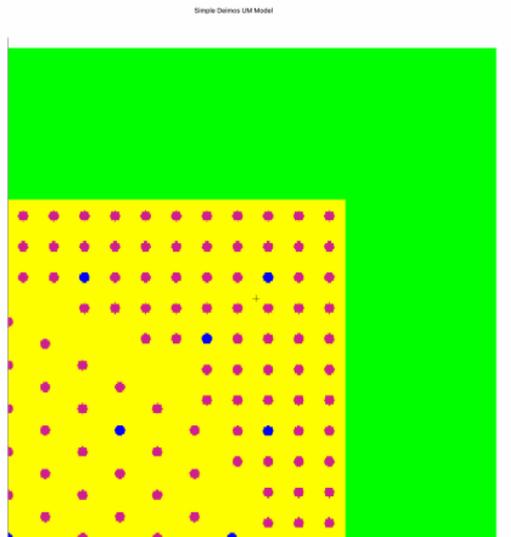
```
1 plt.savefig("eigenvalues.pdf", bbox_inches="tight")
2
3 mat, n_xyz, delta_xyz, origin = extract_fmat("deimos_um_simple.r.h5")
4 plot_eigs(mat, n_xyz, 15, 3)
```

Step 3: Run Python script:

```
> python3 fission_matrix_process.py
```

Fission Matrix Visualization Example

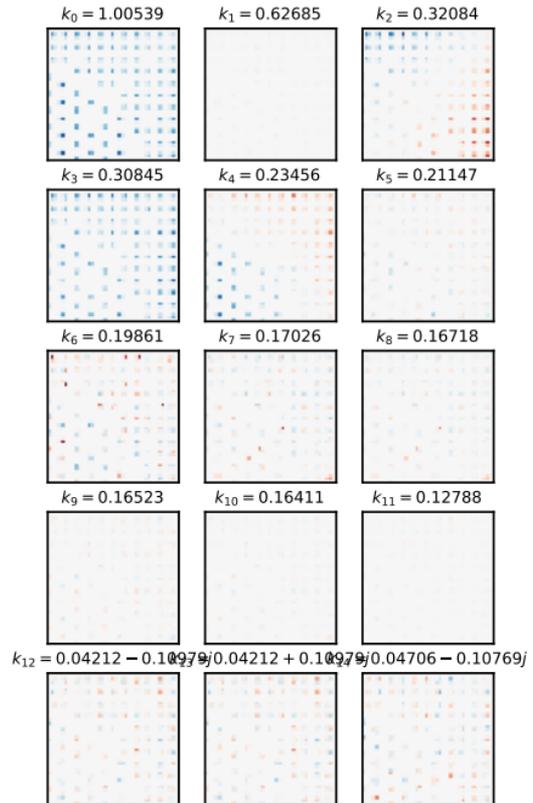
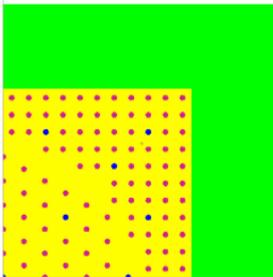
- ▶ Default 3x3x5 fission matrix mesh



Fission Matrix Visualization Example

- For a more refined fission matrix solution, make Shannon entropy / fission matrix mesh finer using the HSRC card:

| | | | | | |
|---|-------|---------------|-----|-----|-----|
| 1 | kopts | fmat=yes | | | |
| 2 | | fmatconvr=yes | | | |
| 3 | | fmataccel=yes | | | |
| 4 | hsrc | 40 | 0 | 40 | |
| 5 | | 40 | 0 | 40 | |
| 6 | | 100 | -50 | 50 | |
| 7 | kcode | 10000 | 1.0 | 100 | 350 |



New Tally Features

FMESH Upgrades and HDF5+XDMF

Introduction

Since MCNP6.2, FMESH has undergone a substantial revision in capabilities.

- ▶ The default FMESH configuration was heavily optimized.
- ▶ 3 new tally backends were added for various needs (mainly, larger tallies).
- ▶ MESHTAL is deprecated and replaced with an HDF5 + XDMF output format.
(This format allows for much faster and easier postprocessing and analysis)

Tally Algorithms

It started as a side project - can we use MPI remote memory access to scale further than ever before?

Implemented 4 algorithms:

History Basic history statistics without optimization.

Fast History A tuned version of 6.2's FMESH algorithm, tracks changed indices to reduce memory bandwidth usage.

Batch Threads share a tally array, so memory usage is reduced.

Batch RMA The Batch algorithm, but using MPI-3 RMA to distribute tallies over all MPI ranks.

Infrastructure Changes

The MCNP code didn't support batch statistics in any way. A number of changes had to be made:

- ▶ NPS now has a batch size option.
- ▶ When any batch tallies are enabled, KCODE will resample the fission bank to a fixed size.

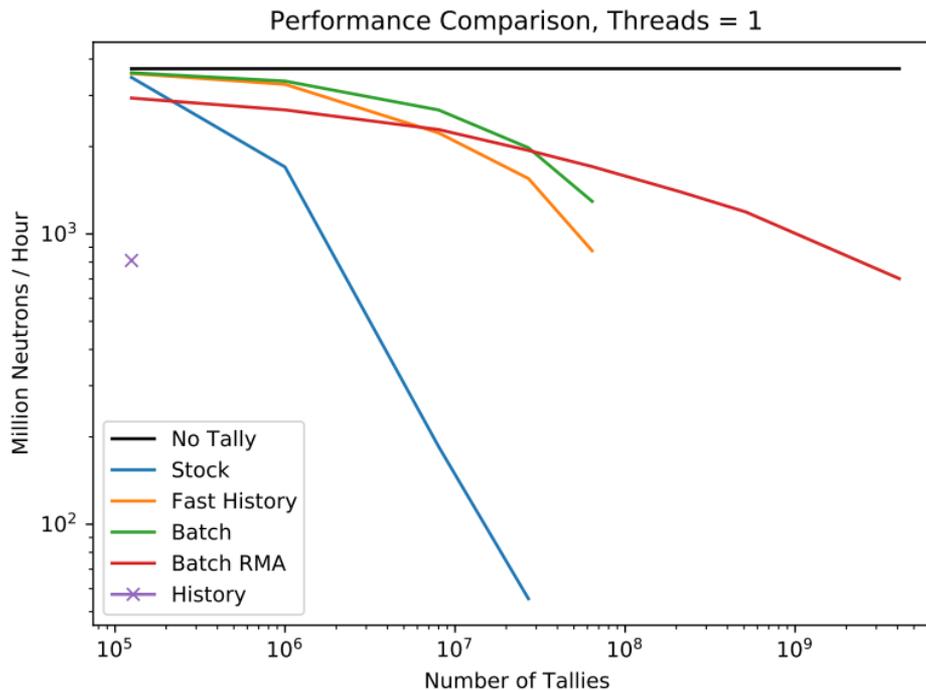
This means the RNG sequence will change if batch tallies are added!

Performance

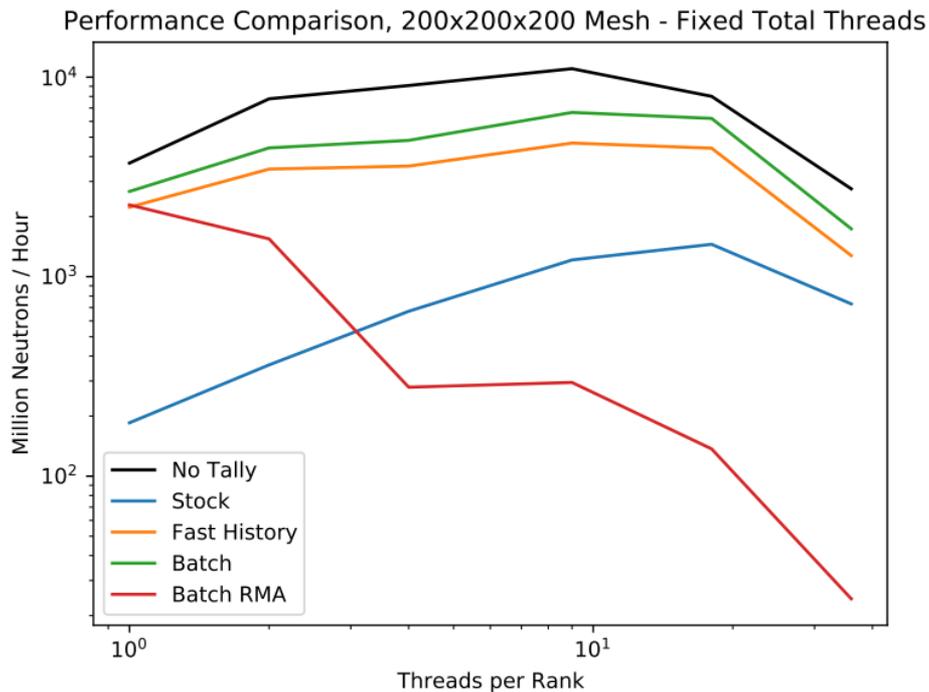
- ▶ Tested on a k -eigenvalue problem with a 10-cm, 10-g/cc ball of ^{235}U .
- ▶ Maximizes the effect of tally performance on the problem.
- ▶ Mesh was scaled from $50 \times 50 \times 50$ to $1600 \times 1600 \times 1600$
- ▶ Neutrons/hr and memory usage tallied
- ▶ Tested on 6 nodes of a cluster with 2 sockets, 18 cores each, 128 GB memory.
- ▶ Ran combinations of MPI, OpenMP.

Note: OpenMPI 3.1.6 + Omni-Path does not support `MPI_THREAD_MULTIPLE`, so threading performance is poor for Batch RMA.

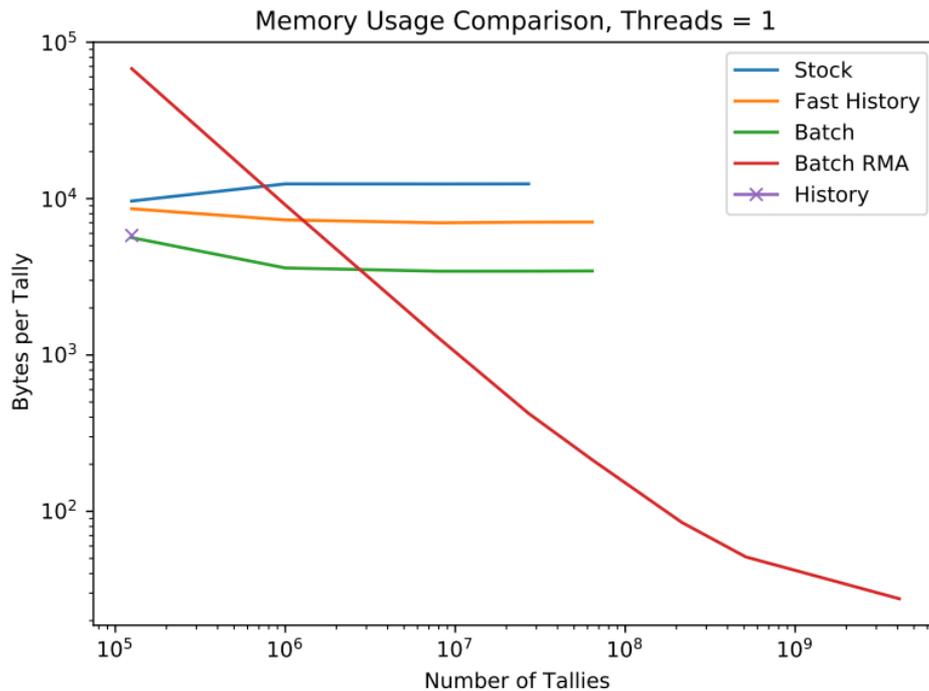
Performance



Performance

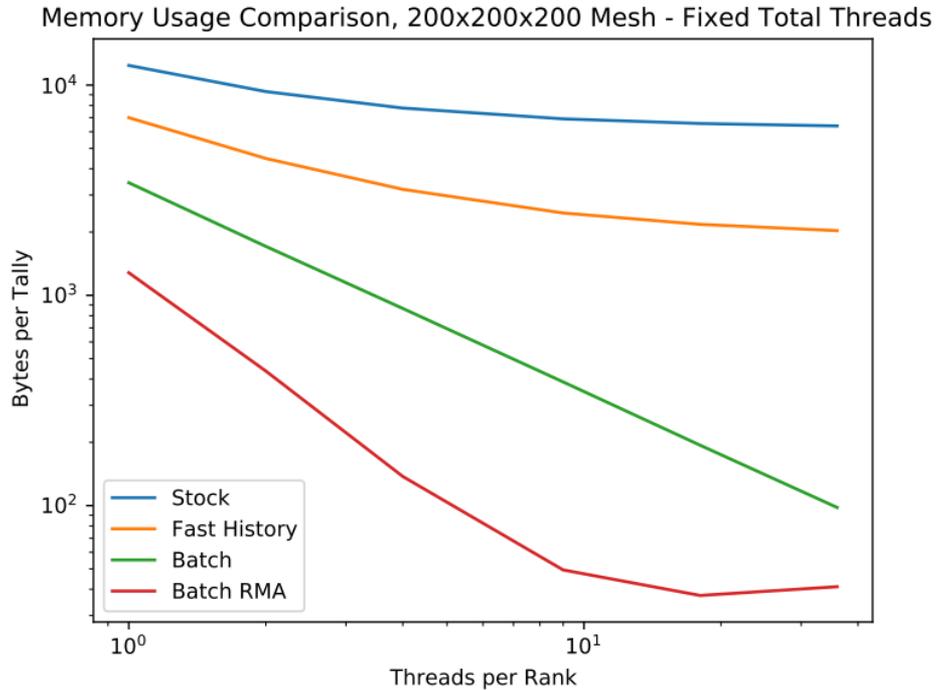


Memory



Batch RMA has high overhead that dissipates for large problems.

Memory



File Formats

Previous versions of the MCNP code used the MESHTAL format:

- ▶ ASCII output results in large file sizes.
- ▶ The binary to ASCII conversion was generally slow.
- ▶ It is tricky to bring into other tools (needs a processing script).

Version 6.3 uses HDF5 + XDMF:

- ▶ Binary file format for smaller sizes and faster IO.
- ▶ Trivial to load into ParaView, VisIt¹, Python, etc.
- ▶ (Optional) parallel HDF5 for even faster performance.

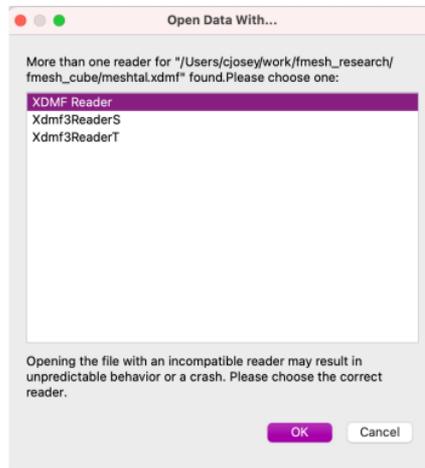
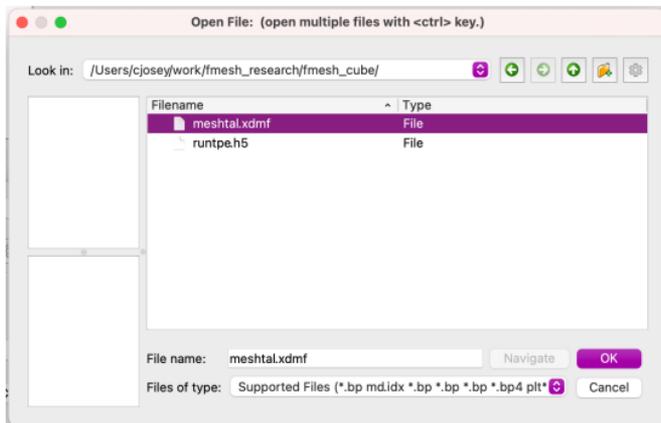
¹ Note that VisIt uses HDF5 1.8 at the time of this writing. MCNP outputs files that use the 1.10 format. Future versions of VisIt will use 1.10+. For now, `h5repack` can be used to convert to a 1.8 file.

File IO Performance

216 million cell mesh, Lustre filesystem, 8 stripes, 1M stripe size, 8 MPI Ranks:

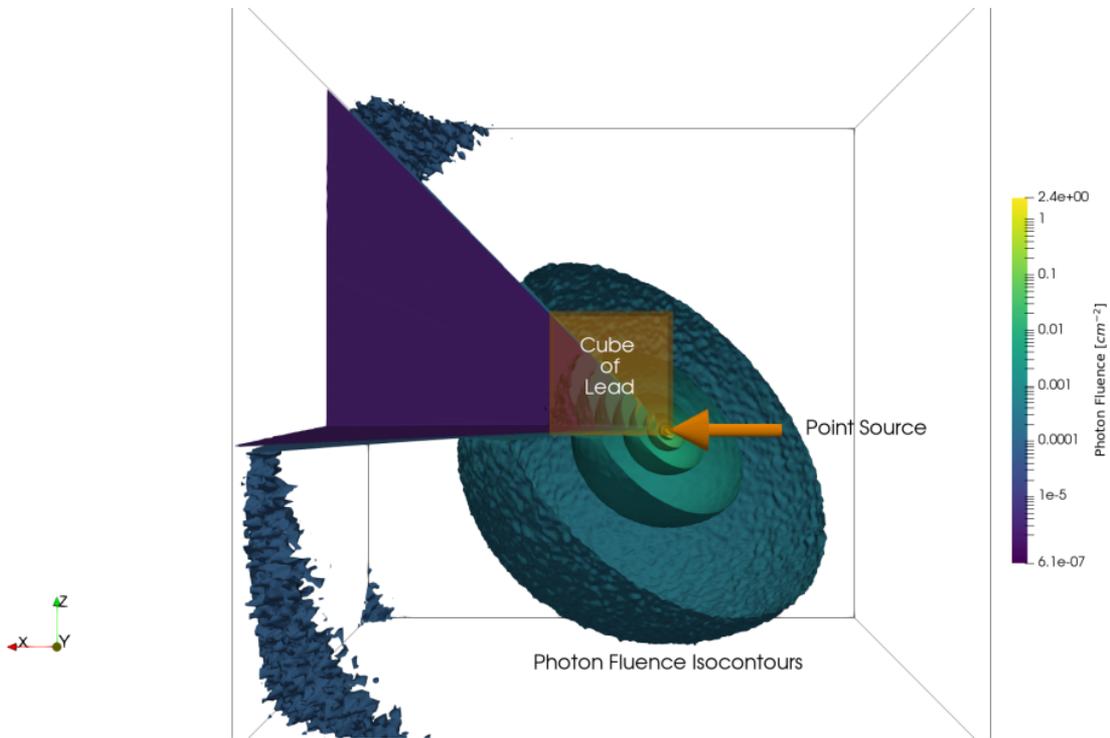
| Method | Time (s) | File Size |
|----------------------|----------|---------------------|
| MESHTAL | 617.5 | 12 GB |
| HDF5 + XDMF | 18.1 | 3.2 GB (in runtape) |
| Parallel HDF5 + XDMF | 7.5 | 3.2 GB (in runtape) |

ParaView Example



Make sure to open with “XDMF Reader”, which is the reader for XDMF version 2 files.

ParaView Example - Point Source on Cube Corner



Python Example

Listing 14: Python 3.6+ Example for Reading FMESH

```
1 import h5py
2 import numpy
3
4 def read_fmash(filename, tally_id):
5     with h5py.File(filename, 'r') as handle:
6         group = handle[f"/results/mesh_tally/mesh_tally_{tally_id}"]
7
8         data = {}
9         # Transpose converts indices to x, y, z, e, t
10        data["mean"] = numpy.transpose(group["mean"][(0)])
11        data["relative_standard_error"] = \
12            numpy.transpose(group["relative_standard_error"][(0)])
13        data["grid_x"] = group["grid_x"][(0)]
14        data["grid_y"] = group["grid_y"][(0)]
15        data["grid_z"] = group["grid_z"][(0)]
16        data["grid_energy"] = group["grid_energy"][(0)]
17        data["grid_time"] = group["grid_time"][(0)]
18
19        return data
20
21 data = read_fmash("runtpe.h5", 4)
```

By slicing `group['mean']` instead of using `[(0)]`, one can load a portion into memory without loading all of it.

Summary

- ▶ New FMESH outperforms MCNP6.2's in most workloads.
- ▶ New modes allow for much lower memory usage for large problems.
- ▶ File formats are fast and easy to work with.

In the future, we expect to extend this capability to more parts of the MCNP code.

Special Tally Treatments for Multigroup Cross Sections

Motivation

- ▶ Historically, MCNP development has not been focused on nuclear reactor applications.
- ▶ Recent institutional investments through the LANL Laboratory Directed Research & Development (LDRD) Program have focused on developing capabilities for nuclear reactor applications.
- ▶ **A new special tally treatment for multigroup cross section calculations is in the MCNP6.3 release**
- ▶ Work is continuing in this area, with a current focus on improved tracking algorithms (i.e., Delta tracking), improved energy deposition and burn-up/activation tallies, and new tooling to support efficient workflows for nuclear reactor applications
 - ▶ Note that several of these newly developed tools and/or improved capabilities will likely have an impact outside of nuclear reactor applications (e.g., radiation protection and shielding, criticality safety)

Background (1)

Multigroup cross sections are typically needed in deterministic transport and diffusion codes for reactor physics applications

- ▶ Accurate multigroup cross sections require the use of an appropriate weighting spectrum
- ▶ The weighting spectrum should be representative of the application that the multigroup cross sections are being used for

In simplified notation, the multigroup cross section can be computed as a ratio integrals,

$$\bar{\Sigma}_{x,g} = \frac{\langle \Sigma_x, \phi \rangle_g}{\langle \phi \rangle_g}, \text{ where}$$

$\bar{\Sigma}$ multigroup cross section

Σ continuous-energy cross section

ϕ weighting spectrum

x reaction channel

g incident-energy group

$\langle a, b \rangle$ the inner product of a and b integrated over all phase space

Background (2)

While the multigroup reaction cross sections are straightforward, the scattering angle/energy and fission energy terms are slightly more involved and limited (analog transport is used)

The multigroup Legendre moment, l , for the scattering matrix defining transitions from group g' to g is

$$\bar{\Sigma}_{sl,g' \rightarrow g} = \frac{\langle \Sigma_{sl}, \phi \rangle_{g' \rightarrow g}}{\langle \phi \rangle_{g'}}.$$

The multigroup fission neutron spectra is

$$\bar{\chi}_g = \frac{\sum_{g'=1}^G \langle \nu \Sigma_f, \phi \rangle_{g' \rightarrow g}}{\sum_{g=1}^G \sum_{g'=1}^G \langle \nu \Sigma_f, \phi \rangle_{g' \rightarrow g}},$$

which produces a normalized fission neutron energy spectrum. To separately obtain prompt and delayed fission neutron energy spectra, the integrals are binned by time.

Multigroup Cross Section Tally Options

Four new tally special treatment options (FT card) have been added to assist with reactor analyses:

SPM Collision exit energy-angle scatter probability matrices

MGC Flux weighted multigroup cross sections

FNS Induced fission neutron spectra

LCS Legendre coefficients for scatter reactions

These new multigroup tally capabilities have been thoroughly described and verified via code-to-code comparisons [6].

Flux-weighted Multigroup Cross Sections

FTn MGC fg

MGC Flux weighted multigroup cross sections

fg Flag for microscopic (barns) or macroscopic (1/cm) cross section calculation

Description of the Multiplier Bins for the MGC FT Option.

| Bin # | Units | Values |
|-------|----------------------------------|--|
| 1 | $n/(\text{cm}^2 \cdot \text{s})$ | Flux (used as a divisor for the other bins) |
| 2 | sh/cm | Inverse velocity |
| 3 | barns | Total cross section |
| 4 | barns | Absorption cross section |
| 5 | barns | Fission cross section |
| 6 | barns | Total or prompt fission production cross section |
| 7 | barns | Delayed fission production cross section |
| 8 | barns | Fission heat production cross section |
| 9 | barns | Capture cross section (Absorption + Fission) |
| 10 | barns | Scatter cross section [Total - (Absorption + Fission)] |

Flux-weighted Scattering Matrices and Fission Spectra

Multigroup scattering matrix options

FTn SPM na (cosine-binned scattering matrices)

SPM Collision exit energy-angle scatter probability matrices

na Integer number of equally-spaced cosine bins

FTn LCS lo (Legendre coefficient scattering matrices)

LCS Legendre coefficients for scatter reactions

lo Integer number of maximum Legendre scattering order

Multigroup fission energy spectra

FTn FNS nt

FNS Induced fission neutron spectra

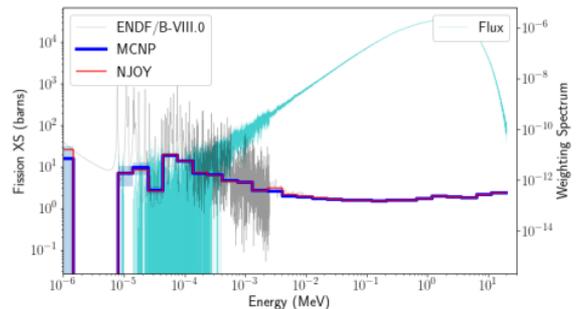
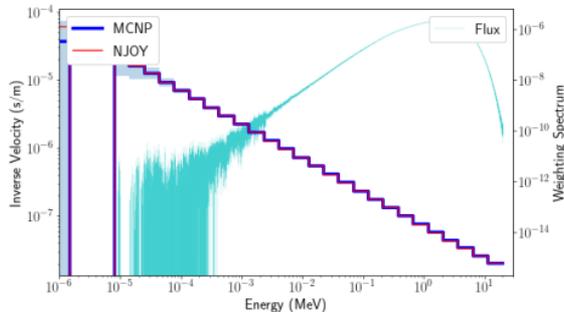
nt Integer number of delayed neutron time bins

- ▶ If nt is not specified, then a T card needs to be used to specify time binning to separate various prompt and delayed neutrons emitted from fission.

Code-to-code Verification Efforts (1)

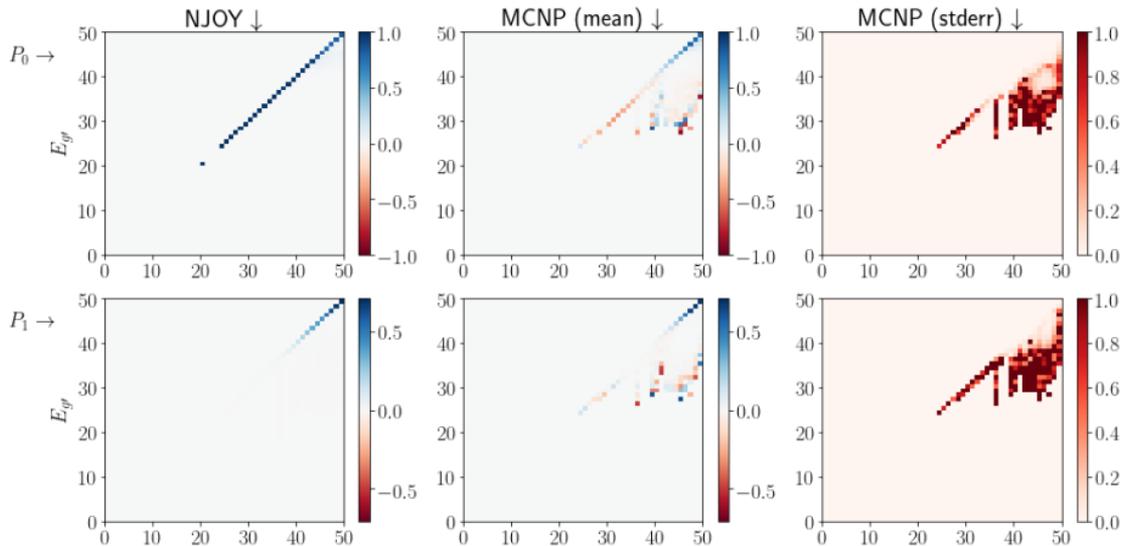
- ▶ Compared new multigroup special treatments to those produced using NJOY
 - ▶ Calculated a fine-group energy weighting spectrum with MCNP
 - ▶ Inserted the weighting spectrum into NJOY
 - ▶ Compared NJOY multigroup data to MCNP multigroup data
- ▶ Compared to other Monte Carlo codes that calculate multigroup cross sections

Multigroup Cross Section MGC Option Verification



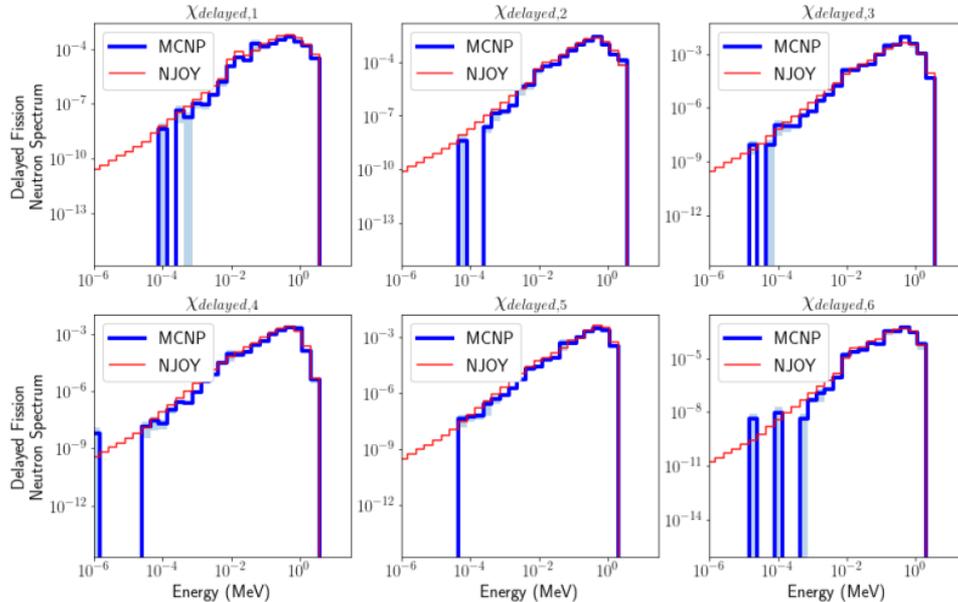
Code-to-code Verification Efforts (2)

Legendre Scattering Coefficient LCS Option Verification



Code-to-code Verification Efforts (3)

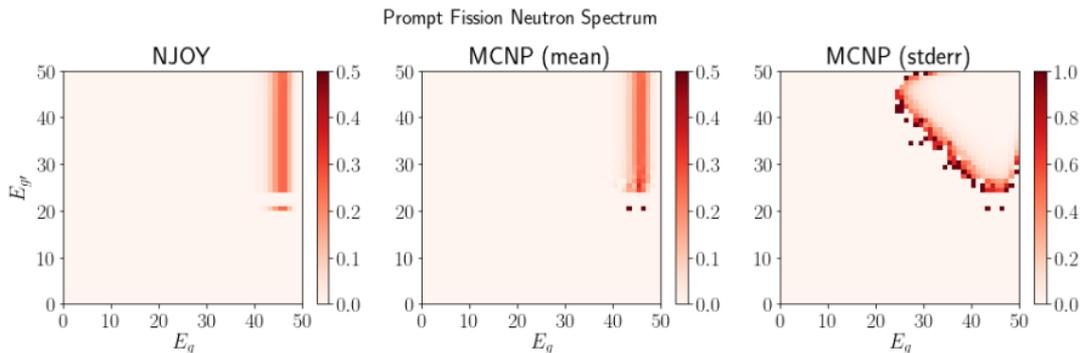
Delayed Fission Spectra FNS Option Verification



Summary

- ▶ The SPM and LCS options were compared to each other for internal consistency
- ▶ Besides any issues using NJOY with a fine-enough weighting spectrum and dealing with the statistical noise in the Monte Carlo tallies, everything looks good between NJOY and MCNP
- ▶ Some reactor pin-cell-like problems were used to compare to multigroup capabilities in other Monte Carlo codes

Prompt Fission Spectra FNS Option Verification



Demo FMESH and Multigroup Cross Section Calculations

Use Mesh Tally XDMF Output Option

Step 1: Open MCNP Input File: `deimos_um_simple.mcnp.inp`

Step 2: Ensure FMESH cards includes `out=xdmf` keyword-value option:

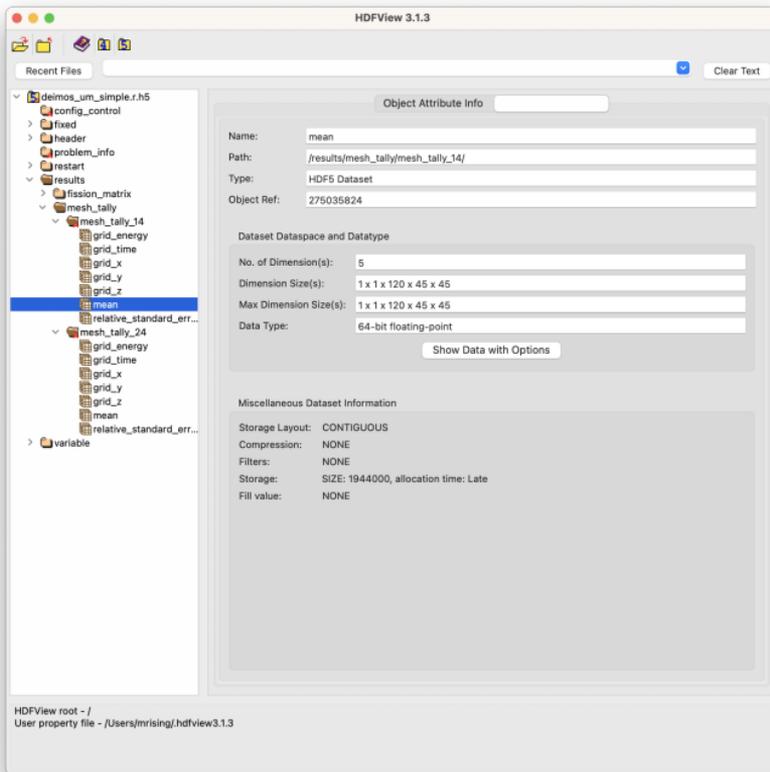
Listing 15: FMESH Card Tally Information

```
1 fmesh14:n geom= xyz   out= xdmf
2           origin= 0.0 0.0 -75.0
3           imesh=  39 57       iints= 40 5
4           jmesh=  39 57       jint= 40 5
5           kmesh= -47 47 87    kints= 10 100 10
6           tally= fast_hist
7 C
8 fmesh24:n geom= xyz   out= xdmf
9           origin= 0.0 0.0 -75.0
10          imesh=  39 57       iints= 40 5
11          jmesh=  39 57       jint= 40 5
12          kmesh= -47 47 87    kints= 10 100 10
13          tally= batch
```

Step 3: Run MCNP (with threads if possible):

```
> mcnp6 i=deimos_um_simple.mcnp.inp tasks 8
```

Mesh Information in the HDF5 Runtime



The screenshot displays the HDFView 3.1.3 interface. On the left, a tree view shows the file structure under 'demos_um_simple.e.h5', with 'mesh_tally_14' selected. The main panel, titled 'Object Attribute Info', shows the following details for the selected dataset:

- Name: mean
- Path: /results/mesh_tally/mesh_tally_14/
- Type: HDF5 Dataset
- Object Ref: 275035824

Below this, the 'Dataset Dataspace and Datatype' section provides the following information:

- No. of Dimension(s): 5
- Dimension Size(s): 1 x 1 x 120 x 45 x 45
- Max Dimension Size(s): 1 x 1 x 120 x 45 x 45
- Data Type: 64-bit floating-point

A 'Show Data with Options' button is located below the dataset information.

The 'Miscellaneous Dataset Information' section at the bottom shows:

- Storage Layout: CONTIGUOUS
- Compression: NONE
- Filters: NONE
- Storage: SIZE: 1944000, allocation time: Late
- Fill value: NONE

At the bottom of the window, the status bar indicates: 'HDFView root - /' and 'User property file - /Users/mrising/hdfview3.1.3'.

View Mesh Tally Results in ParaView

The screenshot displays the ParaView 5.10.1 interface. On the left, the Pipeline Browser shows a tree view with 'builtin:' at the top, followed by 'deimos_um_simple.msht.xdmf' and 'Threshold1'. The Properties panel for 'Threshold1' is open, showing the 'Scalars' section with '14_relative_standard_deviation (part...)'. The 'Lower Threshold' is set to 0.05 and the 'Upper Threshold' is set to 0.142252. The 'Threshold Method' is 'Between'. The 'Display' section shows 'UnstructuredGridRepresent' and 'Surface'. The 'Coloring' section shows '14_relative_standard_deviation' selected. The 'Styling' section shows 'Opacity' set to 1. The main view area is split into two panels: 'Layout #1' and 'Layout #2'. 'Layout #1' shows a 3D plot of a rectangular block with a color gradient from blue to red, representing the '14_tally' quantity. A color bar on the right of this plot ranges from -2.2×10^{-6} to 2.1×10^{-3} . 'Layout #2' shows a 3D plot of a complex mesh structure with a color gradient from blue to red, representing the '14_relative_standard_deviation' quantity. A color bar on the right of this plot ranges from 1.3×10^{-2} to 1.4×10^{-1} . Red arrows point to various UI elements: 'Create Threshold' points to the 'Threshold1' entry in the Pipeline Browser; 'Multiple Layouts' points to the 'Layout #2' tab; 'Threshold Ranges' points to the 'Lower Threshold' and 'Upper Threshold' sliders; 'Auto-scale Color Range' points to the 'Edit' button in the 'Coloring' section; and 'Plot Quantity and Color Bar' points to the color bar in the 'Layout #2' view.

ParaView 5.10.1

Time: 0

14_relative_standard_deviation

Surface

Pipeline Browser

builtin:

deimos_um_simple.msht.xdmf

Threshold1

Properties

Information

Apply

Reset

Delete

Search ... (use Esc to clear text)

Properties (Threshold1)

Scalars

14_relative_standard_deviation (part...)

Lower Threshold 0.05

Upper Threshold 0.142252

Threshold Method Between

All Scalars

Use Continuous Cell Range

Invert

Display (UnstructuredGridRepresent

Representation Surface

Coloring

14_relative_standard_deviation

Edit

Styling

Opacity 1

Create Threshold

Multiple Layouts

Threshold Ranges

Auto-scale Color Range

Plot Quantity and Color Bar

Multigroup Tally Options in Input File

Step 1: Open MCNP Input File: `deimos_um_simple.mcnp.inp`

Step 2: Ensure MGC, SPM, FNS, and LCS special tally treatments and PRDMP card included in input file:

Listing 16: Multigroup Special Tally Treatment Input Information

```
1 C          TRISO Fuel   Inconel      Graphite          Beryllium
2 f34:n (1 92i 94) (95 5i 101) (102 103 104 105) (106)
3 ft34  mgc
4 e34  1e-11 99ilog 30.
5 C
6 f44:n (1 92i 94) (95 5i 101) (102 103 104 105) (106)
7 ft44  spm 20
8 e44  1e-11 99ilog 30.
9 C
10 f54:n (1 92i 94) (95 5i 101) (102 103 104 105) (106)
11 ft54  lcs 8
12 e54  1e-11 99ilog 30.
13 C
14 f64:n (1 92i 94) (95 5i 101) (102 103 104 105) (106)
15 ft64  fns 6
16 e64  1e-11 99ilog 30.
17 C
18 prdmp 2j 1
```

Step 3: Run MCNP (with threads if possible):

```
> mcnp6 i=deimos_um_simple.mcnp.inp tasks 8
```

Plot Cross Section Tallies with MCNP Plotter

- ▶ Can follow the sequence of MCNP plotter commands to visualize multigroup cross sections

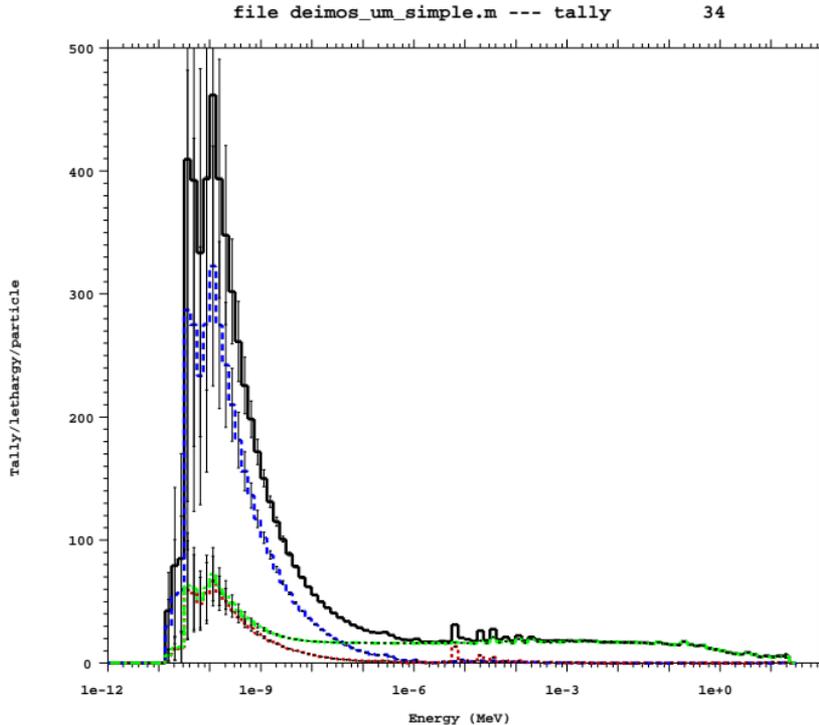
Listing 17: Complete MCNPLOT COM File

```
1  rmctal  deimos_um_simple.m
2  loglin  lethargy
3  xlims  1e-12  1e2
4  thick  0.08
5  file  fixed m 3 label "Fuel Total XS" coplot
6  fixed m 5 label "Fuel Fission XS" coplot
7  fixed m 4 label "Fuel Capture XS" coplot
8  fixed m 10 label "Fuel Scatter XS"
9  end
```

- ▶ Can also drive the MCNP plotter with this COM script:

```
> mcnp6 z com=mcplot_fuel.comin
```

Plot Cross Section Tallies with MCNP Plotter



```
mcnp6      6
probid:04/13/24 08:07:16
tally      34
n
nps        2860000
f(u)=ef(e) bin normed
mctal = deimos_um_simple.m
f Cell      1
d Flag/Dir  1
u User      1
s Segment   1
m Mult      3
c Angle     1
e Energy    *
t Time      1
```

Extracting Multigroup Tally with MCNPTools

Step 1: Ensure `mgxs.py` script and `deimos_um_simple.m` MCTAL file in working directory

Listing 18: Header of Multigroup Tally Extraction Script

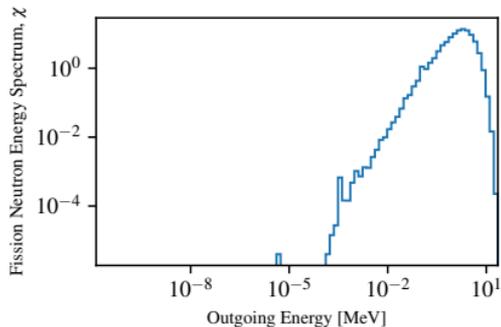
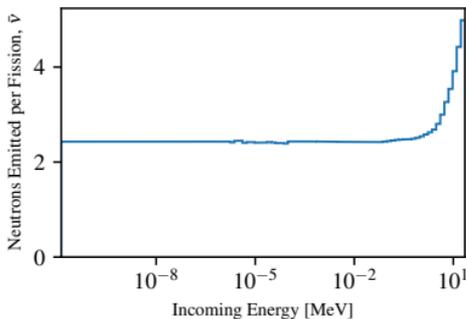
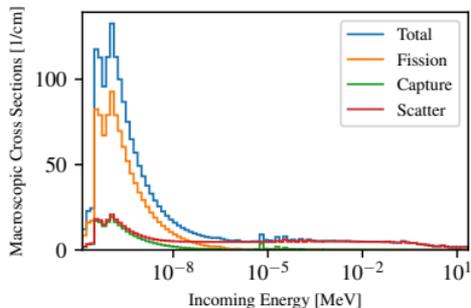
```
1 from mcnpTools import Mctal, MctalTally
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import pylab
```

Step 2: Run script:

```
> python3 mgxs.py
```

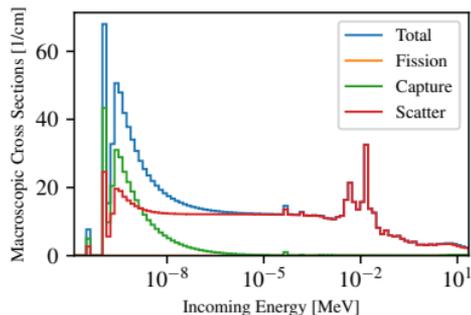
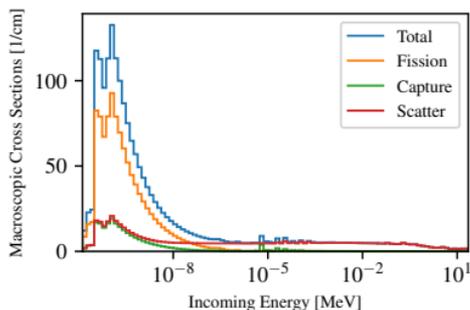
Plot Cross Section Tallies with MCNP Plotter

Triso Fuel Cross Sections

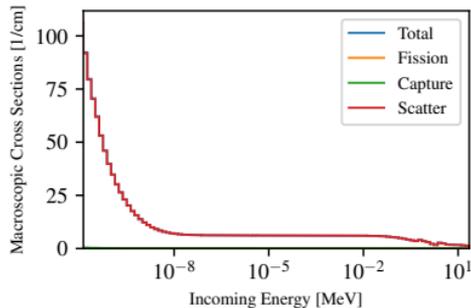
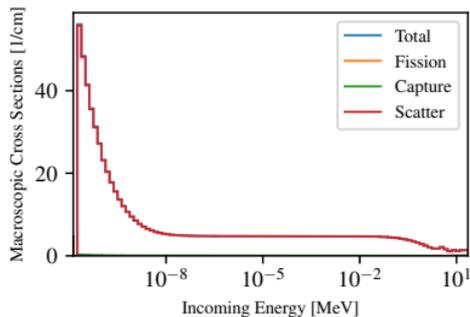


Plot Cross Section Tallies with MCNP Plotter

Triso Fuel (left) and Inconel (right) Cross Sections



Graphite (left) and Beryllium (right) Cross Sections



MCNP6.3 for Nuclear Criticality Safety

Los Alamos Scientific Laboratory, 30 December 1958 [20]

- ▶ On December 30, 1958 a nuclear criticality accident occurred in a processing facility at Los Alamos Scientific Laboratory.
 - ▶ Plutonium organic solution in an organic treatment tank;
 - ▶ single excursion;
 - ▶ one fatality, two significant exposures.
- ▶ Prior to the accident, the vessel contained two layers of aqueous and organic solutions.
 - ▶ A stirrer was started and abruptly forced solution up the tank wall, which caused a critical thickness to form in the central region.
 - ▶ The motion changed the system from subcritical to superprompt critical and an excursion occurred.
 - ▶ The system was driven permanently subcritical by the mixing of the layers.

Analysis Overview

- ▶ What effect does stirring have on a supercritical plutonium solution?
 - ▶ MCNP6.3 can be used to answer this question.
 - ▶ An input deck will be presented that utilizes universes, fills, and cones to model the vessel with the stirrer on and off.
 - ▶ The data card block will include standard cards to run a basic KCODE problem.
 - ▶ Additional data cards will be added to perform:
 - ▶ stochastic geometry calculations,
 - ▶ eigen-decomposition of the fission matrix,
 - ▶ visualization of the XDMF output of a FMESH, and
 - ▶ multigroup cross section generation.

Basic Input Deck

- ▶ A single-sheet cone is used to model the surface of the solution.
- ▶ The two infinite cells fill a finite right circular cylinder.
- ▶ A bounding sphere surrounds the system.

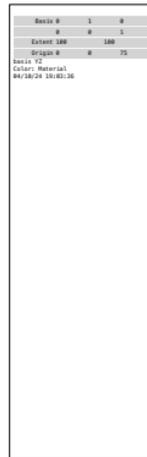
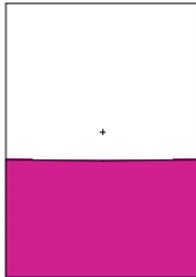
```
1 message: o=output.txt r=runtape.h5
2
3 Los Alamos Scientific Laboratory, 30 December 1958
4 10 1000 -0.98 +100          U=1      IMP:N=1
5 20 0          -100          U=1      IMP:N=1
6 30 0          -200          FILL=1   IMP:N=1
7 40 0          +200 -300      IMP:N=1
8 50 0          +300          IMP:N=0
9
10 100 KZ 60 3300 +1          <--- stirrer off
11 100 KZ 55 33  +1          <--- stirrer on
12 200 RCC 0 0 0 0 0 141.5 50
13 300 SO 1000
14
15 MODE N
16 PRINT
17 PRDMP 2j 1
18 RAND GEN=2 SEED=3 STRIDE=1e6
19 M1000 1001 8 8016 4 94239 1
```

Qt-based Geometry Plotter

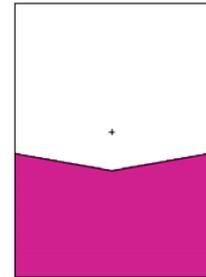
- ▶ The new Qt-based plotter can be accessed from the command line, `mcnp6.qt inp= input.txt ip`, and used to check the geometry.



Los Alamos Scientific Laboratory, 30 December 1958



Los Alamos Scientific Laboratory, 30 December 1958



Stochastic Geometry

- ▶ Stirring the solution creates a vortex but does not create extra mass.
 - ▶ The vertex of the cone must decrease to maintain the mass of the solution.
 - ▶ MCNP6.3 can be used to perform a stochastic volume calculation and determine the vertex coordinate through trial-and-error.

```
1 VOID
2 VOL 1.0
3 SDEF SUR=300 NRM=-1 DIR=D1 WGT=3141592.65359 <--- WGT=PI*(R=1000)**2
4 SI1 0 1
5 SP1 -21 1
6 SB1 -21 2
7 FC4 Solution Volume
8 F4:N ( 10 < 30 )
9 NPS 1E7
```

- ▶ The volume of the solution with the stirrer on and off is $\sim 480,000 \text{ cm}^3$ when the vertex of the cone is 5 cm lower.

Basic KCODE

- ▶ Add a basic KCODE block to the data cards
 - ▶ 10,000 particles/cycle, initial guess of 1.0, discard 50 cycles, 150 total cycles
 - ▶ Start initial fission sites at (0, 0, 50)

```
1 KCODE 10000 1.0 50 150
2 KSRC 0 0 50
```

- ▶ Stirrer off: final $k(\text{col}/\text{abs}/\text{trk len}) = 1.006221$ std dev = 0.001060
- ▶ Stirrer on: final $k(\text{col}/\text{abs}/\text{trk len}) = 0.997261$ std dev = 0.000951
- ▶ MCNP6.3's basic KCODE can answer the original question!
 - ▶ Stirring a supercritical plutonium solution can make the system subcritical.

Advanced KCODE

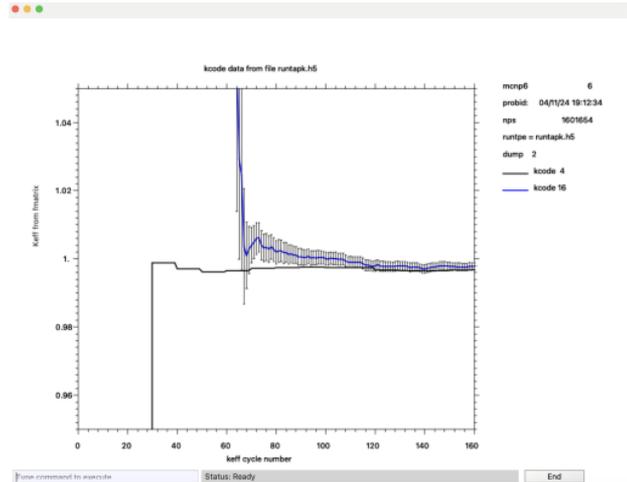
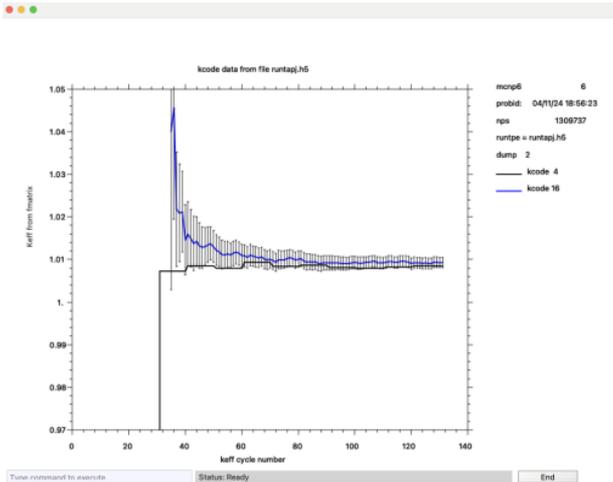
- ▶ Why does stirring the solution cause the system to be subcritical?
 - ▶ The system's fission matrix can be used to investigate this question.
 - ▶ A Shannon entropy grid is specified, the FMAT option is turned on, convergence and acceleration are enabled, and initial fission sites are sampled uniformly throughout the HSRC grid.

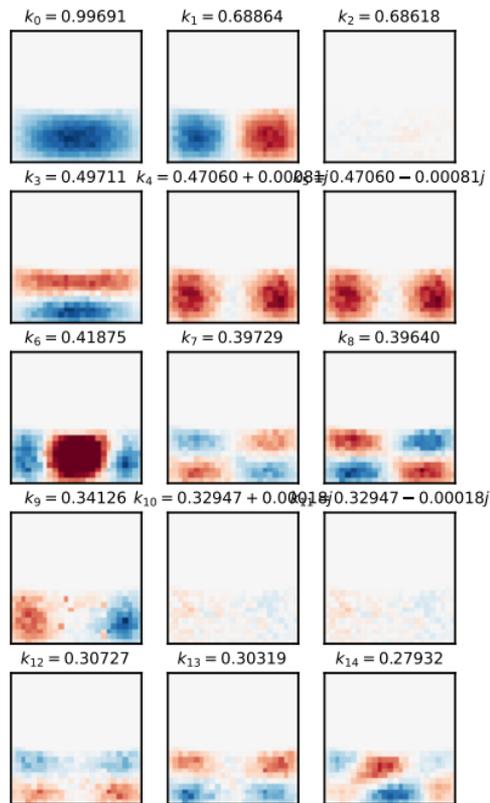
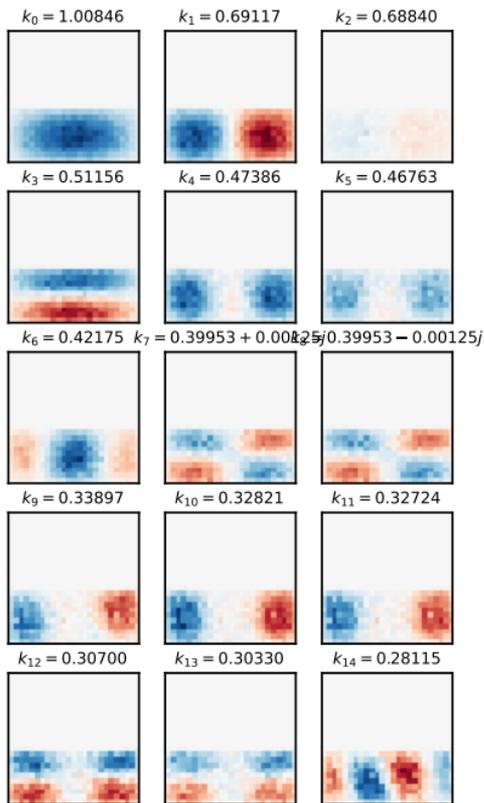
```
1 KCODE 10000 1.0 50 150
2 HSRC 25 -50 50 25 -50 50 25 0 150
3 KOPTS FMAT=YES FMATCONVRG=YES FMATACCEL=YES FMATSRG=YES
```

- ▶ Stirrer off:
 - ▶ Inactive cycles: 31
 - ▶ KCODE results: final k(col/abs/trk len) = 1.009300 std dev = 0.001234
 - ▶ FMAT results: k-effective = 1.00846 dominance ratio = 0.68271
- ▶ Stirrer on:
 - ▶ Inactive cycles: 60
 - ▶ KCODE results: final k(col/abs/trk len) = 0.997763 std dev = 0.001062
 - ▶ FMAT results: k-effective = 0.99691 dominance ratio = 0.69080

Fission Matrix Visualization

- ▶ The eigenvalues and eigenvectors of the fission matrix can be plotted and analyzed from the runtime.



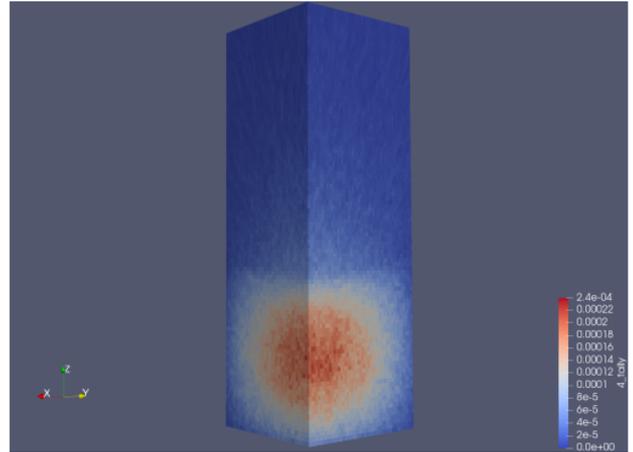
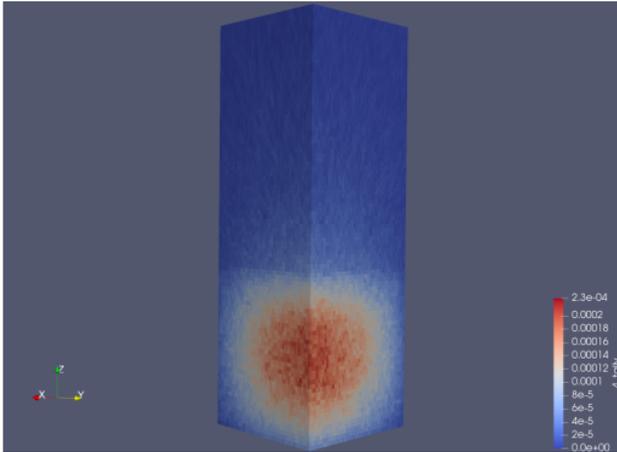


XDMF FMESH Visualization

- ▶ The FMESH tally's XDMF output can be used to generate a 3-dimensional plot of the flux.

```
1 FMESH4 : N  GEOM=XYZ      ORIGIN=-50 -50 0
2              IMESH=50     IINTS=100
3              JMESH=50     JINTS=100
4              KMESH=150    KINTS=100
5              OUT=XDMF
```

- ▶ Plots can be generated with the ParaView software.
 - ▶ Getting it: <https://www.paraview.org/download>
 - ▶ Getting help with it:
<https://www.paraview.org/community-support>
 - ▶ Issues:
<https://gitlab.kitware.com/paraview/paraview/-/issues>

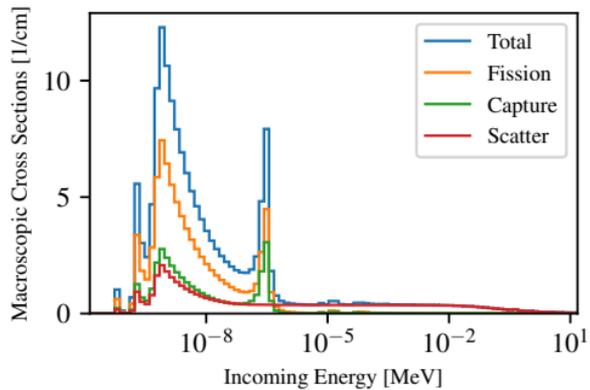
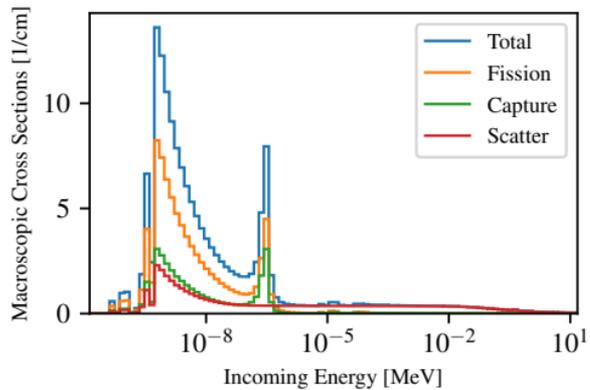


Multigroup XS Generation

- ▶ MCNP6.3 can be used to generate multigroup cross sections for use in deterministic transport software and multiphysics simulations.

```
1 VOL 4.75996E+05  
2 F4:N ( 10 < 30 )  
3 E4 1E-11 99ilog 20  
4 FT4 MGC 1
```

- ▶ MCNPTools can be used to extract the multigroup cross section tallies from a MCTAL file.
 - ▶ Getting it: <https://github.com/lanl/mcnptools>



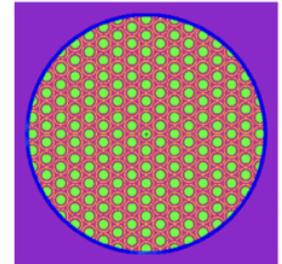
Final Thoughts

- ▶ MCNP6.3 is a powerful tool for analyzing nuclear systems.
- ▶ In KCODE problems, creating the geometry and defining materials are the most user intensive processes.
- ▶ It is easy to add data cards but using and interpreting them correctly can be challenging.
- ▶ Output files can now be analyzed and plotted easily.
- ▶ Multigroup cross sections should be generated for each configuration.
 - ▶ Using the “wrong” multigroup cross sections can introduce unnecessary nuclear data error into a calculation.
- ▶ Consider investigating some of MCNP6.3’s other tools and features such as `mcnp_pstudy` and `KSEN`.

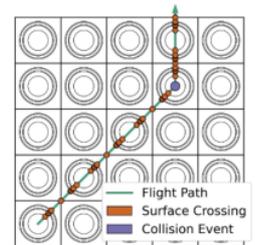
Future Work

Delta Tracking in MCNP6.4

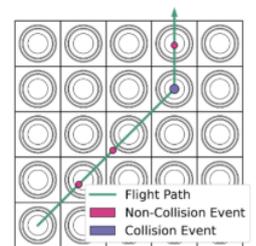
- ▶ Delta tracking (aka Woodcock tracking) is an alternative algorithm to the standard Monte Carlo surface-to-surface tracking algorithm
- ▶ Can be beneficial when collisions occur rarely with respect to the frequency that surfaces are crossed
- ▶ Implemented, tested, documented, and reviewed the Delta tracking capability – will be released in MCNP6.4
- ▶ Figures on right from Kristin Stolte ANS Summer 2023 presentation where discrete modeling of TRISO particles compared both tracking algorithms



Surface Tracking Algorithm

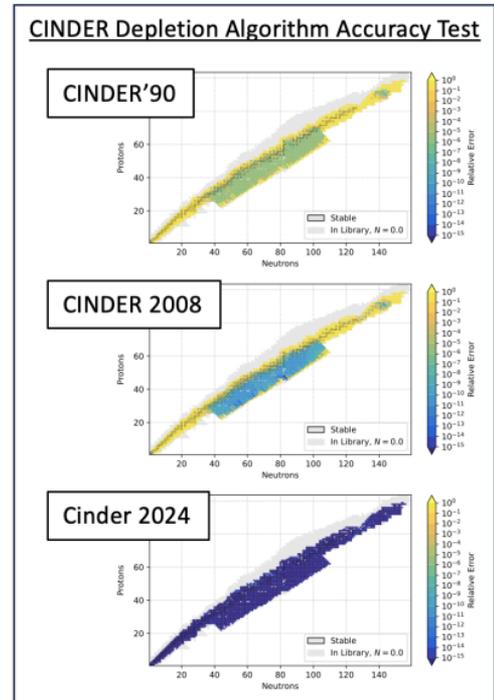


Delta Tracking Algorithm



Cinder Burn-up/Depletion in MCNP6.4

- ▶ The CINDER'90 code, used for material depletion and transmutation physics, was introduced into the MCNPX code in the early 2000's
 - ▶ Used inline for nuclear reactor depletion and activation/delayed particle emission physics
- ▶ Although it was planned, the CINDER 2008 code was never merged into MCNP6
- ▶ We are actively modernizing the code, with plans to release Cinder 2024 including improved numerical algorithms and better nuclear data



Credit: C. Josey CINDER Algorithm Improvements

Other Noteworthy Changes to Expect in MCNP6.4 (or earlier)



- ▶ Generalized ZAID-specifier input processing (in MCNP6.3.1)
- ▶ New random number generator (see RAND GEN=8 in MCNP6.3.1)
- ▶ Extended general sensitivity/uncertainty tally capabilities (FSEN)
- ▶ Completion of FMESH improvements to supplant TMESH capabilities
- ▶ Dynamically-linked source and tally capabilities
- ▶ More output data converted into HDF5-accessible formats
- ▶ And many more code enhancements, bug fixes, etc.
- ▶ For annual updates on MCNP6 code developments, consider attending our MCNP User Symposium (<https://mcnp.lanl.gov/symposia.html>)

Thanks!

Q&A

Backup Slides

References

- [1] M. E. Rising, J. C. Armstrong, S. R. Bolding, F. B. Brown, J. S. Bull, T. P. Burke, A. R. Clark, D. A. Dixon, R. A. Forster, III, J. F. Giron, T. S. Grieve, H. G. Hughes, III, C. J. Josey, J. A. Kulesza, R. L. Martz, A. P. McCartney, G. W. McKinney, S. W. Mosher, E. J. Pearson, C. J. Solomon, Jr., S. Swaminarayan, J. E. Sweezy, S. C. Wilson, and A. J. Zukaitis, “MCNP[®] Code Version 6.3.0 Release Notes,” Tech. Rep. LA-UR-22-33103, Rev. 1, Los Alamos National Laboratory, Los Alamos, NM, USA, Jan. 2023.
- [2] J. A. Kulesza, T. R. Adams, J. C. Armstrong, S. R. Bolding, F. B. Brown, J. S. Bull, T. P. Burke, A. R. Clark, R. A. Forster, III, J. F. Giron, T. S. Grieve, C. J. Josey, R. L. Martz, G. W. McKinney, E. J. Pearson, M. E. Rising, C. J. Solomon, Jr., S. Swaminarayan, T. J. Trahan, S. C. Wilson, and A. J. Zukaitis, “MCNP[®] Code Version 6.3.0 Theory & User Manual,” Tech. Rep. LA-UR-22-30006, Rev. 1, Los Alamos National Laboratory, Los Alamos, NM, USA, Sept. 2022.

References

- [3] J. S. Bull, J. A. Kulesza, C. J. Josey, and M. E. Rising, “MCNP[®] Code Version 6.3.0 Build Guide,” Tech. Rep. LA-UR-22-32851, Rev. 1, Los Alamos National Laboratory, Los Alamos, NM, USA, Dec. 2022.
- [4] C. J. Josey, A. R. Clark, J. A. Kulesza, E. J. Pearson, and M. E. Rising, “MCNP[®] Code Version 6.3.0 Verification & Validation Testing,” Tech. Rep. LA-UR-22-32951, Rev. 1, Los Alamos National Laboratory, Los Alamos, NM, USA, Dec. 2022.
- [5] V. K. Mehta, J. Armstrong, D. V. Rao, and D. Kotlyar, “Capturing Multiphysics Effects in Hydride Moderated Microreactors using MARM,” *Annals of Nuclear Energy*, vol. 172, p. 109067, 2022.
- [6] R. B. Wilkerson, G. McKinney, M. E. Rising, and J. A. Kulesza, “MCNP Reactor Multigroup Tally Options Verification,” Tech. Rep. LA-UR-20-27819, Los Alamos National Laboratory, Oct. 2020.
- [7] C. J. Josey and J. A. Kulesza, “Improved FMESH Capabilities in the MCNP 6.3 Code,” Tech. Rep. LA-UR-21-26363, Los Alamos National Laboratory, July 2021.

References

- [8] J. A. Kulesza, J. C. Armstrong, T. C. McClanahan, and S. Swaminarayan, “MCNP Unstructured Mesh Elemental Quality Assessment,” Tech. Rep. LA-UR-21-26362, Los Alamos National Laboratory, Los Alamos, NM, USA, July 2021. MCNP User Symposium Presentation.
- [9] C. J. Stimpson, C. D. Ernst, P. K. and P. P. Pébay, and D. Thompson, “The Verdict Geometric Quality Library,” Tech. Rep. SAND2007-1751, Sandia National Laboratories, Albuquerque, NM ,USA, Mar. 2007.
- [10] J. A. Kulesza, “MCNP Code Version 6.3.0 Unstructured-mesh Quality Metrics & Assessment,” Tech. Rep. LA-UR-20-27150, Los Alamos National Laboratory, Los Alamos, NM, USA, Sept. 2020.
- [11] P. M. Knupp, “Achieving Finite Element Mesh Quality Via Optimization of the Jacobian Matrix Norm and Associated Quantities. Part II—A Framework for Volume Mesh Optimization and the Condition Number of the Jacobian Matrix,” *International Journal for Numerical Methods in Engineering*, vol. 48, pp. 1165–1185, May 2000.

References

- [12] P. M. Knupp, “Algebraic Mesh Quality Metrics for Unstructured Initial Meshes,” *Finite Elements in Analysis and Design*, vol. 39, pp. 217–241, Jan. 2003.
- [13] P. P. Pébay, D. Thompson, J. Shepherd, P. Knupp, C. Lisle, V. A. Magnotta, and N. M. Grosland, “New Applications of the Verdict Library for Standardized Mesh Verification Pre, Post, and End-to-End Processing,” in *Proceedings of the 16th International Meshing Roundtable* (M. L. Brewer and D. Marcum, eds.), (Seattle, WA, USA; October 15–17), pp. 535–552, 2008.
- [14] R. L. Martz, “The MCNP6 Book On Unstructured Mesh Geometry: User’s Guide For MCNP 6.2,” Tech. Rep. LA-UR-17-22442, Los Alamos National Laboratory, Los Alamos, NM, USA, Mar. 2017.
- [15] J. L. Alwin and J. B. Spencer, “Utilizing Unstructured Mesh Geometry in Criticality Calculations and Criticality Accident Alarm System Analysis,” Presentation LA-UR-21-25913, Los Alamos National Laboratory, Los Alamos, NM, USA, June 2021.

References

- [16] M. R. Dzur, J. C. Armstrong, and C. A. D'Angelo, "CUBIT for MCNP Unstructured Mesh Analysis of Oktavian Benchmarks," Presentation LA-UR-22-30541, Los Alamos National Laboratory, Los Alamos, NM, USA, October 2022.
- [17] I. Remec and F. B. K. Kam, "Pool Critical Assembly Pressure Vessel Facility Benchmark," Tech. Rep. NUREG/CR-6454, Oak Ridge National Laboratory, Oak Ridge, TN, USA, July 1997.
- [18] J. A. Kulesza and R. L. Martz, "Evaluation of the Pool Critical Assembly Benchmark with Explicitly Modeled Geometry Using MCNP6," *Nuclear Technology*, vol. 197, pp. 284–295, Mar. 2017.
- [19] J. A. Kulesza and R. L. Martz, "Evaluation of the Pool Critical Assembly Benchmark with Explicitly Modeled Geometry using MCNP6's Unstructured Mesh Capabilities," in *Proceedings of 16th International Symposium on Reactor Dosimetry (ISR16)*, (Santa Fe, NM, USA; May 7–12), ASTM International, 2017. Accompanying presentation: LA-UR-17-23854.

References

- [20] T. P. McLaughlin, S. P. Monahan, B. G. Ryazanov, N. L. Pruvost, V. V. Frolov, and V. I. Sviridov, *A Review of Criticality Accidents*. 2000.
- [21] U. Ayachit, *The ParaView Guide*. Kitware, Inc., community ed., June 2018.
- [22] J. A. Kulesza and T. C. McClanahan, “A Python Script to Convert MCNP Unstructured Mesh Elemental Edit Output Files to XML-based VTK Files,” Tech. Rep. LA-UR-19-20291, Rev. 2, Los Alamos National Laboratory, Los Alamos, NM, USA, Nov. 2019.
- [23] J. A. Kulesza, J. L. Alwin, J. D. Hutchinson, E. F. Shores, and R. C. Little, “l3d2vtk: An MCNPTools Utility to Enable LNK3DNT File Visualization & Post-processing,” in *Transactions of the Winter 2019 American Nuclear Society Meeting*, vol. 121, pp. 1233–1236, Nov. 2019.

References

- [24] H. Childs, E. Brugger, B. Whitlock, J. Meredith, S. Ahern, D. Pugmire, K. Biagas, M. Miller, C. Harrison, G. H. Weber, H. Krishnan, T. Fogal, A. Sanderson, C. Garth, E. W. Bethel, D. Camp, O. Rübel, M. Durant, J. M. Favre, and P. Navrátil, “VisIt: An End-User Tool For Visualizing and Analyzing Very Large Data,” in *High Performance Visualization—Enabling Extreme-Scale Scientific Insight*, pp. 357–372, Oct. 2012.
- [25] J. A. Clarke and E. R. Mark, “Enhancements to the eXtensible Data Model and Format (XDMF),” in *HPCMP User’s Group Conference 2007. High Performance Computing Modernization Program: A Bridge to Future Defense*, (Pittsburgh, PA, USA; June 18–21), pp. 322–327, 2007.
- [26] “XDMF Model and Format.” Website, Mar. 2019.

Qt-based Geometry and Tally Plotter

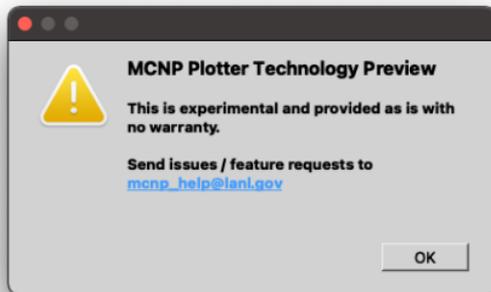
Outline

- ▶ Geometry Plotter
 - ▶ Introduction of new features
 - ▶ Brief look at each panel
 - ▶ Live demo
- ▶ Tally Plotter
 - ▶ New viewport
 - ▶ Live demo

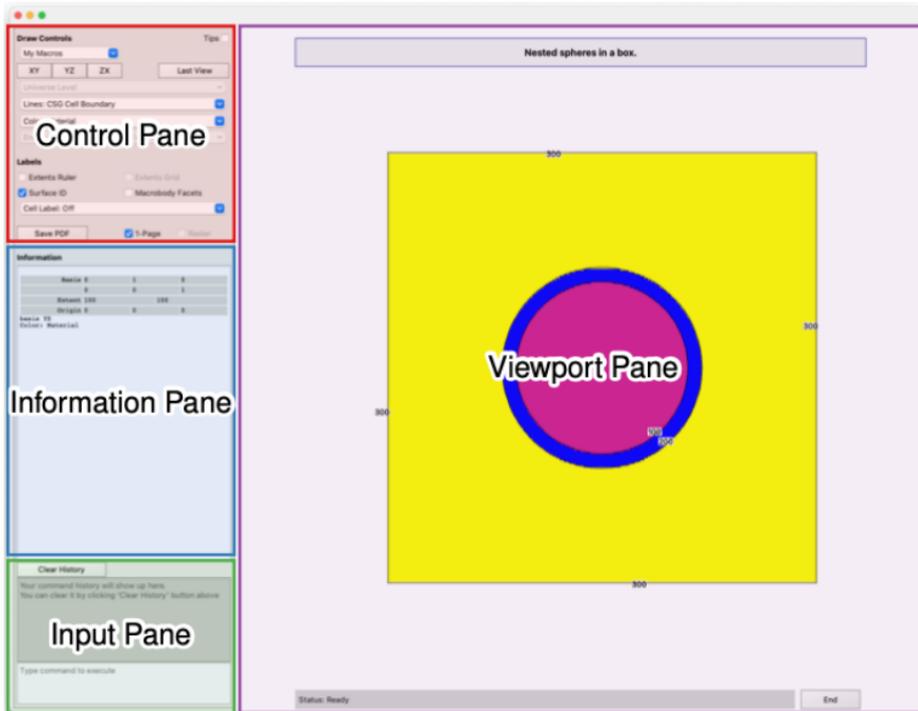
New Plotter Features

- ▶ Qt-based render window (no X-Server required)
- ▶ Modern buttons (geometry plotter)
- ▶ Command box supporting > 29 characters
- ▶ Mouse control to pan, zoom, and rotate slices
- ▶ New plotter view file formats (png, pdf)
- ▶ Macro support

New plotter is shipping with MCNP6.3



Geometry Plotter First Look



Control Pane (1)

The image shows a software control panel titled "Draw Controls" with various settings and options. Callout boxes provide detailed explanations for several key features:

- My Macros:** A user-defined view selector, currently set to "My Macros".
- Views:** Direct access to z, x, and y views, with buttons for "XY", "YZ", and "ZX".
- Universe Level:** A menu that selects the level for repeated structure geometries, currently set to "Universe Level".
- Color Menu:** Controls the filling of cells, currently set to "Lines: Mesh Tally + Cell".
- FMESH Tally:** A tally energy bin selector, currently set to "TMESH 111".
- Energy and Time Bins:** Selectors for "Energy Bin" (No FMESH) and "Time Bin" (No FMESH).
- Labels:** A section with checkboxes for "Extents Ruler", "Extents Grid", "Surface ID", and "Macrobody Facets".
- Cell Label:** A menu for cell values, currently set to "Off".
- PDF Options:** Includes a "Save PDF" button and checkboxes for "1-Page" (checked) and "Raster".
- Tips:** A checkbox to show tool tips by hovering the mouse over graphics elements.
- Last View:** A button to toggle between the current and previous view.
- Line Menu:** A menu that determines outlines on the graph.
- Grid:** A checkbox to draw a grid, which is only active when the ruler is displayed.
- Macrobody Facets:** A checkbox to toggle the labelling of macrobody surfaces.
- PDF Output:** A note indicating that single-page PDF files are the default, and deselecting allows for multiple figures per file.
- Raster PDFs:** A note stating that Save Raster PDFs are not implemented yet.

A general note at the bottom states: "All drop-down menus can be 'torn off' with the dotted line at the top."

Control Pane (2)

- ▶ “My Macros” functionality
 - ▶ Similar to reading a COM file on the execution line
 - ▶ More later
- ▶ “Last View” expands on limitations of “Restore” button
 - ▶ Resets all parameter(s) to previous parameter(s) instead of a select few
- ▶ “Tips” button allows user to hover over a feature for quick help text
- ▶ “Save PDF” button
 - ▶ Available as a new command: `savepdf`
 - ▶ **Raster PDF functionality to come**

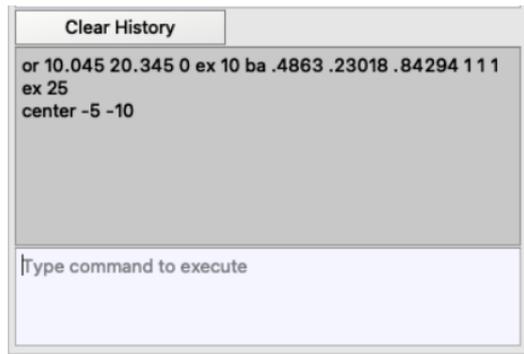
Information Pane

- ▶ Contains all information present in X-based plotter by default
 - ▶ Basis, Extent, Origin
- ▶ Click in a cell for more information on that cell.
 - ▶ Old plotter only showed what was selected in the right-hand control panel
 - ▶ Now shows all information relevant to problem
- ▶ Will also show information on an overlaid FMESH mesh
 - ▶ Runtape information, tally value/error of selected voxel, voxel indices

| Information | | | |
|-----------------|------------|--------|--------|
| Basis | 0 | 1 | 0 |
| | 0 | 0 | 1 |
| Extent | 100 | | 100 |
| Origin | 0 | 0 | 0 |
| basis YZ | | | |
| Color: Material | | | |
| | | | |
| Cell ID 1 | | | |
| Location | 0 | -12.48 | -11.52 |
| Atom Density | 1.54 | | |
| Mass Density | 55.06 | | |
| Volume | 2.5133e+07 | | |
| Mass | 1.3838e+09 | | |
| pwt | -1 | | |
| Material 1 | | | |
| Temperature | 1 2.53e-08 | | |
| n:Importance: | 1 | | |
| p:Importance: | 1 | | |
| h:Importance: | 1 | | |
| d:Importance: | 1 | | |
| t:Importance: | 1 | | |
| n:FCL: | 1 | | |

Input Pane

- ▶ The old “Click here or picture or menu” box only supported 29 characters of input
 - ▶ or 10.045 20.345 0 ex 10 ba
.4863 .23018 .84294 1 1 1
- ▶ New input is “infinite” for all practical one-liners
 - ▶ Also supports command recall with up- and down-arrows
- ▶ Messy history can be cleared with “Clear History” button, but command recall with arrow keys still works
- ▶ Error messages are still presented in the command-prompt window



Viewport Pane

- ▶ Can now interact directly with keyboard and mouse:
 - ▶ Click and drag to pan
 - ▶ Shift-Click and drag to rotate
 - ▶ Ctrl-Click and drag to zoom in and out
 - ▶ Ctrl-[Arrow Key] to pan
 - ▶ Ctrl-Shift-[Arrow Key] to pan more
 - ▶ Ctrl-[+/-] to zoom at origin
 - ▶ Ctrl-Shift-[+/-] to zoom more
- ▶ Status bar now tells you when the plotter is working

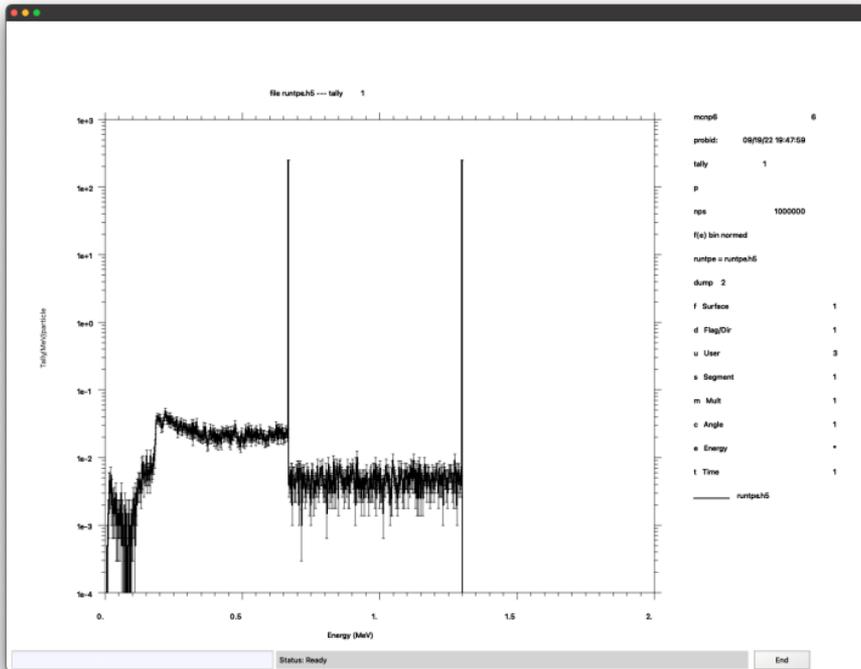
These features will be shown in the demo

Geometry Plotter Live Demo

- ▶ Show macro functionality with lost particle use-case
- ▶ Demo more complex geometry
 - ▶ Compare X-plotter speed
 - ▶ Show new `savepdf`, `savepng` commands

Tally Plotter Viewport

- ▶ Largely the same look as the old tally plotter, though modernized



Tally Plotter Live Demo

- ▶ Show classic tallies
 - ▶ Arrow key recall
- ▶ Show FMESH tally
 - ▶ Tie into geometry plotter

MCNP6.3 Visualization Approaches

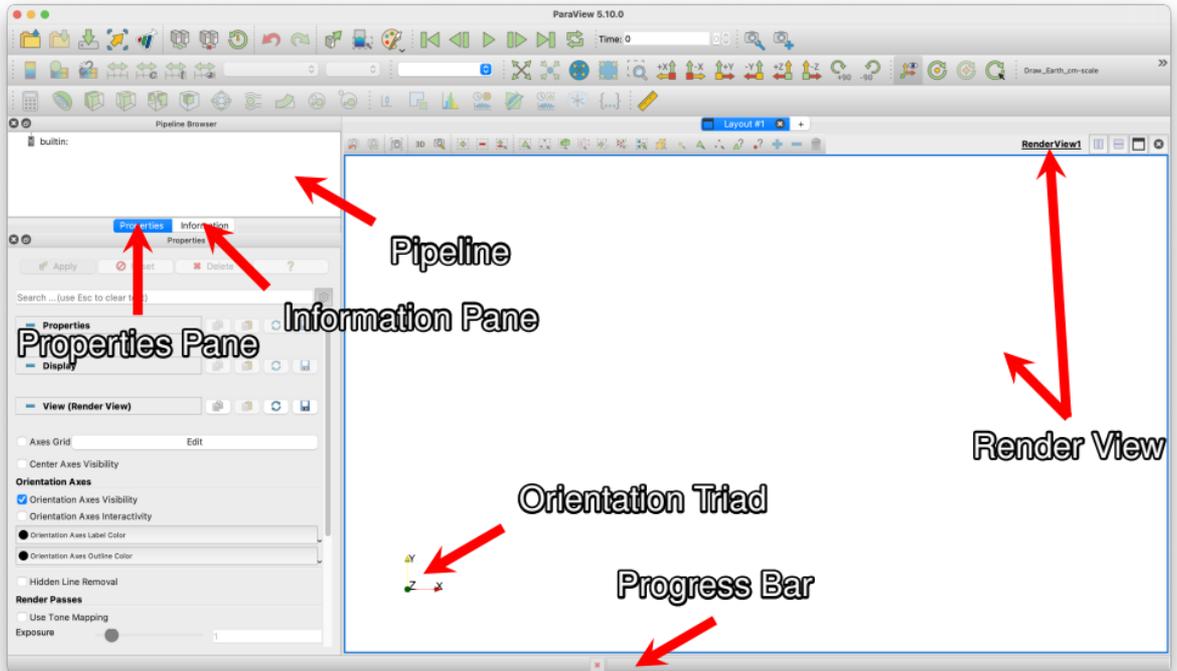
How to Use These Slides

- ▶ Intended to be useful as a reference
 - ▶ Can be used independent of ParaView [21] being open
- ▶ GUI elements labeled to establish nomenclature
 - ▶ When “clicks” are necessary, steps are numbered to give click order
 - ▶ Additional information in lower-right “callout” boxes
- ▶ Recommended approach: focus on presentation and experiment after
 - ▶ Prevents “getting lost” while the presentation is underway
 - ▶ Experimentation is how skill and “muscle memory” is built

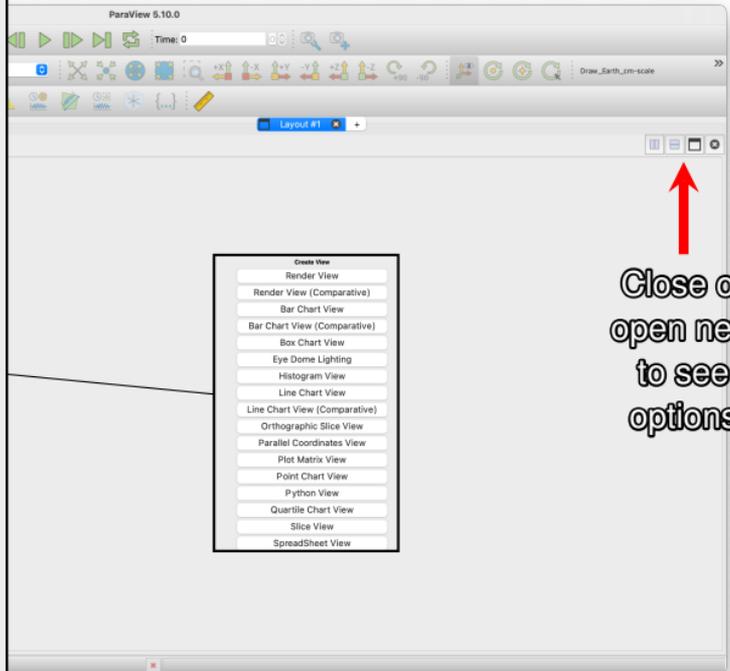
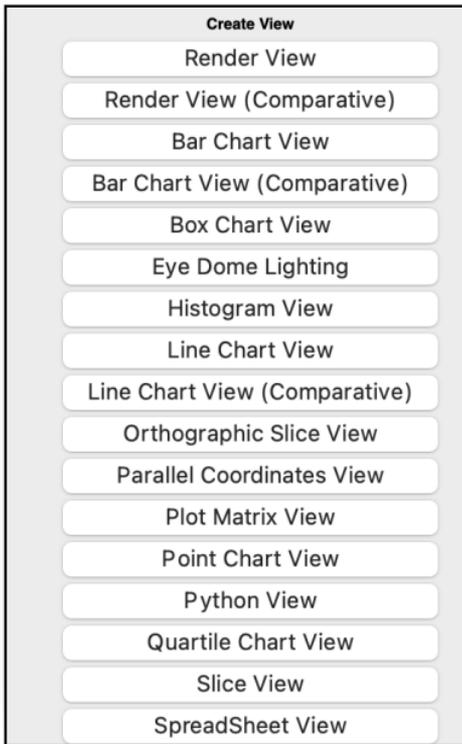
ParaView Introduction

- ▶ Open-source & cross-platform software
- ▶ Maintained by Kitware Inc.
 - ▶ Kitware also maintains CMake, VTK, etc.
- ▶ Supported by, among others:
 - ▶ Advanced Simulation and Computing Program
 - ▶ Army Research Laboratory
 - ▶ Los Alamos National Laboratory
 - ▶ Sandia National Laboratories
- ▶ Getting it: <https://www.paraview.org/download>
- ▶ Getting help with it: <https://www.paraview.org/community-support>
- ▶ Issues: <https://gitlab.kitware.com/paraview/paraview/-/issues>

ParaView Interface Overview—Main Components

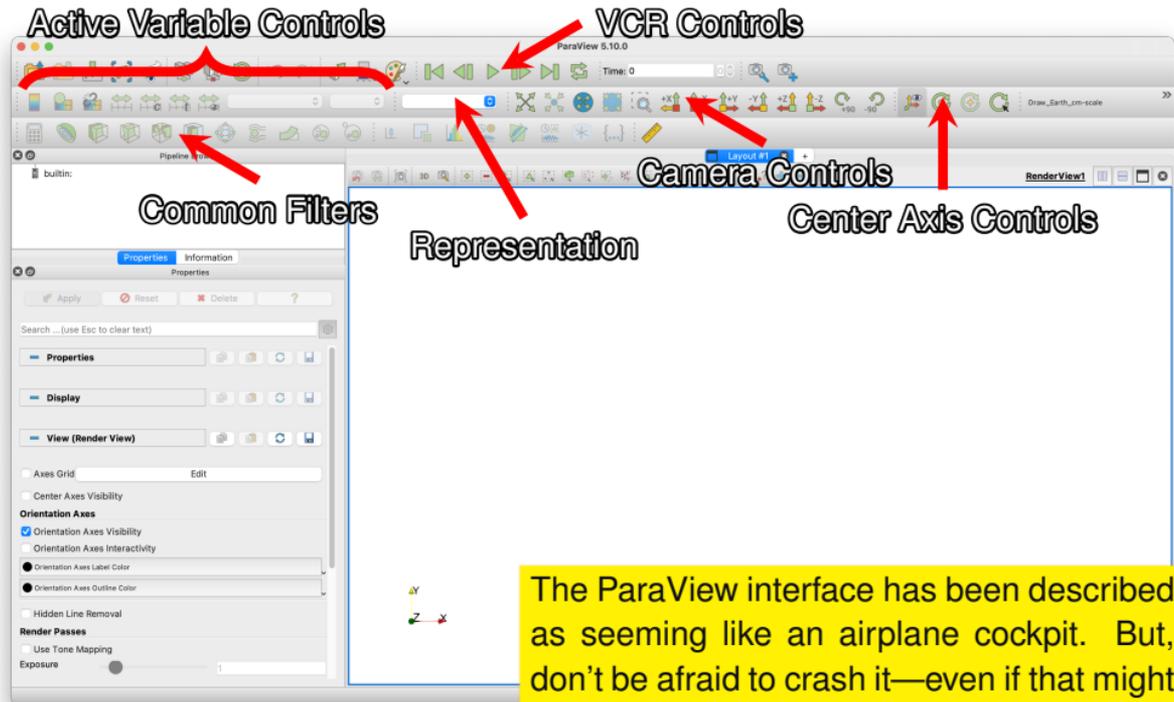


Views other than the Render View



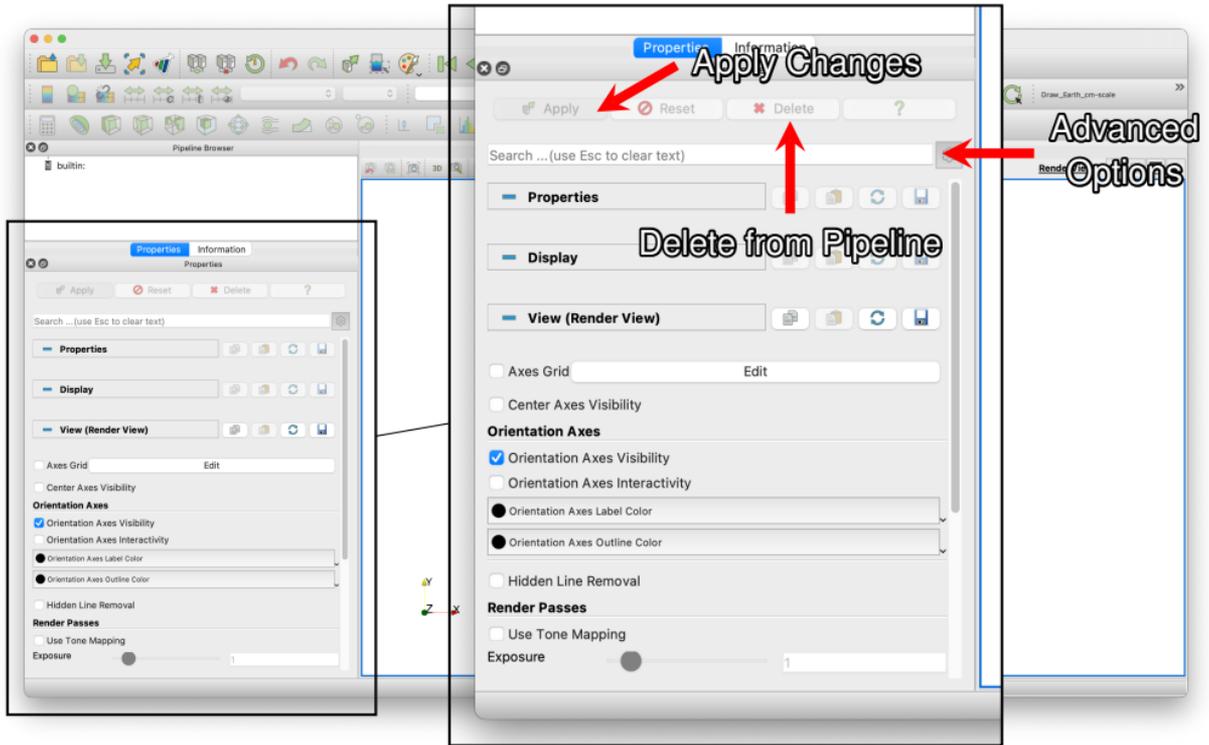
Close or
open new
to see
options

ParaView Interface Overview—Toolbars



The ParaView interface has been described as seeming like an airplane cockpit. But, don't be afraid to crash it—even if that might (rarely) happen from time to time...

ParaView Interface Overview—Properties Pane



Render View Colors (Settings, Color Palette Tab)



Advanced Options

Complement the Background, As Appropriate

Background Color

macOS: ParaView Menu → Preferences
Windows/Linux: Edit Menu → Settings

Moving Around / Controls

ParaView 5.10.0

Time: 0 0 max is 43

Draw_Earth_cm-scale

Pipeline Browser

Layout #1

RenderView1

Data in Pipeline

(1) Click "Apply" to Load

Variables to Initially Load

Nothing showed up because ParaView is giving an opportunity to deselect some data/geometry before loading it. This can save time/memory.

The screenshot shows the ParaView 5.10.0 interface. The Pipeline Browser on the left shows a single data source named 'can.ex2'. The Properties panel below it has an 'Apply' button highlighted with a red arrow and the text '(1) Click "Apply" to Load'. Below the 'Apply' button, there is a search bar and a list of variables under the heading 'Properties (can.ex2)'. A red bracket groups the following variables: 'Block Arrays' (with sub-items 'accl', 'displ', 'vel', 'eqps'), 'Sets' (with sub-items 'nodelist_1', 'nodelist_100', 'surface_4'), and 'Set Arrays'. The text 'Variables to Initially Load' is placed to the right of this bracket. A yellow text box at the bottom right contains the text: 'Nothing showed up because ParaView is giving an opportunity to deselect some data/geometry before loading it. This can save time/memory.'

Moving Around / Controls (Properties Tab)

(1) Color by Block

Note the "Open" Eyeball

Unavailable when Applied

(2) Scroll through the Properties Pane to get a Sense of the Amount of (Advanced) Options

(3) Click and Drag to Rotate View

(4) Advance Timesteps with VCR Controls

Not a Can...

ParaView 5.10.0

Time: 0 max is 43

Layout #1

RenderView1

Properties (can.ex2)

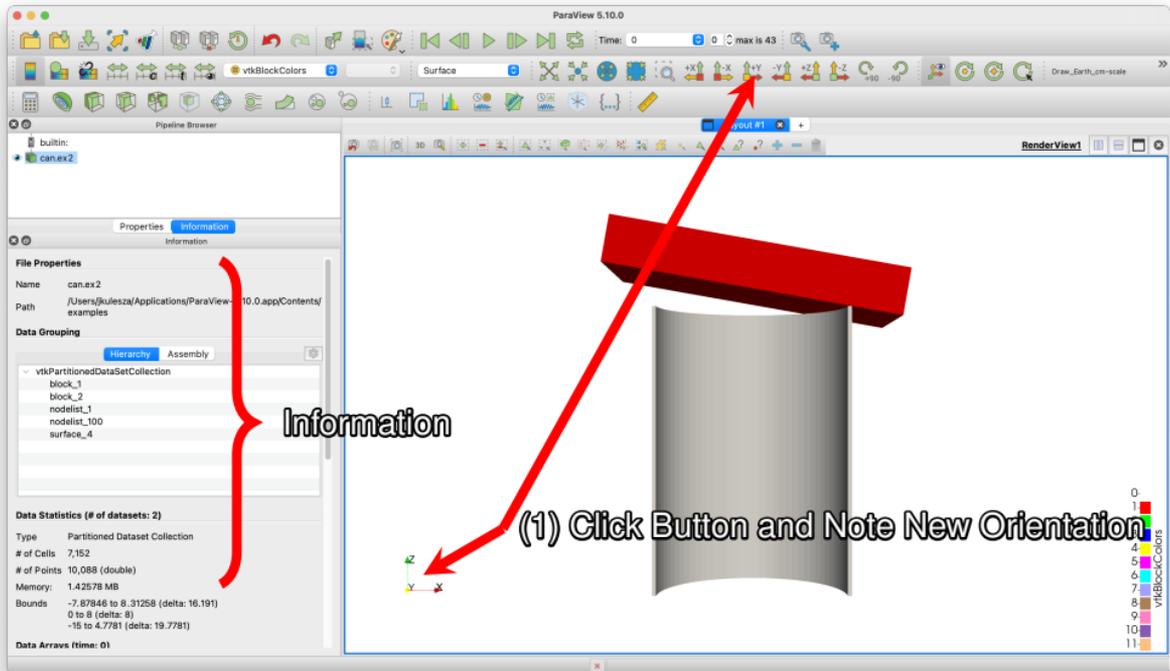
- Blocks
 - block_1
 - block_2
- Block Arrays
 - accl
 - displ
 - vel
 - eqps
- Sets
 - nodelist_1
 - nodelist_100
 - surface_4
- Set Arrays

Files

Run for Selected Files

0
1
2
3
4
5
6
7
8
9
10
11
vtkBlockColors

Moving Around / Controls (Information Tab)



Moving Around / Controls (Coloring & Legends)

(1) Edit Color Map

(2) Representation

(4) Edit Legend Properties

Discrete Coloring
Materials, Parts, etc.

(3) Name Blocks

Legend

Color Map Editor

Array Name: vtkBlockColors

Automatic Rescale Range: Grow and update on 'Apply'

Mode: Interpret Values As Categories

Rescale On Visibility Change

Annotations

| Value | Annotation |
|-------|------------|
| 1 | Block |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 7 | 7 |
| 8 | 8 |
| 9 | 9 |

Color Mapping Parameters

Color Space: RGB

Nan Color:

Nan Opacity: 1

Color Discretization

Can-Block

1 2 3 4 5 6 7 8 9 10 11

vtkBlockColors

File Properties

Name: can.ex2

Path: /Users/kulesza/Applications/ParaView-5.10.0.app/Contents/examples

Data Grouping

Hierarchy Assembly

vtkPartitionedDataSetCollection

- block_1
- block_2
- nodelist_1
- nodelist_100
- surface_4

Data Statistics (# of datasets: 2)

Type: Partitioned Dataset Collection

of Cells: 7,152

of Points: 10,088 (double)

Memory: 1.42578 MB

Bounds: -7.87846 to 8.31258 (delta: 16.191)
0 to 8 (delta: 8)
-15 to 4.7781 (delta: 19.7781)

Data Arrays (Time: 0)

Edit Color Map

Simple / Advanced

Orientation & Position
(Useful for Reproducible Plots)

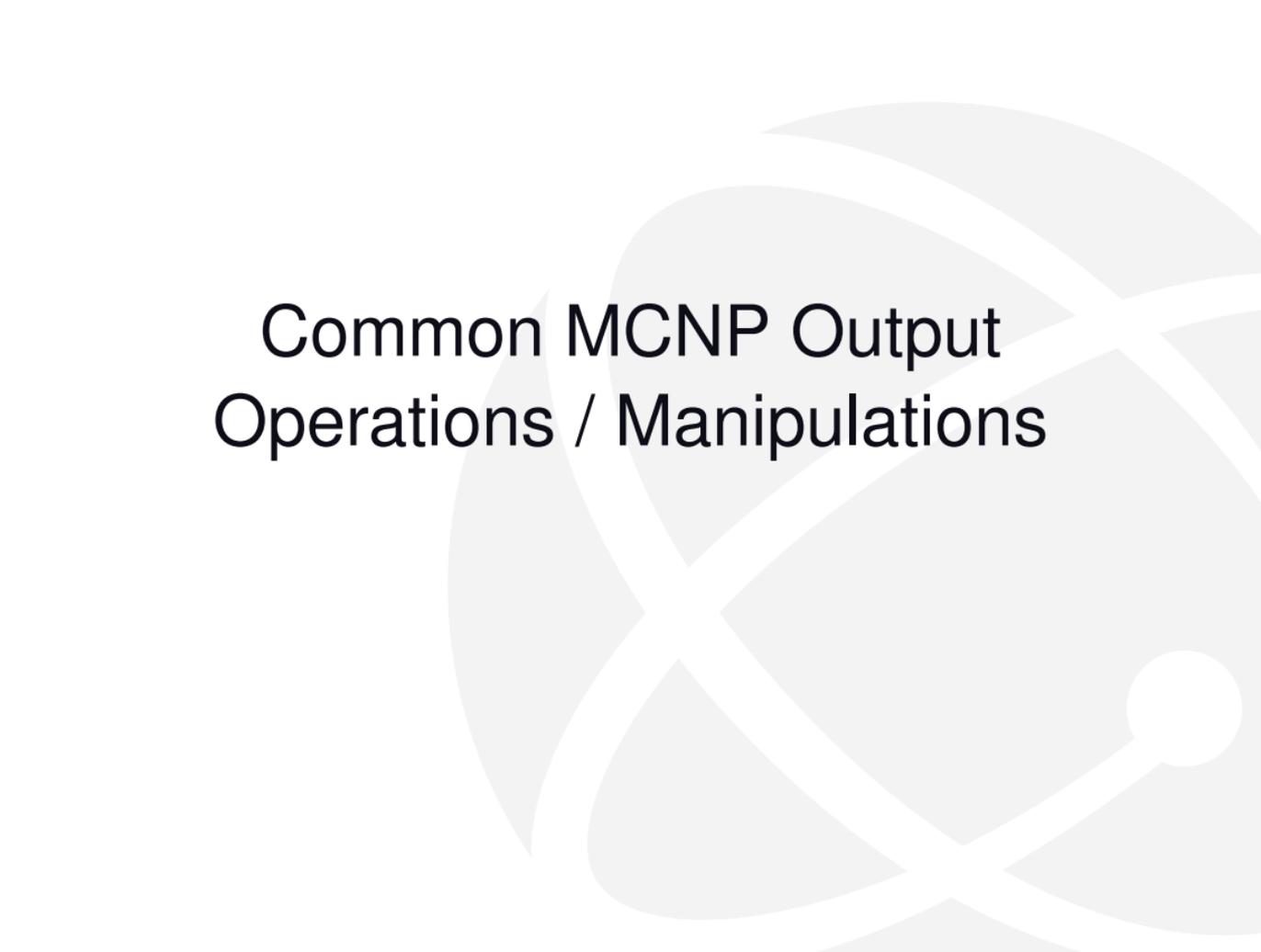
LaTeX Markup (Generally) Available

Few Font Choices (cf. LaTeX Font)

Sizing
(Useful for Reproducible Plots)

The image displays two versions of the 'Edit Color Legend Properties' dialog box. The left window is the 'Simple' view, and the right window is the 'Advanced' view. Red arrows point from the text annotations to the corresponding UI elements in both windows.

- Simple / Advanced:** A red arrow points from the top of the dialog boxes to the text 'Simple / Advanced'.
- Orientation & Position (Useful for Reproducible Plots):** A red arrow points from this text to the 'Orientation' dropdown menu in the 'Simple' view.
- LaTeX Markup (Generally) Available:** A red arrow points from this text to the 'Title Texts' section in the 'Simple' view.
- Few Font Choices (cf. LaTeX Font):** A red arrow points from this text to the font selection dropdown in the 'Text/Annotation Font Properties' section of the 'Simple' view.
- Sizing (Useful for Reproducible Plots):** A red arrow points from this text to the 'Label Format' field in the 'Labels' section of the 'Advanced' view.



Common MCNP Output Operations / Manipulations

Motivating Calculation Geometry & Results

- ▶ Focus: UM geometry and results
 - ▶ EEOU file converted to an ASCII Unstructured VTK File (.vtu) [22]
 - ▶ HDF5+XDMF output files (not available in MCNP6.2 and earlier)
- ▶ Focus: FMESH-based mesh tallies
 - ▶ MESHTAL file converted to an ASCII Structured VTK File (.vts)
 - ▶ HDF5+XDMF output files (not available in MCNP6.2 and earlier)
- ▶ The needed HDF5+XDMF files are provided
 - ▶ Conversion processes shown next for completeness
- ▶ Other possibilities (not covered here)
 - ▶ Particle tracks
 - ▶ Point data (collisions, fissions, etc.)
 - ▶ Voxelized CSG representations [23]
 - ▶ Weight-window mesh files

MCNP UM File Format Conversion

- ▶ MCNP6.2 and earlier: use conversion script electronically attached to [22]
- ▶ MCNP6.3: direct embed XDMF+HDF5 output via `hdf5file=filename`
- ▶ Example conversion execution looks like:

```
1 > ./Convert_MCNP_eeout_to_VTK.py caas_hybrid.mcnp.eeout
2 Processing caas_hybrid.mcnp.eeout...
3 Found 1 edit(s).
4 Processing ENERGY_6 edit...
5 Processing & Validating EDIT_6_RESULT_...
6 Maximum value: 1.19052e-02
7 Minimum positive value: 3.21896e-14
8 Minimum value: 0.00000e+00
9 Processing & Validating EDIT_6_ERROR_...
10 Maximum value: 1.00000e+00
11 Minimum positive value: 8.86076e-04
12 Minimum value: 8.86076e-04
```

- ▶ Resulting file: `caas_hybrid.mcnp.eeout.vtu`

MCNP Mesh Tally File Format Conversion

- ▶ MCNP6.2 and earlier: use `meshtal2vtk` script with MCNPTools 5.2.1+
- ▶ MCNP6.3: direct `fmesh` XDMF+HDF5 output via `out=xdmf`
- ▶ Example `meshtal2vtk` execution looks like:

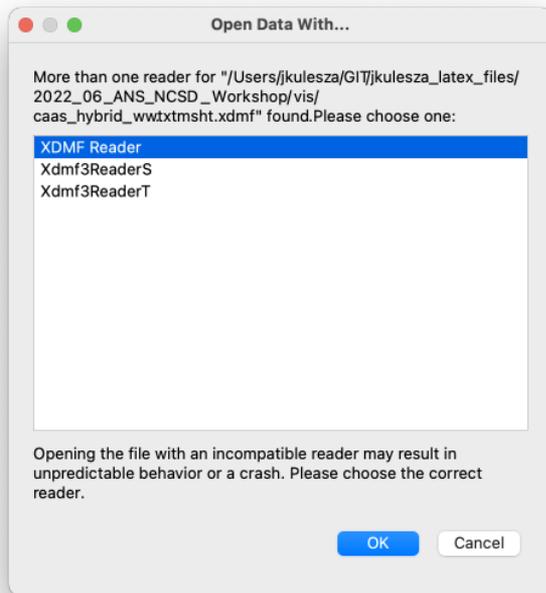
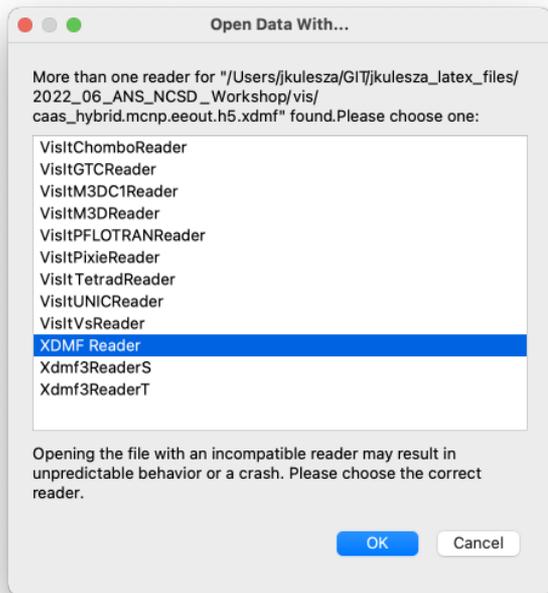
```
1 > meshtal2vtk meshtal
2 Processing mesh tally: 14 -> meshtal_14.vts
3   Smallest non-zero tally value: 1.58150e-03
4   Largest tally value: 3.57728e-02
```

- ▶ As before: the values are useful for setting log-scaled colorbar bounds

Introduction to XDMF [25, 26]

- ▶ ASCII, XML-formatted, file
 - ▶ Standard, albeit not actively developed, file format
 - ▶ Supports various structured and unstructured geometries
 - ▶ Can contain data and/or **contain pointers into HDF5 files**
- ▶ Cross-platform & open-source support (direct visualization of output)
 - ▶ ParaView [21]
 - ▶ VisIt [24]
- ▶ No additional MCNP library dependencies
 - ▶ MCNP outputs are XDMF version 2

Loading XDMF Version 2 Files



Logarithmically Scaled Coloring & Rename Legend

(1) Rename Legend

(2) Set Custom Range
 $(9e-13 < D < 1.2e-2)$
 (or $1e-12 < D < 1e-2$)

(3) Enable Log Scaling

(4) Choose Different Color Map

Color Map Editor

Array Name: edit_6_particle_1_value

Automatic Rescale Range Mapping: Grow and update on Array

Mapping Data: Select a color map to use for this array

Data:

- Enable Freehand Drawing Of Opacity Transfer Function
- Use Log Scale When Mapping Data To Colors
- Enable Opacity Mapping For Surfaces
- Use Log Scale When Mapping Data To Opacity

Data Histogram

- Display Data Histogram
- Automatically recalculate data histogram

Number of Bins: 10

Color Mapping Parameters

Color Space: Diverging

Energy Deposition (eV)

1.0e-02
0.001
0.0001
1e-6
1e-7
1e-8
1e-9
1e-10
1e-11
1.0e-12

| Name | Type | Ranges |
|-------------------------|--------|---|
| edit_6_particle_1_error | double | [0.00028047175555678344,1] |
| edit_6_particle_1_value | double | [0.0,0.018783942675251] |
| element_volume | double | [94378.22625211787,2765914.0] |
| cell_id | int | [1,9] |
| mass_density | double | [2.3856386267549374,2.3856386267549374] |
| material_id | int | [1,1] |

Color Map Selection

(1) Search for, and select, "Plasma"

Search ... (use Esc to clear text) Default

Options to load:
 Colors
 Opacities
 Use preset range

Actions on selected:
 Show current preset in default mode

| Presets | Presets |
|--|---|
|  Cool to Warm |  Divergent Color Map (Plasma) |
|  Black-Body Radiation |  X Ray |
|  Inferno (matplotlib) |  Ramped Color Map (Black, Blue and White) |
|  Blue Orange (divergent) |  Viridis (matplotlib) |
|  Gray and Red |  Linear Green (Gr4L) |
|  Cold and Hot |  Blue - Green - Orange |
|  Rainbow Desaturated |  Ramped Color Map (Yellow - Gray - Blue) |
|  Rainbow Uniform |  Turbo |

(2) Apply Color Map

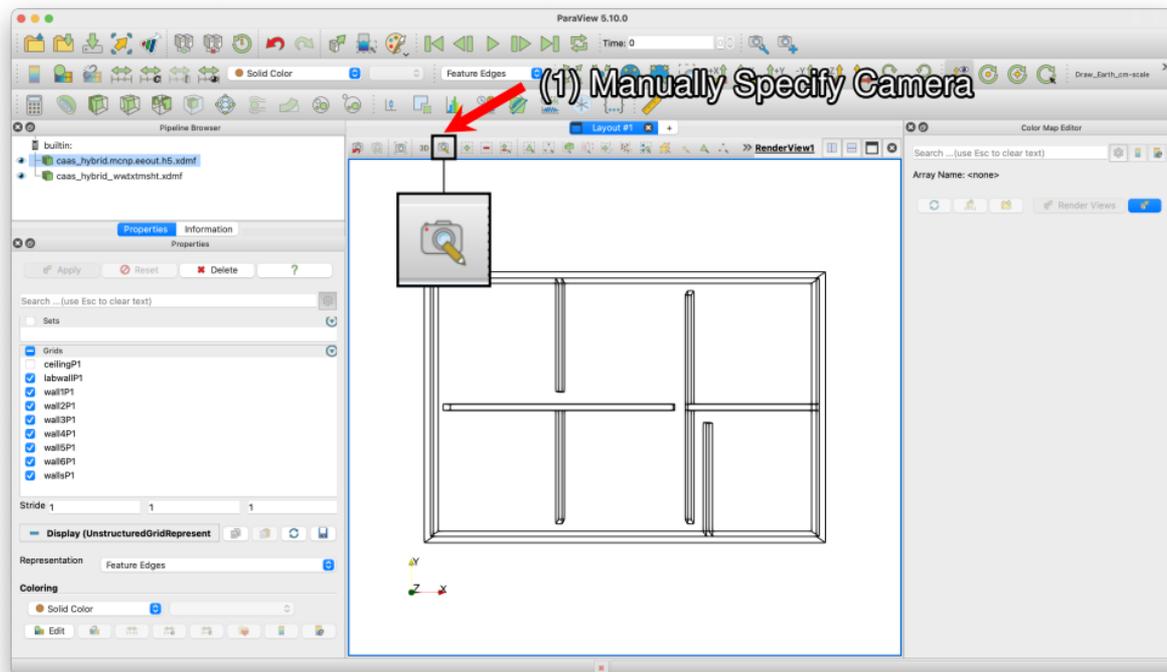
Can Import Custom JSON-formatted Maps

Apply
Import
Export
Remove
Close

Tip: <click> to select, <double-click> to apply a preset.

Lots of Options!

UM Feature-edges Only



Specify Camera Settings

Adjust Camera

Standard Viewpoints

Custom Viewpoints

1 + ← (2) Save Camera Preset

Center of Rotation

300 -825 130

Reset Center of Rotation When Camera is Reset

Rotation Factor

1

Camera Parameters

| | | | |
|-------------|-----------|----------------|----------|
| Position | 2560 | -3730 | 2220 |
| Focal Point | 450 | -950 | -150 |
| View Up | -0.349159 | 0.438962 | 0.827889 |
| View Angle | 30.00 | Eye Angle | 2.00 |
| Focal Disk | 1.00 | Focal Distance | 0.00 |

Load Save

Manipulate Camera

Roll 0.00 degrees

Elevation 0.00 degrees

Azimuth Roll Focus Zoom In

Close

ParaView 5.10.0

Time: 0

Layout #1

RenderView1

(1) Specify Camera Settings

Operations of Interest

- ▶ Clips and slices
- ▶ Isocontours
 - ▶ Cell-to-point data conversion
 - ▶ Lighting considerations
 - ▶ Overlaying relative fractional uncertainties
 - ▶ Probing & manually annotating values
- ▶ Assessing mesh quality
 - ▶ Segregating and finding particular elements

Clip (Hide Half Space)

(1) Click "Clip" Filter
(or Select from "Filter" Menu)

(2) Disable Plane to Avoid Accidents

(3) Specify Position & Orientation

Useful Options

ParaView 5.10.0
Time: 0
Draw_Earth_om_scale

Pipeline Browser
builtin:
caas_hybrid_mcnp.eeout.h5.xdmf
caas_hybrid_vvwtmsht.xdmf
Clip1

Properties
Apply Reset Delete ?
Search ... (use Esc to clear text)

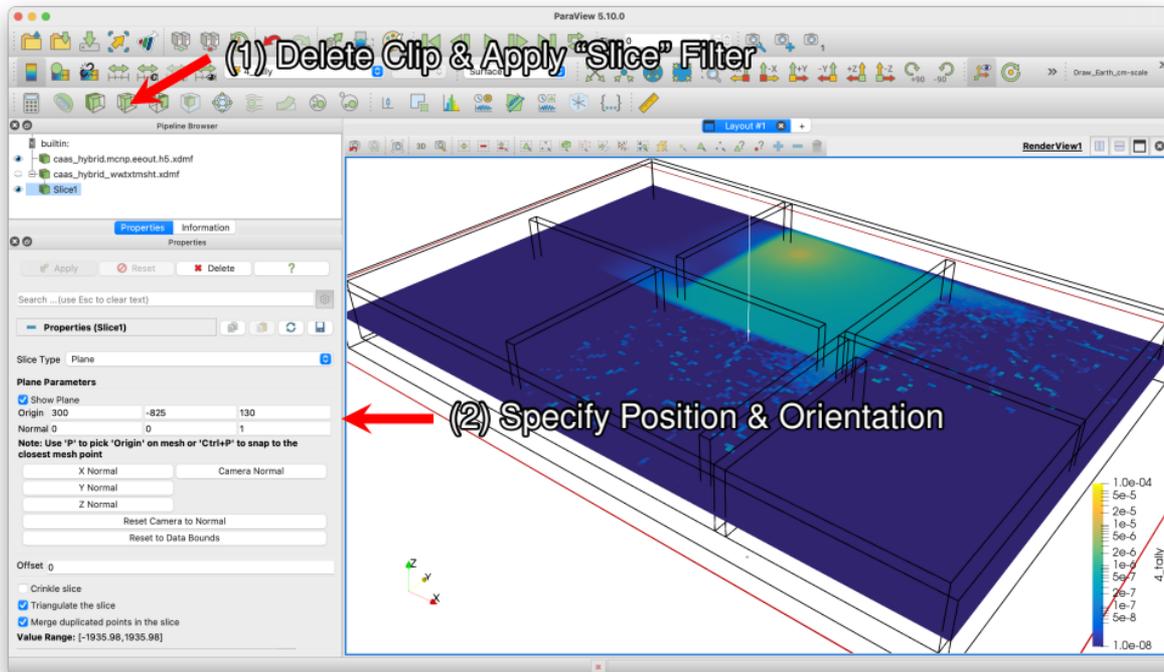
Properties (Clip1)
Clip Type: Plane
Plane: meters
 Show Plane
Origin: 300 -825 130
Normal: 0 0 1
Note: Use 'P' to pick 'Origin' on mesh or 'Ctrl+P' to snap to the closest mesh point
X Normal: Camera Normal
Y Normal:
Z Normal:
Reset Camera to Normal
Reset to Data Bounds

Offset: 0
 Invert
 Crinkle clip

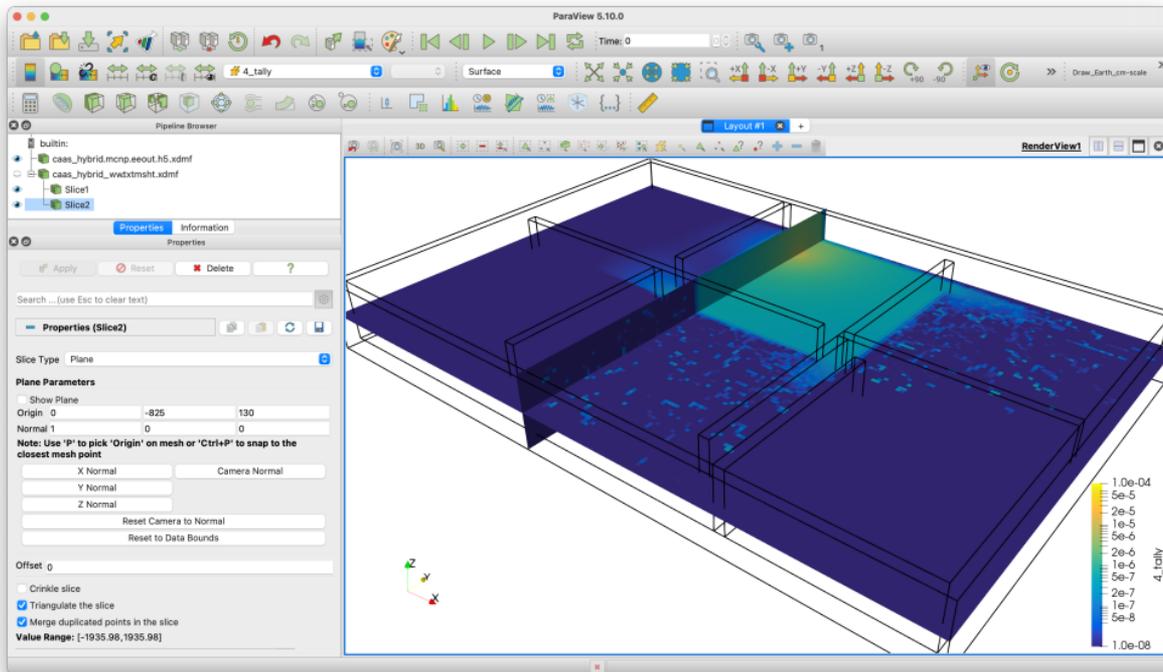
Display (UnstructuredGridRepresent)

RenderView1
1.0e-04
5e-5
2e-5
1e-5
5e-6
2e-6
1e-6
5e-7
2e-7
1e-7
5e-8
1.0e-08
4, fully

Slice



Another Slice ($x, y, z = 0, -825, 130 \perp x$)





Isocontours & Lighting Considerations

Apply Contour Filter and “Contour By” Value

(1) Convert Cell Data to Point Data (via Filter)

(2) Apply Contour Filter

(3) Contour by Values

ParaView 5.10.0

Time: 0

4_tally (?)

Surface

Layout #1

RenderView1

built-in:

- cas_hybrid_mcnp.eeout.h5.xdmf
- cas_hybrid_id_wvtxmsh.t.xdmf
- CellDataToPointData1
- Contour1

Properties

Apply Reset Delete ?

Search ... (use Esc to clear text)

Properties (Contour1)

Contour By: 4_tally

Compute Normals

Compute Gradients

Compute Scalars

Output Points: Same as input

Precision: [dropdown]

Generate Triangles

Isosurfaces

Value Range: [0.7, 8543e-05]

Point Locator

Point Merge Method: Uniform Binning

Divisions: 50 50 50

Number of points: ..

1.0e-04

5e-5

2e-5

1e-5

5e-6

2e-6

1e-6

5e-7

2e-7

1e-7

5e-8

2e-8

1.0e-08

4_tally

Choose Isocontour Values

(3) Enter Upper/Lower Extreme Values

(4) Select Spacing

(5) Select Number of Levels

(6) Generate to Create The Levels

(7) Apply

(1) Remove Existing Value(s)

(2) Generate Number Series for Contour Levels

Generate Number Series

Range: 1e-08 - 0.0001

Type: Logarithmic

Number of Samples: 14

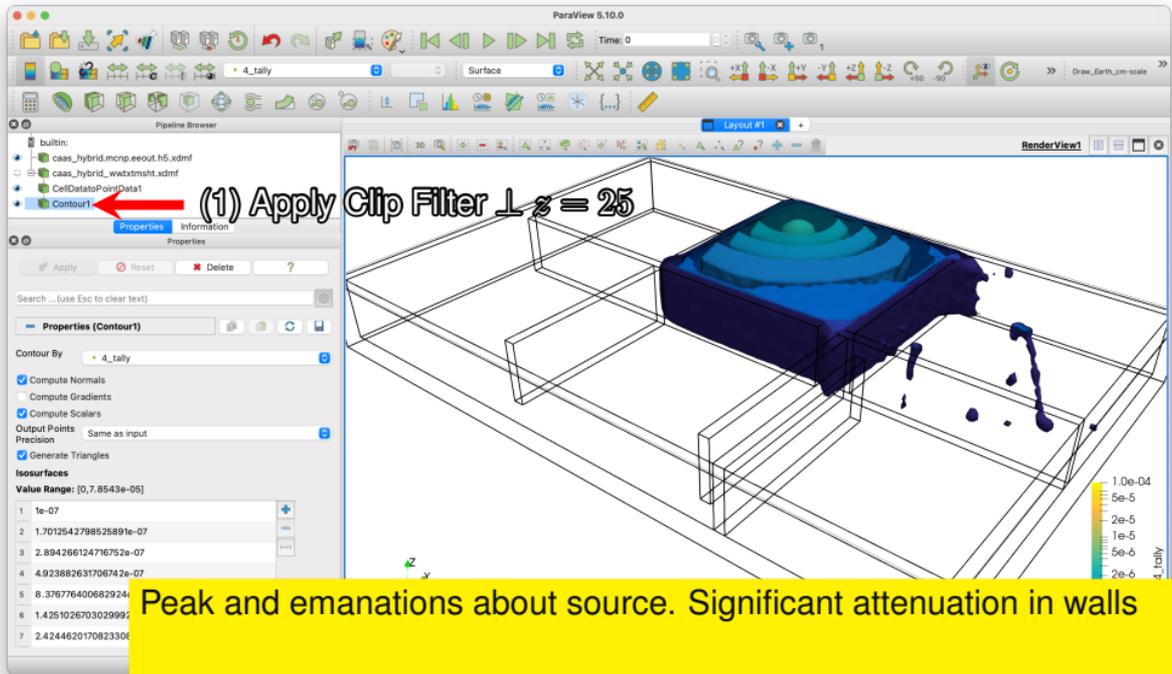
Sample series: 1e-08, 2.03092e-08, 4.12463e-08, 8.37678e-08, 1.70125e-07, 3.45511e-07, 7.01704e-07, 1.4251e-06, 2.89427e-06, 5.87802e-06, 1.19378e-05, 2.42446e-05, 4.92388e-05, 0.0001

Cancel Generate

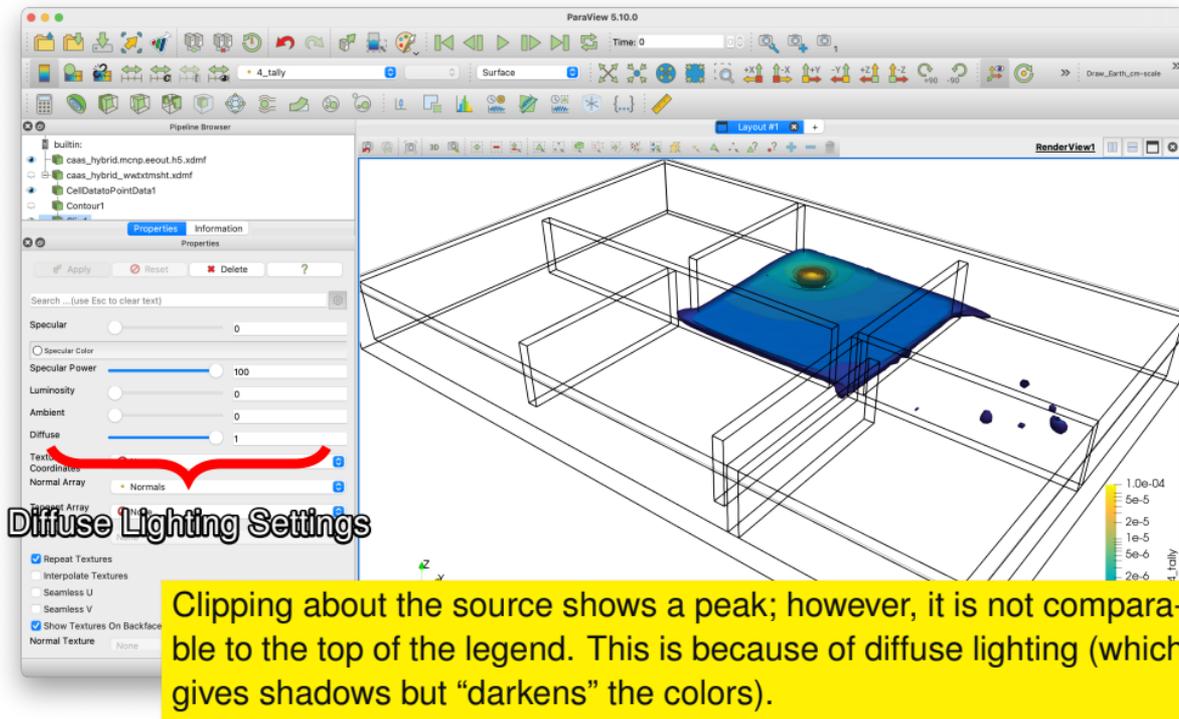
1.0e-04
5e-5
2e-5
1e-5
5e-6
2e-6
1e-6
5e-7
2e-7
1e-7
5e-8
2e-8
1.0e-08

4, tally

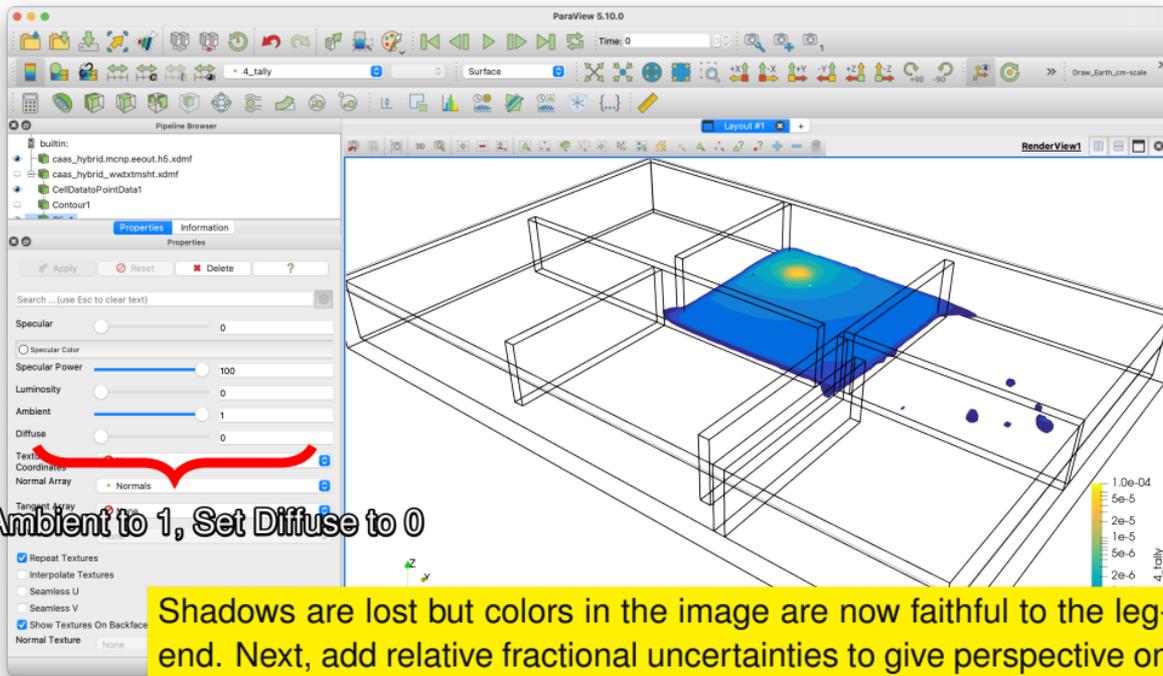
Observe Isocontour Results & Trends



Clip Isocontour & Observe Shadows



Adjust from Diffuse to Ambient Lighting



Add Relative Fractional Uncertainties

(4) Set Appropriate Range of Values

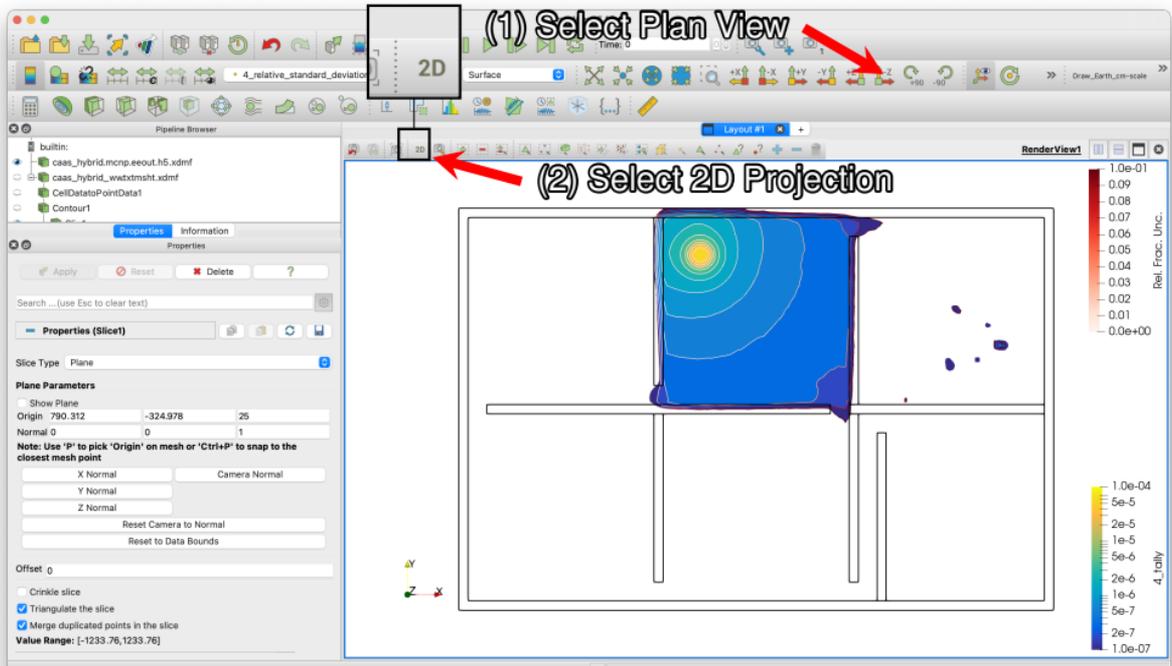
(2) Select Tally "Errors"

(1) Apply Slice Filter to Contour

(3) Ensure Consistent Position with Earlier Clip

(5) Set Ambient to 1, set Diffuse to 0, set line width to 2.

2D Plan View



Gives a more “traditional” representation of geometry and results.

Assessing Mesh Quality



What is “Mesh Quality”

- ▶ Mesh element “quality” can be assessed by a variety of metrics
 - ▶ Deterministically calculated
 - ▶ Useful for mesh-generation algorithms
 - ▶ Useful for deterministic engineering calculations
- ▶ Body of knowledge mainly for linear tetrahedra and hexahedra
 - ▶ Verdict library [9] summarizes these metrics
 - ▶ ParaView incorporates Verdict
- ▶ Useful for Monte Carlo transport to identify undesirable elements
 - ▶ Negative-Jacobian elements have negative volume
 - ▶ Problematic for edits
 - ▶ “Paper-thin” elements can challenge particle tracking

Mesh Quality Filter & Finding Elements

The image shows a screenshot of the ParaView 5.10.0 software interface. The main window displays a 3D model of a meshed rectangular structure with a color scale on the right ranging from 10 to 4.2e+01. The interface includes a Pipeline Browser on the left, a Properties panel, and a RenderView window. Three red arrows point to specific elements in the interface, labeled with numbered instructions:

- (1) Add Mesh Quality Filter: Points to the 'MeshQuality1' filter in the Pipeline Browser.
- (2) Select Metric: Points to the 'Radius Ratio' metric in the Properties panel under 'Triangle Quality Measure'.
- (3) Select Coloring, Scaling, etc.: Points to the 'Quality' coloring option in the Properties panel.

A yellow banner at the bottom of the screenshot contains the following text: "Identify elements with high Radius Ratios (poorest quality). To do this: (4) Edit menu: Find Data."

Find Data Dialog Use

The screenshot shows the 'Find Data' dialog box with the following configuration:

- Create Selection** (tab)
- Selection Criteria**: Data Producer: MeshQuality1, Element Type: Cell
- Selection Qualifier**: Quality is >= 42
- Selection Qualifier** (list):
 - Block Selection: vtkMultiBlock (checked)
 - wallsP1 (checked)
 - labwallP1 (checked)
 - wall1P1 (checked)
 - wall2P1 (checked)
 - wall3P1 (checked)
 - wall4P1 (checked)
 - wall5P1 (checked)
 - wall6P1 (checked)
 - ceilingP1 (checked)
- Find Data** button (highlighted with a red arrow)
- Selected Data (MeshQuality1)** (tab)
- Attribute: Cell Data** (table)

| Block Name | Cell ID | Cell Type | Quality | edit_6_particle_t_error | edit_6_particle_t_value | element_volume |
|------------|---------|-------------|---------|-------------------------|-------------------------|----------------|
| wall4P1 | 21 | tetrahedron | 42.1136 | 0.00178762 | 0.000325242 | 556914 |
| wall3P1 | 22 | Tetrahedron | 42.1136 | 0.0421464 | 2.1389e-05 | 556914 |
| wall4P1 | 9 | Tetrahedron | 42.0761 | 0.129316 | 1.70814e-06 | 568086 |
| wall6P1 | 9 | Tetrahedron | 42.1136 | 0.364664 | 5.44778e-07 | 556914 |
| wall6P1 | 10 | Tetrahedron | 42.0761 | 0.284701 | 1.97301e-07 | 568086 |

Buttons: Freeze, Extract, Plot Over Time

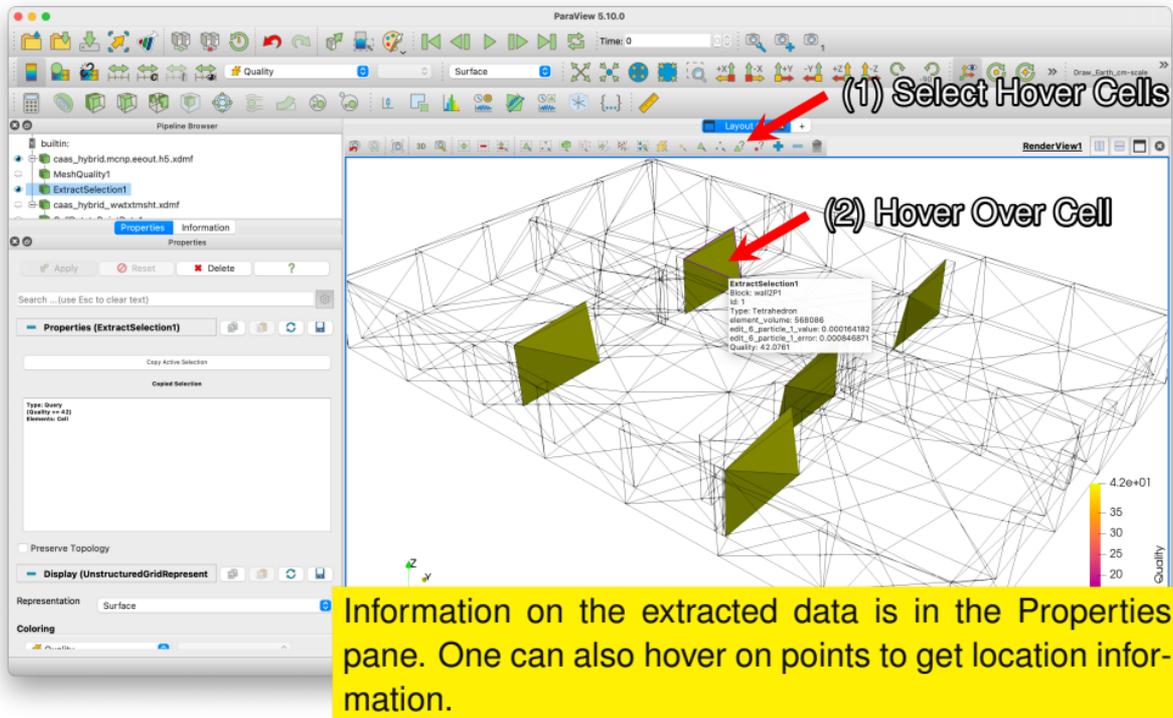
(1) Select Pipeline Item (points to MeshQuality1)

(2) Select Data Set & Conditions (points to the Selection Qualifier list)

(3) Execute Selection (points to the Find Data button)

(4) Extract Selection. (highlighted in yellow)

Viewing & Interrogating Specific Elements



(1) Select Hover Cells

(2) Hover Over Cell

Information

Properties

Apply Reset Delete ?

Search ... (use Esc to clear text)

Properties (ExtractSelection1)

Copy Active Selection

Copy Selection

Type: Quality (Quality == A3)
Elements: Cell

Preserve Topology

Display (UnstructuredGridRepresent)

Representation Surface

Coloring

4.2e+01
35
30
25
20
Quality

ExtractSelection1
Block: wal2P1
id: 1
Type: Tetrahedron
element_volume: 568086
edit_0_particle_1_value: 0.000164182
edit_0_particle_1_error: 0.000846871
Quality: 42.0761

Information on the extracted data is in the Properties pane. One can also hover on points to get location information.

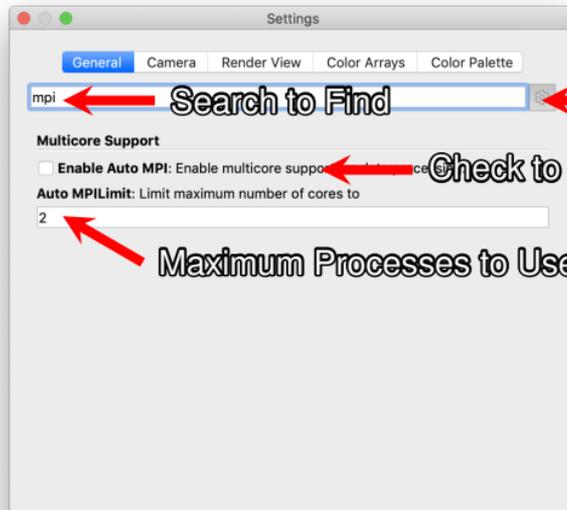
Other (Advanced?) ParaView Topics



Brief Overview of Other Topics

- ▶ MPI Parallelism
- ▶ Toggling Geometry via Blocks
- ▶ Saving ParaView State
- ▶ Camera Linking
- ▶ Recording Traces (Macros)
- ▶ Client-server Connections
- ▶ Animation

MPI Parallelism (Settings, General Tab)



Advanced
Options

Check to Enable, Restart to Use

Maximum Processes to Use

Requires MPI on your system (and running). Also, depending on your system firewall settings, you may see a number of MPI processes launch and be forced to dismiss allow/block requests. This means: start with a few processes to see how your system behaves.

Saving ParaView State (File Menu)

The image shows the ParaView 5.10.0 interface. The File menu is open, with 'Save State...' highlighted. A red arrow points to this option. A text box 'Can Reduce Rework' with a pencil icon is positioned over the menu. Below the main menu, the 'Load State Options' dialog is open, with 'Use File Names From State' selected. A red arrow points to this option. A text box 'Can Also Load State and Select Different Data' is positioned over the dialog. The background shows a 3D wireframe model of a rectangular structure with several green tetrahedron elements highlighted. A tooltip for one of these elements is visible, showing details like 'Type: Tetrahedron', 'element_volume: 568086', and 'Quality: 42.0761'. A color scale legend on the right indicates 'Quality' values from 0 to 42e+01.

Can Reduce Rework

Can Also Load State and Select Different Data

File Menu Options:

- Open... ⌘O
- Recent Files
- Reload Files
- Load State...
- Save State...**
- Save Data... ⌘S
- Save Screenshot...
- Export Scene...
- Save Animation...
- Save Geometry...
- Export Now
- Import Cinema Image Database...
- Load Path Tracer Materials...
- Load Window Arrangement...
- Save Window Arrangement...
- Connect...
- Disconnect

Load State Options:

- Use File Names From State
- Search files under specified directory
- Choose File Names

Camera Linking (Tools Menu)

(1) Select A Render View, and Click This

Copy/Paste Properties to Propagate Settings

(2) Name the Link

Click on another view to link with.

Name:

Interactive View Link

Cancel

Tools Menu:

- Create Custom Filter...
- Add Camera Link...**
- Link with Selection
- Manage Custom Filters...
- Manage Links...
- Manage Plugins...
- Manage Favorites...
- Customize Shortcuts...
- Record Test...
- Play Test...
- Lock View Size
- Lock View Size Custom...
- Timer Log
- Start Trace

Display (UnstructuredGridRepresent)

Representation: Surface

Coloring

Quality

Scalar Coloring

- Map Scalars
- Interpolate Scalars Before Mapping

Styling

Reset Camera

Recording Macros (Traces, Tools Menu)

- Create Custom Filter...
- Add Camera Link...
- Link with Selection
- Manage Custom Filters...
- Manage Links...
- Manage Plugins...
- Manage Favorites...
- Customize Shortcuts...
- Record Test...
- Play Test...
- Lock View Size
- Lock View Size Custom...
- Timer Log
- Start Trace**



Trace Options

General options

Properties To Trace On Create: Select which properties to save in trace

any *modified* properties

Color maps, color bar etc.

Fully Trace Supplemental Proxies: Fully trace supplemental objects, such as color maps, color bars, etc. when accessed the first time in trace.

Miscellaneous

Show Incremental Trace: Show trace as it is being generated.

Cancel OK



Stop Trace

```
untitled.py - Script Editor
# trace generated using paraview version 5.8.0
#
# To ensure correct image size when batch processing, please search
# for and uncomment the line " # renderView1.ViewSize = [*,*]"
#### import the simple module from the paraview
from paraview.simple import *
#### disable automatic camera reset on 'Show'
paraview.simple.DisableFirstRenderCameraReset()

# get active source
run3mcppeouth5xdmf = GetActiveSource()

# get active view
renderView1 = GetActiveViewOrCreate('RenderView')
# uncomment following to set a specific view size
# renderView1.ViewSize = [2392, 1454]

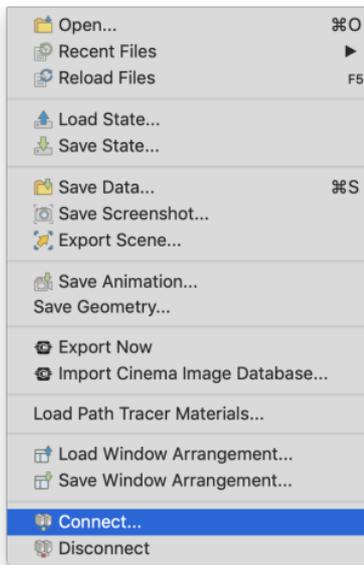
# get layout
layout1 = GetLayout()

# get display properties
run3mcppeouth5xdmf[Display] = GetDisplayProperties(run3mcppeouth5xdmf, view=renderView1)

# change representation type
Ready
```

Client-server Connections

- ▶ Negates downloading data
- ▶ Provide parallel processing via MPI
- ▶ Institution-specific configuration
- ▶ LANL uses a PVSC configuration file
- ▶ https://hpc.lanl.gov/paraview_usage
- ▶ Remote memory monitoring
 - ▶ View: Memory Inspector



Animation

(1) Enable View

(2) Select Track

(3) Select Type

Python tracks can perform scripted actions every frame (every tick).