LA-UR-22-31281
Approved for public release; distribution is unlimited.

Title: MCNP6.3 Fission Matrix Interrogation

Author(s): Kulesza, Joel A.
Josey, Colin James

Intended for: 2022 MCNP User Symposium, 2022-10-17/2022-10-21 (Los Alamos, New Mexico, United States)

Issued: 2022-10-24

Los Alamos NATIONAL LABORATORY

MCNP6.3 Fission Matrix Interrogation

Joel A. Kulesza & Colin J. Josey

Los Alamos National Laboratory

2022 MCNP® User Symposium
October 17–21, 2022

Outline

Motivation and Introduction to HDF5

MCNP HDF5 Runtape Organization Overview
- An Aside on Compressed Sparse-row Storage

Fission Matrix Example: Pool Critical Assembly
1. Extract Mesh Data
2. Calculate and Sort Eigenvalues & Eigenvectors
3. Plot Eigenvalues & Eigenvectors

Motivation

- MCNP output exists in many forms with various uses
  - Output file (`outp`)
    - Human-readable collection of relevant results
  - Data files (EEOUT, `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
    - **The fission matrix is stored in the runtape**
  - Other files undergoing development and implementation

Introduction to HDF5 [1]

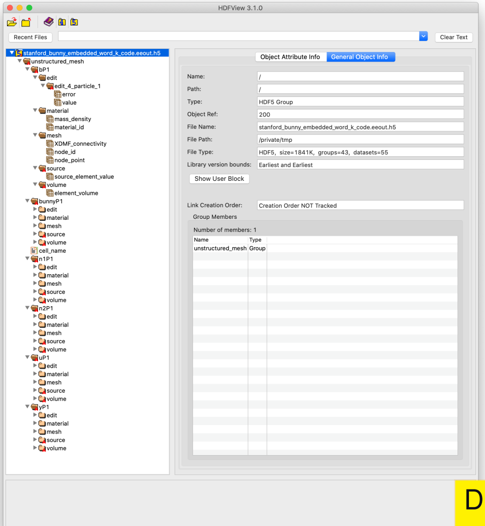
- HDF: Hierarchical Data Format
- Developed by The HDF Group
  - Non-profit organization
  - Spun off from Nat'l Center for Supercomp. Appl. at Univ. of Illinois
  - Central authority to ensure quality and prevent fragmentation
- BSD-like license, freely available, portable, numerous APIs
  - Official APIs: C, C++, Fortran, Java
  - Unofficial APIs: Julia, Matlab, Mathematica, Perl, Python, R
- Developed with speed and scalability in mind
- Three major objects: **groups**, **datasets**, and **attributes**
  - Groups are containers for datasets or other groups
  - Datasets are homogeneous n-dimensional arrays
    - Can contain complex objects, e.g., images
  - Attributes can be added to either groups or datasets

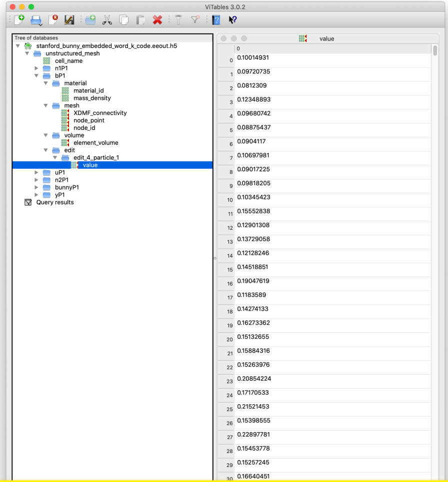
History of HDF5

- 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

Interactive Interrogation—HDFView & ViTables

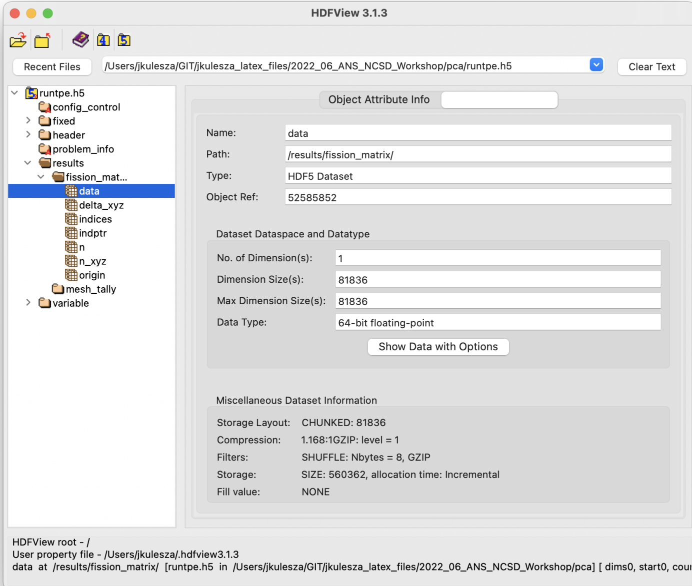




Despite being binary, there are a variety of cross-platform and freely available tools to interactively explore the HDF5 data files.

https://www.hdfgroup.org/downloads/hdfview
https://vitables.org/Download

MCNP HDF5 Runtape Organization



`runtpe.h5/results/fission_matrix`

1-D data? Compressed-sparse storage!

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)

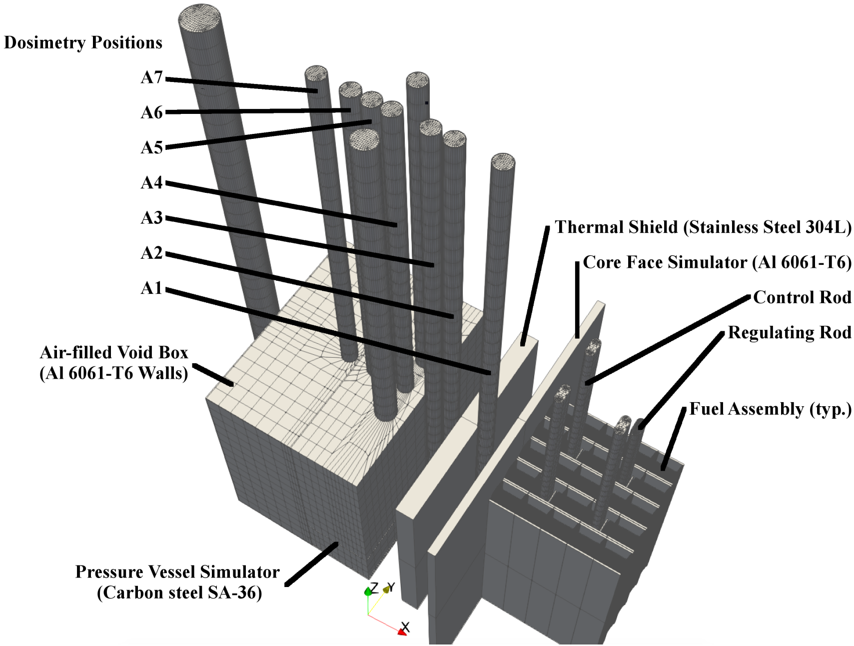
`colind:` (0 1 2 3 0 1 2 0 1 2 0 3 4 5 4 5)

`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)

Los Alamos NATIONAL LABORATORY

Image from:
Elafrou et al., “A lightweight optimization selection method for Sparse Matrix-Vector Multiplication”, https://arxiv.org/abs/1511.02494.

Fission Matrix Interrogation Example [2–4]



Input file available at https://doi.org/10.2172/1601379

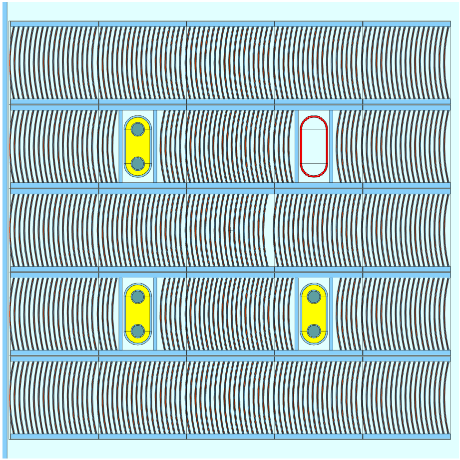
Fission Matrix Input Example

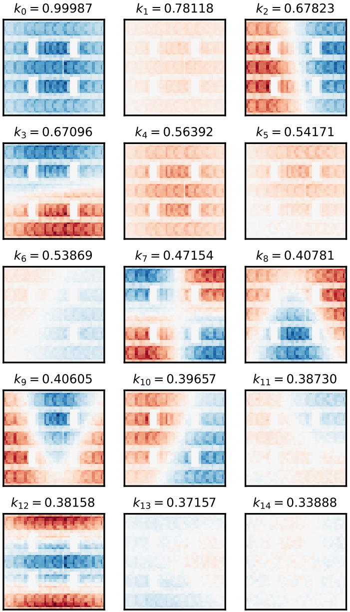
- Direct control can create a “preferred” fission-matrix mesh
- ORNL PCA example:

```
kcode 250000 1.000 50 250
kopts fmat=yes
hsrc   40 -20.41    20.25
       40 -19.275   19.275
       75 -36.35375 36.35375
```

- Beware of file size: this HDF5 runtape is several gigabytes

Fission Matrix Visualization Example





Process to generate: extract mesh data, calculate and sort eigenvalues and eigenvectors, and plot (coming up next...).

Python & MCNP Runtape Preliminaries

```
#!/usr/bin/env python3

import h5py
import matplotlib.pyplot as plt
import numpy as np
import scipy.sparse as sparse
import scipy.sparse.linalg as sla
from matplotlib.cm import get_cmap

SUPPORTED_RUNTAPE = [1, 0, 0]
```


Complete script is electronically attached to this PDF and is from Kulesza and Josey, "Fission Matrix Processing Using the MCNP6.3 HDF5 Restart File", https://doi.org/10.2172/1867160.

1. Extract Mesh Data

```python
def extract_fmat(runtape):
    """Returns the last saved fission matrix as a scipy.sparse.csr_matrix"""
    with h5py.File(runtape, "r") as handle:
        # Check runtape version
        version_file = handle["config_control"].attrs["version_file"]
        if any(SUPPORTED_RUNTAPE != version_file):
            print("Possibly incompatible runtape detected.")

        fmat = handle["results/fission_matrix"]

        n_dim = fmat["n"][()]
        indices = fmat["indices"][:]
        indptr = fmat["indptr"][:]
        data = fmat["data"][:]

        n_xyz = fmat["n_xyz"][:]
        delta_xyz = fmat["delta_xyz"][:]
        origin = fmat["origin"][:]

    return ( sparse.csr_matrix((data, indices, indptr), shape=(n_dim, n_dim)),
        n_xyz, delta_xyz, origin )
```


SciPy CSR matrix:
https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.csr_matrix.html

2. Calculate and Sort Eigenvalues & Eigenvectors

```
def plot_eigs(mat, n_xyz, n_tot=6, n_col=2):
    """Retrieve eigenvalues/vectors, sort, reshape to 3-d object, and plot."""
    eigenvalues, eigenvectors = sla.eigs(mat, k=n_tot)

    # Clean up and sort the eigenvectors.
    sorted_eigvals = []
    sorted_eigvecs = []
    for i in np.argsort(-np.abs(eigenvalues)):
        val = eigenvalues[i]
        val = np.real(val) if np.real(val) == val else val
        sorted_eigvals.append(val)
        vec = np.real(eigenvectors[:, i].reshape(n_xyz[::-1]).transpose())
        if len(sorted_eigvals) == 1:
            vec = np.abs(vec)
        sorted_eigvecs.append(vec)
```


SciPy sparse linear algebra eigenvalue calculation:
https://docs.scipy.org/doc/scipy/reference/generated/scipy.sparse.linalg.eigs.html

3. Plot Eigenvalues & Eigenvectors

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

Questions?

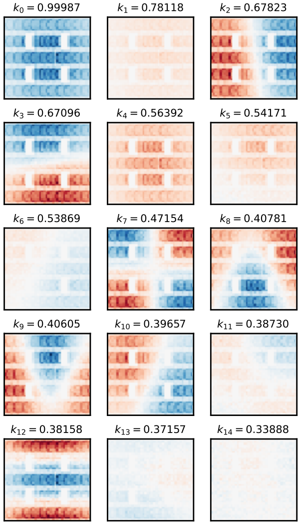
Motivation and Introduction to HDF5

MCNP HDF5 Runtape Organization Overview

- An Aside on Compressed Sparse-row Storage

Fission Matrix Example: Pool Critical Assembly

1. Extract Mesh Data
2. Calculate and Sort Eigenvalues & Eigenvectors
3. Plot Eigenvalues & Eigenvectors



Backup Slides

Table of Contents

References

[1] The HDF Group, “Hierarchical Data Format, Version 5,” Website, Apr. 2020. [Online]. Available: http://www.hdfgroup.org/HDF5/

[2] I. Remec and F. B. K. Kam, “Pool Critical Assembly Pressure Vessel Facility Benchmark,” Oak Ridge National Laboratory, Oak Ridge, TN, USA, Tech. Rep. NUREG/CR-6454, Jul. 1997. [Online]. Available: https://www.osti.gov/biblio/515584-pool-critical-assembly-pressure-vessel-facility-benchmark

[3] J. A. Kulesza and R. L. Martz, “Evaluation of the Pool Critical Assembly Benchmark with Explicitly Modeled Geometry Using MCNP6,” *Nuclear Technology*, vol. 197, no. 3, pp. 284–295, Mar. 2017. [Online]. Available: http://www.ans.org/pubs/journals/nt/a_39691

[4] ——, “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 (ISRD16)*. Santa Fe, NM, USA; May 7–12: ASTM International, 2017, accompanying presentation: LA-UR-17-23854.