LA-UR-23-31033

Approved for public release; distribution is unlimited.

Title: Neutron (and other Particle) Transport at LANL: An Overview

Author(s): Ortega, Mario Ivan

Intended for: University of Tennessee Nuclear Engineering Colloquium

Issued: 2023-09-27

Los Alamos NATIONAL LABORATORY

Neutron (and other Particle) Transport at LANL: An Overview

Mario I. Ortega
University of Tennessee Nuclear Engineering Colloquium, Knoxville, TN

10/4/2022
LA-UR-23-xxxxx

Abstract

For decades, Los Alamos National Laboratory has been at the forefront of neutron transport methods research and code development. One such code is PARTISN, the LANL parallel time-dependent discrete ordinate neutron transport code. In this presentation, we describe the various research efforts currently underway by the PARTISN and other code teams. Some examples of current research are a block automated mesh refinement scheme, the application of tensor trains to the discretized neutron transport equation, and GPU code porting. The block automated mesh refinement scheme uses cross section information to refine and coarsen the solution mesh to improve time to solution and reduce memory. The tensor train approach expresses discretized transport operators as tensor products of vectors and matrices to compress the size of linear systems being solved by transport codes. Rather than relying on matrix-free methods such as the transport sweep, we have access to an operator that can be inverted, reshaped, or manipulated algebraically. Finally, we describe how PARTISN is used, what problems we are looking to solve, and what the future holds for neutron transport at LANL. In addition to this, we briefly describe the various research efforts in other particle transport teams using both deterministic and Monte Carlo methods. In the presentation, we list possible opportunities for collaboration between the laboratory and faculty and students.

Outline

- Overviews
  - About Me
  - Los Alamos National Laboratory
  - Computational Physics and Methods Group, CCS-2
- Radiation Transport
  - Thermal X-ray transport
    - CAPSAICIN and JAYENNE
  - Neutron transport
    - PARTISN
- Block AMR – Block Automated Mesh Refinement
- Tensor Train Neutron Transport
- Machine Learning for Group Structures
- Come work for us!
- Questions

About Me

- Scientist at LANL/PARTISN team member since October 2019.
- BS and MS in Nuclear Engineering from the University of New Mexico.
- PhD in Nuclear Engineering from the University of California, Berkeley
- Alumnus of the DOE Computational Science Graduate Fellowship.

Los Alamos National Laboratory

- Originally founded during WWII as part of the Manhattan Project
- Currently about 14,500 employees, about 1/5 of which are Ph.D. researchers (460 postdocs)
- About 1,800 students during the summer (high school, undergraduate, and graduate)
- Run by Triad National Security (Battelle, Texas A&M, and University of California)
- Funded primarily by the U.S. Department of Energy National Nuclear Security Administration

LANL Science Mission

- Our science mission is organized into four pillars
- **Information Science and Technology** - leverages advances in theory, algorithms, and the exponential growth of high-performance computing to accelerate the integrative and predictive capability of the scientific method.
- **Materials for the Future** - seeks to optimize materials for national security applications by predicting and controlling their performance and functionality through discovery science and engineering.
- **Nuclear and Particle Futures** - applies science and technology to intransigent problems of system identification and characterization in areas of global security, nuclear defense, energy, and health.
- **Science of Signatures** - integrates nuclear experiments, theory, and simulation to understand and engineer complex nuclear phenomena.

Computational Physics and Methods Group, CCS-2

“Performing innovative simulations of physics phenomena on tomorrow’s scientific computing platforms.”

- CCS Division was formed to strengthen the visibility and impact of computer science and computational physics research on strategic directions for the Laboratory.
- Both computer science and computational science are now central to scientific discovery and innovation. They have become indispensable tools for all other scientific missions at the Laboratory.
- CCS Division forms a bridge between external partners and Laboratory programs, bringing new ideas and technologies to bear on today’s important problems and attracting high-quality technical staff members to the Laboratory.
- **The Computational Physics and Methods Group CCS-2 conducts methods research and develops scientific software aimed at the latest and emerging HPC systems.**

Overview of the Transport Project

- Delivers radiation transport capabilities software libraries:
  - CAPSAICIN: X-ray thermal transport using finite-element methods on unstructured meshes
  - JAYENNE: X-ray thermal transport using Implicit Monte Carlo (IMC)
  - PARTISN: Neutron/gamma transport using deterministic methods on structured meshes
  - MCATK: Neutron transport using continuous-energy Monte Carlo
- Each library is massively-parallel and runs on the latest high-performance computing hardware
- 20+ staff, + students, + post-docs
- Multidisciplinary: **nuclear** and other engineering disciplines, physicists, mathematicians, and computer scientists.

Our codes approximate variants of the Boltzmann Transport Equation

- Linear radiation transport equation for neutrons

$$\left(\frac{1}{v}\frac{\partial}{\partial t} + \Omega \cdot \nabla + \Sigma_t\right)\psi(\boldsymbol{r}, \boldsymbol{\Omega}, \mathrm{E})$$
$$= \int_{4\pi}\int_0^{\infty} \Sigma_s(\boldsymbol{\Omega}' \rightarrow \boldsymbol{\Omega}, E' \rightarrow E)\psi(\boldsymbol{r}, \boldsymbol{\Omega}', \mathrm{E}')dE'd\Omega'$$
$$+ \frac{\chi(E)}{k}\int_{4\pi}\int_0^{\infty} \nu\Sigma_f(E')\psi(\boldsymbol{r}, \boldsymbol{\Omega}', \mathrm{E}')dE'd\Omega' + Q$$

- Nonlinear radiation transport equation for thermal x-rays

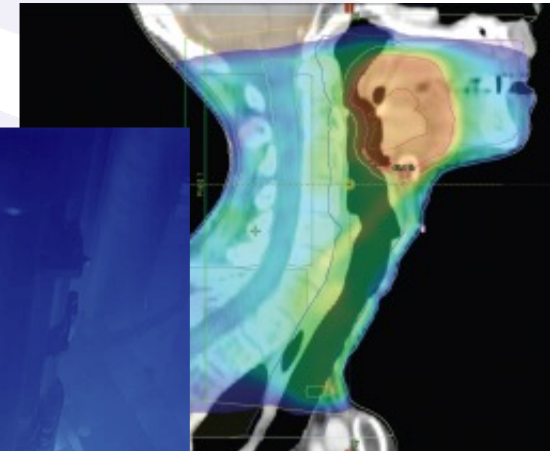
$$\frac{1}{c}\frac{\partial I}{\partial t} + \boldsymbol{\Omega} \cdot \nabla I + \sigma(\boldsymbol{r}, \boldsymbol{\Omega}, \mathrm{v})\mathrm{I}(\mathbf{r}, \boldsymbol{\Omega}, \mathrm{v}, \mathrm{t}) = \sigma B(v, T) + \frac{Q_r}{4\pi}$$

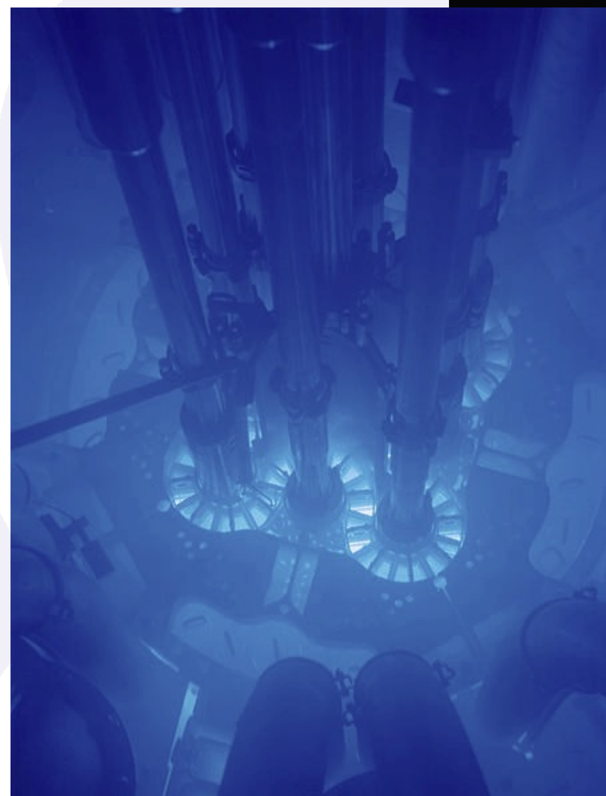
  coupled with hydrodynamic motion at temperature T

- The seven-dimensional solution domain (space+velocity+time) often results in billions, sometimes trillions, of unknowns.

Radiation Transport Applications in CCS-2

- The transport project in CCS-2 develops neutron and x-ray transport codes
- Neutral particle, linear transport
  - Nuclear reactor simulations
    - Criticality (k-eff)
    - Isotope depletion
    - Shielding
  - Medical, e.g., brachytherapy
  - Oil well logging
  - Supernova explosion
- Nonlinear transport of thermal x-rays
  - Inertial confinement fusion
    - National Ignition Facility (LLNL)
    - Laboratory for Laser Energetics (Rochester)
  - Astrophysics (Center for Theoretical Astrophysics, LANL)
  - Z Pulsed-Power machine (Sandia)



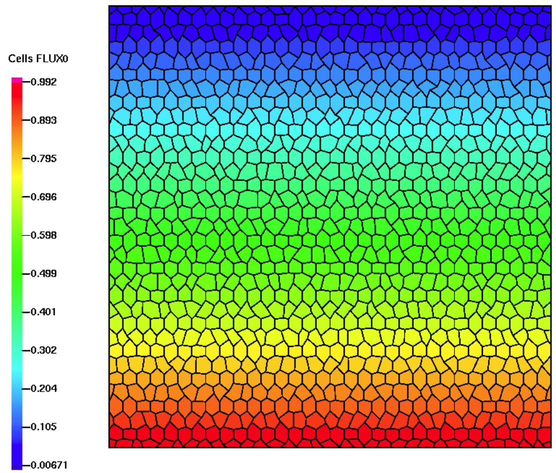


Los Alamos NATIONAL LABORATORY

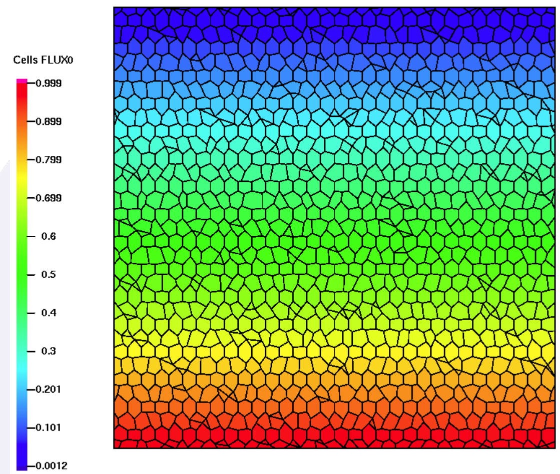
CAPSAICIN – X-ray Thermal Transport

CAPSAICIN is a finite-element transport library for general mesh topologies

- A collection of object-oriented C++ components.
  - 3-temperature (ion, electron, radiation), time-dependent radiative transfer
  - Steady-state radiation transport
  - Non-local tensor diffusion
- We make heavy use of the third-party software libraries; e.g., Trilinos, ParMetis, and SuperLU_DIST. We collaborate with the developers of many of these libraries.



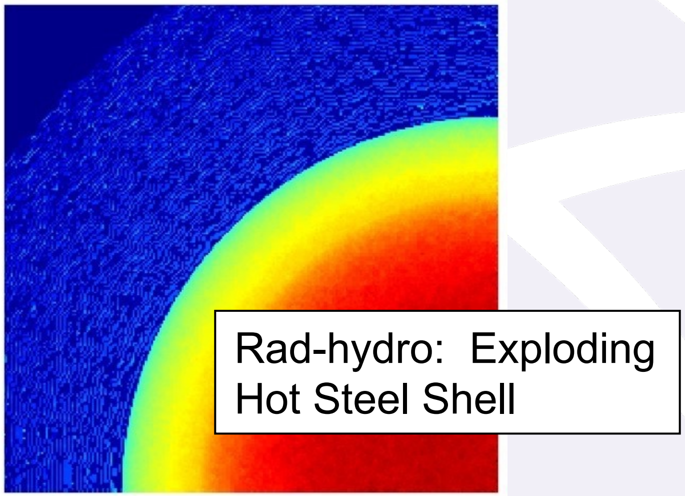
non-convex mesh

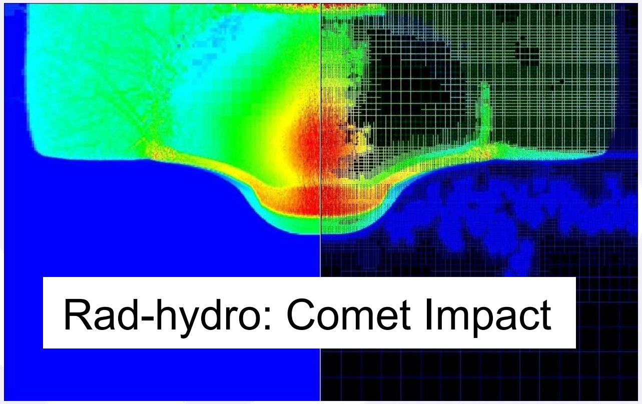


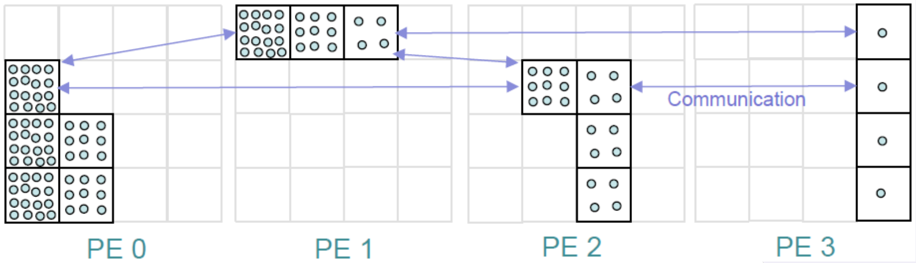
convex mesh

JAYENNE: X-Ray Thermal Transport using IMC

- Implicit Monte Carlo for x-ray transport, nonlinearly coupled with hydrodynamics for high-energy density physics simulations
- Object-oriented C++, sharing much code with Capsaicin (via open-source Draco library)
- Multiphysics: Jayenne interfaces with rad-hydro application codes
- Massively Parallel (MPI; MPI+X)
  - Mesh replication or decomposition for distributing particle workload
  - Adapted to Roadrunner's IBM Cell chip, NVidia GPU, KNL, etc.
  - *Branson* mini-app (https://github.com/lanl/branson)

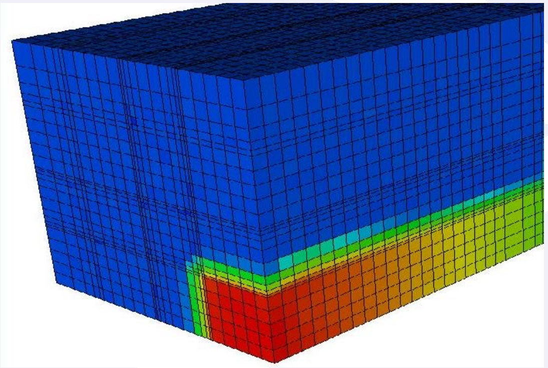




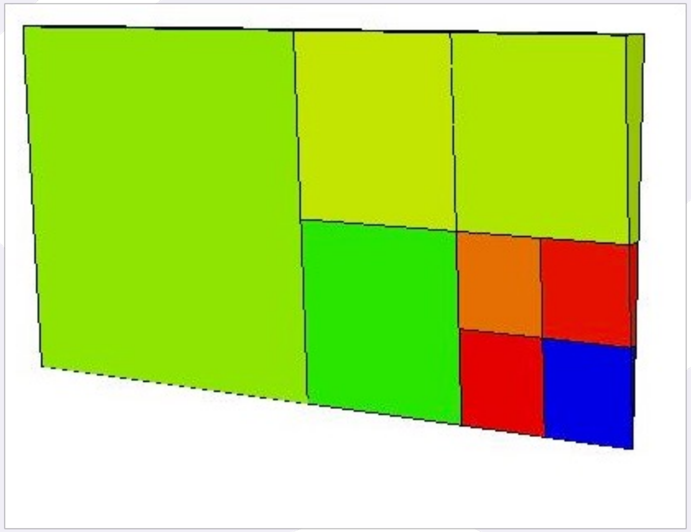


PARTISN – Neutron/Gamma Deterministic Transport

The PARTISN team covers a wide variety of research:



- We focus on providing computational solutions to the linear Boltzmann transport equation for neutral particles using the S_N method.
- Our latest research includes novel material-motion discretizations and energy group treatments.
- We use traditional structured and block Adaptive Mesh Refinement structured meshes, for static and time-dependent problems.



Los Alamos NATIONAL LABORATORY

PARTISN Code Capabilities

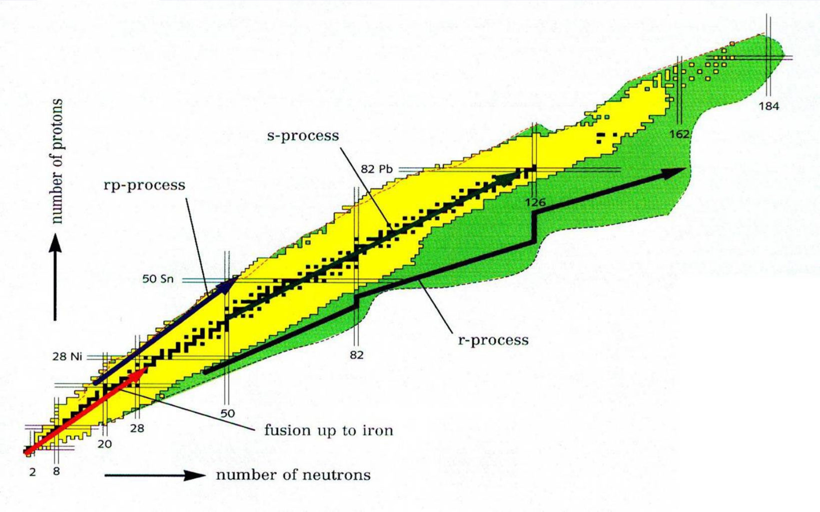
- PARTISN: PARallel TIme-dependent SN
- Solves both the static and time-dependent forms of the linear Boltzmann transport equation
  - Forward and adjoint mode
  - Static eigenvalues: k and α
  - Time-dependent stochastic neutron transport equations:
    - Probability of Initiation (static) and Probability of Survival: probability of obtaining a divergent chain
    - Moments of the (instantaneous) neutron and (total) fission numbers
  - Discretization strategies:
    - Discrete ordinate approximation for treating the angular variable
    - Multigroup in energy (with the Bondarenko method for self-shielding)
    - Diamond difference (standard or adaptive weighted) for the spatial variable
    - Crank-Nicholson in time
    - Set-to-zero fixup to maintain positivity (because of negative sources, strong gradients, large spatial cells)
  - Acceleration via diffusion synthetic acceleration method or the transport synthetic acceleration
  - Parallelization is performed via 2-D spatial decomposition ("KBA") and/or Jacobi iteration for energy decomposition

PARTISN – Neutron/Gamma Deterministic transport

Our team adds new physics, develops better solution methods, translates these into software and helps our users understand their results

- Our worldwide users perform a wide variety of 1-D/2-D/3-D calculations, so we require robust and efficient methods for reducing transport solution iterations (e.g., "KBA" sweeps, Diffusion Synthetic Acceleration)
- We have long been involved with work and research in nuclear application areas such as criticality safety, reactor engineering/shielding, sensitivity analysis, inverse problems, etc.
- We are also applying our methods and solvers to non-traditional areas such as the r-process in core-collapse in supernovae and neutron-star mergers, which requires material-motion corrections.

Figure: R-Process Nucleosynthesis
Rapid neutron capture in these types of events is believed to account for the creation of roughly have the abundance of materials heavier than Fe



PARTISN Code Generally

- Six dedicated team members (+/- students!) who spend varying amounts of time working on the code and supporting LANL students
  - Erin Davis (CCS-2 Group Leader), Nathan Hart (MMC on GPUs), Mario Ortega (Block AMR), Thomas Saller (Team Lead/Cross sections), Joe Zerr (GPUs), and with help from Jon Dahl (general expertise and years of code knowledge)
  - Students: Jawad Moussa (Ph.D.)
  - Currently, about 122k lines of (mostly) modern Fortran
  - Methods development
    - Material-motion advection algorithms
    - Using machine learning to improve multigroup data (my work!)
    - Self-shielding corrections for multigroup data
    - Solutions to the stochastic neutronics equations
    - Use of Block Adaptive Mesh Refinement (AMR) (my work)
    - Efficient and scalable computational transport solution methods for massively parallel architectures using a “MPI+X” paradigm
  - Software maintenance
    - Bug fixes
    - Code modernization (goodbye ancient Fortran!)
  - User support
    - Help our users run the code *and* understand the results
  - The code is available through RSICC and is installed on LANL and LLNL machines

The PARTISN Team

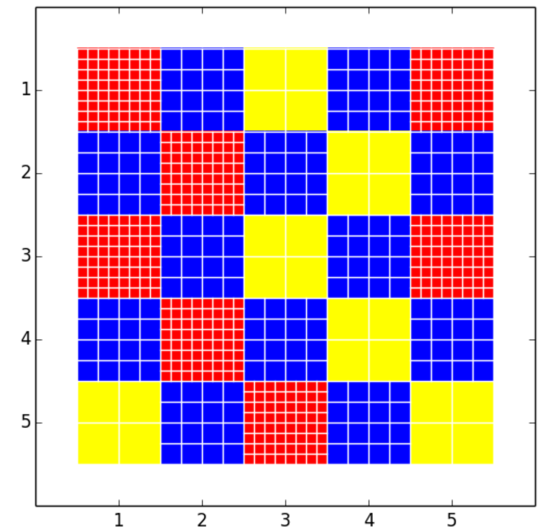
- Six dedicated team members from different nuclear engineering programs and with very different PhD projects.

- We all do very different things now and our diverse backgrounds and experiences bring valuable insight to writing really good neutron transport code and developing interesting methods.

  - Erin Davis (CCS-2 Group Leader)
    - PhD, University of New Mexico, Stochastic Methods for Uncertainty Quantification in Radiation Transport
  - Thomas Saller (Team Lead)
    - PhD, University of Michigan, Asymptotic Homogenized SP2 Approximations to the NTE
  - Joe Zerr
    - Penn St., Solution of the Within-Group Discrete Ordinates Equation on Massively Parallel Architectures
  - Nathan Hart
    - NC State, Spatial Error Estimators for SN Neutron Transport
  - Mario Ortega
    - University of California, Berkeley, A Rayleigh Quotient Fixed Point Method for Criticality Eigenvalue Problems in Neutron Transport
  - Jon Dahl
    - University of Arizona, Positive Anisotropic Scattering Sources for Discrete Ordinate Methods

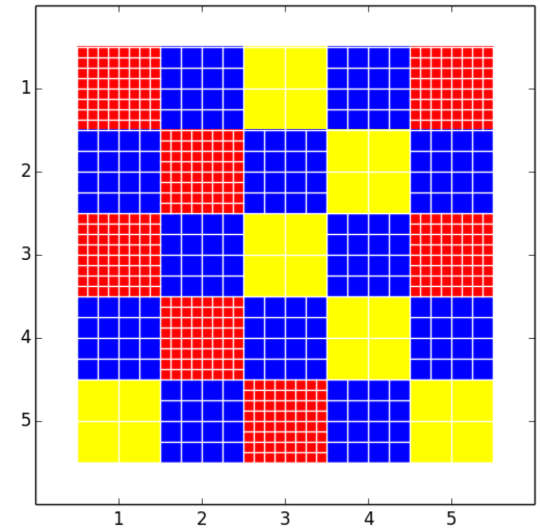
My Current Work – Block Automated Mesh Refinement (Block AMR)

- Splits the spatial domain into block regions with a refinement level.
- This refinement level is a function of material cross sections. We add refinement where we need it and coarsen where we can get away it.
- Saves memory and improved time to solution (sometimes…).
- Load balancing remains an open question.



My Current Work – Block AMR Load Balancing

- Block AMR can decompose the spatial mesh across MPI rank for increased parallelism.
- Currently we decompose our block mesh along the x-axis. That means each MPI rank owns a row of blocks.
- This leads to potential issues where one MPI rank has a lot of work and others have little work.
- Currently investigating load balancing schemes that respect the ability to sweep across the mesh.



My Current Work – Tensor Train Neutron Transport

We are always solving a linear system of equations when solving the discretized transport equation. There is structure to these equations that we leverage using the tensor train format.

$$
\frac{\mu_\ell}{4\Delta x_i}\left[\sum_{k'=k-1}^{k}\sum_{j'=j-1}^{j}\psi_{g,\ell,k',j',i}-\psi_{g,\ell,k',j',i-1}\right]+\frac{\eta_\ell}{4\Delta y_j}\left[\sum_{k'=k-1}^{k}\sum_{i'=i-1}^{i}\psi_{g,\ell,k',j,i'}-\psi_{g,\ell,k',j-1,i'}\right]+
$$

$$
\frac{\xi_\ell}{4\Delta z_k}\left[\sum_{j'=j-1}^{j}\sum_{i'=i-1}^{i}\psi_{g,\ell,k,j',i'}-\psi_{g,\ell,k-1,j',i'}\right]+\frac{\sigma_{g,k,j,i}}{8}\left[\sum_{k'=k-1}^{k}\sum_{j'=j-1}^{j}\sum_{i'=i-1}^{i}\psi_{g,\ell,k',j',i'}\right]=
$$

$$
\frac{1}{k_{\text{eff}}}\frac{1}{8}\sum_{g'=1}^{G}\chi_{g'g}\nu\sigma_{f,g',k,j,i}\sum_{\ell'=1}^{L}w_{\ell'}\left[\sum_{k'=k-1}^{k}\sum_{j'=j-1}^{j}\sum_{i'=i-1}^{i}\psi_{g',\ell',k',j',i'}\right]+
$$

$$
\frac{1}{8}\sum_{g'=1}^{G}\sigma_{s,g,g',k,j,i}\sum_{\ell'=1}^{L}w_{\ell'}\left[\sum_{k'=k-1}^{k}\sum_{j'=j-1}^{j}\sum_{i'=i-1}^{i}\psi_{g',\ell',k',j',i'}\right],
$$

for $g=1,\dots,G$, $i=1,\dots,M$, $j=1,\dots,J$, $k=1,\dots,K$, and $\ell=1,\dots,L$.

Tensor Train Neutron Transport – A Matrix Approach

- We can always write transport operators as the products of matrices and vectors.
- We can write matrix equations for the leakage/absorption, scattering, and fission operators.
- However, as the number of unknowns increases, the size of the linear system becomes untenable.

$$H_\mu = I_G \otimes \hat{\mu} \otimes Z(S_z \otimes S_y \otimes \Delta x^{-1} D_x)$$

$$H_\eta = I_G \otimes \hat{\eta} \otimes Z(S_z \otimes \Delta y^{-1} D_y \otimes S_x)$$

$$H_\xi = I_G \otimes \hat{\xi} \otimes Z(\Delta z^{-1} D_z \otimes S_y \otimes S_x)$$

$$\Sigma = (I_G \otimes I_L \otimes Z)(I_L \otimes diag(\Sigma_1, \dots, \Sigma_G))(I_G \otimes I_L \otimes S)$$

$$\mathcal{S} = (I_G \otimes I_L \otimes Z)(I_G \otimes L^{0,+})\Sigma_s(I_G \otimes L^0)(I_G \otimes I_L \otimes S)$$

$$\mathcal{F} = (I_G \otimes I_L \otimes Z)(I_G \otimes L^{0,+})\Sigma_f(I_G \otimes L^0)(I_G \otimes I_L \otimes S)$$

$$\mathcal{H}\Psi = \mathcal{S}\Psi + \frac{1}{k}\mathcal{F}\Psi$$

Tensor Train Neutron Transport – A Tensor Approach

- Instead of flattening the matrix operators, work on them as tensors.
- You will have different matrix operators based on your boundary conditions but you can construct the operators as tensors.
- Using tensors as opposed to matrices reduces the storage required for each operator.

$$\underbrace{\begin{bmatrix} a \\ b \\ c \end{bmatrix} \otimes \begin{bmatrix} d \\ e \\ f \end{bmatrix} \otimes \begin{bmatrix} g \\ h \\ i \end{bmatrix} \otimes \begin{bmatrix} l \\ m \\ n \end{bmatrix}}_{\mathbf{N=3,\, d=4 \Rightarrow \color{red}{d \cdot N} = 4 \cdot 3 = 12 \text{ elements}}} \equiv \underbrace{\begin{bmatrix} adgl & adgm & adgn & adhl & adhm & adhn & adil & adim & adin \\ aegl & aegm & aegn & aehl & aehm & aein & aeil & aeim & aehn \\ afgl & afgm & afgn & afhl & afhm & afin & afil & afim & afhn \\ bdgl & bdgm & bdgn & bdhl & bdhm & bdin & bdil & bdim & bdhn \\ bfgl & bfgm & bfgn & bfhl & bfhm & bfin & bfil & bfim & bfhn \\ begl & begm & begn & behl & behm & behn & beil & beim & bein \\ cdgl & cdgm & cdgn & cdhl & cdhm & cdin & cdil & cdim & cdhn \\ cfgl & cfgm & cfgn & cfhl & cfhm & cfin & cfil & cfim & cfhn \\ cegl & cegm & cegn & cehl & cehm & cehn & ceil & ceim & cein \end{bmatrix}}_{\mathbf{N=3,\, d=4 \Rightarrow \color{red}{N^d} = 3^4 = 81 \text{ elements}}}$$

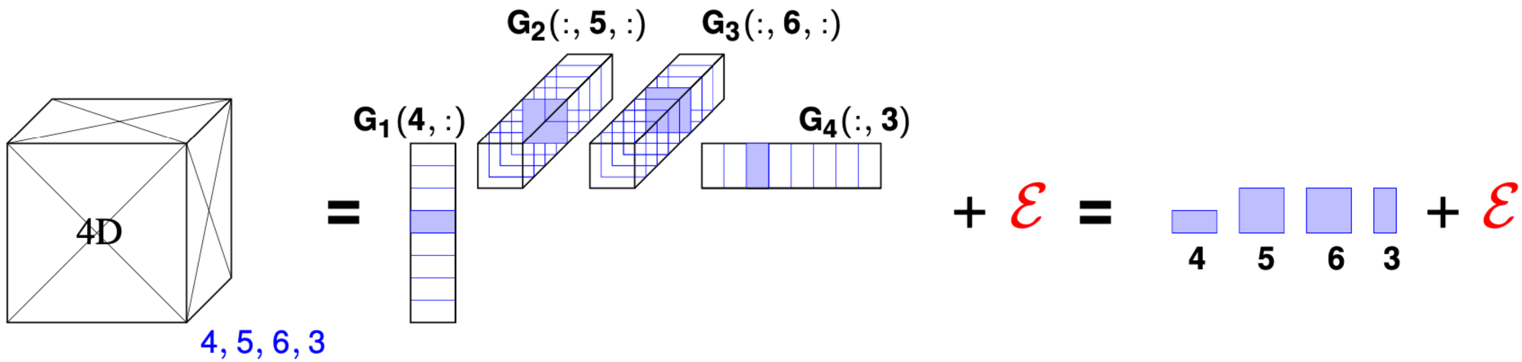
Tensor Train Neutron Transport – A Tensor Approach

- Instead of flattening the matrix operators, work on them as tensors.
- We have explicit representations of the transport operators as Kronecker products of matrices and vectors.
- In essence, we don't have to work with the full system and just do math on the individual components themselves.

$$\underbrace{\begin{bmatrix} adgl & adgm & adgn & adhl & adhm & adhn & adil & adim & adin \\ aegl & aegm & aegn & aehl & aehm & aein & aeil & aeim & aehn \\ afgl & afgm & afgn & afhl & afhm & afin & afil & afim & afhn \\ bdgl & bdgm & bdgn & bdhl & bdhm & bdin & bdil & bdim & bdhn \\ bfgl & bfgm & bfgn & bfhl & bfhm & bfin & bfil & bfim & bfhn \\ begl & begm & begn & behl & behm & behn & beil & beim & bein \\ cdgl & cdgm & cdgn & cdhl & cdhm & cdin & cdil & cdim & cdhn \\ cfgl & cfgm & cfgn & cfhl & cfhm & cfin & cfil & cfim & cfhn \\ cegl & cegm & cegn & cehl & cehm & cehn & ceil & ceim & cein \end{bmatrix}}_{\mathbf{N=3,\ d=4 \Rightarrow N^d = 3^4 = 81\ elements}} = \underbrace{\begin{bmatrix} \tilde{a} \\ \tilde{b} \\ \tilde{c} \end{bmatrix} \otimes \begin{bmatrix} \tilde{d} \\ \tilde{e} \\ \tilde{f} \end{bmatrix} \otimes \begin{bmatrix} \tilde{g} \\ \tilde{h} \\ \tilde{i} \end{bmatrix} \otimes \begin{bmatrix} \tilde{l} \\ \tilde{m} \\ \tilde{n} \end{bmatrix}}_{\mathbf{N=3,\ d=4 \Rightarrow d \cdot N = 4 \cdot 3 = 12\ elements}} + [\mathcal{E}]$$

Tensor Formats: The Tensor Train Format

- Every tensor element is the product of much smaller objects: matrices and 3D tensors.
- We introduce an error when doing this sort of decomposition. However, it is bounded and can be mitigated.
- For neutron transport, the decomposition is **exact**.



$$\mathcal{A}(4,5,6,3) = G_1(4,:)G_2(:,5,:)G_3(:,6,:)G_4(:,3)$$

My Current Work – Tensor Train Neutron Transport

The TT representation of the leakage/absorption operator $\mathbb{H}$, $\mathbb{H}^{TT}$, is then:

$$\mathbb{H} = \sum_{j \in \{x,y,z,\sigma\}} \sum_{b=1}^{8} \mathbb{H}_j^{bci}$$

where $\mathbb{H}_j^b$ is a matrix operator for a variable $j \in \{x, y, z, \sigma\}$ with a boundary condition b.

The $\mathbb{H}_j^1$, for the first boundary condition, $\mu_\ell < 0, \eta_\ell < 0, \xi_\ell < 0$, is defined as:

$$\mathbb{H}_x^1 = \mathbf{D}_x^- \circ \mathbf{Ip}_y^- \circ \mathbf{Ip}_z^- \circ \mathbf{Q}_\mu \circ \mathbf{I}_{G\times G}$$
$$\mathbb{H}_y^1 = \mathbf{Ip}_x^- \circ \mathbf{D}_y^- \circ \mathbf{Ip}_z^- \circ \mathbf{Q}_\eta \circ \mathbf{I}_{G\times G}$$
$$\mathbb{H}_z^1 = \mathbf{Ip}_x^- \circ \mathbf{Ip}_y^- \circ \mathbf{D}_z^- \circ \mathbf{Q}_\mu \circ \mathbf{I}_{G\times G}$$
$$\mathbb{H}_\sigma^1 = \mathbf{Ip}_x^- \circ \mathbf{Ip}_y^- \circ \mathbf{Ip}_z^- \circ (\mathbf{I}_{bc1} \otimes \mathbf{I}_{N^2/4}) \circ \mathrm{diag}(\sigma).$$

My Current Work – Tensor Train Neutron Transport

Each operator H_j^1 can be converted into the TT-representation for matrices, where each cores are reshapes of matrices from the above formulas. For example,

$$\mathbb{H}_x^{TT,1} = G_1 G_2 G_3 G_4 G_5$$

$$G_1 = \mathbf{D}_x^- \in \mathbb{R}^{(M+1)\times(M+1)\times 1}$$

$$G_2 = \mathbf{Ip}_y^- \in \mathbb{R}^{1\times(J+1)\times(J+1)\times 1}$$

$$G_3 = \mathbf{Ip}_z^- \in \mathbb{R}^{1\times(K+1)\times(K+1)\times 1}$$

$$G_4 = \mathbf{Q}_\mu \in \mathbb{R}^{1\times L\times L\times 1}$$

$$G_5 = \mathbf{I}_{G\times G} \in \mathbb{R}^{1\times G\times G}$$

Other $\mathbb{H}_j^{TT,bci}$ are built in a similar fashion.

My Current Work – Tensor Train Neutron Transport

- Results from a 3D Benchmark Problem:
  - We solved a 3D problem with 256 energy groups, and 512 spatial cells in each direction. We used an S8 quadrature.
  - This results in an angular flux solution vector that would be approximately 4.4 TB.
  - We would never form the matrix operators. We instead rely on matrix-free methods like those normally used in neutron transport.
  - However, using a MATLAB implementation of the tensor train operators, we can solve the entire problem on a 10-core i9 processor and 32 GB RAM
  - We see massive speedups and reductions in memory! This is very promising.

My Current Work – Tensor Train Neutron Transport: Results for a 3D Benchmark Problem

Results for Tensor Train Approach

$$\text{speed-up factor} = \frac{\text{PARTISN elapsed time} \times \text{Number of cores used by PARTISN}}{\text{TT/QTT elapsed time} \times \text{Number of cores used by TT/QTT}}$$

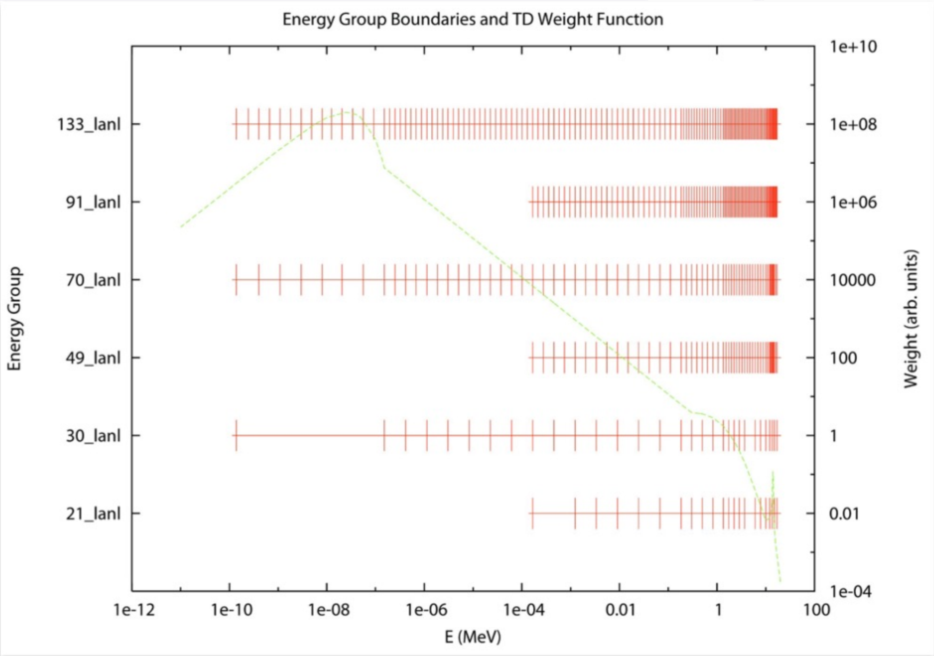
Grid Size	Number of Iterations	Elapsed Time (seconds)	Speed-up factor	fullsize of H (Zetabyte)	Compression Ratio of H	Ψ^{TT} Memory Storage (MB)	Eigenvalue Error
128	16	37.58	216.75	32	8.95e-18	1.03	6.53e-5
256	22	57.07	1348.5	2048	1.60e-19	0.98	3.26e-5
512	18	92.32	7564.4	131072	2.81e-21	1.15	3.44e-5

Results for PARTISN

Grid Size	Number of Iterations	Elapsed Time (seconds)	Time per Iter (seconds)	MPI Ranks	Scalar Flux Memory Storage (GB)	Eigenvalue
128	15	1131.323	75.42	72	4.29	1.0275823
256	15	5344.328	356.29	144	34.36	1.0276025
512	15	6061.987	404.13	1152	274.88	1.0276074

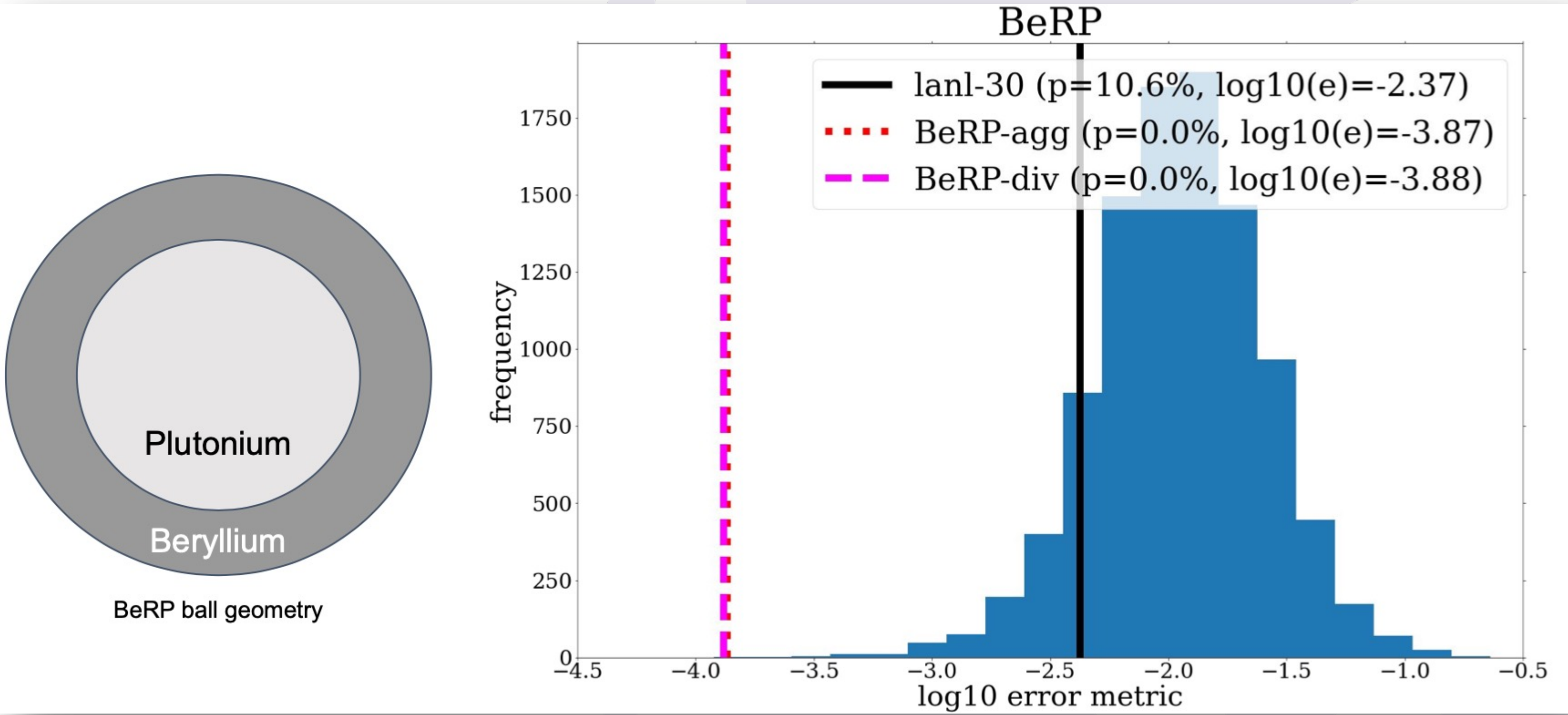
Previous Work – Machine Learning for Group Structures

- Post-Bac-led work (Leo Tunkle, now at Umich): Use ML to determine optimal group structures for problems of interesting using criticality benchmarks.
- LANL has traditionally used a 30 group structure that has worked well enough for us.
- However, we were curious if ML could give us additional insight. Can we get away with using fewer, better groups?

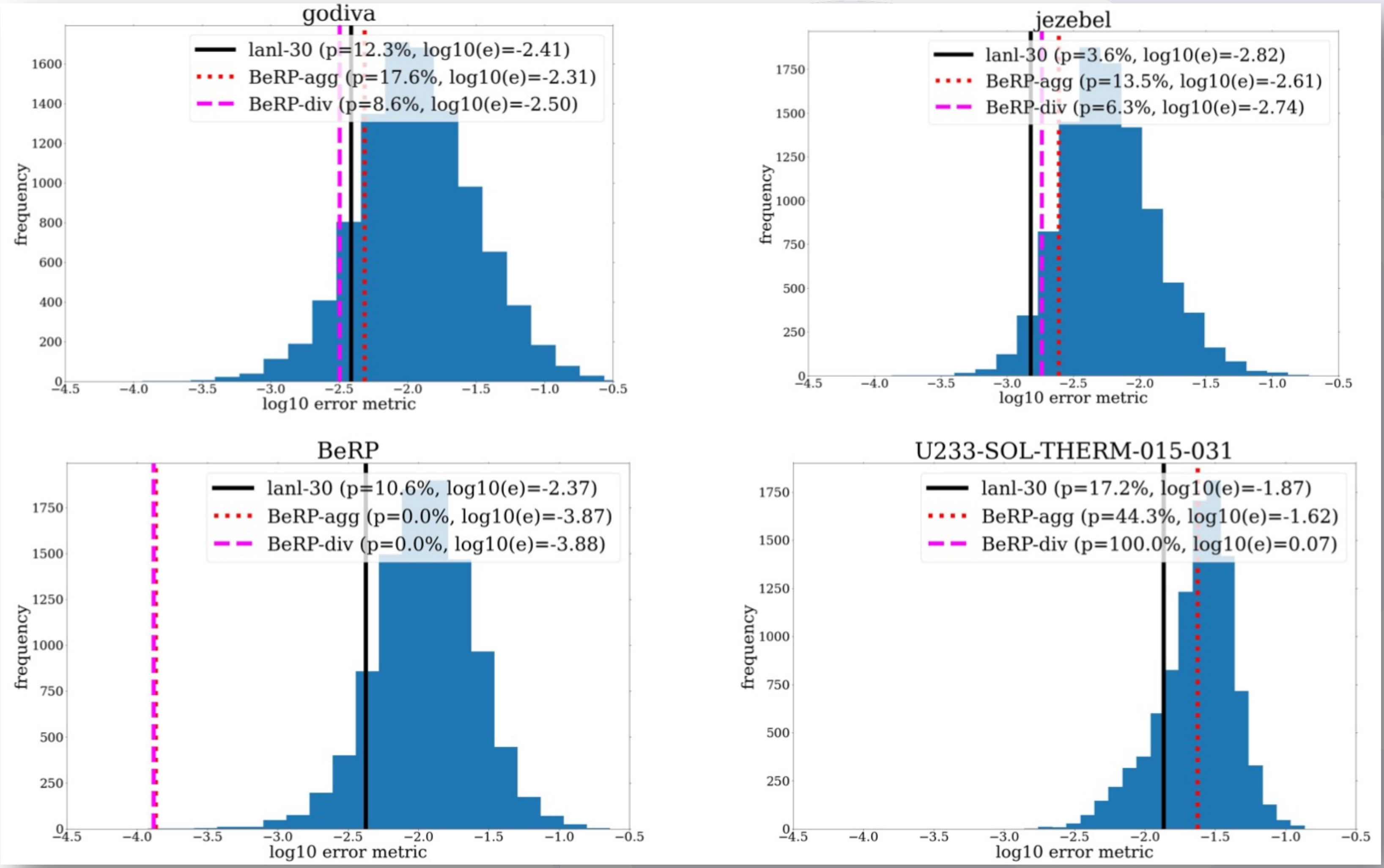


Previous Work – Machine Learning for Group Structures

- We trained on a critical assembly (BeRP) using hierarchical division, hierarchical agglomeration, and randomly generated group structures.



Previous Work – Machine Learning for Group Structures



Previous Work – Machine Learning for Group Structures

- Using ML, we can do pretty well for the set of problems we studied!
- However, random group structure generation can sometimes do better. Why? The ML algorithms we were looking at were greedy so we potentially ended up in local minima and did not find the global minimum.
- The error function is very important: you need to know what is important to your problems. Is it k-effective, reaction rates, leakage? ML does not absolve you of making decisions!
- We are doing similar work looking at the weight functions used to generate multi-group cross sections.

PARTISN Future Research Interests

- Solution methods for the transport equation with neutron-neutron scattering (for the R-process), currently being worked on by Zach Hardy (post-doc).
- Time-dependent multigroup weighting functions
- Relativistic moving material corrections
- Collaborations with NEN-2 (Advanced Nuclear Technology) on FVM (?), Rossi-alpha, etc.
- Neutron-Diagnosed Sub-critical Experiments (NDSE)
- Anisotropic fission (once data is available)

Internship Opportunities at LANL

- PARTISN has had students from many nuclear engineering programs.
- Recently, we've hosted students from UNM, UMich, Imperial School London, and Texas A&M.
- No students from Tennessee since I have been at the lab. Are you interested?

Summer Student Research Ideas

- Moving material corrections:
  - 2D momentum advection on GPUs
- Stochastic neutronics
  - Adding dead-time corrections to counts moments calculations
  - Modeling 3D point sources in reduced geometries
- Higher-order alpha eigenvalues
- Multigroup data
  - Application-dependent weight functions (+ optional machine learning aspect)
  - Using ultra-fine group data and collapsing on-the-fly
  - Generating new fission product data for LANL
- Anisotropic fission
  - Lack of data, but still want to evaluate importance

So What are the Opportunities You Speak Of?

Summer Schools
- A mix of lectures and project work
- Competitive selection process
- Size is between 20 and 30 students
- Students are paired with a mentor for project work

Graduate internships
- Mentor led
- Less structure than the schools, no formal lecture
- Students are encouraged to attend any talks/lectures/tours that are offered.

Post-bachelors/master
- Generally 1-2 years between degrees

Post-doctoral
- Generally 2-3 years to work on research with a designated mentor
- We are always looking for good postdoc candidates from varied fields.

Los Alamos
NATIONAL LABORATORY

Thank you!

Questions?