ORNL/TM-2021/2056

Implementation of the Windowed Multipole Method in Shift

Elliott D. Biondo

July 15, 2021

OAK RIDGE National Laboratory

ORNL IS MANAGED BY UT–BATTELLE, LLC, FOR THE U.S. DEPARTMENT OF ENERGY

ORNL/TM-2021/2056

Nuclear Energy and Fuel Cycle Division

IMPLEMENTATION OF THE WINDOWED MULTIPOLE METHOD IN SHIFT

Elliott D. Biondo

Date Published: July 15, 2021

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-Battelle, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The windowed multipole (WMP) method has been implemented in the Shift Monte Carlo (MC) radiation transport code with support for both CPU and GPU execution. With this method, small WMP data libraries (~100 MB) can be used to accurately Doppler broaden cross sections to arbitrary temperatures "on the fly" during an MC simulation. This approach yields significant memory savings relative to traditional methods, making it ideal for high-fidelity analysis such as coupled multiphysics simulations.

This document provides the exact forms of the WMP equations used by Shift, as well as a detailed description of the structure of WMP HDF5 data files provided by the Massachusetts Institute of Technology (MIT). The Shift implementation has been validated against the OpenMC radiation transport code, with excellent agreement demonstrated for 70 nuclides across an operative range of temperatures. CPU and GPU performance testing using a small module reactor (SMR) problem demonstrated that this method decreases the neutron tracking rate by a factor of ~2 on the Summitdev machine. A new set of WMP data being developed in-house will employ novel methods to improve tracking rates.

1. INTRODUCTION

Properly accounting for the temperature dependence of neutron cross sections is essential for criticality safety and multiphysics coupling applications. Neutron cross sections describe the probability per unit length that an incident neutron at energy E will interact with a target nucleus within some medium with a bulk temperature T. This probability is dependent on the relative speed of the interaction, i.e., the magnitude of the difference between the incident neutron velocity vector and target nucleus velocity vector. Target speeds follow a Maxwellian distribution, centered around T. At higher T, target speeds are more widely distributed, leading to a wider distribution of possible relative speeds. As a result, at higher temperatures, neutron cross section resonances become shorter and wider, a phenomenon known as Doppler broadening. This effect is especially important in the resolved resonance region (RRR), where the magnitude of cross sections can vary by an order of magnitude or more over small energy intervals ($\sim$1 eV).

For Monte Carlo (MC) neutron transport simulations, Doppler broadening is traditionally accounted for by creating separate sets of point-wise "continuous energy" (CE) cross section libraries for each temperature within a temperature grid, using the SIGMA1 [1] algorithm for Doppler broadening. Temperature-dependent cross sections are then calculated by either looking up the cross section at the nearest temperature grid point or linearly interpolating between cross sections at two temperature grid points. When calculations require a fine temperature grid to accurately capture temperature effects, cross section storage requirements on the order of $\sim$1 GB per temperature point make this method impractical.

The windowed multipole (WMP) method [5] can be used to calculate cross sections at any temperature "on the fly" during a Monte Carlo simulation. This can be accomplished from a single set of nuclear data on the order of $\sim$100 MB. This method calculates cross sections using pole and residue parameters from an alternative, mathematically equivalent nuclear data representation [4], and it has previously been implemented in the OpenMC [11] Monte Carlo radiation transport code.

The WMP method has been implemented on both the CPU and GPU sides of Shift MC radiation transport code [10], to be used with a set of WMP data generated by the Massachusetts Institute of Technology (MIT) Computational Reactor Physics Group. The remainder of this document is organized as follows. Section 2 describes the original non-windowed multipole method, followed by the WMP and the final forms of the WMP equation used in Shift. Section 3 describes the format of the HDF5-based WMP files provided by MIT. Section 4 describes the implementation of the WMP method in Shift and subsequent validation against OpenMC. Section 5 describes performance testing of the CPU and GPU implementation using a small modular reactor (SMR) problem. Section 6 provides concluding remarks.

2. THEORY

In this section, the original non-windowed version of the multipole method is first introduced. The windowed multipole method is then described, and the final answers used in the Shift implementation are presented.

2.1 ORIGINAL MULTIPOLE METHOD (NON-WINDOWED)

The multipole method, originally proposed by Hwang [4], is based on the observation that the standard R-matrix [12] form of the nuclear cross section can be reasonably approximated as a rational function in $\sqrt{(E)}$ space[1]. A partial fraction expansion can be performed on this rational function using the residue method [8]. This provides an expression for the cross section in terms of poles — singularities in the complex plane — and residues, which are proportional to the path integrals around these singularities. For some reaction x, the 0 K cross section is

$$\sigma_x(E, 0\,\mathrm{K}) = \frac{1}{E} \sum_j \Re \left[\frac{-i r_{x,j}}{p_j - \sqrt{E}} \right], \tag{1}$$

as seen in Forget et al. equation 4 [2]. In this equation p_j are the poles, $r_{x,j}$ are reaction-specific residues, i is the imaginary unit, and $\Re$ provides the real part of a complex-valued argument. To calculate the total cross section, two additional quantities are required: $\sigma_p(E)$, the potential cross section; and $e^{-2i\varphi_l}$, which is referred to as the "total cross section factor," where φ_l is the hard-sphere phase shift for the angular momentum quantum number l, noting that each resonance has an associated l. The total cross section is given by

$$\sigma_t(E, 0\,\mathrm{K}) = \sigma_p(E) + \frac{1}{E} \sum_j \Re \left[e^{-2i\varphi_l} \frac{-i r_{x,j}}{p_j - \sqrt{E}} \right], \tag{2}$$

as seen in Forget et al. equation 5. In Josey et al. [5], a slightly different but equivalent formulation is used with the $-i$ omitted. This document uses the Forget et al. formulation. The φ_l is calculated using the recursive relationship

$$\varphi_l = \varphi_{l-1} - \tan^{-1}\left(\frac{P_{l-1}}{l - S_{l-1}} \right), \quad \varphi_0 = \rho_l, \tag{3}$$

where the penetration factor is given by

$$P_l = \frac{\rho_l^2 P_{l-1}}{(l - S_{l-1})^2 + P_{l-1}^2}, \quad P_0 = \rho_l, \tag{4}$$

and the shift factor is given by

$$S_l = \frac{\rho_l^2 (l - S_{l-1})}{(l - S_{l-1})^2 + P_{l-1}^2} - l, \quad S_0 = 0. \tag{5}$$

[1] A rational function is one that can be expressed as the ratio of two polynomials.

Values of φ_l for $l = [0, 3]$ are

$$\varphi_l = \begin{cases} \rho_0, & l = 0, \\ \rho_1 - tan^{-1}(\rho_1), & l = 1, \\ \rho_2 - tan^{-1}\left(\frac{3\rho_2}{3-\rho_2^2}\right), & l = 2, \\ \rho_3 - tan^{-1}\left(\frac{\rho_3(15-\rho_3^2)}{15-6\rho_3^2}\right), & l = 3, \end{cases} \tag{6}$$

as seen in the SAMMY manual [6]. The quantity ρ must be provided by the WMP data.

These cross sections can be Doppler broadened to temperature T using the following equations:

$$\sigma_x(E, \xi) = \frac{1}{2E\sqrt{\xi}} \sum_j \Re\left[r_{x,j}\sqrt{\pi}W(z_{0,j}) - \frac{-ir_{x,j}}{\sqrt{\pi}} C\left(\frac{p_j}{\sqrt{\xi}}, \frac{\sqrt{E}}{2\sqrt{\xi}}\right)\right], \tag{7}$$

$$\sigma_t(E, \xi) = \sigma_p(E) + \frac{1}{2E\sqrt{\xi}} \sum_j \Re\left[e^{-2i\varphi_l}\left[r_{x,j}\sqrt{\pi}W(z_{0,j}) - \frac{-ir_{x,j}}{\sqrt{\pi}} C\left(\frac{p_j}{\sqrt{\xi}}, \frac{\sqrt{E}}{2\sqrt{\xi}}\right)\right]\right], \tag{8}$$

with

$$\xi = \frac{kT}{4A}, \tag{9}$$

and

$$z_{0,j} = \frac{\sqrt{E} - p_j}{2\sqrt{\xi}}, \tag{10}$$

as seen in Forget et al. Equations 7, 8, and 9. In these equations, W is the Faddeeva function. The function C can be neglected for "most practical cases" [4] and is therefore excluded henceforth.

2.2 WINDOWED MULTIPOLE METHOD

Using the original multipole method described in Section 2.1, the Faddeeva function must be evaluated for each pole. This approach is prohibitively computationally expensive, as important nuclides such as ^{238}U have $O(10^3)$ poles. With the windowed multipole method, the summations in Equations 7 and 8 are truncated so that the Faddeeva function is only evaluated for the subset of the poles. The poles outside of this subset are accounted for using an approximation.

With this approach, energy-space is subdivided into a series of non-overlapping "inner windows" spaced uniformly in $\sqrt{E}$-space. For each inner window, an outer window is created which contains all of the poles that significantly contribute to the cross section within the inner window. For some energy E within the inner window, the cross section is calculated in terms of contributions from poles inside the outer window, and outside of the outer window:

$$\sigma = \sigma_{\text{inside}} + \sigma_{\text{outside}}. \tag{11}$$

The σ_{inside} is calculated using the equations provided in Section 2.1, with summation indices from w_{start} to w_{end}. The contributions from poles outside the outer window are treated using a polynomial fitting function with N terms in the form, as adapted from Josey et al., Equation 14.

$$\sigma_{outside}(E, 0\,\text{K}) = \sum_{n=0}^{N-1} a_{w,n} E^{(n-2)/2}, \tag{12}$$

where values of $a_{w,n}$ are specific to reactions. In order to Doppler broaden $\sigma_{outside}$, the same $a_{w,n}$ values are used in the equation

$$\sigma_{outside}(E, \xi) = \sum_{n=0}^{N-1} a_{w,n} \mathfrak{D}_n, \tag{13}$$

where

$$\mathfrak{D}_n = \begin{cases} \frac{1}{E}\text{erf}\left(\frac{\sqrt{E}}{2\sqrt{\xi}}\right), & n = 0, \\ \frac{1}{\sqrt{E}}, & n = 1, \\ (2\xi + E)\frac{1}{E}\text{erf}\left(\frac{\sqrt{E}}{2\sqrt{\xi}}\right) + 2\sqrt{\frac{\xi}{\pi E}}\text{e}^{-\frac{E}{4\xi}}, & n = 2, \\ \left[2\xi(2n-3) + E\right]\mathfrak{D}_{n-2}, & n = 3, \\ \left[2\xi(2n-3) + E\right]\mathfrak{D}_{n-2} - (2\xi)^2(n-2)(n-3)\mathfrak{D}_{n-4}, & n \geq 4. \end{cases} \tag{14}$$

as adapted from Josey et al., Equations 15 and 16.

2.2.1 FINAL ANSWERS

The final answers for the 0 K cross sections can be found by combining the expressions for the cross sections inside and outside the window:

$$\sigma_x(E, 0\,\text{K}) = \frac{1}{E}\sum_{j=w_{\text{min}}}^{w_{\text{max}}} \Re\left[\frac{-ir_{x,j}}{p_j - \sqrt{E}}\right] + \sum_{n=0}^{N-1} a_{x,w,n} E^{(n-2)/2}, \tag{15}$$

$$\sigma_t(E, 0\,\text{K}) = \frac{1}{E}\sum_{j=w_{\text{min}}}^{w_{\text{max}}} \Re\left[e^{-2i\varphi_l}\frac{-ir_{x,j}}{p_j - \sqrt{E}}\right] + \sum_{n=0}^{N-1} a_{t,w,n} E^{(n-2)/2}. \tag{16}$$

The final answers for the Doppler broadened cross sections are as follows:

$$\sigma_x(E, \xi) = \frac{1}{2E\sqrt{\xi}}\sum_{j=w_{\text{min}}}^{w_{\text{max}}} \Re\left[\sqrt{\pi} r_{x,j} W(z_{0,j})\right] + \sum_{n=0}^{N-1} a_{x,w,n}\mathfrak{D}_n, \tag{17}$$

$$\sigma_t(E, \xi) = \frac{1}{2E\sqrt{\xi}}\sum_{j=w_{\text{min}}}^{w_{\text{max}}} \Re\left[e^{-2i\varphi_l}\left[\sqrt{\pi} r_{x,j} W(z_{0,j})\right]\right] + \sum_{n=0}^{N-1} a_{t,w,n}\mathfrak{D}_n. \tag{18}$$

The potential cross section σ_p, which appears in Equation 8 is accounted for by the fitting parameters [7].

Up to this point, it has been assumed that poles and residues use the Reich Moore (RM) formalism. In some cases, the nuclear data available necessitates the use of the Multi-Level Breit-Wigner (MLBW) formalism. When using MLBW poles and residues, an additional residue, r_c, is necessary to represent the "competitive" cross section. This affects the calculation of the total cross section, as shown below:

$$\sigma_t(E, 0\,\mathrm{K}) = \frac{1}{E} \sum_{j=w_{\min}}^{w_{\max}} \Re \left[e^{-2i\varphi_l} \frac{-i(r_{x,j} + r_{c,j})}{p_j - \sqrt{E}} \right] + \sum_{n=0}^{N-1} a_{t,w,n} E^{(n-2)/2}. \tag{19}$$

$$\sigma_t(E, \xi) = \frac{1}{2E\sqrt{\xi}} \sum_{j=w_{\min}}^{w_{\max}} \Re \left[\sqrt{\pi} \Big[e^{-2i\varphi_l} r_{x,j} + r_{c,j} \Big] W(z_{0,j}) \right] + \sum_{n=0}^{N-1} a_{t,w,n} \mathfrak{D}_n. \tag{20}$$

3. MIT FILE FORMAT

The equations used to calculate the cross section described in Section 2 require custom nuclear data libraries containing poles, corresponding residues, and other parameters. In January 2018, a WMP library containing 71 nuclides in separate HDF5 files was provided by the MIT Computational Reactor Physics Group. Subsequently, the complete version of this library, containing 423 nuclides, was provided by email in April, 2018. The total size of these 423 HDF5 files on disk is 91 MB. The format of these files is described below. Subsequent versions of this library are publicly available on GitHub [9], but contain files in a slightly different format.

Data for each nuclide was supplied in files named as `ZZZAAA.h5`, where `ZZZ` and `AAA` always have 3 digits. Each of these files is independent and contains all data necessary to calculate cross sections using the WMP method. Some of the datasets in these HDF5 files are extraneous, and others store necessary data in an idiosyncratic way. Table 1 lists only the datasets that are read from these HDF5 files by Shift. Note that the descriptions only apply to the MIT file format; they do not describe how data are actually stored in Shift.

To use the MIT-supplied data, all of the individual HDF5 files must be linked into a single HDF5 file using external links. This can be done using the code below. The resulting single file can be specified in the Omnibus using the `ce_pole_data_path` parameter.

Listing 1. Python code for creating a top-level HDF5 file.

```
from os import listdir
from h5py import File, ExternalLink

data_directory = "/path/to/data/directory"
output = "pole2.h5"
with File(output,'w') as f:
    for h5 in listdir(data_directory):
        f[h5.split(".")[0]] = ExternalLink(h5, "nuclide")
```

Table 1. MIT file format parameters (Note: this is the presumed format of files that are read by Shift; it does not reflect how data are stored internally in Shift)

dataset(s)	type	description
num_l	int	The number of possible l quantum states
fissionable	bool	Whether or not the nuclide is fissionable
formalism	int	Either 2 for MLBW or 3 for RM
spacing	double	The width of the windows in $\sqrt{E}$-space
start_E, end_E	double	The minimum and maximum incident neutron energies for which the data is valid
pseudo_k0RS	double	Equivalent to $\rho/\sqrt{E}$, an intermediate quantity used to calculate ϕ_l,
sqrt_AWR	double	The square root of the atomic weight ratio
length	int	The number of poles
windows	int	The number of windows
fit_order	int	Add 1 to this field to get the number of terms in the polynomial
l_value	1D array of int	Subtract 1 from each value in the array to get the l quantum number for each pole
w_start, w_end	1D array of int	The indices of the first and last poles (inclusive) included in each window, indexed from 1
broaden_poly	1D array of bool	Whether or not to broaden the polynomial within each window
data	2D array of complex<double>	The pole and reside data. The first dimension is the pole index. The second dimension provides the poles and the residues. For RM formalism, the order the data appears in the second dimension is pole, total residue, absorption residue, fission residue (when applicable). For MLBW formalism, the order is pole, total residue, competitive residue, absorption residue, fission residue (when applicable).
curvefit	3D array complex<double>	The polynomial data. The first dimension is the window index, the second dimension is the reaction index (appearing in the same order as data), and the third dimension is the polynomial index.

3.1 BOUND VS. UNBOUND HYDROGEN

The MIT WMP library contains a single file for ^{1}H. This file, `001001.h5`, contains data for unbound ^{1}H. This was initially a source of confusion, as in SCALE "1001" denotes bound hydrogen, and "8001001" denotes unbound hydrogen. Furthermore, pole data is not valid for bound nuclides in the $S(\alpha,\beta)$ region. To remedy this, the `001001.h5` file was modified to represent bound hydrogen by moving the lower energy bound (i.e., `start_E`) to be the lower energy bound of the second window, well above the $S(\alpha,\beta)$ region. This was done using the script below.

Listing 2. Python code for changing the lower bound of the 1001 data.

```
import h5py
import numpy as np

f = h5py.File('001001.h5', 'r+')

num_windows = f['/nuclide/windows'].value
start_E = f['/nuclide/start_E'].value
spacing = f['/nuclide/spacing'].value

new_start_E = (spacing + np.sqrt(start_E))**2
new_num_windows = num_windows - 1

print(new_start_E, new_num_windows)

f['/nuclide/windows'][...] = new_num_windows
f['/nuclide/start_E'][...] = new_start_E

f.close()
```

4. IMPLEMENTATION AND VALIDATION

The WMP method was implemented in Shift by creating three main classes— `Pole_Reader`, `Pole_Data`, and `Pole_XS_Calculator`—with the latter two implemented on both the CPU and the GPU. The `Pole_Reader` class reads the top-level HDF5 file of the MIT WMP data and stores pertinent information in `Pole_Data` objects (one `Pole_Data` object per nuclide). The `Pole_XS_Calculator` class calculates the cross section at a given energy and temperature using `Pole_Data` objects.

Although the functionality added to Shift is unit tested, additional validation was performed by comparing cross sections produced using the WMP method in Shift to those produced using the WMP method in OpenMC. For 70 of the 71 original nuclides, cross sections were calculated with Shift and OpenMC at each energy point in the continuous energy data between `start_E` and `end_E`. The `001001.h5` file was not included in this analysis because of the (now resolved) issue described in Section 3.1. The maximum relative difference between the results was then calculated using the formula

$$\text{maximum relative difference} = \max_{[E_{min},\, E_{max}]} \left(\frac{|\sigma_{\text{Shift}}(E) - \sigma_{OpenMC}(E)|}{\sigma_{\text{OpenMC}}(E)} \right). \tag{21}$$

This was done for all applicable reactions at three different temperatures: $T = 0$ K, $T = 300$ K, and $T = 1000$ K. These results appear in Appendix A. For the majority of the combinations of nuclide, reaction, and temperature, the maximum relative difference was less than 1×10^{-6}. The one notable outlier was ^{52}Cr (ZAID=24052). Maximum relative differences of 2.18×10^{-3} and 2.64×10^{-3} were observed for the absorption cross sections at $T = 300$K and $T = 1000$K, respectively.

Further investigation was conducted to determine the source of the discrepancy in the $T = 300$K case. It was found that this difference occurs at an energy of 572104.75001 eV, where Shift calculates the cross section to be 5.42641×10^{-4} b, and OpenMC calculates the cross section to be 5.41210×10^{-4} b. This energy is within a window where no poles are present and `broaden_poly` is true, meaning that cross section comes entirely from the contribution of the polynomial via Equation 13. The a_n and $\mathfrak{D}_n$ terms in this equation, as calculated by OpenMC and Shift, are shown in Table 2. These values are all identical within 12 decimal places, which indicates that the discrepancy in the cross section is a floating-point artifact that arises from Equation 13, which involves the addition and subtraction of values that vary over many orders of magnitude.

Table 2. Curvefit polynomial parameters produced by OpenMC and Shift

	a_n		$\mathfrak{D}_n$	
n	OpenMC	Shift	OpenMC	Shift
0	$3.563678816713\times10^{14}$	$3.563678816713\times10^{14}$	$1.747931650598\times10^{-6}$	$1.747931650598\times10^{-6}$
1	$-1.885377990642\times10^{12}$	$-1.885377990642\times10^{12}$	$1.322093661810\times10^{-3}$	$1.322093661810\times10^{-3}$
2	$3.740499363623\times10^{9}$	$3.740499363623\times10^{9}$	$1.000000001463\times10^{0}$	$1.000000001463\times10^{0}$
3	$-3.298209410335\times10^{6}$	$-3.298209410335\times10^{6}$	$7.563760671983\times10^{2}$	$7.563760671983\times10^{2}$
4	$1.090581541091\times10^{3}$	$1.090581541091\times10^{3}$	$5.721047550304\times10^{5}$	$5.721047550304\times10^{5}$

5. PERFORMANCE ASSESSMENT

The CPU and GPU implementations of the WMP method in Shift were performance tested by measuring neutron tracking rates during k-eigenvalue calculations of a quarter-core reactor model based on the NuScale SMR [3], as shown in Figure 1. Simulations were run using both fresh and depleted fuels. For the fresh fuel case, a single fuel material was used, consisting of 5 nuclides. The depleted fuel case consists of the fresh fuel material depleted for 30 days with 5-day time steps, resulting in 12,570 unique fuel materials containing an average of 240 nuclides.

All transport was performed at 595 K, corresponding to a SCALE CE library temperature. This provides the most conservative timing comparison, as it is the best scenario for CE data, and only a single temperature library needs to be loaded. Trials were run on a single node of the Summitdev machine consisting of 2 IBM Power8+ CPUs and 4 NVIDIA Tesla P100 GPUs. Results are are shown in Figures 2 and 3.

These results show a consistent factor of ~2 decrease in the tracking rate when using the WMP method on both the CPU and the GPU. This was investigated using `callgrind` on the CPU with the fresh fuel case to identify opportunities for future improvements. The resultant `callgrind` graph is shown in Figure 4 and is summarized Figure 5.

The results in Figure 5 show that 36% of execution time is spent evaluating the Faddeeva function and math built-ins. Work is currently underway to mitigate this issue. Reasonable approximations can be made to eliminate error function evaluations from the broadened polynomial. Improved windowing schemes will reduce the average and maximum number of poles per window, which should reduce the number of Faddeeva evaluations per cross section calculation. Progress on this forthcoming WMP library will be reported in a future document.

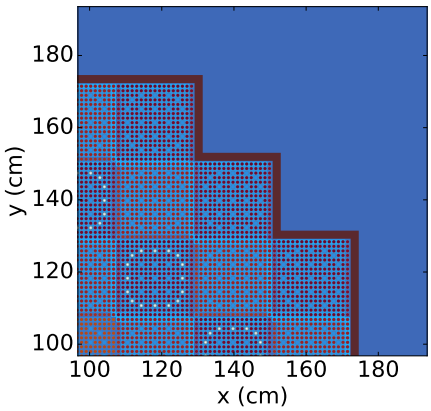

Figure 1. Quarter-core SMR model [3] used for tracking rate assessment.

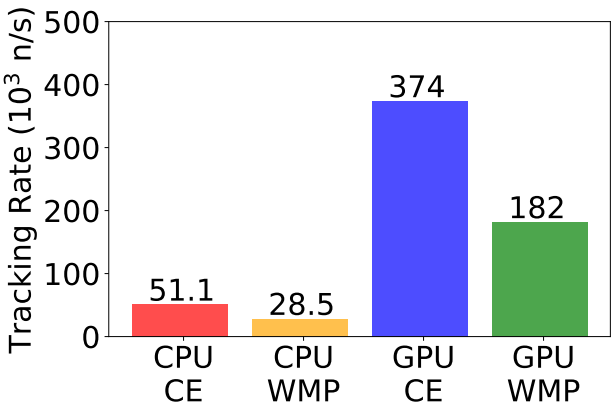

Figure 2. Tracking rate results for the fresh fuel case.

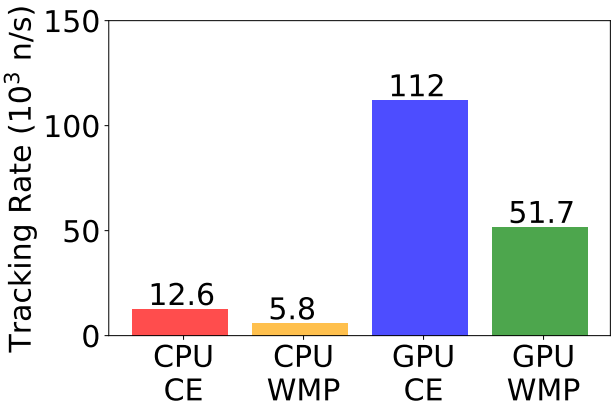

Figure 3. Tracking rate results for the depleted fuel case.

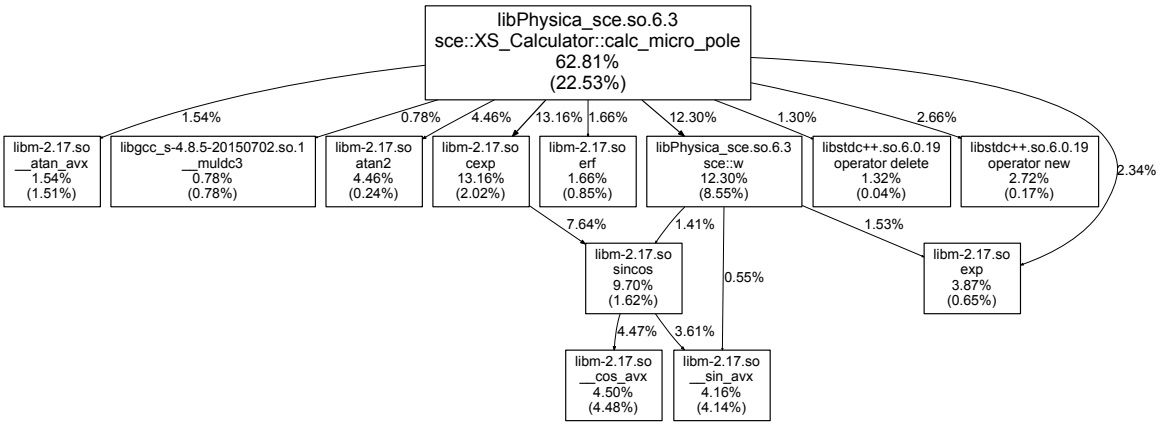

Figure 4. The `callgrind` graph for the CPU version fresh core SMR problem.

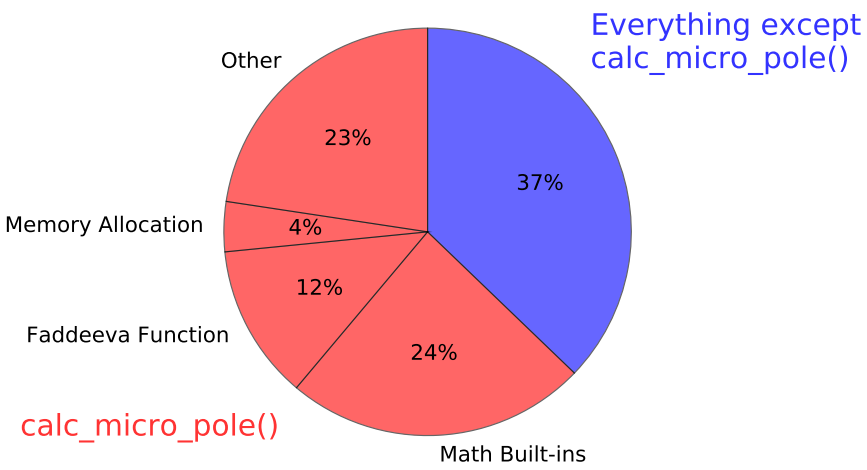

Figure 5. Summary of `callgrind` results for the fresh core SMR problem.

6. CONCLUSION

The windowed multipole method has been implemented on the CPU and GPU sides of Shift, allowing for accurate Doppler broadening at arbitrary temperatures with less than 100 MB of RAM. This memory savings comes with a trade-off: the factor of ~2 reduction in the CPU and GPU particle tracking rate. When combined with the complete 423-nuclide WMP data library supplied by MIT, this capability is ready for production-level use. Future work is currently underway to generate WMP data in-house in order to create data sets that will not incur the same performance penalties.

7. ACKNOWLEDGMENTS

This research was supported by the Exascale Computing Project (ECP), Project Number: 17-SC-20-SC, a collaborative effort of two DOE organizations: the Office of Science and the National Nuclear Security Administration. These offices are responsible for the planning and preparation of a capable exascale ecosystem—including software, applications, hardware, advanced system engineering, and early testbed platforms—to support the nation's exascale computing imperative.

This work was supported by Oak Ridge National Laboratory, which is managed and operated by UT-Battelle, LLC, for the US Department of Energy under Contract No. DEAC05-00OR22725. This research used resources of the Oak Ridge Leadership Computing Facility at Oak Ridge National Laboratory, which is supported by the Office of Science of the US Department of Energy under Contract No. DE-AC05-00OR22725.

REFERENCES

[1] D. E. Cullen and C. R. Weisbin. Exact Doppler broadening of tabulated cross sections. *Nuclear Science and Engineering*, 60(3):199–229, 1976.

[2] B. Forget, S. Xu, and K. Smith. Direct Doppler broadening in Monte Carlo simulations using the multipole representation. *Annals of Nuclear Energy*, 64:78 – 85, 2014.

[3] S. P. Hamilton and T. M. Evans. Continuous-energy Monte Carlo neutron transport on GPUs in the Shift code. *Annals of Nuclear Energy*, 128:236 – 247, 2019.

[4] R. N. Hwang. A rigorous pole representation of multilevel cross sections and its practical applications. *Nuclear Science and Engineering*, 96(3):192–209, 1987.

[5] C. Josey, P. Ducru, B. Forget, and K. Smith. Windowed multipole for cross section Doppler broadening. *Journal of Computational Physics*, 307:715 – 727, 2016.

[6] N. M. Larson. Updated users' guide for SAMMY: Multilevel R-matrix fits to neutron data using Bayes' equations. Technical Report ORNL/TM-9179/R8, Oak Ridge National Laboratory, 2008.

[7] J. Liang. Private Communication, February 2018.

[8] MathWorks. Partial fraction expansion (partial fraction decomposition). https://www.mathworks.com/help/matlab/ref/residue.html.

[9] MIT Computational Reactor Physics Group. Windowed multipole library. https://github.com/mit-crpg/WMP_Library.

[10] T. M. Pandya, S. R. Johnson, T. M. Evans, G. G. Davidson, S. P. Hamilton, and A. T. Godfrey. Implementation, capabilities, and benchmarking of Shift, a massively parallel Monte Carlo radiation transport code. *Journal of Computational Physics*, 308:239–272, Mar. 2016.

[11] P. K. Romano and B. Forget. The OpenMC Monte Carlo particle transport code. *Annals of Nuclear Energy*, 51:274, 2013.

[12] E. P. Wigner and L. Eisenbud. Higher angular momenta and long range interaction in resonance reactions. *Phys. Rev.*, 72:29–41, Jul 1947.

APPENDIX A. SHIFT VALIDATION RESULTS

APPENDIX A. SHIFT VALIDATION RESULTS

Table 3. Maximum relative difference between Shift results and OpenMC results over all

	T = 0 K			T = 300 K			T = 1000 K		
ZAID	total	absorption	fission	total	absorption	fission	total	absorption	fission
13027	5.82×10^{-8}	5.92×10^{-8}	-	2.11×10^{-7}	4.44×10^{-7}	-	1.65×10^{-7}	4.86×10^{-7}	-
14028	5.90×10^{-8}	5.85×10^{-8}	-	1.59×10^{-7}	7.95×10^{-7}	-	1.60×10^{-7}	9.03×10^{-7}	-
14029	5.77×10^{-8}	5.81×10^{-8}	-	1.49×10^{-7}	1.77×10^{-7}	-	1.63×10^{-7}	2.54×10^{-7}	-
14030	5.93×10^{-8}	5.87×10^{-8}	-	2.58×10^{-7}	2.95×10^{-7}	-	3.40×10^{-7}	4.08×10^{-7}	-
16032	5.90×10^{-8}	5.89×10^{-8}	-	2.58×10^{-7}	5.38×10^{-7}	-	2.46×10^{-7}	6.90×10^{-7}	-
16033	5.91×10^{-8}	5.90×10^{-8}	-	1.24×10^{-7}	8.20×10^{-8}	-	1.40×10^{-7}	1.20×10^{-7}	-
16034	5.83×10^{-8}	5.72×10^{-8}	-	1.30×10^{-7}	5.80×10^{-8}	-	1.50×10^{-7}	5.87×10^{-8}	-
20040	5.90×10^{-8}	5.91×10^{-8}	-	3.14×10^{-7}	8.68×10^{-7}	-	5.24×10^{-7}	9.41×10^{-7}	-
20042	5.93×10^{-8}	5.91×10^{-8}	-	4.97×10^{-7}	4.84×10^{-7}	-	6.10×10^{-7}	5.61×10^{-7}	-
20043	5.94×10^{-8}	5.88×10^{-8}	-	2.30×10^{-7}	3.44×10^{-7}	-	2.14×10^{-7}	4.58×10^{-7}	-
20044	5.86×10^{-8}	5.85×10^{-8}	-	3.41×10^{-7}	5.27×10^{-7}	-	5.88×10^{-7}	6.68×10^{-7}	-
20048	5.83×10^{-8}	5.71×10^{-8}	-	9.79×10^{-8}	5.86×10^{-8}	-	1.14×10^{-7}	6.32×10^{-8}	-
22046	5.92×10^{-8}	5.94×10^{-8}	-	3.33×10^{-7}	6.70×10^{-7}	-	4.11×10^{-7}	7.68×10^{-7}	-
22047	5.89×10^{-8}	5.86×10^{-8}	-	1.71×10^{-7}	5.24×10^{-7}	-	2.23×10^{-7}	5.59×10^{-7}	-
22048	5.95×10^{-8}	5.90×10^{-8}	-	4.65×10^{-7}	8.83×10^{-7}	-	6.31×10^{-7}	8.43×10^{-7}	-
22049	9.79×10^{-8}	1.09×10^{-7}	-	1.97×10^{-7}	6.03×10^{-7}	-	1.94×10^{-7}	6.96×10^{-7}	-
22050	5.85×10^{-8}	5.90×10^{-8}	-	4.16×10^{-7}	8.07×10^{-7}	-	3.87×10^{-7}	8.59×10^{-7}	-
23050	5.92×10^{-8}	9.49×10^{-8}	-	6.22×10^{-8}	1.43×10^{-7}	-	7.96×10^{-8}	1.61×10^{-7}	-
23051	5.85×10^{-8}	5.89×10^{-8}	-	2.13×10^{-7}	5.36×10^{-7}	-	1.76×10^{-7}	5.70×10^{-7}	-
24050	8.34×10^{-6}	6.23×10^{-5}	-	6.37×10^{-6}	6.23×10^{-5}	-	4.44×10^{-6}	6.22×10^{-5}	-
24052	4.60×10^{-5}	9.97×10^{-4}	-	6.04×10^{-5}	2.18×10^{-3}	-	9.14×10^{-5}	2.64×10^{-3}	-
24053	9.70×10^{-6}	9.73×10^{-5}	-	2.21×10^{-5}	9.73×10^{-5}	-	3.27×10^{-5}	9.73×10^{-5}	-
24054	5.92×10^{-8}	5.93×10^{-8}	-	4.01×10^{-7}	8.00×10^{-7}	-	3.22×10^{-7}	9.35×10^{-7}	-
25055	5.00×10^{-7}	7.97×10^{-7}	-	4.66×10^{-7}	8.19×10^{-7}	-	4.54×10^{-7}	8.54×10^{-7}	-
26054	5.94×10^{-8}	5.91×10^{-8}	-	2.73×10^{-7}	5.27×10^{-7}	-	3.13×10^{-7}	6.29×10^{-7}	-
26056	5.93×10^{-8}	5.93×10^{-8}	-	3.31×10^{-7}	7.74×10^{-7}	-	4.47×10^{-7}	7.99×10^{-7}	-
26057	1.34×10^{-5}	3.63×10^{-5}	-	4.44×10^{-5}	4.47×10^{-5}	-	7.98×10^{-5}	4.10×10^{-5}	-
26058	5.91×10^{-8}	5.87×10^{-8}	-	6.43×10^{-7}	6.71×10^{-7}	-	7.86×10^{-7}	7.83×10^{-7}	-
28058	5.94×10^{-8}	5.93×10^{-8}	-	5.59×10^{-7}	6.66×10^{-7}	-	5.43×10^{-7}	7.14×10^{-7}	-
28060	5.94×10^{-8}	5.93×10^{-8}	-	5.37×10^{-7}	8.36×10^{-7}	-	4.09×10^{-7}	8.78×10^{-7}	-
28061	5.91×10^{-8}	5.93×10^{-8}	-	2.47×10^{-7}	4.18×10^{-7}	-	2.68×10^{-7}	5.23×10^{-7}	-
28062	6.45×10^{-8}	2.53×10^{-7}	-	1.66×10^{-7}	6.66×10^{-7}	-	2.74×10^{-7}	8.01×10^{-7}	-
28064	5.90×10^{-8}	5.99×10^{-8}	-	2.04×10^{-7}	3.19×10^{-7}	-	2.49×10^{-7}	3.26×10^{-7}	-
29063	5.94×10^{-8}	1.27×10^{-7}	-	3.15×10^{-7}	6.27×10^{-7}	-	3.08×10^{-7}	6.64×10^{-7}	-
29065	9.03×10^{-8}	5.99×10^{-6}	-	2.12×10^{-7}	5.99×10^{-6}	-	2.81×10^{-7}	5.99×10^{-6}	-
40090	5.89×10^{-8}	5.87×10^{-8}	-	2.92×10^{-7}	5.95×10^{-7}	-	4.15×10^{-7}	7.25×10^{-7}	-
40091	5.92×10^{-8}	5.90×10^{-8}	-	2.98×10^{-7}	6.23×10^{-7}	-	3.43×10^{-7}	7.19×10^{-7}	-
40092	5.96×10^{-8}	5.91×10^{-8}	-	3.96×10^{-7}	7.48×10^{-7}	-	4.42×10^{-7}	8.26×10^{-7}	-
40094	5.89×10^{-8}	5.93×10^{-8}	-	4.34×10^{-7}	7.34×10^{-7}	-	5.62×10^{-7}	7.48×10^{-7}	-
40096	5.88×10^{-8}	5.79×10^{-8}	-	2.06×10^{-7}	5.46×10^{-7}	-	3.70×10^{-7}	6.69×10^{-7}	-
42092	5.84×10^{-8}	5.93×10^{-8}	-	4.05×10^{-7}	6.62×10^{-7}	-	2.88×10^{-7}	7.85×10^{-7}	-
42094	5.90×10^{-8}	5.95×10^{-8}	-	5.16×10^{-7}	7.26×10^{-7}	-	4.30×10^{-7}	8.61×10^{-7}	-
42096	5.93×10^{-8}	5.89×10^{-8}	-	5.28×10^{-7}	7.99×10^{-7}	-	5.90×10^{-7}	8.93×10^{-7}	-
42097	5.93×10^{-8}	5.94×10^{-8}	-	3.36×10^{-7}	4.86×10^{-7}	-	4.21×10^{-7}	6.00×10^{-7}	-
42098	5.93×10^{-8}	8.80×10^{-8}	-	5.07×10^{-7}	8.01×10^{-7}	-	6.83×10^{-7}	9.17×10^{-7}	-
42100	4.95×10^{-6}	2.86×10^{-6}	-	1.91×10^{-6}	5.44×10^{-6}	-	1.21×10^{-6}	8.33×10^{-6}	-
47107	5.91×10^{-8}	5.94×10^{-8}	-	4.58×10^{-7}	6.39×10^{-7}	-	4.70×10^{-7}	6.96×10^{-7}	-
47109	2.73×10^{-5}	1.76×10^{-4}	-	2.00×10^{-5}	1.69×10^{-4}	-	1.63×10^{-5}	1.28×10^{-4}	-
48106	5.94×10^{-8}	5.94×10^{-8}	-	5.22×10^{-7}	6.25×10^{-7}	-	6.11×10^{-7}	7.22×10^{-7}	-
48108	5.92×10^{-8}	5.93×10^{-8}	-	5.77×10^{-7}	6.65×10^{-7}	-	7.36×10^{-7}	7.70×10^{-7}	-
48110	5.89×10^{-8}	5.92×10^{-8}	-	6.36×10^{-7}	8.01×10^{-7}	-	6.78×10^{-7}	8.93×10^{-7}	-
48111	5.89×10^{-8}	5.93×10^{-8}	-	4.65×10^{-7}	5.59×10^{-7}	-	4.86×10^{-7}	6.56×10^{-7}	-

48112	5.94×10^{-8}	5.94×10^{-8}	-	5.95×10^{-7}	7.87×10^{-7}	-	6.80×10^{-7}	8.77×10^{-7}	-
48113	8.20×10^{-6}	1.37×10^{-4}	-	5.94×10^{-6}	1.34×10^{-4}	-	5.11×10^{-6}	1.18×10^{-4}	-
48114	5.92×10^{-8}	5.92×10^{-8}	-	6.16×10^{-7}	8.57×10^{-7}	-	7.13×10^{-7}	9.57×10^{-7}	-
48116	5.81×10^{-8}	5.92×10^{-8}	-	6.54×10^{-7}	8.68×10^{-7}	-	6.99×10^{-7}	9.88×10^{-7}	-
49113	5.90×10^{-8}	5.95×10^{-8}	-	4.32×10^{-7}	5.22×10^{-7}	-	4.75×10^{-7}	6.52×10^{-7}	-
49115	6.45×10^{-8}	3.65×10^{-7}	-	4.05×10^{-7}	5.45×10^{-7}	-	4.13×10^{-7}	6.00×10^{-7}	-
50112	5.87×10^{-8}	5.79×10^{-8}	-	4.62×10^{-7}	4.63×10^{-7}	-	5.99×10^{-7}	5.83×10^{-7}	-
50114	5.90×10^{-8}	5.76×10^{-8}	-	5.63×10^{-7}	6.03×10^{-7}	-	6.03×10^{-7}	7.30×10^{-7}	-
50115	5.75×10^{-8}	5.67×10^{-8}	-	4.30×10^{-7}	3.76×10^{-7}	-	4.61×10^{-7}	5.16×10^{-7}	-
50116	5.94×10^{-8}	5.95×10^{-8}	-	6.39×10^{-7}	9.36×10^{-7}	-	6.73×10^{-7}	1.03×10^{-6}	-
50117	5.91×10^{-8}	5.93×10^{-8}	-	5.27×10^{-7}	5.54×10^{-7}	-	4.96×10^{-7}	6.63×10^{-7}	-
50118	5.73×10^{-8}	5.82×10^{-8}	-	6.30×10^{-7}	6.76×10^{-7}	-	6.83×10^{-7}	8.35×10^{-7}	-
50119	5.86×10^{-8}	5.95×10^{-8}	-	4.20×10^{-7}	5.76×10^{-7}	-	5.11×10^{-7}	6.79×10^{-7}	-
50120	5.90×10^{-8}	1.84×10^{-7}	-	6.39×10^{-7}	9.79×10^{-7}	-	7.07×10^{-7}	1.09×10^{-6}	-
50122	6.00×10^{-8}	6.89×10^{-6}	-	5.20×10^{-7}	6.89×10^{-6}	-	4.91×10^{-7}	6.89×10^{-6}	-
50124	7.79×10^{-6}	8.85×10^{-5}	-	2.89×10^{-6}	8.85×10^{-5}	-	1.70×10^{-6}	8.85×10^{-5}	-
92234	5.91×10^{-8}	5.93×10^{-8}	8.14×10^{-8}	6.52×10^{-7}	7.53×10^{-7}	8.07×10^{-7}	6.99×10^{-7}	8.44×10^{-7}	9.07×10^{-7}
92235	1.69×10^{-7}	9.87×10^{-7}	6.04×10^{-6}	2.72×10^{-7}	4.91×10^{-7}	7.91×10^{-7}	2.81×10^{-7}	3.64×10^{-7}	7.53×10^{-7}
92238	5.94×10^{-8}	5.96×10^{-8}	6.59×10^{-7}	6.64×10^{-7}	9.47×10^{-7}	1.17×10^{-6}	6.85×10^{-7}	9.56×10^{-7}	1.28×10^{-6}